ТРУДЫ

ИНСТИТУТА СИСТЕМНОГО ПРОГРАММИРОВАНИЯ РАН

PROCEEDINGS OF THE INSTITUTE FOR SYSTEM PROGRAMMING OF THE RAS

ISSN Print 2079-8156
Том 36 Выпуск 1

ISSN Online 2220-6426
Volume 36 Issue 1

Институт системного
программирования
им. В.П. Иванникова РАН

Москва, 2024

ИСП РАН

ISSN 2079-8156 (Print), ISSN 2220-6426 (Online)

Труды Института системного программирования РАН
Proceedings of the Institute for System Programming of the RAS

Труды ИСП РАН – это издание с двойной анонимной системой рецензирования, публикующее научные статьи, относящиеся ко всем областям системного программирования, технологий программирования и вычислительной техники. Целью издания является формирование научно-информационной среды в этих областях путем публикации высококачественных статей в открытом доступе.
Издание предназначено для исследователей, студентов и аспирантов, а также практиков. Оно охватывает широкий спектр тем, включая, в частности, следующие:

- операционные системы;
- компиляторные технологии;
- базы данных и информационные системы;
- параллельные и распределенные системы;
- автоматизированная разработка программ;
- верификация, валидация и тестирование;
- статический и динамический анализ;
- защита и обеспечение безопасности ПО;
- компьютерные алгоритмы;
- искусственный интеллект.

Журнал издается по одному тому в год, шесть выпусков в каждом томе.
Поддерживается открытый доступ к содержанию издания, обеспечивая доступность результатов исследований для общественности и поддерживая глобальный обмен знаниями.
Труды ИСП РАН реферируются и/или индексируются в:

Proceedings of ISP RAS are a double-blind peer-reviewed journal publishing scientific articles in the areas of system programming, software engineering, and computer science. The journal's goal is to develop a respected network of knowledge in the mentioned above areas by publishing high quality articles on open access.
The journal is intended for researchers, students, and practitioners. It covers a wide variety of topics including (but not limited to):

- Operating Systems.
- Compiler Technology.
- Databases and Information Systems.
- Parallel and Distributed Systems.
- Software Engineering.
- Software Modeling and Design Tools.
- Verification, Validation, and Testing.
- Static and Dynamic Analysis.
- Software Safety and Security.
- Computer Algorithms.
- Artificial Intelligence.

The journal is published one volume per year, six issues in each volume.
Open access to the journal content allows to provide public access to the research results and to support global exchange of knowledge. **Proceedings of ISP RAS** is abstracted and/or indexed in:

УДК004.45

Адрес: 109004, г. Москва, ул. А. Солженицына, дом 25.
Телефон: +7(495) 912-44-25
E-mail: info-isp@ispras.ru
Сайт: http://www.ispras.ru/proceedings/

Address: 25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.
Tel: +7(495) 912-44-25
E-mail: info-isp@ispras.ru
Web: http://www.ispras.ru/en/proceedings

Содержание

Table of Contents

DOI: 10.15514/ISPRAS-2024-36(1)-1

Нейросетевые методы сжатия векторов для задачи приближенного поиска ближайших соседей

[1] *И.О. Буянов*, ORCID: 0009-0000-6994-151X, <*buyanov.igor.o@yandex.ru*>
[1] *В.В. Ядринцев*, ORCID: 0000-0002-4981-2515, <*vvyadrincev@gmail.com*>
[1,2,3,4] *И.В. Соченков,* ORCID: 0000-0003-3113-3765, <*sochenkov@isa.ru*>
[1] *Федеральный исследовательский центр Информатика и Управление РАН,*
119333, Россия, г. Москва, ул. Вавилова, д. 44, кор. 2
[2] *Институт системного программирования РАН,*
109004, Россия, г. Москва, ул. А. Солженицына, д. 25.
[3] *Университет Иннополис,*
420500, Россия, р. Татарстан, г. Иннополис, ул.Университетская, д. 1.
[4] *Сеченовский Университет*
119048, Россия, г. Москва, ул. Трубецкая, д. 8, стр. 2

Аннотация. В статье проверяется гипотеза применимости нейросетевых автокодировщиков как метод векторного сжатия для задачи приближенного поиска ближайших соседей. Проверка проводилась на нескольких больших датасетах с различными архитектурами автокодировщиков и индексов. Она показала, что, хотя ни одна из комбинаций автокодировщиков и индексов не может полностью превзойти чистые решения, в некоторых случаях они могут быть полезными. Мы также выявили некоторые эмпирические связи оптимальной размерности скрытого слоя и внутренней размерности наборов данных. Было также показано, что функция потерь является определяющим фактором качества сжатия.

Ключевые слова: приближенный поиск соседей; автокодировщики; крупномасштабный набор данных.

Для цитирования: Буянов И.О., Ядринцев В.В., Соченков И.В. Нейросетевые методы сжатия векторов для задачи приближенного поиска ближайших соседей. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 7–22. DOI: 10.15514/ISPRAS–2024–36(1)–1.

Благодарности: Работа выполнена при поддержке Фонда содействия инновациям (Договор № 12ГУКодИИС12-D7/72692 о предоставлении гранта на выполнение проекта открытых библиотек от 27 декабря 2021 г).

Neural Vector Compression In Approximate Nearest Neighbor Search On Large Datasets

[1] *I.O. Buyanov*, ORCID: 0009-0000-6994-151X, <*buyanov.igor.o@yandex.ru* >
[1] *V.V. Yadrinsev*, ORCID: 0000-0002-4981-2515, <*vvyadrincev@gmail.com* >
[1,2,3,4] *I.V. Sochenkov*, ORCID: 0000-0003-3113-3765, <*sochenkov@isa.ru* >
[1] *Federal Research Center "Computer Science and Control" of the Russian,*
44, Vavilov st., Moscow, 119333, Russia
[2] *Institute for System Programming of the Russian Academy of Sciences,*
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.
[3] *Innopolis University,*
1, Universitetskaya st., Innopolis, 420500, Russia.
[4] *Sechenov University,*
8, Trubetskaya st., Moscow, 119048, Russia.

Abstract. The paper examines the hypothesis of the applicability of neural autoencoders as a method of vector compression in the pipeline of approximate nearest neighbor search. The evaluation was conducted on several large datasets using various autoencoder architectures and indexes. It has been demonstrated that, although none of the combinations of autoencoders and indexes can fully outperform pure solutions, in some cases, they can be useful. Additionally, we have identified some empirical relationships between the optimal dimensionality of the hidden layer and the internal dimensionality of the datasets. It has also been shown that the loss function is a determining factor for compression quality.

Keywords: approximate nearest neighbor search; autoencoders; large datasets.

For citation: Buyanov I.O., Yadrinsev V.V., Sochenkov I.V. Neural vector compression in Approximate Nearest Neighbor Search on Large Datasets. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 7-22 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-1.

Acknowledgements. We thank Foundation for Assistance to Small Innovative Enterprises for funding this work (Agreement №12GUKodIIS12-D7/72692).

1. Введение

В настоящее время объем данных, доступный в Интернете, растет экспоненциально. Этот рост создает потребность в эффективных методах поиска информации, способных справиться с такими объемами. Информационный поиск (Information Retrieval, IR) – это процесс поиска и получения информации, соответствующей потребностям пользователя. Он включает в себя как поиск, так и фильтрацию больших объемов данных, чтобы представить наиболее релевантные результаты. Это особенно важно в таких областях, как научные исследования, где точность и актуальность полученной информации могут существенно повлиять на результативность проекта. К тому же экспоненциальный рост объема данных ведет к усложнению задачи поиска релевантной информации из-за постоянного ее роста.

Одной из наиболее активных областей исследований в области информационного поиска является задача приближенного поиска ближайших соседей (Approximate Nearest Neighbor, ANN). Она возникает в большом числе приложений: от систем рекомендаций до компьютерного зрения, где целью является поиск ближайших соседей для запроса в большой базе векторов, где точные алгоритмы поиска ближайших соседей могут быть вычислительно затратными и непрактичными для больших наборов данных. Это требует от исследователей поиска приближенных алгоритмов, которые могут возвращать приближенное решение с приемлемой точностью. Данные методы обычно включают построение такой структуры данных, как дерево или граф для эффективного сужения пространства поиска и избегания проверки всех возможных кандидатов. Другим возможным направлением является исследование различных методов сжатия векторов.

В частности, проблема с большими наборами данных возникает из-за того, что общая размерность векторов варьируется от сотен до нескольких тысяч. Это ведет к высоким требованиям к размеру хранилища, а также к оперативной памяти. Несмотря на то, что графические процессоры позволяют проводить быстрые матричные вычисления высокой размерности, способы сжатия векторного представления для экономии ресурсов представляют практический интерес.

В данной работе мы исследуем нейронные сети с архитектурой автокодировщика в качестве компрессора векторов для поиска ближайших соседей. Вероятно, они могут расцениваться неподходящими для этой задачи в связи с высокими вычислительными затратами. Тем не менее мы решили исследовать, какие возможные преимущества могут принести автокодировщики в роли векторного компрессора.

В статье мы проверяем гипотезу об использовании автокодировщиков в качестве метода сжатия векторов путем проведения обширных тестов различных методов приближенного поиска ближайших соседей с использованием нескольких автокодировщиков на нескольких наборах данных. Мы также покажем, что существует граница сжатия, пересечение которой на некоторых алгоритмах поиска вызывает быстрое падение качества независимо от типа автокодировщиков, показывающих определенный уровень качества до этой границы. Мы также показываем эмпирически, как это может быть связано с внутренней размерностью данных.

2. Смежные работы

Первые алгоритмы поиска были основаны на древовидных методах, таких как KD-tree [1] и Ball tree [2], которые организовывают данные в древовидную структуру для результативного поиска. Эти методы широко применяются для приближенного поиска ближайших соседей в пространствах низкой и средней размерности, где данные могут быть эффективно организованы в такую структуру. Предполагается несколько вариантов древовидных методов для улучшения их производительности, включая использование случайных KD-trees и иерархической кластеризации. Эти методы эффективны и действенны для данных с более низкой размерностью.

Как было упомянуто ранее, существует множество исследований по методам сжатия. Один из наиболее популярных методов сжатия – квантизация произведений (product quantization, PQ) [3]. Метод подразумевает разделение исходного пространства данных на подпространства и последующую их квантизацию независимо друг от друга. Метод показал высокую эффективность для высокоразмерных данных и применяется в таких областях, как поиск изображений и распознавание речи.

Еще одной значимой группой алгоритмов являются графовые методы. Например, NSW (Navigable Small World) [4] использует структуру сети "малого мира" для эффективной «навигации» по данным. Суть этой структуры заключается в поддержании баланса между локальными и глобальными связями. Этот метод доказал свою эффективность для высокоразмерных данных и используется в таких областях, как поиск изображений и поиск текста. HNSW [5] (Hierarchical Navigable Small World) является расширением NSW и использует иерархическую структуру, что улучшает эффективность исходного подхода. Графы HNSW разделяют данные на несколько уровней, каждый из которых содержит графы разного масштаба. Этот подход можно сравнить с почтой: сначала письмо пересылается между странами (граф стран), затем между городами (граф городов) и затем уже между городскими отделениями (граф отделений). Таким образом метод позволяет с высокой точностью осуществлять поиск по данным на разных масштабах и повышает общую эффективность приближенного поиска ближайших соседей. HNSW показал свою эффективность для высокоразмерных данных и применяются в таких областях, как поиск изображений и системы рекомендаций.

Еще одно направление исследований, которое также смежно с нашим, – это создание векторных представлений, потенциально пригодных для приближенного поиска ближайших соседей. Например, в работе [6] авторы предлагают обучаемый слой индексирования векторных представлений. Вместе со специальной регуляризации функции потерь, этот слой может быть встроен в любую глубокую нейросеть для задачи информационного поиска, позволяя совместно обучать векторные представления и поисковый индекс для них.

3. Проблема поиска ближайших соседей

Поиск ближайших соседей является основной проблемой во многих приложениях машинного обучения и анализа данных, где целью является поиск k ближайших соседей заданной точки запроса в большом наборе данных. Пусть $X = \{x_1, x_2, \ldots, x_n\}$ – набор из n векторов в d-мерном пространстве, а q – вектор запроса в d-мерном пространстве. K ближайших соседей q в X определяются как k векторов x_i в X, минимизирующих функцию расстояния $dist(q, x_i)$:

$$min_{\{i=1,2,..,k\}} dist(q, x_i)$$

где $dist(q, x_i)$ – функция расстояния, которая измеряет сходство или различие между вектором запроса q и вектором базы данных x_i. В данном случае используется евклидово расстояние.

Прямой подход, вычисляющий расстояния между точкой запроса и каждой точкой в наборе данных, имеет временную сложность $O(nd)$, что является непрактичным для больших наборов данных. Алгоритмы ANN стремятся найти приближенный набор из k ближайших соседей, которые находятся в непосредственной близости к истинным ближайшим соседям, тем самым существенно снижая вычислительную сложность.

Точность алгоритмов ANN обычно измеряется в терминах полноты (recall), которая представляет собой долю найденных истинных ближайших соседей, относительно всех истинных ближайших соседей. Пусть $NN(q)$ обозначает множество истинных k ближайших соседей вектора q в X. Полнота $R@k$ алгоритма ANN, который возвращает множество k векторов $S(q)$ для данной точки запроса q, определяется следующим образом:

$$R@k = \frac{|S(q) \cap NN(q)|}{|NN(q)|}$$

где $|S(q) \cap NN(q)|$ – количество элементов в пересечении множеств $S(q)$ и $NN(q)$, а $|NN(q)|$ – количество элементов в множестве $NN(q)$.

Эффективность алгоритмов ANN измеряется по времени запроса, то есть времени, необходимого для поиска k ближайших соседей для данной точки запроса. Целью алгоритмов ANN является достижение разумного соотношения между точностью и эффективностью за счет баланса между качеством приближенного решения и вычислительной сложностью его нахождения.

4. Использование автокодировщика для сжатия набора данных

Нейронные автокодировщики [7] представляют собой тип нейронных сетей, используемых для обучения без учителя. Они состоят из кодирующей (энкодера) и декодирующей (декодера) частей. Цель заключается в восстановлении входных данных на выходе. Энкодер отображает входные данные в скрытое пространство меньшей размерности, в то время как декодер отображает векторы из скрытого пространства обратно в исходное. Мы полагаем, что возможно использовать энкодер для снижения размерности векторизованных наборов данных.

Формально пусть $\boldsymbol{x}$ будет входным вектором, а $\boldsymbol{h}$ – вектор в скрытом пространстве. Энкодер отображает $\boldsymbol{x}$ в $\boldsymbol{h}$ с помощью функции $f(\boldsymbol{x}) = \boldsymbol{h}$. Аналогично, пусть $\mathbf{y}$ будет выходным вектором, а $\mathbf{g}$ – функция, отображающая $\mathbf{h}$ обратно в $\mathbf{y}$, $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{h})$. Цель автокодировщика

состоит в минимизации ошибки восстановления (the reconstruction loss), которая является разницей между входными данными **x** и выходными данными **y**. Одним из распространенных способов измерения ошибки восстановления является среднеквадратическая ошибка (MSE):

$$L(x, y) = \frac{1}{n}\sum_{\{i=1\}}^{\{n\}}(\boldsymbol{x}_i - \boldsymbol{y}_i)^2$$

где n – количество векторов на входе. Основное свойство, которым должен обладать автокодировщик для задачи поиска ближайших соседей, – сохранять отношения между векторами при преобразовании векторного пространства. В экспериментах мы используем несколько модификаций описанного автокодировщика, который мы будем называть Vanilla. Полагаем, что модифицированные варианты сохраняют отношения между точками лучше, чем Vanilla.

Для начала рассмотрим автокодировщик "The Neighborhood Reconstructing Autoencoders" [8], идея которого заключается в использовании аппроксимации функции декодера с помощью локальных графов. Эти графы отражают локальную геометрию распределения данных, что позволяет сделать автокодировщик более устойчивым к переобучению и проблемам связности. Для этого метода строится полностью связанный граф соседства, который используется в функции потерь:

$$L = \sum_i \sum_{\{x \in N(x_i)\}} ||x - f_\theta(g_{\phi(x)}; g_{\phi(x_i)})||$$

где $N(x_i)$ – множество соседних точек для x_i, а $\tilde{f}(.; g_\phi(x_i))$ – аппроксимация декодера $f_\phi(x)$ относительно закодированной точки $g_\phi(x)$.

Другая модификация – DCEC [9] (Deep Clustering with Convolutional Autoencoder), где в архитектуру автокодировщика Vanilla был добавлен слой кластеризации после энкодера. Этот слой является обучаемыми центроидами для кластеров в данных, что, как мы считаем, позволит проводить более детальное разделение векторного пространства и, как следствие, улучшит качество алгоритмов поиска, не основанных на графе. Математически слой кластеризации, как было сказано выше, представляет собой центроиды кластеров в виде обучаемых весов и отображает каждый скрытый вектор z_i в нестрогую метку q_i, которая рассчитывается следующим образом:

$$q_{\{ij\}} = \frac{\left(1 + \left|\left|z_i - \mu_i\right|\right|^2\right)^{-1}}{\sum_j \left(1 + \left|\left|z_j - \mu_j\right|\right|^2\right)^{-1}}$$

Функция потерь кластеризации определяется как: $L_c = KL(P||Q)$ где P – целевое распределение. Авторы предлагают эвристическое целевое распределение (см. уравнение 8). Тем не менее распределение может быть любым, которое (1) способно создавать нестрогие метки, (2) улучшать чистоту (purity) кластеров, (3) обращать внимание на точки с высокой уверенностью и (4) нормализовывать влияние центроидов, чтобы предотвратить искажение векторного пространства. Наконец, функция потерь кластеризации добавляется к функции потерь восстановления с помощью гиперпараметра Y, который контролирует ее влияние:

$$L = L_{rec} + \gamma L_c$$

Наконец, мы рассмотрим гиперболический автокодировщик (Hyperbolic Autoencoder [10]), работающий в гиперболическом пространстве. Авторы этого подхода используют модель сферы Пуанкаре, которая определяется как $B^n = \{x \in R^n: ||x||\}$ с метрическим тензором Римана. Особенностью этого автокодировщика является метрика расстояния между двумя точками (см. уравнение 2), которая делает расстояния близкимит в евклидовом пространстве экспоненциально большими в гиперболическом пространстве. Это позволяет эффективно моделировать сложные сети и структуры похожие на деревья.

5. Описание наборов данных

Чтобы сделать наш тест достаточно обширным, мы используем шесть наборов данных, основные статистические показатели которых приведены в табл. 1. Описание используемых датасетов:

- SIFT-Small Dataset [3]: компактная версия набора данных SIFT-1M, который состоит из 10 000 дескрипторов SIFT, извлеченных из набора изображений.
- SIFT-1M Dataset: один миллион дескрипторов SIFT, извлеченных из набора изображений. Обычно используется в качестве стандарта для оценки производительности алгоритмов поиска ближайших соседей.
- GIST Dataset [3]: набор данных из 1 миллиона дескрипторов GIST. Другой часто используемый эталон для алгоритмов ANN.
- GIST Dataset 1B: такой же, как датасет GIST, но с количеством в один миллиард дескрипторов.
- Wiki-LASER Dataset: набор данных, который мы создали на основе статей из Википедии. Мы разбили каждый текст на предложения и закодировали их с использованием метода LASER (Language Agnostic Sentence Representations) [11].
- Wiki-LASER Dataset Small: уменьшенная версия датасета Wiki-LASER Dataset.
- Open Images Dataset (OID) [12]: набор данных, который содержит более 9 миллионов изображений с более чем 30 миллионами ограничительных рамок. Набор разработан задач обнаружения объектов, классификации и обнаружения визуальных отношений. Мы используем сеть ResNet18 для получения векторных представлений изображений, которые затем были нормализованы и преобразованы в единый вид.
- Open Images Dataset small (OID small): уменьшенная версия датасета OID.

Табл. 1. Сводная статистика наборов данных. BPC означает "байт на компонент"
Table 1. Statistic of the datasets. BPC means "byte per component"

Название	Размерность	Vec cnt	BPC	Query vec cnt	Learn vec cnt	Orig size
siftsmall	128	10,000	4	100	25,000	5,120,000
sift1m	128	1,000,000	4	10000	100,000	512,000,000
gist	960	1,000,000	4	1000	500,000	3,840,000,000
sift1b	128	1,000,000,000	1	10000	100,000,000	128,000,000,000
OID small	512	1,012,239	4	1012	101,223	2,073,065,472
OID	512	8,390,600	4	83906	843,480	17,183,948,800
Wiki-LASER Dataset small	1024	576,940	4	576	57,694	2,363,146,240
Wiki-LASER Dataset	1024	87,817,400	4	87783	1,317,261	359,700,070,400

6. Подготовка к тестированию

Для тестирования мы выделили разные флаги и компоненты и составляем из них разные комбинации. Результаты перспективных комбинаций мы свели в таблицы для каждого конкретного набора данных (см. Табл. 3-10 в конце статьи). Таблицы разделены на две части. Первая часть (табл. 3, 5, 7, 9) состоит из результатов, вторая (табл. 4, 6, 8, 10) описывает

некоторые параметры комбинации. Строки могут быть сопоставлены по столбцу ID. Мы используем реализации различных индексов из библиотеки Faiss [13]. Тестовые случаи делятся на две группы: первая группа – это случаи, когда используется автокодировщик в сочетании с индексом и/или квантования из Faiss, а вторая группа – это случаи, когда используются только компоненты Faiss. В качестве индексов мы используем Flat, IVFPQ, HNSW и NSG (столбец index). В качестве автокодировщиков (столбец enc) мы используем все описанные варианты: Vanilla, DCEC, HyperAE, NRAE.

Кроме того, экспериментируем с нормализацией входных и выходных векторов (столбцы norm inp vect, norm embs и norm out vect), скрытой размерностью автокодировщиков (столбец hidden dim) и установкой размерности скрытого слоя, равной размерности входа автокодировщика (столбец set l hidden). Эти параметры представлены в виде столбцов во второй таблице. Если столбец не указан, это означает, что он имеет одно и то же значение в каждой ячейке. По умолчанию отсутствие значения означает то, что действие не было выполнено. Также указываем константы, такие как размер пакета (batch size), равный 8, количество эпох обучения, равное 5, параметр nprobe, равный 20.

Мы вычисляем несколько метрик, таких как различные варианты полноты (recall), которые мы кодируем как $n - R@k$, где n – это количество истинных ближайших соседей, а k – это топ релевантных векторов, возвращенных индексом. Мы также вычисляем коэффициент сжатия (cr), экономию пространства (ss), размер индекса на диске (index size), время поиска одного вектора (o-v-s), общее время поиска (s) и также время обучения и индексации (t_i). Все эксперименты были проведены в два этапа. Сначала на небольшом подмножестве датасета мы ищем лучшие комбинации параметров, затем запускаем отобранные варианты на всем наборе данных.

7. Выбор размерности автокодировщика

Важным моментом является выбор размерности скрытого слоя, который эквивалентен коэффициенту сжатия. Нет четкого способа определить оптимальную размерность скрытого слоя, которая бы давала максимальное сжатие при минимальной потере качества.

В ходе предварительного исследования было обнаружено, что существует граница сжатия, пересечение которой вызывает стремительное снижение качества, независимо от типа автокодировщика. Это характерно для алгоритмов поиска, показывающих определенный уровень качества до достижения этой границы, например, HNSW на рис. 1. Изучение этого явления привело к выводу, что эта граница является внутренней размерностью, которая определяется как размерность многообразия, к которому принадлежит вложенное пространство [14]. По альтернативному определению, это можно рассматривать как минимальное число измерений, необходимое для представления данных.

Мы провели эксперименты на синтетически сгенерированных данных с известной внутренней размерностью. Для генерации мы использовали пакет scikit-dimension [15], в котором воспользовались функцией генерации данных в режиме «Nonlinear manifold». Затем обучили автокодировщик Vanilla на сгенерированном наборе и вычислили функцию потерь на тестовом наборе, отобранном из сгенерированных данных, при этом размер скрытого слоя варьируется от исходной размерности до 10. В качестве внутренних размерностей тестируем значения 8, 32 и 64. На рис. 2 видно, что значение функции потерь начинает расти после того, как размер скрытого слоя становится меньше внутренней размерности. Мы повторили этот эксперимент с реальными данными, в частности с наборами данных SIFT1M и Open Images Dataset. Мы обнаружили, что потери автокодировщика начинают увеличиваться приблизительно в том же диапазоне, в котором ухудшается качество поиска.

Поскольку экспериментально показано существование оптимальной размерность для сжатия, мы исследуем все алгоритмы оценки внутренней размерности, реализованные в пакете scikit-dimension. Суть эксперимента очень проста – подать на вход каждого алгоритма из

программного пакета датасет, для которого мы имеем гипотезу о внутренней размерности, а затем сравнить результат выдачи алгоритма с нашей гипотезой.

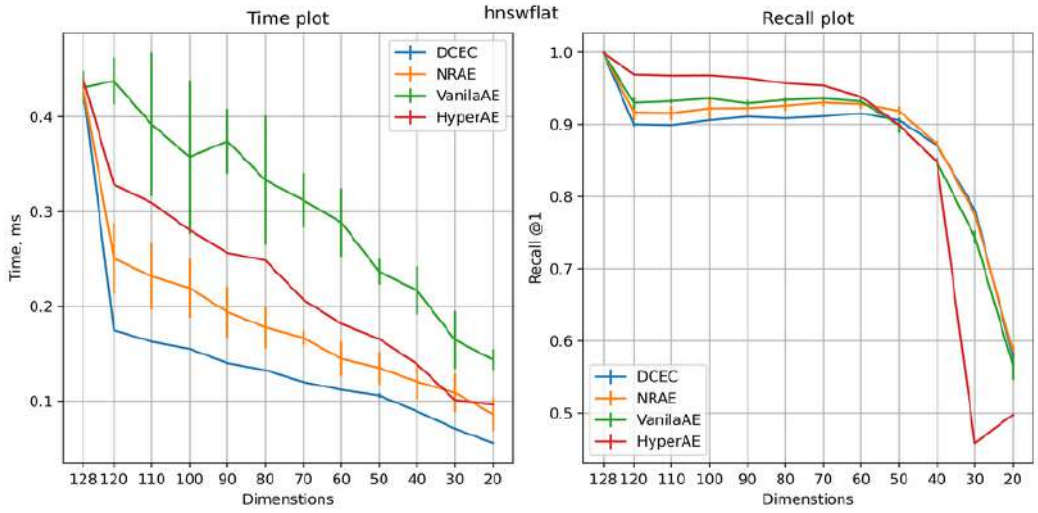


Рис. 1. Тест HNSW с автокодировщиками.
Левый график отображает время вывода. Правый график отображает показатель полноты (recall)
Fig. 1. Test of HNSW with autoencoder.
Left plot shows the inference time, right plot shows performance in terms of recall

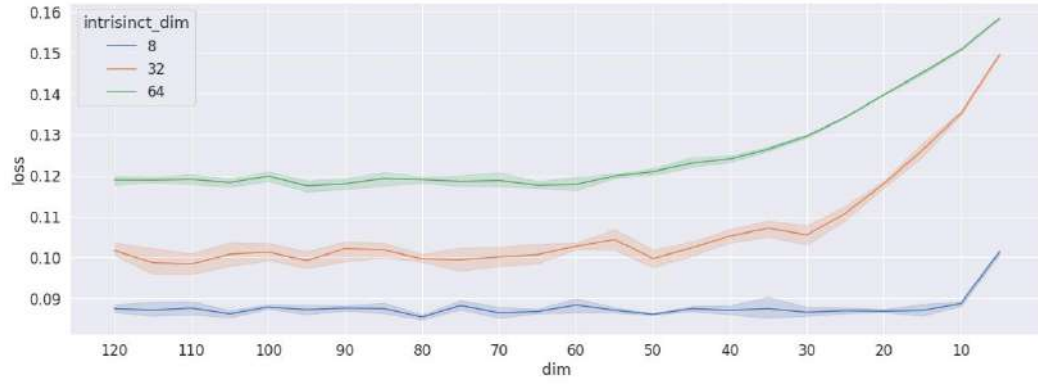


Рис. 2. Тестирование функции потерь автокодировщика Vanilla при различных внутренних размерностях синтетических наборов данных
Fig. 2. Results of the Vanilla autoencoder testing with synthetic datasets with various intrinsic dimensions

Исходя из эмпирического исследования, мы считаем, что внутренняя размерность SIFT1M составляет около 60. Тем не менее все алгоритмы показали значения существенно далекие от 60, как показано в табл. 2 в столбце «Размерность». Кроме того, некоторые алгоритмы требуют значительное время на выполнение, как видно в столбце «Время выполнения».

Кроме того, экспериментально не удалось подтвердить, что полученное нами число является внутренней размерностью. Возможно, это число им не является, а возможно алгоритмы не справились с задачей, учитывая какой большой разброс оценок мы получили. Стоит отметить, что мы не калибровали параметры алгоритмов и запускали их со стандартными значениями. Мы оставляем вопросы автоматического выбора оптимальной размерности, а также природу этой оптимальности для будущих исследований.

Табл. 2. Результаты теста алгоритма оценки внутренней размерности на наборе данных SIFT1M
Table 2. Results of the intrinsic dimension estimation algorithms on SIFT1M dataset

Имя	Размерность	Время выполнения, с
ESS	25.4422	9324.32
DANCo	nan	265.773
CorrInt	9.99273	420.92
FisherS	3.11696	304.551
KNN	3	695.634
lPCA	10	0.88
MADA	21.9273	1956.41
MiND_ML	1	255.691
MLE	0	256.54
MOM	19.5008	253.178
TLE	19.1451	288.2
TwoNN	10.4751	831.291

8. Влияние функции потерь на качество методов поиска

Известно, что функция потерь для обучения во многом определяет свойства и поведение нейронной сети. Для достижения наилучшего качества поиска необходимо, чтобы векторы в сжатом пространстве находились в таких же отношениях, как и в исходном. Можно попытаться сформулировать это требование в виде функции потерь. В работе [16] предлагается функция потерь, которая заставляет автокодировщик напрямую воссоздавать отношения расстояний, сравнивая матрицы расстояний в исходном и сжатом пространствах.

$$L_{distance}(\phi;\, X, Z) \;=\; \sum_{(i,j)\in\, G_X} \left| d_X(x_i, x_j) - \gamma\; d_Z(z_i, z_j) \right|^2$$

Мы провели тестирование этой функции потерь на двух датасетах SIFT1M и GIST. Эмпирически получив для них оптимальную размерность, мы обучили автокодировщики со среднеквадратичной ошибкой (MSE) и вышепредставленной функцией потерь. В качестве методов поиска мы использовали HNSW и IVFPQ. Результаты представлены на рис. 3. Из него видно, что новая функция потерь обеспечивает существенный прирост по качеству для обоих датасетов. Результаты показывают, что выбор функции потерь может существенно влиять на конечный результат. Это тестирование было проведено после получения главных результатов, поэтому дальнейшее изучение дизайна функции потерь мы оставим на будущее.

9. Основные результаты

Как видно из результатов, методы поиска на основе графов NSG и HNSW показывают практически идеальные результаты по всем вариантам полноты для всех наборов данных (табл. 4, 6, 8). Недостатками этих методов является то, что они используют векторы без изменений и добавляют избыточные затраты памяти, что можно видеть по метрикам cs и ss. Они также требуют меньше всего времени при непосредственном использовании и имеют среднюю скорость индексирования.

Ни одна из комбинаций, в которых присутствуют автокодировщики, не показывает результатов, которые полностью превосходят чистую систему Faiss. Можно сделать вывод, что сжатие автокодировщика сильно искажает векторное пространство. Это выражается в нарушении локальных отношений: векторы, являющиеся соседями в исходном пространстве,

могут оказаться далеко друг от друга в сжатом пространстве. При сравнении только автокодировщиков заметно, что один автокодировщик работает лучше других в зависимости от набора данных. Соответственно, можно выделить, что HyperAE работает лучше в большинстве случаев. В отличие от него, NRAE не показал приемлемых результатов ни в одном варианте.

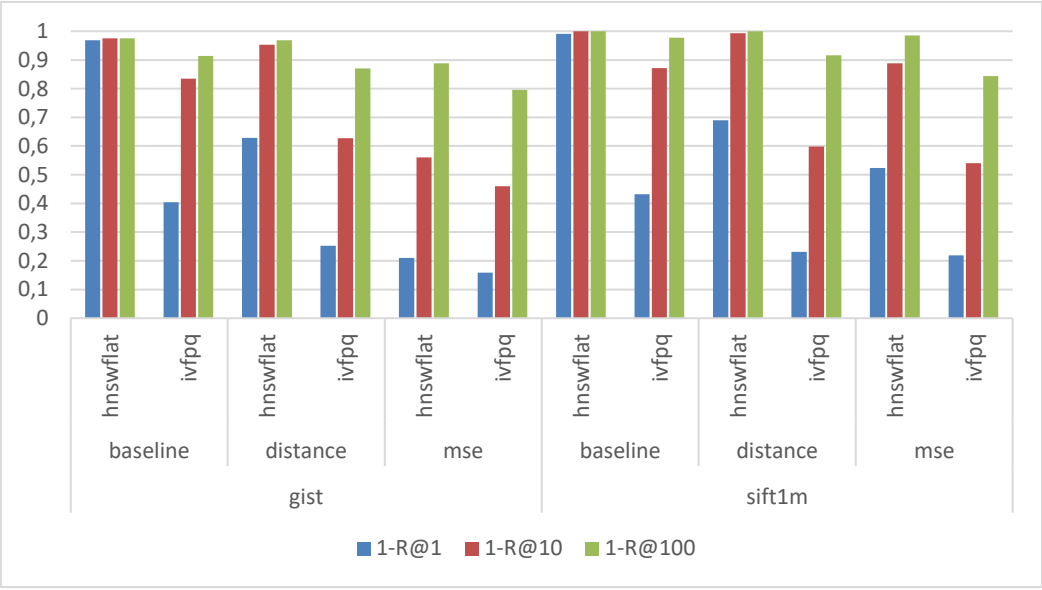


Рис. 3. Сравнение качества методов ANN в зависимости от разных функций потерь (ФП). Результаты без сжатия указаны как baseline

Fig. 3. Results of the influence testing of various losses on ANN performance. Baseline is compressless result

Для наборов данных Wiki и Open Images комбинации показывают идентичные результаты по метрике $1 - R@k$, поэтому возможно сократить сравнение до $10 - R@k$. Можно отметить, что комбинация Vanilla AE с индексом Flat сравнима с HNSW, но при этом обеспечивает двукратное уменьшение размера. Однако скорость работы комбинации с использованием Vanilla AE гораздо медленнее.

10. Заключение

В данной статье мы исследовали применимость нейронных автокодировщиков в качестве компонента сжатия векторов в конвейере поиска соседей. Рассмотрели автокодировщик Vanilla и его варианты, которые лучше сохраняют геометрию пространства при сжатии, с различными методами индексирования.

Мы обнаружили, что автокодировщики не смогли сжать исходное векторное пространство таким образом, чтобы полученное пространство точно сохраняло локальные отношения между векторами, хотя некоторые из них делают это лучше других. В ходе работы также отметили связь между нижней границей сжатия автокодировщщиков и оценкой внутренней размерности. Также показали, что возрастание функции потерь автокодировщика может служить хорошим сигналом о достижении внутренней размерности, в то время как ни один из протестированных алгоритмов ее оценки не может обработать такую высокую размерность.

Эксперименты показали, что использование кодировщиков в некоторых случаях позволяет сохранить вектор меньшей размерности с уверенностью, что выбранная метрика качества

сохраняет тот же уровень. Таким образом, это может дать преимущество, например, в задаче хранения сжатых векторов.

В дальнейшей работе мы будем продолжать искать автокодировщик, который минимально искажает пространство при сжатии данных настолько, насколько это возможно. Как видно из экспериментов, прежде всего необходимо уделить внимание функции потери. У нас есть гипотеза, согласно которой точки из плотных областей интерферируют между собой, что приводит к ошибкам. Если это так, то стоит рассмотреть добавление регуляризатора к функции потерь. Мы также попытаемся разработать способы автоматического выбора размерности автокодировщика.

Табл. 3. Основные результаты для SIFT1M
Table 3. Main results for SIFT1M

id	1-R@100	1-R@10	1-R@1	10-R@100	10-R@10	100-R@100	index_size	cr	ss	o-v s	s	t_i
0	0.56	0.33	0.16	0.38	0.15	0.18	2.56E+08	2	0.5	0.0282	58.2	80.57
1	0.56	0.33	0.16	0.38	0.15	0.17	5.28E+08	0.97	-0.03	0.0058	0.67	98.56
2	1	0.93	0.57	0.98	0.62	0.65	2.56E+08	2	0.5	0.0259	57.75	289.5
3	1	0.94	0.58	0.98	0.63	0.67	2.56E+08	2	0.5	0.0257	57.88	299.66
4	1	0.93	0.56	0.97	0.61	0.65	2.56E+08	2	0.5	0.0283	57.87	108.23
5	0.98	0.82	0.41	0.92	0.49	0.56	24082612	21.26	0.95	0.0192	12.76	131.78
6	1	0.92	0.56	0.97	0.61	0.65	72133300	7.1	0.86	0.0257	17.92	190.7
7	1	0.93	0.56	0.97	0.61	0.65	2.56E+08	2	0.5	0.0283	57.87	108.23
8	1	0.93	0.56	0.97	0.61	0.65	2.56E+08	2	0.5	0.0283	57.87	108.23
9	0.99	0.87	0.44	0.96	0.54	0.62	24164532	21.19	0.95	0.0097	4.65	28.87
10	1	1	0.81	1	0.87	0.89	72264372	7.09	0.86	0.0099	4.75	99.95
11	1	1	0.81	1	0.85	0.85	72663732	7.05	0.86	0.0069	1.79	108.84
12	1	1	0.9	1	0.86	0.66	7.84E+08	0.65	-0.53	0.007	0.74	31.54
13	1	1	0.99	1	0.99	0.94	5.96E+08	0.86	-0.16	0.007	1.04	114.66

Табл. 4. Вариации параметров для SIFT1M. Скрытый слой равен 64
Table 4. Parameter variations for SIFT1M. Hidden dim is 64

id	index	norm embs	encoder
0	Flat	false	dcec
1	HNSW32	false	dcec
2	Flat	false	hyperae
3	Flat	true	hyperae
4	Flat	false	vanae
5	IVF64,PQ16	false	vanae
6	IVF256,PQ64	false	vanae
7	Flat	false	vanae
8	Flat	false	vanae
9	IVF64,PQ16	false	-
10	IVF256,PQ64	false	-
11	IVF1024,PQ64	false	-
12	HNSW32	false	-
13	NSG32	false	-

Табл. 5. Основные результаты для GIST
Table 5. Main results for GIST

id	1-R@100	1-R@10	1-R@1	10-R@100	10-R@10	100-R@100	index_size	cr	ss	o-v s	s	t_i
0	0.5	0.23	0.09	0.31	0.09	0.12	1.3E+08	29.43	0.97	0.0167	1.58	860.27
1	0	0	0	0	0	0	1.23E+09	3.12	0.68	0.0061	0.58	2,111.7
2	0	0	0	0	0	0	1.23E+09	3.12	0.68	0.0063	0.59	2,100.57
3	0	0	0	0	0	0	1.23E+09	3.12	0.68	0.0061	0.58	2,047.03
4	0	0	0	0	0	0	1.23E+09	3.12	0.68	0.0072	0.57	1,969.21
5	0.07	0.04	0.03	0.03	0.01	0.02	2.19E+09	1.75	0.43	0.006	0.59	2,189.44
6	0.06	0.04	0.02	0.03	0.01	0.02	1.3E+08	29.43	0.97	0.0179	1.68	2,086.03
7	0.06	0.04	0.03	0.03	0.01	0.02	2.5E+08	15.33	0.93	0.029	3.08	2,112.66
8	0.5	0.23	0.09	0.31	0.09	0.12	1.3E+08	29.43	0.97	0.0167	1.58	860.27
9	0.5	0.23	0.09	0.31	0.09	0.12	1.3E+08	29.43	0.97	0.0167	1.58	860.27
10	0.5	0.23	0.09	0.31	0.09	0.12	1.3E+08	29.43	0.97	0.0167	1.58	860.27
11	0.98	0.85	0.42	0.94	0.52	0.59	1.3E+08	29.55	0.97	0.0155	1.47	190
12	1	0.98	0.63	1	0.69	0.69	2.53E+08	15.18	0.93	0.0114	1.15	454.85
13	1	0.99	0.67	1	0.6	0.43	4.11E+09	0.93	-0.07	0.0067	0.68	390.93
14	1	0.98	0.63	1	0.69	0.69	2.53E+08	15.18	0.93	0.0114	1.15	454.85

Табл. 6. Вариации параметров для GIST
Table 6. Parameter variations for GIST

id	index	norm embs	enc	hidden dim	norm inp vect	norm out vect	set l hidden
0	IVF1024,PQ120x8	false	dcec	480	false	false	true
1	HNSW32	false	hyperae	240	false	false	false
2	HNSW32	false	hyperae	240	false	false	true
3	HNSW32	true	hyperae	240	true	true	false
4	HNSW32	true	hyperae	240	true	true	true
5	HNSW32	false	vanae	480	false	false	true
6	IVF1024,PQ120x8	false	vanae	480	false	false	true
7	IVF1024,PQ240x8	false	vanae	480	false	false	true
8	IVF1024,PQ120x8	false	dcec	480	false	false	true
9	IVF1024,PQ120x8	false	dcec	480	false	false	true
10	IVF1024,PQ120x8	false	dcec	480	false	false	true
11	IVF256,PQ120x8	false	-	-	-	-	-
12	IVF1024,PQ240x8	false	-	-	-	-	-
13	HNSW32	false	-	-	-	-	-
14	IVF1024,PQ240x8	false	-	-	-	-	-

Табл. 7. Основные результаты для Open Image Dataset
Table 7. Main results for Open Image Dataset

id	1-R@100	1-R@10	1-R@1	10-R@100	10-R@10	100-R@100	index_size	cr	ss	o-v s	s	t_i
0	0.99	0.99	0.98	0.83	0.54	0.45	1.09E+10	1.57	0.36	0.0147	4.79	4,096.41
1	0.99	0.99	0.98	0.83	0.54	0.45	1.09E+10	1.57	0.36	0.0157	4.83	4,111.58
2	1	1	1	0.8	0.49	0.47	6.1E+08	28.18	0.96	0.0163	75.98	3,462.03
3	1	1	1	0.8	0.49	0.47	6.1E+08	28.18	0.96	0.0163	75.8	3,484.63
4	1	1	1	0.81	0.51	0.48	6.12E+08	28.07	0.96	0.0163	76.52	4,180.76
5	1	1	1	0.81	0.5	0.48	6.12E+08	28.07	0.96	0.0165	76.81	4,178.86
6	0.99	0.99	0.98	0.83	0.54	0.45	1.09E+10	1.57	0.36	0.0147	4.79	4,096.41
7	0.99	0.99	0.98	0.83	0.54	0.45	1.09E+10	1.57	0.36	0.0157	4.83	4,111.58
8	1	0.99	0.92	0.99	0.79	0.58	1.96E+10	0.88	-0.14	0.0171	8.2	1,731.1
9	1	1	1	0.95	0.58	0.57	6.17E+08	27.87	0.96	0.009	20.54	1,487.1
10	1	0.99	0.92	0.99	0.79	0.58	1.96E+10	0.88	-0.14	0.0171	8.2	1,731.1

Табл. 8. Вариации параметров для Open Image Dataset
Table 8. Parameter variations for Open Image Dataset

id	index	enc	hidden dim	set l hidden
0	HNSW32	hyperae	256	false
1	HNSW32	hyperae	256	true
2	IVF4096,PQ64	hyperae	128	false
3	IVF4096,PQ64	hyperae	128	true
4	IVF4096,PQ64	hyperae	256	false
5	IVF4096,PQ64	hyperae	256	true
6	HNSW32	hyperae	256	false
7	HNSW32	hyperae	256	true
8	HNSW32	-	-	-
9	IVF4096,PQ64	-	-	-
10	HNSW32	-	-	-

Табл. 9. Основные результаты для Wiki
Table 9. Main results for Wiki

id	1-R@100	1-R@10	1-R@1	10-R@100	10-R@10	100-R@100	index_size	cr	ss	o-v s	s	t_i
0	0.99	0.98	0.97	0.44	0.34	0.24	6.34E+09	56.73	0.98	0.0412	268.86	22,424.1
1	0.99	0.98	0.97	0.33	0.3	0.19	6.34E+09	56.73	0.98	0.0427	263.97	21,576.6
2	0.99	0.98	0.97	0.44	0.34	0.24	6.34E+09	56.73	0.98	0.0412	268.86	22,424.1
3	0.99	0.98	0.97	0.87	0.58	0.52	6.39E+09	56.28	0.98	0.024	103.61	62,943.41

Табл. 10. Вариации параметров для Wiki. Скрытый слой равен 256
Table 10. Parameter variations for Wiki. Hidden dim is 256

id	index	enc	set l hidden
0	IVF16384,PQ64	dcec	true
1	IVF16384,PQ64	dcec	false

2	IVF16384,PQ64	dcec	true
3	IVF16384,PQ64	-	-

Список литературы / References

[1]. J. L. Bentley. Multidimensional binary search trees used for associative searching. Commun. ACM 18, 509–517 (1975).
[2]. S. M. Omohundro. Five balltree construction algorithms (2009).
[3]. H. J´egou, M. Douze, C. Schmid. Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis and Machine Intelligence 33, 117–128 (2011).
[4]. P. Alexander, Y. Malkov, L. Andrey, V. Krylov. Approximate nearest neighbor search small world approach (2011).
[5]. Y. A. Malkov, D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchicalnavigable small world graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 42, 824–836 (2016).
[6]. H. Zhang, H. Shen, Y. Qiu, Y. Jiang, S. Wang, S. Xu, Y. Xiao, B. Long, W. Yang. Joint learning of deep retrieval model and product quantization based embedding index. Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (2021).
[7]. D. E. Rumelhart, G. E. Hinton, R. J. Williams. Learning internal representations by error propagation (1986).
[8]. Y. Lee, H. Kwon, F. C. Park. Neighborhood reconstructing autoencoders. In Neural Information Processing Systems (2021).
[9]. X. Guo, X. Liu, E. Zhu, J. Yin. Deep clustering with convolutional autoencoders. In International Conference on Neural Information Processing (2017).
[10]. L. Mirvakhabova, E. Frolov, V. Khrulkov, I. Oseledets, A. Tuzhilin. Performance of hyperbolic geometry models on top-n recommendation tasks. Proceedings of the 14th ACM Conference on Recommender Systems (2020).
[11]. H. Gong, H. Schwenk. Multimodal and multilingual embeddings for large-scale speech mining. In Neural Information Processing Systems (2021).
[12]. A. Kuznetsova, H. Rom, N. G. Alldrin, J. R. R. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, V. Ferrari. The open images dataset v4. International Journal of Computer Vision 128, 1956–1981 (2018).
[13]. J. Johnson, M. Douze, H. J´egou. Billion-scale similarity search with gpus. IEEE Transactions on Big Data 7, 535–547 (2017).
[14]. V. Erba, M. Gherardi, P. Rotondo. Intrinsic dimension estimation for locally undersampled data. Scientific Reports 9 (2019).
[15]. J. Bac, E. M. Mirkes, A. N. Gorban, I. Y. Tyukin, A. Y. Zinovyev. Scikit-dimension: A python package for intrinsic dimension estimation. Entropy 23 (2021).
[16]. N. Chen, P. van der Smagt, B. Cseke. Local distance preserving auto-encoders using continuous k-nearest neighbours graphs. ArXiv abs/2206.05909 (2022), https://api.semanticscholar.org/CorpusID:249626370.

Информация об авторах / Information about authors

Игорь Олегович БУЯНОВ – аспирант ФИЦ ИУ РАН, старший разработчик в MTS AI. Сфера научных интересов: обработка естественного языка, анализ пространств эмбеддингов, вычислительная психология.

Igor Olegovich BUYANOV – postgraduate student at FRC CSC RAS, senior developer at MTS AI. Research interests: natural language processing, embedding space analysis, computational psychology.

Василий Владимирович ЯДРИНЦЕВ – младший научный сотрудник в отделе 73 ФИЦ ИУ РАН: интеллектуальных технологий и систем. Сфера научных интересов: индексация векторных данных, обработка естественного языка.

Vasiliy Vladimirovich YADRINSEV – junior researcher at department 73 intellectual technologies and systems of FRC CSC RAS. Research interests: data vector indexing, natural language processing.

Илья Владимирович СОЧЕНКОВ – кандидат физико-математически наук, ведущий научный сотрудник ФИЦ ИУ РАН, ведущий научный сотрудник ИСП РАН, ведущий эксперт-консультант Университета Иннополис. Сфера научных интересов: обработка естественного языка, детекция плагиата.

Ilia Vladimirovich SOCHENKOV – Cand. Sci. (Phys.-Math.), lead researcher at FRC CSC RAS, lead researcher at ISP RAS, lead expert at Innopolis University. Research interests: natural language processing, plagiarism detection.

DOI: 10.15514/ISPRAS-2024-36(1)-2

Исследование вопросов учёта нагрузок в программно-конфигурируемых сетях

[1] *И. Б. Бурдонов,* ORCID: 0000-0001-9539-7853 <*igor@ispras.ru*>
[1,2] *Н. В. Евтушенко,* ORCID: 0000-0002-4006-1161 <*evtushenko@ispras.ru*>
[1] *А. С. Косачев,* ORCID: 0000-0001-5316-3813 <*kos@ispras.ru*>

[1] *Институт системного программирования РАН им. В.П. Иванникова, 109004, Россия, г. Москва, ул. А. Солженицына, д. 25.*
[2] *Национальный исследовательский университет «Высшая школа экономики», 101000, Россия, г. Москва, ул. Мясницкая, д. 20.*

Аннотация. Описывается исследование возможности реализации виртуальных сетей с учётом различных параметров и их корректировки в программно-конфигурируемых структурах, моделируемых взвешенным графом плоскости данных. В работе исследуются параметры двух типов: «ресурс» и «стоимость». Для параметра типа «ресурс» с ребром ассоциируется его «ёмкость», и число путей, проходящих через ребро, не должно превышать ёмкость ребра. Для параметра типа «стоимость» с ребром ассоциируется его «цена», «цена» пути есть сумма «цен» его рёбер, и ставится задача минимизации суммарной «цены» всех путей. Для реализации на взвешенном графе плоскости данных предложены алгоритм корректировки виртуальной сети с учётом параметров типа «ресурс» и два алгоритма построения виртуальной сети с учётом параметров типа «стоимость». В последнем случае один алгоритм строит для каждого хоста один путь из него в один хост из заданного подмножества целевых хостов; другой алгоритм строит для каждого хоста множество путей: по одному пути в один хост из каждого множества из семейства множеств целевых хостов.

Ключевые слова: распределённые и параллельные вычисления; программно-конфигурируемые сети; маршрутизация пакетов; взвешенный граф.

Для цитирования: Бурдонов И.Б., Евтушенко Н.В., Косачев А.С. Исследование вопросов учёта нагрузок в программно-конфигурируемых сетях. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 23–34. DOI: 10.15514/ISPRAS–2024–36(1)–2.

Studying Load Issues in Software-Defined Networks

[1] *I.B. Burdonov* ORCID: 0000-0001-9539-7853 <*igor@ispras.ru*>
[1,2] *N.V. Yevtushenko* ORCID: 0000-0002-4006-1161 <*evtushenko@ispras.ru*>
[1] *A.S. Kossatchev* ORCID: 0000-0001-5316-3813 <*kos@ispras.ru*>

[1] *Institute for System Programming of the Russian Academy of Sciences,*
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.
[2] *National Research University Higher School of Economics,*
20, Myasnitskaya st., Moscow, 101000, Russia.

Abstract. The purpose of the work is to study the possibility of implementing virtual networks taking into account various parameters and their adjustments in software-configurable structures modeled by a weighted data plane graph. The work examines parameters of "resource" and "cost" types. For a resource type parameter, an edge is augmented with its "capacity," and the number of paths passing through the edge must not exceed the edge's capacity. For a parameter of the "cost" type, the path weight is the sum of the weights of the edges and the task of minimizing the weight of the path is set. For implementing a virtual network on a weighted graph, an algorithm for adjusting a virtual network taking into account parameters of the "resource" type and two algorithms for constructing a virtual network taking into account parameters of the "cost" type are proposed. In the latter case, one algorithm builds one path from each host to one host of the given subset of target hosts; another algorithm builds a set of paths for each host: one path to one host from each set of a family of sets of target hosts.

Keywords: distributed and parallel computations; software-defined networks (SDN); packet routing; weighted graph.

For citation: Burdonov I.B., Yevtushenko N.V., Kossatchev A.S. Studying load issues in the software-defined networks. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 23-34 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-2.

1. Введение

В программно-конфигурируемых сетях (SDN) с разделёнными плоскостями данных и управления (см., например, [1-5]) пакеты между хостами пересылаются на плоскости данных через промежуточные коммутаторы, настройка которых осуществляется SDN-контроллерами. Настройка коммутаторов определяет множество путей, по которым между хостами пересылаются пакеты с заданными идентификаторами. Набор реализуемых путей моделирует виртуальную сеть на плоскости данных, которая в работах [4-6] рассматривается как невзвешенный граф связей. При исследовании вопросов надёжности, безопасности и оптимизации нагрузки на хосты и коммутаторы необходимо рассматривать дополнительные параметры сети, которые игнорируются в предыдущих работах авторов [4-7]. Для учёта параметров типа «ресурс» и «стоимость» можно использовать взвешенный граф связей для плоскости данных сети, тем самым налагаются ограничения на маршрутизацию пакетов.

Для параметра типа «ресурс» с ребром ассоциируется его «ёмкость», и число путей, реализующих виртуальную сеть и проходящих через ребро, не должно превышать ёмкости ребра. В разделе 3 предлагается алгоритм корректировки виртуальной сети с учётом параметров типа «ресурс» при реализации на взвешенном графе плоскости данных. Предлагаемый алгоритм достаточно просто расширяется на случай реализации нескольких виртуальных сетей.

Для параметра типа «стоимость» ребро имеет «цену», и «цена» пути есть сумма «цен» его рёбер. Ставится задача минимизации суммарной «цены» путей, реализующих виртуальную сеть. В разделе 4 предлагаются два алгоритма построения виртуальной сети с учётом параметров типа «стоимость» при реализации на взвешенном графе плоскости данных. Один алгоритм строит замкнутое по дугам множество путей без зацикливания и дублирования и такое, что из каждого хоста ведёт ровно один путь в один из хостов заданного подмножества

целевых хостов. Другой алгоритм строит замкнутое по дугам множество путей без зацикливания и дублирования и такое, что из каждого хоста ведёт ровно один путь в хост каждого множества из заданного семейства множеств целевых хостов.

2. Определения и обозначения

Графом физических связей (далее просто графом связей) будем называть связный неориентированный граф $G = \{V, E\}$ без кратных рёбер и петель, где V – множество коммутаторов и хостов, $E \subseteq V \times V$ – множество рёбер, моделирующих физические связи между коммутаторами и между коммутаторами и хостами. Поскольку ребро, соединяющее вершины a и b, неориентированное и нет кратных рёбер, его можно обозначать как ab, так и ba. Поскольку нет петель, рёбер вида aa в E нет. Поскольку нет кратных рёбер, путь как последовательность смежных рёбер однозначно задаётся последовательностью вершин $a_1 \ldots a_n$, через которые он проходит. Если путь проходит по ребру ab из a в b, то будем говорить, что он проходит дугу ab. Если x и y хосты, то путь из x в y, в котором все остальные вершины коммутаторы, будем называть *полным* и обозначать как xy-путь. Путь, в котором вершины (дуги) не повторяются, называется *вершинно-простым* (*рёберно-простым*[1]). Вершины графа будем обозначать строчными буквами $a, b, c, \ldots, x, y, z$, пути – жирными строчными буквами $\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{r}, \ldots$, а множества путей – прописными буквами: $P, Q, R, \ldots$ На рисунках коммутаторам соответствуют белые кружки, а хостам – чёрные кружки; если безразлично, является вершина графа коммутатором или хостом, рисуются серые кружки.

Мы будем предполагать, что каждый хост x соединён в точности с одним коммутатором [4], и, как показано в [4-5], в этом случае достаточно рассматривать графы, в которых терминальные вершины суть хосты. Множества хостов и коммутаторов обозначается через H и S, соответственно; $H \cup S = V$, $H \cap S = \varnothing$, и под виртуальной сетью понимается конечное множество P полных путей между парами различных хостов, которое контроллер должен реализовать на плоскости данных, задавая нужные правила в коммутаторах.

В общем случае правило коммутатора b имеет вид σabc, где a и c соседи b, а σ вектор значений параметров заголовка пакета, которые используются в правилах. Такое правило означает, что коммутатор b, получив пакет с вектором σ от соседа a, пересылает его соседу c. В настоящей работе предполагается, что коммутатор не меняет σ. Тем самым, для вектора σ порождаются полные пути вида $a_1 \ldots a_n$, где для $i = 2..n - 1$ в коммутаторе a_i есть правило $\sigma a_{i-1} a_i a_{i+1}$, хост a_1 соединён с коммутатором a_2 и хост a_n соединён с коммутатором a_{n-1}. Если в коммутаторе есть два правила σabc и $\sigma abc`$, где $c \neq c`$, говорят, что пакет *клонируется*, то есть пересылается обоим соседям c и $c`$.

Вектора σ и $\sigma`$ считаются эквивалентными, если они определяют одни и те же пути, то есть для каждого правила $\sigma abc$ есть правило $\sigma` abc$ и, наоборот, для каждого правила $\sigma` abc$ есть правило σabc. В модели для множества всех векторов задаётся его разбиение на непересекающиеся подмножества, каждый из которых является подмножеством класса эквивалентных векторов. Каждому такому подмножеству ставится во взаимно-однозначное соответствие идентификатор пакета. Таким образом, каждому идентификатору соответствует множество путей, определяемых правилами коммутаторов, хотя разным идентификаторам может соответствовать одно и то же множество путей, если эти идентификаторы определяют подмножества векторов из одного и того же класса эквивалентных векторов. Если вектору σ соответствует идентификатор d, то вместо правила σabc рассматривается правило $dabc$. В [6]

[1] В литературе часто даётся другое определение: "путь рёберно-простой, если он не проходит дважды по одному ребру" (а не дуге), то есть путь, проходящий дважды по ребру, но в противоположных направлениях, не считается рёберно-простым. Однако для нашего определения «замкнутости по дугам» (см. ниже) удобнее использовать наше определение рёберно-простого пути.

показано, каким образом результаты по построению виртуальной сети можно распространить на реализацию конечного множества виртуальных сетей.

Заданное множество P полных путей однозначно определяет необходимый набор правил коммутаторов, порождающий все пути из P, однако при этом могут порождаться и новые пути, не принадлежащие P [4,5], которые не обязательно являются рёберно-простыми, то есть некоторые пакеты могут находиться в сети сколь угодно долго. В работах показано, что для точной реализации на плоскости данных множество путей должно быть замкнутым по дугам.

Множество рёберно-простых полных путей, то есть путей между хостами на плоскости данных через коммутаторы, называется *замкнутым по дугам*, если для любых двух путей множества с различными начальными и конечными хостами выполняется следующее: если после слияния на некоторой дуге эти пути разделяются (после этой же или другой дуги), то все четыре пути должны принадлежать исходному множеству [4,5]. Если множество P полных рёберно-простых путей замкнуто по дугам, то на плоскости данных дополнительных путей, а следовательно (для конечного P) и зацикливаний, не появляется; в [3] предложен алгоритм проверки, является ли заданное множество путей замкнутым по дугам.

Для учёта различных параметров при реализации виртуальных сетей и их корректировки в программно-конфигурируемых структурах плоскость данных моделируется взвешенным графом. Таким образом, мы полагаем, что задан граф связей, в котором каждому ребру приписано натуральное число («вес» ребра), то есть граф является взвешенным. В работе исследуются параметры двух типов: «ресурс» и «стоимость». Для параметра типа «ресурс» под весом ребра понимается его «ёмкость», и число путей, проходящих через ребро, не должно превышать ёмкость ребра. Для параметра типа «стоимость» под весом ребра понимается «цена» прохождения пути через ребро, «цена» пути — это сумма «цен» его рёбер, и ставится задача минимизации суммарной «цены» путей, реализующих виртуальную сеть.

3. Алгоритм корректировки виртуальной сети с учётом параметров типа «ресурс» при реализации на взвешенном графе плоскости данных

В этом разделе мы полагаем, что во взвешенном графе связей каждому ребру приписан некоторый вес, соответствующий параметру типа «ресурс», значение которого равно натуральному числу. Превышение допустимого числа путей, проходящих по ребру, может привести к потере пакетов или задержке при их отправке. Замкнутое по дугам множество путей P, то есть путей, не порождающих циклы в графе, называется *ресурсно-допустимым*, если число путей из P, проходящих по каждому ребру, не превышает вес ребра.

Мы предлагаем осуществлять построение ресурсно-допустимого множества путей в два этапа. На первом этапе строится замкнутое по дугам множество рёберно-простых путей, соединяющих требуемые пары хостов [5]. Если построенное множество не является ресурсно-допустимым, то предлагается простой способ для изменения дуг пути (рис. 1), используя рёбра взвешенного графа связей, которые отсутствуют в путях P (множество N в алгоритме 1). Замена ребра происходит, если число путей, проходящих через ребро, больше веса, сопоставленного ребру.

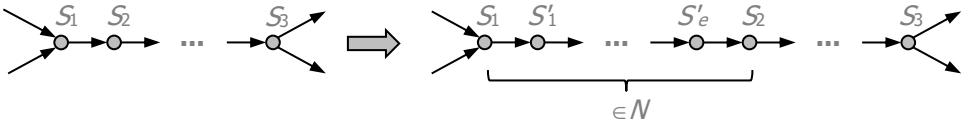


Рис. 1. Замена дуги s_1s_2 в путях множества P
Fig. 1. Replacing the arc s_1s_2 in the paths of P

Алгоритм 1: Модификация пути посредством изменения ребра или последовательности рёбер, при этом сохраняя головной и хвостовой узлы исходного пути.

***Вход**: непустое множество P рёберно-простых полных путей, замкнутое по дугам, непустое множество N рёбер между коммутаторами графа связей, которые не используются в путях из множества P.*

***Выход**: сообщение «множество P путей не может быть модифицировано» или модифицированное замкнутое по дугам множество Q, где головная и хвостовая вершины каждого изменённого пути совпадают с таковыми из исходного множества.*

while существует ребро (s_1, s_2) такое, что суммарное число k дуг s_1s_2 и s_2s_1, входящих в пути из множества P, больше ёмкости ребра (s_1, s_2)
do
  if в N существует подмножество рёбер $\{(s_1, s_1'), (s_1', s_2'), \ldots, (s_l', s_2)\}$, в котором ёмкость каждого ребра не меньше k,
  then
    В каждом пути из множества P заменить дугу s_1s_2 на путь $s_1s_1's_2' \ldots s_l's_2$,
    а дугу s_2s_1 – на путь $s_2s_l' \ldots s_2's_1's_1$;
    Удалить рёбра (s_1, s_1'), (s_1', s_2'), …, (s_l', s_2) из N;
  else
    модификация невозможна;

Можно показать, что если алгоритм 1 возвращает набор путей Q, то Q есть ресурсно-допустимое замкнутое по дугам множество рёберно-простых полных пути, причём для каждого модифицированного пути головная и хвостовая вершины совпадают с таковыми из исходного множества P. Действительно, рассмотрим два пути из множества P, которые пересекаются по дуге s_1s_2, и суммарное число k дуг s_1s_2 и s_2s_1, входящих в пути из множества P, больше ёмкости соответствующего ребра. Если в N существует подмножество рёбер $\{(s_1, s_1'), (s_1', s_2'), \ldots, (s_l', s_2)\}$, такое, что ёмкость каждого ребра не меньше k, то в каждом пути из множества P дуга s_1s_2 заменяется на путь $s_1s_1's_2' \ldots s_l's_2$, а дуга s_2s_1 – на путь $s_2s_l' \ldots s_2's_1's_1$. Поскольку каждый модифицируемый путь является рёберно-простым, и ни в каком другом месте модифицируемого пути дуг s_1s_1', $s_1's_2'$, …, $s_l's_2$ и s_2s_l', …, $s_1's_1s_1$ нет, то получим множество рёберно-простых путей Q, в котором для каждого модифицированного пути головная и хвостовая вершины совпадают с таковыми из исходного множества P. Более того, если до модификации для любых двух путей, для которых имелось слияние по некоторой дуге, а потом расхождение, все четыре пути принадлежали P, то и после модификации это свойство охраняется, то есть модифицированное множество путей является замкнутым по дугам. Сложность алгоритма $\boldsymbol{O}(m^2)$, где m число рёбер графа.

В алгоритме 1 мы рассматриваем достаточно простую эвристику для модификации путей, и результат существенно зависит от порядка перебора рёбер путей в P. Необходимы дополнительные исследования, чтобы детально исследовать вопрос о порядке такого перебора. Заметим также, что параметр типа «ресурс» можно ввести для дуг, заменив ребро в графе связей двумя соответствующими дугами, каждая из которых имеет свой ресурс, а также ввести такой параметр для коммутаторов. В обоих случаях можно использовать алгоритм 1 с небольшими модификациями.

4. Построение виртуальной сети на взвешенном графе физических связей с учётом параметров типа «стоимость»

Учёт параметров типа «стоимость» – это задача снижения нагрузки на сеть, когда нагрузка на сеть понимается как суммарное число прохождений пакетов по рёбрам графа: пара (пакет, ребро) учитывается, когда пакет проходит по ребру. Число пакетов, одновременно циркулирующих в сети SDN, зависит от поведения хостов, формирующих эти пакеты, и

потому является величиной переменной. Более грубая оценка характеризует виртуальную сеть независимо от движения пакетов по ней, она учитывает суммарное число не пакетов, а путей, по которым пакеты движутся. Пара (путь, ребро) учитывается, если путь проходит по ребру. Множество путей, реализуемых SDN, определяется настройкой коммутаторов, которая меняется по командам с верхнего уровня (SDN-контроллеры), что происходит значительно реже, и потому это множество путей в рамках рассматриваемой задачи может считаться статическим. Прохождение пути по разным рёбрам вносит в суммарную нагрузку слагаемые разной величины, что моделируется понятием веса ребра, понимаемым как «цена» его прохождения.

В нашей работе [7] рассматривалось выполнение на плоскости данных SDN программы задания, которая понималась в духе парадигмы объектно-ориентированного программирования как состоящая из объектов и сообщений, которыми объекты могут обмениваться. Объекты реализуется в хостах, причём в одном хосте может быть реализовано несколько разных объектов, а один и тот же объект может быть реализован в нескольких хостах. Сообщения между объектами, реализованными в разных хостах, помещаются в пакеты, двигающиеся по сети из начального хоста в конечный хост. В работе решались две задачи: 1) минимизация числа идентификаторов пакетов, соответствующих объектам, реализованным в хостах, 2) настройка коммутаторов (реализации путей, которые должны проходить пакеты) для выбранного соответствия идентификаторов пакетов и объектов. Эти задачи решались в двух случаях: A) без клонирования пакетов: пакет, предназначенный для некоторого объекта, должен попасть ровно в один хост, и в этом хосте должен быть реализован нужный объект, B) с клонированием пакетов: пакет может попадать в несколько хостов, но в одном и только одном из них должен быть реализован нужный объект.

Решение задачи 1 зависит от распределения объектов по хостам, реализующим эти объекты, и не зависит от графа физических связей. Решение задачи 2 использует результаты решения задачи 1 и зависит от графа физических связей. В случае A решением задачи 1 является непустое множество хостов $U(d)$, определяемое для каждого идентификатора пакета d. Требовалось провести пути из каждого хоста в один и только один хост из $U(d)$. В случае B решением задачи 1 является непустое семейство непустых непересекающихся множеств хостов $U(d)$, определяемое для каждого идентификатора пакета d. Хосты одного множества из семейства $U(d)$ реализуют одно и то же множество объектов, а каждый объект реализован в каждом хосте одного и только одного множества из семейства $U(d)$. Требовалось провести пути так, чтобы для каждого хоста h и каждого множества $U(d)_i$ из семейства $U(d)$ множество путей, начинающихся в хосте h, содержало один и только один путь, заканчивающийся в некотором хосте из множества $U(d)_i$.

В работе [7] нас не интересовали вопросы оптимизации нагрузки на SDN, то есть мы не учитывали веса рёбер (граф считался невзвешенным) при рассмотрении возможности реализации требуемой настройки коммутаторов. Однако такой учёт был специально указан как возможное направление дальнейших исследований, то есть рассмотрение взвешенного графа физических связей.

В данной работе мы корректируем решение задачи 2 для случая взвешенного графа с весами типа «стоимость». В предложенных в [7] решениях задачи 2 для невзвешенного графа строились кратчайшие пути, удовлетворяющие условиям задачи (случай A или B). Для взвешенного графа аналогичная задача 2 формулируется как задача минимизации суммарного веса путей, то есть задача о кратчайших взвешенных путях, то есть о путях с минимальной суммой весов входящих в эти пути рёбер.

4.1. Решение задачи построения путей с минимальной суммарной стоимостью для случая положительных целочисленных весов рёбер

В случае, когда веса рёбер выражаются натуральными числами (для невзвешенного графа веса всех рёбер равны 1), задача построения путей с минимальной суммарной стоимостью сводится к аналогичной задаче для невзвешенного графа с соответствующим увеличением сложности алгоритма. Для этого ребро с весом p преобразуется в вершинно-простой путь длиной p введением p - 1 промежуточных вершин степени 2 (рис. 2). Опишем это формально.

Под взвешенным графом будем понимать граф $G = (V, E, f)$, где V множество вершин, $E \subseteq V \times V$ множество рёбер, $f: E \to N$ функция веса ребра, N множество натуральных чисел. Задача построения кратчайших взвешенных путей в графе G сводится к задаче построения невзвешенных кратчайших путей в невзвешенном графе $G^* = (V^*, E^*)$, который получается из G удалением всех рёбер и добавлением для каждого ребра $ab \in E$ с весом $p = f(ab)$ дополнительных вершин $V(ab) = \{ c_i : i = 1..p - 1 \}$ и, обозначая $c_0 = a$, $c_p = b$, дополнительных рёбер $E(ab) = \{c_i c_{i+1} : i = 0..p\}$, где $c_i \notin V$, $i = 1..p - 1$. Тогда $V^* = V \cup \cup\{ V(ab) : ab \in E \}$ и $E^* = \{ E(ab) : ab \in E \}$.

Рис. 2. Преобразование ребра ab с весом p в путь из a в b длиной p
Fig. 2. Replacing edge ab of weight p with path from a to b of length p

Поскольку сложность алгоритма решения задачи 2A построения кратчайших путей для невзвешенного графа равна $\boldsymbol{O}(m)$, где m число рёбер, сложность соответствующего алгоритма для взвешенного графа равна $\boldsymbol{O}(M)$, где M сумма весов рёбер графа. Сложность алгоритма решения задачи 2B для невзвешенного графа равна $\boldsymbol{O}(km)$, где k мощность семейства U целевых хостов (U результат решения задачи 1B), а m число рёбер графа. Поэтому сложность соответствующего алгоритма для взвешенного графа равна $\boldsymbol{O}(kM)$.

Отсутствие зацикливания, дублирования и замкнутость по дугам строящегося множества путей непосредственно следует из того, что алгоритм для каждой вершины определяет в случае без клонирования единственный кратчайший путь в заданное множество U, а в случае с клонированием – ровно по одному пути в каждое множество из семейства U.

4.2. Оптимизация решения задачи построения путей с минимальной суммарной стоимостью для случая положительных целочисленных весов рёбер

Если сумма весов рёбер графа $M >> nm$, где n число вершин графа, то возможна оптимизация алгоритма со сложностью $\boldsymbol{O}(nm)$ для случая A или, учитывая, что $m = \boldsymbol{O}(n^2)$, со сложностью $\boldsymbol{O}(n^3)$, и со сложностью $\boldsymbol{O}(knm)$ для случая B, или, учитывая, что $m = \boldsymbol{O}(n^2)$, со сложностью $\boldsymbol{O}(kn^3)$.

Для невзвешенного графа алгоритм 2A основан на обходе неориентированного графа в ширину и может рассматриваться как модификация алгоритма Дейкстра, но вместо вычисления длины кратчайшего пути выполняется настройка коммутаторов вдоль пути. Идея алгоритма заключается в следующем. Начиная с множества хостов U, по графу распространяется «волна» построения путей с помощью пометок вершин. Фронт F этой волны состоит из помеченных вершин a с одним и тем же расстоянием до ближайшего хоста из множества U, тогда как остальные помеченные вершины находятся на меньшем расстоянии от U. Для каждой вершины a из фронта волны просматриваются её соседи и для каждого соседа b, который ещё не помечен, выставляется пометка и запоминается его сосед

$p(b) = a$, через которого ведёт кратчайший путь из вершины b до ближайшего хоста из множества U. Если вершина a коммутатор, в нём устанавливается правило $d\ b\ a\ p(a)$, означающее: при получении коммутатором a пакета с идентификатором d от вершины b следует направить этот пакет вершине $p(a)$. Если вершина b коммутатор, она помещается в новый фронт волны F_{next}. Когда просмотрены все вершины фронта F, начинается новый шаг, на котором фронтом волны становится F_{next}. Алгоритм завершает свою работу, когда просмотрены все вершины и фронт F стал пустым.

На каждом шаге алгоритма имеется множество помеченных вершин и его подмножество, фронт волны, содержащее вершины, у которых могут быть не просмотренные инцидентные им рёбра (эти рёбра могут вести в непомеченные вершины). На каждом шаге алгоритма просматриваются вершины фронта волны и не просмотренные инцидентные им рёбра. В новый фронт волны помещаются непомеченные вершины на концах таких рёбер. В случае взвешенного графа мы модифицируем граф G в граф G^*, превращая каждое ребро ab веса p в вершинно-простой путь $E(ab)$ из a в b длиной p - 1, добавляя дополнительные вершины c_i, $i = 1..p - 1$. Из-за этого несколько шагов подряд во фронт волны могут попадать только дополнительные вершины. Оптимизация заключается в том, чтобы объединить несколько шагов в один шаг так, чтобы на каждом объединённом шаге во фронт волны попадала хотя бы одна исходная (не дополнительная) вершина.

Пусть в графе G^* вершина v принадлежит фронту волны, $v \in F$, и ребро vu не просмотрено. Ребро vu лежит на пути $(d_1 = v)(d_2 = u)$, d_2d_3, ..., $d_L(d_{L+1} = w)$, в котором конечная вершина w исходная (не дополнительная), а все промежуточные вершины d_i, $i = 1..k$, дополнительные. Длина этого пути равна L, обозначим её $L(vu)$. Минимум $l = \{ L(vu) \mid v \in F$ & ребро vu не просмотрено $\}$ определяет число шагов алгоритма, которые объединяются в один шаг. На этом объединённом шаге для каждого не просмотренного ребра vu помечается вершина на расстоянии l от вершины v вдоль пути, начинающегося ребром vu, то есть вершина d_{l+1}. На каждом объединённом шаге помечается хотя бы одна непомеченная ранее исходная (не дополнительная) вершина.

Заметим, что для определения минимальной длины $L(vu)$ не обязательно просматривать весь путь от вершины v до исходной (не дополнительной) вершины w, начинающийся ребром vu. Когда ребро ab с весом p превращается в путь $(c_0 = a)c_1$, c_1c_2, ..., $c_{p-1}(c_p = b)$ длиной p, в каждой вершине c_i, $i = 0..p$, можно хранить её расстояние до конца b этого пути, которое равно $p - i$. Тогда, если $v = c_i$, а $u = c_{i+1}$, то $L(vu) = p - i$.

Оценка сложности этой оптимизации. На каждом шаге просматривается не более m рёбер. Также на каждом шаге помечается хотя бы одна исходная (не дополнительная) вершина. Следовательно, сложность алгоритма не превышает $\boldsymbol{O}(nm) = \boldsymbol{O}(n^3)$. В случае B алгоритм делает то же самое, но для каждого из k множеств семейства U. Тем самым, оценка сложности равна $\boldsymbol{O}(knm) = \boldsymbol{O}(kn^3)$.

4.3. Решение задачи построения путей с минимальной суммарной стоимостью для случая произвольных неотрицательных весов рёбер

Это решение основано не на модификации графа, как в подразделах 4.1-4.2, а на модификации самих алгоритмов построения путей.

Поскольку рёбра имеют разные веса, кратчайший путь (путь с минимальным числом рёбер) может не совпадать с кратчайшим взвешенным путём (путь с минимальной суммой весов его рёбер). Поэтому предлагается следующая модификация алгоритма.

Каждый раз, когда из вершины a просматривается ребро ab, не обращаем внимания на то, помечена вершина b или нет, то есть пометки вершин не используются. Вместо этого в каждой вершине v хранится её взвешенное расстояние $L(v)$, как минимальная сумма весов рёбер пути до множества хостов U, вычисленное на данный момент времени. Если вершина

ещё не просматривалась, это расстояние считается бесконечным. Когда из вершины a просматривается ребро ab, проверяется условие $L(b) > L(a) + f(ab)$. Если условие выполнено, $L(b)$ меняется: $L(b) = L(a) + f(ab)$, и вершина b, если она коммутатор, помещается в новый фронт волны. Заметим, что условия $L(b) > L(a) + f(ab)$ и $L(a) > L(b) + f(ab)$ не могут одновременно выполняться, если вес ребра неотрицательный. Тем самым, вершина может несколько раз оказываться во фронте волны со строго убывающей последовательностью её расстояний до целевых хостов.

Оценим сложность алгоритма для случая *A*. Пусть максимальная невзвешенная длина пути от хоста до хоста из U равна l_{max}. Очевидно, что алгоритм закончит работу через l_{max} шагов. На каждом шаге просматривается не более m рёбер. Поскольку $l_{max} < n$, где n число вершин, оценка сложности равна $\boldsymbol{O}(nm) = \boldsymbol{O}(n^3)$. Для случая B то же самое делается для каждого из k множеств хостов из семейства U: каждый раз, когда в алгоритме 2A ребро просматривается один раз, в алгоритме 2B оно просматривается k раз. Поэтому оценка сложности равна $\boldsymbol{O}(knm) = \boldsymbol{O}(kn^3)$. Ниже приведены формальные описания алгоритмов.

Алгоритм_2A (*V, NB, f, d, U, R*) /* Настройка коммутаторов для кратчайшего взвешенного доступа (замкнутого по дугам, без зацикливания и дублирования) пакетов с идентификатором d от каждого хоста до хоста из множества U */

***Вход*:**

V – множество вершин графа физических связей;

NB – функция, задающая для каждой вершины a множество $NB(a)$ её соседей;

f – функция, задающая для каждого ребра ab его вес $f(ab)$;

d – идентификатор пакетов;

U – непустое множество целевых хостов, соответствующее d;

R – текущая настройка коммутаторов, задающая для каждого коммутатора s текущее множество правил вида $d'a\,s\,b$, где $d' \neq d$;

***Выход*:** новая настройка коммутаторов R.

$p(a)$ – сосед вершины a на пути к ближайшему (с учётом весов) хосту из U.

$L(a)$ – взвешенное расстояние от вершины a до хоста из U; $L(a) = \infty$ означает, что вершина ещё не просматривалась; вес ребра меньше ∞;

F – фронт (множество вершин) волны от хостов из U до других хостов.

F_{next} – множество F на следующем шаге.

```
F = ∅; F_next = ∅;
for all a ∈ V do L(a) = ∞;                  /* все вершины ещё не просматривались */
for all a ∈ U do                            /* хосты из U помещаются в фронт волны */
└  L(a) = 0; F = F ∪ { a };
while F ≠ ∅ do                              /* пока фронт волны не пуст */
│  for all a ∈ F do                         /* просмотр вершин a из фронта волны */
│  │  for all b ∈ NB(a) do                  /* просматриваются соседи b вершины a */
│  │  │  if L(b) > L(a) + f(ab) then        /* расстояние соседа b станет меньше */
│  │  │  │  L(b) = L(a) + f(ab); p(b) = a;
│  │  │  │  if | NB(a) | > 1 then           /* a коммутатор */
│  │  │  │  └  R(a) = R(a) ∪ { d b a p(a) }; /* правило коммутатора a */
│  │  │  │  if | NB(b) | > 1 then           /* b коммутатор */
└  └  └  └  └  F_next = F_next ∪ { b };     /* помещаем b в следующий фронт */
return R;
```

Алгоритм_2В (*V, NB, f, d, U, R*) /* Настройка коммутаторов для кратчайшего взвешенного доступа (замкнутого по дугам, без зацикливания и дублирования) от каждого хоста пакетов с идентификатором *d* до одного хоста из каждого множества хостов в семействе множеств хостов *U* */

***Вход*:**

V – множество вершин графа физических связей;

NB – функция, задающая для каждой вершины *a* множество *NB*(*a*) её соседей;

f – функция, задающая для каждого ребра *ab* его вес *f*(*ab*);

d – идентификатор пакетов;

U – непустое семейство непустых множеств хостов, соответствующее *d*;

$k = |U|$;

R – текущая настройка коммутаторов, задающая для каждого коммутатора *s* текущее множество правил вида $d'\,a\,s\,b$, где $d' \neq d$;

***Выход*:** новая настройка коммутаторов *R*.

$p(a) = \{ p(a)(1), ..., p(a)(k) \}$ – вектор соседей вершины *a* на путях к ближайшим (с учётом весов) хостам из множеств, входящих в *U*,

$L(a) = \{ L(a)(1), ..., L(a)(k) \}$ – вектор взвешенных расстояний от вершины *a* до ближайшего (с учётом весов) хоста из множеств, входящих в *U*; $L(a)(i) = \infty$ означает, что вершина ещё не просматривалась; вес ребра меньше ∞;

F – фронт волны (множество вершин) от хостов из *U* до других хостов,

F_{next} – множество *F* на следующем шаге.

```
F = ∅; F_next = ∅;
for all a ∈ V do
|   for all i ∈ { 1, ..., k } do
|   |   L(a)(i) = ∞;                    /* все вершины ещё не просматривались */
for all i ∈ { 1, ..., k } do             /* индексы i */
|   for all a ∈ U(i) do                  /* вершины в U(i)*/
|   |   L(a)(i) = 0; F = F ∪ { a };      /* хосты из U(i) помещаются в фронт волны */
while F ≠ ∅ do                                  /* пока фронт волны не пуст */
|   for all a ∈ F do                           /* просмотр вершин a из фронта волны */
|   |   for all b ∈ N(a) do                    /* просмотр соседей b вершины a */
|   |   |   for all i ∈ { 1, ..., k } do      /* индексы i */
|   |   |   |   if L(b)(i) > L(a)(i) + f(ab) then   /* расстояние соседа b станет меньше */
|   |   |   |   |   L(b)(i) = L(a)(i) + f(ab); p(b)(i) = a;
|   |   |   |   |   if | NB(a) | > 1 then                  /* a коммутатор */
|   |   |   |   |   |   R(a) = R(a) ∪ { ( d b a p(v)(i) };  /* правило коммутатора a */
|   |   |   |   |   if | NB(b) | > 1 then                  /* b коммутатор */
|   |   |   |   |   |   F_next = F_next ∪ { b };            /* помещаем b в следующий фронт */
return R;
```

5. Заключение

В предыдущих работах [4-7] авторов не интересовали вопросы оптимизации нагрузки на SDN, и соответственно не учитывался вес ребра в графе связей, то есть граф считался невзвешенным при рассмотрении возможности реализации требуемой настройки коммутаторов. В настоящей работе рассматривается взвешенный граф связей для плоскости данных сети, в котором веса рёбер соответствуют параметрам типа «ресурс» и «стоимость», что налагает ограничения на маршрутизацию пакетов.

Вообще говоря, набор нефункциональных параметров, используемых для исследования вопросов надежности, безопасности и оптимизации SDN, может быть расширен, а кроме

того, могут использоваться сочетания параметров разных типов. Мы предполагаем продолжить наши исследования различных нефункциональных параметров и их сочетаний, а также их влияния на качество реализуемых SDN.

Список литературы / References

[1]. Sezer. S, Scott-Hayward. S, Chouhan P.K., Fraser B., Lake D., Finnegan J., Viljoen N., Miller M. and Rao N. Are we ready for sdn? Implementation challenges for software-defined networks IEEE Communications Magazine, 2013, 51 (7), pp. 36-43.

[2]. Mohammed, A. H., Khaleefah, R. M., k. Hussein, M., and Amjad Abdulateef, I. A review software defined networking for internet of things. In 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2020, pp. 1–8.

[3]. OpenNetworkingFoundation (2012). Software-defined networking: The new norm for networks. ONF White Paper. 2012.

[4]. Burdonov, I.; Kossachev, A.; Yevtushenko, N.; López, J.; Kushik, N. and Zeghlache, D. (2021). Preventive Model-based Verification and Repairing for SDN Requests. In Proceedings of the 16th International Conference on Evaluation of Novel Approaches to Software Engineering - ENASE, ISBN 978-989-758-508-1 ISSN 2184-4895, pages 421-428. DOI: 10.5220/0010494504210428.

[5]. Igor B. Burdonov**,** Nina Yevtushenko, Alexandre Kossachev*:* Verifying Multiple Virtual Networks in Software Defined Networks. Proceedings 2021 IEEE East-West Design & Test Symposium (EWDTS2021).

[6]. Igor Burdonov, Nina Yevtushenko and Alexander Kossatchev. Implementing a virtual network on the SDN data plane. Proceedings 2020 IEEE East-West Design & Test Symposium (EWDTS). 2020, pp. 279-283.

[7]. Бурдонов И.Б., Евтушенко Н.В., Косачев А.С. Реализация распределенных и параллельных вычислений в сети SDN. Труды института системного программирования. 2022. Т. 34. № 3. С. 159-172.

Информация об авторах / Information about authors

Игорь Борисович БУРДОНОВ – доктор физико-математических наук, главный научный сотрудник ИСП РАН. Научные интересы: формальные спецификации, генерация тестов, технология компиляции, системы реального времени, операционные системы, объектно-ориентированное программирование, сетевые протоколы, процессы разработки программного обеспечения.

Igor Borisovich BURDONOV – Dr. Sci. (Phys.-Math.), a Leading Researcher of ISP RAS. Research interests: formal specifications, test generation, compilation technology, real-time systems, operating systems, object-oriented programming, network protocols, software development processes.

Нина Владимировна ЕВТУШЕНКО, доктор технических наук, профессор, главный научный сотрудник ИСП РАН, до 1991 года работала научным сотрудником в Сибирском физико-техническом институте. С 1991 г. работала в ТГУ профессором, зав. кафедрой, зав. лабораторией по компьютерным наукам. Её исследовательские интересы включают формальные методы, теорию автоматов, распределённые системы, протоколы и тестирование программного обеспечения.

Nina Vladimirovna YEVTUSHENKO, Dr. Sci. (Tech.), Professor, a Leading Researcher of ISP RAS, worked at the Siberian Scientific Institute of Physics and Technology as a researcher up to 1991. In 1991, she joined Tomsk State University as a professor and then worked as the chair head and the head of Computer Science laboratory. Her research interests include formal methods, automata theory, distributed systems, protocol and software testing.

Александр Сергеевич КОСАЧЕВ – кандидат физико-математических наук, ведущий научный сотрудник ИСП РАН. Научные интересы: формальные спецификации, генерация тестов,

технология компиляции, системы реального времени, операционные системы, объектно-ориентированное программирование, сетевые протоколы, процессы разработки программного обеспечения.

Alexander Sergeevitch KOSSATCHEV – Cand. Sci. (Phys.-Math.), a Leading Researcher of ISP RAS. Research interests: formal specifications, test generation, compilation technology, real-time systems, operating systems, object-oriented programming, network protocols, software development processes.

DOI: 10.15514/ISPRAS-2024-36(1)-3

Виды атак на федеративные нейросети и способы защиты

В.А. Костенко, ORCID: 0000-0002-7895-2322 <*kostmsu@gmail.com*>
А.Е. Селезнева, ORCID: 0009-0005-8480-8182 <*alice.in.moskow@gmail.com*>

Московский государственный университет имени М.В. Ломоносова,
Россия, 119991, Москва, Ленинские горы, д. 1.

Аннотация. Федеративное обучение — это технология обучения с сохранением конфиденциальности в распределенных системах хранения данных. Такое обучение позволяет создать общую модель прогнозирования, сохраняя все данные в своих системах хранения. В обучении общей модели учувствуют несколько устройств, при этом каждое устройство имеет свои уникальные данные, на которых обучается нейросеть. Взаимодействие устройств происходит только для корректировки весов общей модели. После чего, обновленная модель передается на все устройства. Обучение на нескольких устройствах рождает множество возможностей для атак на этот тип сетей.

Ключевые слова: нейронные сети; федеративные классификаторы; безопасность нейронных сетей; атаки на нейросети; защита нейросетей; атаки отравлением; атаки уклонения; атаки логического вывода; атаки восстановления данных.

Для цитирования: Костенко В.А., Селезнева А.Е. Виды атак на федеративные нейросети и способы защиты. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 35–44. DOI: 10.15514/ISPRAS–2024–36(1)–3.

Types of Attacks on Federated Neural Networks and Methods of Protection

V.A. Kostenko, ORCID: 0000-0002-7895-2322 <kostmsu@gmail.com>
A.E. Selezneva, ORCID: 0009-0005-8480-8182 <alice.in.moskow@gmail.com>

Lomonosov Moscow State University,
GSP-1, Leninskie Gory, Moscow, 119991, Russia.

Abstract. Federated learning is a technology for privacy-preserving learning in distributed storage systems. This training allows you to create a general forecasting model, storing all the data in your storage systems. Several devices take part in training the general model, and each device has its own unique data on which the neural network is trained. The interaction of devices occurs only to adjust the weights of the general model. After which, the updated model is transmitted to all devices. Training on multiple devices creates many attack opportunities against this type of network. After training on a local device, model data is sent via some type of communication to a central server or global model. Therefore, vulnerabilities in a federated network are possible not only at the training stage on a separate device, but also at the data exchange stage. All this together increases the number of possible vulnerabilities of federated neural networks. As is known, not only neural networks, but also other models can be used to build federated classifiers. Therefore, the types of attacks directly on the network also depend on the type of model used. Federated neural networks are a rather complex design, different from neural networks and other classifiers, which can be vulnerable to various types of attacks because training occurs on different devices, and both neural networks and simpler algorithms can be used. In addition, it is necessary to ensure data transfer between devices. All attacks come down to several main types that exploit

classifier vulnerabilities. It is possible to implement protection against attacks by improving the architecture of the classifier itself and paying attention to data encryption.

Keywords: machine learning; federal neural networks; neural network attacks; neural network protections; poisoning attacks; evasion attacks; logical inference attacks; data recovery attacks.

For citation: Kostenko V.A., Selezneva A.E. Types of attacks on federated neural networks and methods of protection. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 35-44 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-3.

1. Введение

Федеративные нейросети обучаются распределенно и позволяют множеству участников вместе обучать единую модель, не раскрывая своих уникальных наборов данных.

Существует несколько подходов для создания федеративных сетей. Один из подходов – это обучение под управлением центрального сервера. Такая архитектура нейронной сети подразумевает, что данные обучения хранятся на каждом устройстве только локально. Эти данные не передаются ни другим устройствам, ни центральному серверу. Центральный сервер принимает на вход только обновленные параметры модели машинного обучения, тем самым, не взаимодействуя с данными, на которых обучались локальные модели [1].

Другой вид федеративного обучения – децентрализованное обучение. Так же обучается глобальная модель, но взаимодействие между узлами сети происходит непосредственно друг с другом, без участия центрального сервера. Данные обучения каждого устройства так же остаются локальными, передаются только параметры модели машинного обучения [1].

Один из примеров федеративных нейросетей – Google клавиатура на устройствах Android. По умолчанию на устройствах Android установлена клавиатура со стандартными подсказками при вводе слов и при составлении предложений. Со временем, для каждого пользователя формируется собственный набор предсказания слов. Этот набор становится уникальным и зависит только от того, насколько часто конкретный пользователь использовал слова в определенном контексте. Клавиатура обучается локально на каждом устройстве, сервер получает информацию только о данных модели обучения, без каких-либо обучающих данных [2].

Федеративные нейросети обучаются на множестве устройств, на каждом из которых происходит собственное обучение. После обучения на локальном устройстве, данные модели пересылаются по каким-либо видам связи на центральный сервер или глобальную модель. Поэтому уязвимости федеративной сети возможны не только на этапе обучения на отдельном устройстве, но и на этапе обмена данными. Все это в совокупности увеличивает количество возможных уязвимостей федеративных нейросетей.

Как известно, для построения федеративных классификаторов могут быть использованы не только нейросети, но и другие модели и методы. Поэтому виды атак непосредственно на сеть также зависят от типа используемых методов и моделей. В данной работе будут рассматриваться федеративные классификаторы, построенные на основе нейросетей.

2. Виды атак и способы защиты

Атаки на федеративные классификаторы могут иметь различные цели – атаки непосредственно на локальное устройство с целью его захвата, получение данных о самой модели или же целью может выступать сам классификатор. Уязвимы могут быть каналы передачи данных и большинство локальных устройств вместе с центральным сервером, если он присутствует у классификатора. Под классификатором понимается обученная модель, управляемая центральным или распределенным сервером, которая обновляется за счет локальных устройств с децентрализованными данными обучения [1].

На рис. 1 схематично отображены уязвимые моменты в работе федеративного классификатора. Можно увидеть, что атаки бывают нацелены на локальные устройства, например, целью может выступать – отравление данных одного или нескольких устройств. Далее атаки переходят на попытки отравления модели или моделей – как локальных устройств, так и центрального классификатора. При взаимодействии локальных устройств и центрального классификатора, возможно, как прослушивание любых передаваемых данных, так и попытки сделать выводы о самих данных и (возможны) нарушения конфиденциальных данных [3].

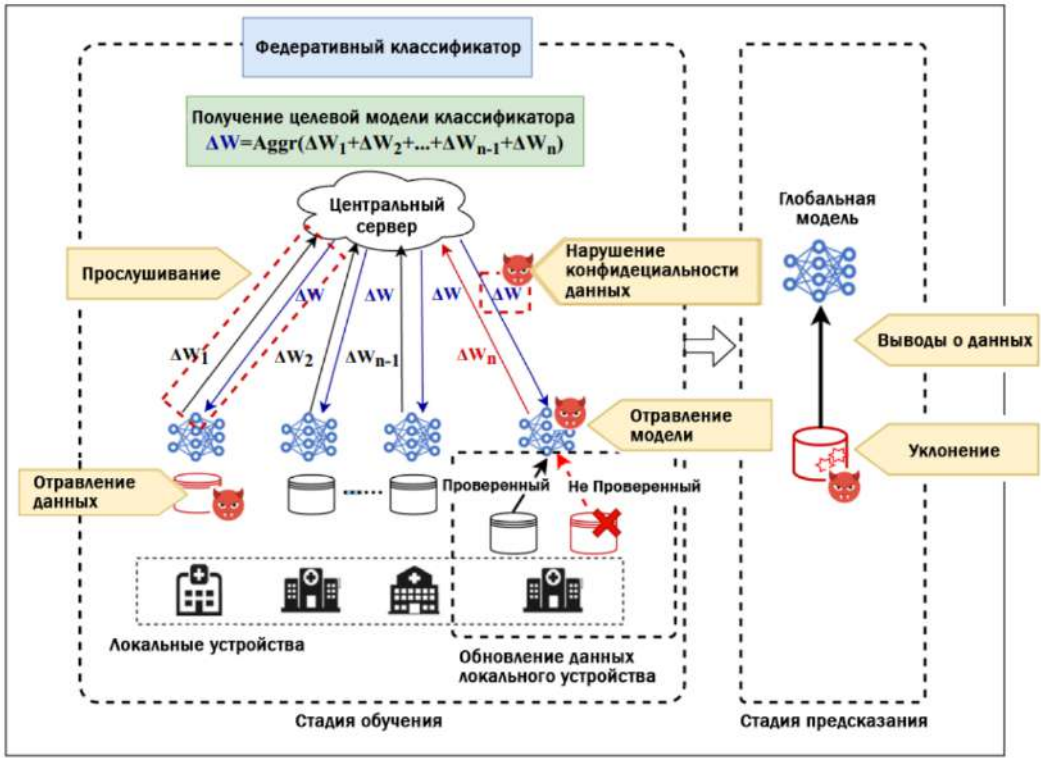


Рис. 1. Схема федеративного классификатора, обозначение уязвимостей и возможных атак
Fig. 1. Scheme of federated neural networks, designation of vulnerabilities and possible attacks

2.1 Атаки отравлением

Под отравлением данных понимается злонамеренное изменение или добавление данных в обучающую выборку, что в конечном итоге, приводит к захвату локального устройства или сервера. Атаки могут быть случайными или целенаправленными, как пример, можно привести бэкдор атаки [3]. Бэкдор атаки – это форма состязательных атак на нейронные сети, во время которых злоумышленник использует зараженные данные. Зараженные данные – это данные, которые способны повлиять на целостность и качество модели или классификатора [1]. Зараженные данные используются злоумышленником для обучения модели. Сначала подкладываются зараженные данные в процесс обучения модели, а затем, злоумышленник в любой момент может активировать атаку, с помощью определенного шаблона-триггера. Активировать атаку – значит передать на вход классификатору определенные данные, то есть триггер. Триггер связан с заранее зараженными данными. Зараженные данные заставляют классификатор обучаться неправильно, при этом в результате, злоумышленник получает ожидаемый результат [4].

Отравление глобальной модели осуществляется злоумышленником, который контролирует небольшое количество вредоносных агентов (локальных устройств). Отравление происходит путем передачи глобальной модели параметров локальных сетей, которые обучены на отравленных данных. Цель такой атаки -заставить глобальную модель ошибочно классифицировать набор выбранных входных данных с высокой степенью достоверности. Атаки отравлением можно поделить на чистые и грязные [5].

Атаки чистого отравления предполагают, что злоумышленник не может изменить весовые коэффициенты модели с помощью обучающих данных. В модели существует процесс, с помощью которого, данные можно валидировать, как принадлежащие к какому-либо определенному классу. Атаки с помощью грязного отравления подразумевают введение в обучающую выборку нескольких копий примеров, которые злоумышленник хочет неправильно классифицировать с указанием желаемого класса [6].

Возможно отравление как самих данных, например, добавление шума, так и отравление целой модели. В работе [5] отмечалось, что атаки грязного отравления данных не особенно эффективны для отравления всего федеративного классификатора, в отличие от отравления локального классификатора. Отравленный классификатор будет продолжать функционировать и посылать невалидные веса в центральную модель.

В федеративном классификаторе захват нескольких устройств для отравления модели центрального сервера так же является одной из возможных атак [7].

Возможны другие виды атак с помощью отравления данных. Рассмотрим атаки отравления под видом возможных ошибок классификатора. Цель злоумышленника – реализовать неправильную классификацию. Этот вид атаки возможен только для многоклассовой классификации. Атака может быть как целенаправленной, так и случайной. Особенность этих атак заключается в том, что они имитируют ошибки при нормальном функционировании сети [3].

Еще один вид атаки с помощью отравления данных – это атака на основе отравления градиента. Эта атака может быть весьма неэффективна. Однако, если использовать обратный градиент, то процедуру обучения можно проследить в обратном направлении, а затем вычислить значения градиента в обратном порядке. Это дает возможность осуществить атаку для большинства градиентных методов, таких как – градиентный спуск с фиксированным шагом [6].

2.2 Способы защиты от отравляющих атак

Первый способ борьбы с атаками отравления для федеративного классификатора – это независимая проверка центральным сервером точности глобальной модели с помощью тестовой выборки. Центральный сервер также может выполнять статистические проверки, сравнивая отличия обновлений локальных устройств между собой, тем самым он способен обнаружить вредоносное устройство [5].

Второй способ описан в работе [8], где было показано, что для атак отравления могут быть эффективны методы защиты с помощью дифференциальной конфиденциальности. Защита заключается в добавлении случайного шума к градиентам модели каждого локального устройства, причем устройства сами могут контролировать эти помехи. Это позволяет в федеративном классификаторе добиться полной уникальности добавления шума на каждом устройстве [9].

Третий способ – это добавление валидации на центральном сервере. В этом случае, отравление центрального сервера возможно только с захватом более 50% работающих устройств [10].

2.3 Атака уклонения

Атака уклонения – это тип атаки, при котором злоумышленник пытается обмануть целевую модель классификатора путем создания конкретных образцов, называемых состязательными примерами. Обычно небольшой шум, добавленный к входным данным у локальной модели, не может быть обнаружен людьми. Это приводит к тому, что такая модель получает неправильные результаты классификации [11]. Обычно это происходит на этапе прогнозирования, когда модель закончила обучение. Результат такого рода атаки – неправильная классификация у глобального классификатора. Одна из главных особенностей этой атаки – широкий разброс возможных опасностей. В качестве примера, можно рассмотреть неправильное распознавание дорожных знаков беспилотными автомобилями, некорректное распознавание лиц, неверная работа системы распознавания речи и прочее. Атака уклонения – это атака на целостность за счет подмены модели [12].

2.4 Способы защиты от атак уклонения

Первый способ защиты – эмпирическая защита, при которой, например, предполагается предварительная обработка исходных данных и преобразование признаков. Этот способ эффективен только если злоумышленнику не будет известно об этом механизме защиты. Другой способ эмпирической защиты – сокрытие полной информации о локальной модели. Используется слияние локальных моделей, при передаче данных глобальной модели, при наложении градиентной маски и другие методы [13].

Второй способ защиты – добавление случайного сглаживания исходного изображения с помощью Гауссовского шума на этапе классификации. Принуждение локального классификатора правильно классифицировать входные данные с учетом добавленного шума – тоже улучшит защищенность модели. При применении этого способа защиты – наблюдается устойчивость к атакам уклонения [14].

2.5 Атаки логического вывода с нарушением приватности

Можно выделить несколько типов логических атак с нарушением приватности.

Первый, нацелен на получение данных о том, является ли конкретный объект частью обучающей выборки. Злоумышленник обучает несколько теневых моделей для имитации поведения локального классификатора и обучает собственную модель на основе данных, полученных из выходных данных теневых моделей [15]. Он может использовать как черный ящик (злоумышленник ограничивается произвольным набором входных данных, пытаясь сделать на основе этого какие-либо выводы), так и белый ящик (злоумышленник получает доступ к самой модели, включая ее параметры, которые необходимы для классификации, поэтому, для любых входных данных он может получить помимо непосредственного результата, все промежуточные вычисления модели) [3]. Также злоумышленник может делать оценку принадлежности объекта к определённому классу, если классификатор осуществляет классификацию объектов [12].

Второй тип атаки реализуется, если злоумышленник нацелен на извлечение самой модели. Он пытается получить информацию о модели с помощью циклических запросов. С помощью атаки с извлечением параметров модели злоумышленник восстанавливает параметры модели за счет доступа к классификатору. Основная задача атакующего – построить состязательную модель, то есть модель, которая состоит из двух сетей: первая генерирует образцы (эту роль выполняет злоумышленник), вторая пытается их классифицировать (атакуемый классификатор) [16].

Третий тип атак – прослушивающие атаки на вывод свойств. Одна из уязвимостей федеративного классификатора – передача данных модели между устройствами. Возможны атаки с инверсией модели, которые используют, в основном, некоторые прикладные

системы, если они используются федеративным классификатором для получения или обмена данными о модели. С помощью этой информации злоумышленники могут проанализировать модель, чтобы получить соответствующую информацию об исходных данных, например, получить свойства модели [17]. Злоумышленники также могут прослушивать канал связи, чтобы получить любую информацию о самом федеративном классификаторе или о локальной модели [12].

Прослушивающие атаки можно разделить по вмешательству в работу классификатора на пассивные и активные. Во время пассивных атак злоумышленник только наблюдает за обновлением существующей модели и параллельно обучает собственный классификатор. На основе полученной модели он может сделать выводы о существующей модели. Во время активного способа атаки злоумышленник пытается обмануть модель, чтобы лучше получить целевые данные атакуемого объекта [18].

2.6 Способы защиты от логических атак

В качестве защиты рассматриваются методы защиты непосредственно передаваемых данных. Предполагается, что злоумышленник может каким-то образом получить доступ к каналу связи. Обеспечить безопасность передачи данных можно за счет, например, выбора безопасных протоколов с шифрованием. Мы рассмотрим только способы защиты непосредственно классификатора, без защиты канала передачи информации.

Первая возможная защита – это защита структуры модели, то есть возможно снижение чувствительности модели к обучающим выборкам и к переобучению модели. Следующий вариант – это защита от обфускации данных классификатора, способ добиться безопасности через неясность, например, введением в заблуждение с помощью изменения выходных данных модели [3].

Еще один способ добиться конфиденциальности данных – обфускация любых данных путем добавления случайного шума при обучении как к самим данным, так и к целевой функции, градиентам, параметрам и выходным данным. Этот метод снижает производительность, но повышает безопасность классификатора. Предыдущий способ защиты является частным случаем данного [19].

2.7 Атаки восстановления данных

Атаки восстановления данных могут позволить получить исходную информацию об обучающей выборке путем сбора любой доступной информации о классификаторе, например, предсказываемых значений, параметров и градиентов модели. Один из возможных сценариев атаки – это использование генеративно-состязательных сетей, за счет которых может быть получен доступ к данным других участников. При такой атаке злоумышленнику не обязательно иметь полное представление о федеративном классификаторе [20].

Другой тип атаки предполагает возможность восстановления исходных данных на основе информации о градиенте модели [21]. Градиент вычисляется через обратное распространение ошибки от последнего слоя к первому. Градиент конкретного слоя получается с использованием функции активации этого слоя и полученной ошибки от его верхнего слоя. Если следить за градиентами, можно вывести значения признаков, которые получаются непосредственно из исходных обучающих данных [18]. Во время использования градиента может происходить синтез пар фиктивных входных данных и меток путем сопоставления их фиктивных градиентов, близких к реальным через задачу оптимизации:

$$\arg\min_{x} || \nabla_\theta L_\theta(x, y) - \nabla_\theta L_\theta(x^*, y)||^2$$

где (x, y) – фиктивные данные и метки, (x*, y) – реальные данные и метки переданного градиента, $\nabla_\theta L_\theta(x^*, y)$ – передаваемый градиент, $\nabla_\theta L_\theta(x, y)$ – градиент фиктивных данных и

меток, $L_\theta(x, y)$ и $L_\theta(x^*, y)$ – некоторые функции потерь. Таким образом, можно восстановить исходные данные за счет перебора фиктивных данных и меток [3].

Еще один тип атаки восстановления данных нацелен на инвертирование градиентов. Эта атака подразумевает наличие полного доступа злоумышленника к данным модели. Инвертирование градиентов порождает противоположные значения, за счет чего модель перестает корректно работать [22].

2.8 Виды защиты от атак, нацеленных на восстановление данных

Один из способов защиты данных от атак, нацеленных на восстановление данных – использование возможности сжатия или разрежения градиентов при их передаче. Такой способ защиты позволяет защитить информацию при передаче градиентов моделей между устройствами. Другой способ защиты – отбрасывание случайных значений градиентов. Поскольку злоумышленник не знает, какой параметр отброшен и отброшен ли вообще, то задача восстановления данных для него сильно осложняется, поскольку ему придется использовать неполные данные. Недостаток такой защиты заключается в том, что небольшое изменение градиента ухудшает производительность классификаторов, как по времени, так и возможно по памяти [18].

Еще один способ защиты – это шифрование градиентов. Способы шифрования можно разделить на гомоморфное шифрование и безопасные многосторонние вычисления. Гомоморфное шифрование позволяет кодировать и обрабатывать зашифрованные данные так, что при этом расшифрованный результат будет эквивалентен результату, полученному на исходных данных. Данный алгоритм не изменяет исходную информацию, поэтому гарантируется отсутствие потери точности [23]. Безопасные многосторонние вычисления позволяют отдельным устройствам выполнять совместные вычисления на основе своих исходных данных, не раскрывая собственной информации другим участникам. Таким образом, они обеспечивают высокую степень конфиденциальности. Но получается, что каждое устройство должно согласовывать свои действие с соседними, что может негативно сказываться на производительности всего классификатора [24].

Еще один способ защиты был описан выше – добавление случайного шума к любым данным [25].

3. Заключение

Федеративные нейросети – это довольно сложная конструкция, отличающаяся от нейронных сетей и других классификаторов, которые могут быть уязвимы для различного рода атак, потому что обучение происходит на различных устройствах. Могут использоваться как нейросети, так и более простые алгоритмы. Помимо этого, необходимо обеспечивать передачу данных между устройствами.

Все атаки сводятся к нескольким основным типам, которые используют уязвимости классификатора. Можно реализовать защиту от атак с помощью усовершенствования архитектуры самого классификатора или уделить внимание шифрованию данных.

Список литературы / References

[1]. Kairouz, Peter; Brendan McMahan, H.; Avent, Brendan; Bellet, Aurélien; Bennis, Mehdi; Arjun Nitin Bhagoji; Bonawitz, Keith; Charles, Zachary; Cormode, Graham; Cummings, Rachel; D'Oliveira, Rafael G. L.; Salim El Rouayheb; Evans, David; Gardner, Josh; Garrett, Zachary; Gascón, Adrià; Ghazi, Badih; Gibbons, Phillip B.; Gruteser, Marco; Harchaoui, Zaid; He, Chaoyang; He, Lie; Huo, Zhouyuan; Hutchinson, Ben; Hsu, Justin; Jaggi, Martin; Javidi, Tara; Joshi, Gauri; Khodak, Mikhail; et al. (10 December 2019). "Advances and Open Problems in Federated Learning". arXiv:1912.04977, DOI: 10.48550/arXiv.1912.04977.

[2]. Federated Learning: Collaborative Machine Learning without Centralized Training Data (online) https://ai.googleblog.com/2017/04/federated-learning-collaborative.html — 01.12.2023.
[3]. Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives - Peng Liu, Xiangru Xu, Wen Wang, Cybersecurity 5, 4 (2022), DOI: 10.1186/s42400-021-00105-6.
[4]. Aniruddha Saha; Akshayvarun Subramanya; Hamed Pirsiavash; (2019), - "Hidden Trigger Backdoor Attacks" - arXiv:1910.00033v2, DOI: 10.48550/arXiv.1910.00033.
[5]. Bhagoji AN, Chakraborty S, Mittal P, Calo SB - Analyzing federated learning through an adversarial lens. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning, ICML 2019, 9–15 June 2019, Long Beach, California, USA, volume 97 of proceedings of machine learning research, pp 634–643, PMLR 97:634-643, 2019.
[6]. Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted backdoor attacks on deep learning systems using data poisoning, arXiv:1712.05526, 2017a, DOI: 10.48550/arXiv.1712.05526.
[7]. Clement Fung, Chris J. M. Yoon, and Ivan Beschastnikh. The limitations of federated learning in sybil settings. In 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020), pp. 301–316, San Sebastian, October 2020. USENIX Association. ISBN 978-1-939133-18-2.
[8]. Naseri M, Hayes J, De Cristofaro E (2020) Toward robustness and privacy in federated learning: experimenting with local and central differential privacy. CoRR arXiv:2009.03561, DOI: 10.48550/arXiv.2009.03561
[9]. Abadi M, Chu A, Goodfellow I, McMahan HB, Mironov I, Talwar K, Zhang L (2016) Deep learning with differential privacy. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pp 308–318, DOI: 10.48550/arXiv.1607.00133.
[10]. Kostenko V.A., Tankaev I.R., Federated Learning Using Simple Voting Scheme; 2022 - ISSN 1060-992X
[11]. Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow IJ, Fergus R (2014) Intriguing properties of neural networks. In: Bengio Y, LeCun Y (eds) 2nd international conference on learning representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, conference track proceedings, arXiv:1312.6199, DOI: 10.48550/arXiv.1312.6199.
[12]. Yao Chen1, Yijie Gui1, Hong Lin1, Wensheng Gan1,2∗, Yongdong Wu; Federated Learning Attacks and Defenses: A Survey – 2022; arXiv:2211.14952v1, DOI: 10.48550/arXiv.2211.14952.
[13]. JI Shou-Ling, Du Tian-Yu, Li Jin-Feng, Shen Chao, Li Bo - Security and privacy of machine learning models: a survey. Ruan Jian Xue Bao/J Softw 32(1):41–67, 2021, DOI: 10.13328/j.cnki.jos.006131.
[14]. Cohen JM, Rosenfeld E, Kolter JZ - Certified adversarial robustness via randomized smoothing. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on machine learning, ICML 2019, 9–15 June 2019, Long Beach, California, USA, volume 97 of proceedings of machine learning research. PMLR, pp 1310–1320, DOI: 10.48550/arXiv.1902.02918.
[15]. Nasr M, Shokri R, Houmansadr A (2019) Comprehensive privacy analysis of deep learning: passive and active white-box inference attacks against centralized and federated learning. In: 2019 IEEE symposium on security and privacy (SP). IEEE, pp 739–753, DOI: 10.1109/SP.2019.00065.
[16]. Ren K, Meng QR, Yan SK - Survey of artificial intelligence data security and privacy protection. Chin J Netw Inf Secur 7(1):1–10, 2021, DOI: 10.11959/j.issn.2096-109x.2021001.
[17]. Jayaraman B, Evans D (2019) Evaluating differentially private machine learning in practice. In: Heninger N, Traynor P (eds) 28th USENIX security symposium, USENIX security 2019, Santa Clara, CA, USA, August 14–16, 2019. USENIX Association, pp 1895–1912, DOI: 10.48550/arXiv.1902.08874.
[18]. Melis L, Song C, De Cristofaro E, Shmatikov V (2019) Exploiting unintended feature leakage in collaborative learning. In: 2019 IEEE symposium on security and privacy (SP). IEEE, pp 691–706. DOI: 10.48550/arXiv.1805.04049.
[19]. Papernot N, McDaniel PD, Sinha A, Wellman MP (2018) Sok: security and privacy in machine learning. In: 2018 IEEE European symposium on security and privacy, EuroS&P 2018, London, United Kingdom, April 24–26, 2018. IEEE, pp 399–414. https://doi.org/10.1109/EuroSP.2018.00035
[20]. B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in ACM SIGSAC Conference on Computer and Communications Security, 2017, pp. 603–618.
[21]. L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," Advances in Neural Information Processing Systems, vol. 32, 2019.
[22]. Geiping J, Bauermeister H, Dröge H, Moeller M (2020) Inverting gradients—how easy is it to break privacy in federated learning? arXiv preprint arXiv:2003.14053, DOI: 10.48550/arXiv.2003.14053.
[23]. Fang H, Qian Q (2021) Privacy preserving machine learning with homomorphic encryption and federated learning. Future Internet 13(4):94, DOI: 10.3390/fi13040094.

[24]. Li Y, Zhou Y, Jolfaei A, Dongjin Y, Gaochao X, Zheng X (2020) Privacy-preserving federated learning framework based on chained secure multi-party computing. IEEE Internet Things J. DOI: 10.1109/JIOT.2020.3022911.

[25]. Lyu L, Yu H, Ma X, Sun L, Zhao J, Yang Q, Yu PS (2020) Privacy and robustness in federated learning: attacks and defenses. arXiv preprint arXiv:2012.06337, DOI: 10.48550/arXiv.2012.06337.

Информация об авторах / Information about authors

Валерий Алексеевич КОСТЕНКО – кандидат технических наук, доцент кафедры автоматизации систем вычислительных комплексов (АСВК) факультета Вычислительной математики и кибернетики (ВМК), Московский государственный университет имени М.В. Ломоносова. Сфера научных интересов: теория расписаний, методы машинного обучения, вычислительные системы реального времени, планирование вычислений в центре обработки данных.

Valery Alekseevich KOSTENKO – Cand. Sci. (Tech.), Associate Professor of the Department of Automation of Computer Complex Systems, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University. Research interests: scheduling theory, machine learning methods, real-time computing systems, scheduling calculations in a data center.

Алиса Евгеньевна СЕЛЕЗНЕВА – студентка факультета Вычислительной математики и кибернетики (ВМК), Московский государственный университет имени М.В. Ломоносова. Сфера научных интересов: модели и методы машинного обучения, федеративные нейросети, планирование вычислений в центре обработки данных.

Alisa Evgenievna SELEZNEVA – student of the Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University. Research interests: machine learning models and methods, federated neural networks, scheduling calculations in a data center.

DOI: 10.15514/ISPRAS-2024-36(1)-4

Фаззинг полиморфных систем в структурах микросервисов

А.С. Юрьев, ORCID: 0009-0003-0369-0422 <*forhhpurpose@yandex.ru*>
Институт системного программирования РАН,
109004, Россия, г. Москва, ул. А. Солженицына, д. 25.

Аннотация. Сегодня фаззинг, фаззинг-тестирование является основной техникой тестирования программного обеспечения, систем и функций, в том числе и как часть динамического анализа. Фаззинг позволяет выявлять дефекты информационной безопасности или отказы. Однако такая практика может требовать привлечений больших ресурсов и вычислительных мощностей для проведения работ в крупных организациях, где количество систем может быть большим. Командам разработки и специалистам информационной безопасности требуется одновременно соблюдать сроки, требования различных регуляторов и рекомендации стандартов. Для решения задач по фаззинг-тестированию при одновременном соблюдении сроков, предлагается метод фаззинг-тестирования, который следует применять сразу ко всей информационно-вычислительной сети крупных организаций, которые используют микросервисы. Под полиморфными системами в настоящей статье понимаются такие системы, которые содержат реализацию различных функций, принимающих на вход различные типы данных, не в рамках одного программного обеспечения, а в рамках подсистем с набором нескольких микросервосов. В этом случае могут использоваться различные сетевые протоколы, форматы и типы данных. При таком многообразии особенностей, возникает проблема выявлений дефектов в составе систем, поскольку при разработке не всегда предусматриваются интерфейсы отладки или обратной связи. Для её решения в настоящей статье предлагается использовать метод сбора и анализа статистики временных интервалов обработки мутированных данных миросервисами. Для фаззинг-тестов предлагается использовать мутированные запросы, где начальное состояние данных для мутации – полезная нагрузка известных или типовых дефектов информационной безопасности. С помощью анализа временных интервалов между клиент-серверным запросом и ответом удалось выявить закономерности, которые показали наличие потенциально опасных дефектов. В рамках статьи рассматривается фаззинг прикладных функций по протоколу HTTP. Предлагаемый подход не оказывает отрицательных влияний на эффективность и сроки разработки. Описанный в статье метод и решение рекомендуется применять в крупных организациях, как дополнительное или основное решение по обеспечению информационной безопасности для того, чтобы предотвращать критичные отказы инфраструктуры и финансовые потери.

Ключевые слова: фаззинг; дефекты информационной безопасности; микросервисная архитектура.

Для цитирования: Юрьев А.С. Фаззинг полиморфных систем в структурах микросервисов. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 45–60. DOI: 10.15514/ISPRAS–2024–36(1)–4.

Fuzzing of Polymorphic Systems within Microsevice Structures

A.S. Yurev, ORCID: 0009-0003-0369-0422 <*forhhpurpose@yandex.ru*>
Institute for System Programming of the Russian Academy of Sceinces,
25, Alexander Solzhenitsyn st, Moscow,109004, Russia.

Abstract. Today fuzzing (fuzzing-testing) is the main technique for testing software, systems and code functions. Fuzzing allows identify vulnerabilities or software failures. However, this practice may require the large resources involvement and network performance in large organizations where the number of systems may

be large. Developers and information security specialists are simultaneously required to comply with time-to-market deadlines, requirements of various regulators and recommendations of standards. In current paper is proposed new fuzzing method, which is designed to solve the problem above. In current aproach is proposed use fuzzing testing for whole computing network at ones in large organizations if them operate with microservices. Polymorphic systems in this paper are understood like systems that consist of various API (Application Programming Interface) functions that operate with various types of data, not within single software, but inside subsystems with a set of several microservices. In this case, a lot of various network protocols, data types and formats can be used. With such a variety of features, there is a problem of detecting errors or vulnerabilities inside systems, beacause debugging or trace interfaces are not always developed in the microservice softwares. So, in this paper it is proposed to use also the method of collecting and analyzing statistics of time intervals of processing mutated data by microservices. For fuzzing tests, it is proposed to use mutated lists of exploit payloads. Time analyzing between client-server requests and the responses helps to identify patterns that showed the presence of potentially dangerous vulnerabilities. This paper discribes fuzzing of API functions only in the HTTP protocol (Hypertext Transfer Protocol). Current approach does not have a negative impact on the effectiveness of development or deadlines. Methods and solution described in the paper are recommended to be used in large organizations as an additional or basic information security solution in order to prevent critical infrastructure failures and financial losses.

Keywords: fuzzing; information security; micro-service architecture.

For citation: Yurev A.S. Fuzzing of polymorphic systems within microservice structures. Trudy ISP RAN/Proc. ISP RAS, vol. 36, issue 1, 2024. pp. 45-60 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-41.

1. Введение

Сегодня в крупных организациях часто используют микросервисные принципы построения информационно-вычислительной архитектуры [1]. Микросервисы – это набор небольших модулей, с помощью которых выполняется непрерывная поставка и развертывание больших и сложных приложений (программного обеспечения). Чаще всего, это веб-сервис (Web-service), отвечающий за один элемент логики в определенной предметной области. Приложения на таких принципах представляют из себя комбинацию микросервисов, каждый из которых предоставляет определенные функциональные возможности пользователям [2]. Под полиморфными системами в настоящей статье понимаются такие системы и микросервисы, которые состоят из множеств различных API (Application Programming Interface) функций, принимающих на вход различные типы данных, не в рамках одного программного обеспечения (ПО), а в рамках автоматизированных систем (АС) с набором нескольких ПО. В этом случае могут использоваться различные сетевые протоколы, ролевые модели, форматы и типы данных, что порождает проблемы исследований и аудита информационной безопасности (ИБ) в крупных организациях, поскольку количество объектов и их особенностей может быть большим. В первую очередь, проблемы связаны с тем, что при проведении работ по анализу защищенности специалистами ИБ требуется соблюдать сроки (time-to-market). Например, в финансовых организациях соблюдение сроков является острой проблемой, поскольку рынок финансовых услуг очень динамичный и резко меняется. Кроме того, в соответствии с принципами SSDL (Secure Software Development Lifecycle) [3] и требованиями регуляторов, каждая новая версия разработки ПО обязана проходить проверки ИБ на регулярной основе в рамках аудита защищенности различных АС и для решения задач по анализу отказоустойчивости. Для этого проводится фаззинг-тестирование (англ. Fuzzing) [4]. В рамках настоящей статьи фаззинг-текстирование рассматривается, как один из вариантов практики динамического сканирования – DAST (Dynamic Application Security Testing) [5]. В основе техники фаззинга лежит метод подачи на вход функций ПО некорректных, случайных или нестандартных данных с целью обнаружения ошибок [6-7]. Практика фаззинг-тестирования регламентирована рекомендациями различных стандартов [8-10]. Инструменты фаззинг-тестирования входят в стек технологий цикла безопасной разработки, регламентированного нормативными документами ведомств-регуляторов, проведение таких процедур предусмотрено ГОСТ Р

56939-2016, как необходимой меры разработки безопасного ПО [6,11]. Поэтому сегодня вопрос проведения фаззинг-тестирований в крупных организациях является крайне актуальным. Для выполнения рекомендаций и требований при одновременном соблюдением сроков ввода ПО в промышленную эксплуатацию, в рамках данной статьи предлагается использовать новый метод фаззинга для структур микросервисов сразу ко всей АС, поскольку именно такой подход используется злоумышленниками, в соответствии с моделью внешнего нарушителя [12]. В этом случае, с точки зрения тестирования на проникновение, АС будет атакована по методологии «черного» или «серого ящика» [13], так как нарушителю или не известна архитектура систем в организации, или он каким-то образом получил всё множество API-функций и их параметры. В настоящей статье проводится анализ различных инструментов, применительно, к микросервисным структурам и разрабатывается метод обнаружения дефектов ИБ, основанный на анализе времени ответов сервисов на запросы. В рамках статьи рассматривается фаззинг API-функций в составе протокола HTTP (Hypertext Transfer Protocol), поскольку сегодня микросервисы преимущественно работают по спецификациям Swagger [14] или OpenAPI [15]. Доказывается эффективность метода мутации полезных нагрузок (payloads) [16-17] известных дефектов ИБ на практике. Исследуются проблемы фаззинга АС в крупных организациях.

2. Описание проблемы фаззинг-тестирования в крупных организациях

В крупных организациях, где информационные системы построены на микросервисных принципах, в АС с течением времени разрабатывается множество новых API, системных интеграций, обновлений, постоянно ведутся разработки новых функций, микросервисы создают полиморфизм и их количество может возрастать в разных АС непрерывно и независимо. Часто, каждый сервис в составе АС существует только в рамках своего жизненного цикла (SDLC) [18] и не зависит от разработок в других частях АС. Бизнес-функции, команды разработчиков и стек технологий также могут быть разными.

В соответствии с этим, внедрение фаззинг-тестирований в отдельности для каждого микросервиса или ПО создает предпосылки для снижения эффективности разработки. Это обусловлено тем, что может потребоваться:

1) Проводить предварительный анализ исследуемой системы, определение объектов фаззинга, анализ обновления;
2) Тонко настраивать инструменты фаззинга и вносить в контур разработки дополнительные библиотеки;
3) Создавать агенты фаззинга и доставлять их на сервера разных систем;
4) Настраивать авторизацию в системах;
5) Создавать дополнительных тестовые контура для предотвращения отказов в промышленной среде во время фаззинга;
6) Создавать механизмы контроля состояния систем во время проведения работ;
7) Непрерывно взаимодействовать с командами разработчиков, с отрывом от выполнения основных задач;
8) Расширять штат специалистов ИБ;
9) и выполнять другие активности.

Предполагается, что вышеперечисленные особенности не только могут увеличить сроки, но и потребуют дополнительных ресурсов, которые в свою очередь, увеличивают финансовые расходы.

Кроме того, существует и другая проблема, которая связана с наличием обратной связи в исследуемых системах. Не всегда разработчики предусматривают интерфейсы, которые

позволяют выявлять дефекты ИБ или ошибки. К ним можно отнести системные журналы, логи ошибок (syslog), обработчики ошибок в составе конечного ПО. Иногда, в составе различных сервисов используются коммерческие решения, включая аппаратные (hardware), исходный код которых недоступен разработчикам организации. Некоторые прикладные решения, например, такие как Web-серверы, базы данных или ПО, которые работают по протоколу HTTP, подвержены дефектам ИБ, которые не приводят к отказам, однако уязвимы. В этом случае входные данные воспринимаются функциями ПО, как обычные, стандартные. К таким дефектам ИБ можно отнести SQL-инъекции (Structured Query Language Injections) [19], RCE (Remote Command Execution/Remote Code Execution) [19], LFI/RFI (Remote/Local File Inclusion) [19], IDOR (Insecure Direct Object Reference) [19] и другие. Из-за отсутствия обратной связи и возникает проблема выявления дефектов ИБ. Доработки в реализации обратной связи могут повлечь дополнительные расходы в организациях.

Еще одной проблемой могут быть ограничения операционных систем и их низкая производительность. Множество сервисов могут работать на виртуальных компонентах, например, такие как Docker-контейнеры [20]. Часто, для таких компонентов могут не создавать высоких вычислительных мощностей, а память (RAM, ROM) может быть сильно ограничена. Масштабирование систем влечет за собой увеличение расходов в организациях. Некоторые рудиментарные и устаревшие микросервисы могут работать на старых операционных системах, заменить или обновить, ввиду своей экономической и практической значимости, может быть невозможно из-за особенностей АС. Поэтому предлагается рассматривать вопрос фаззинга не с точки зрения подстройки систем, а с точки зрения настройки инструментов фаззинга для всех систем сразу в совокупности.

Также, еще одной проблемой может стать анализ полноты покрытия объектов фаззинга для качественной оценки проведенных работ. В рамках крупной организации – это может быть сложной и объемной задачей, поскольку это требует дополнительных ресурсов для проведения исследований ИБ применительно к каждому ПО или сервису, где потребуется глубокий разбор исходного кода и привлечение команд разработки.

Для решения вышеописанных проблем в рамках настоящей статьи предлагается проводить фаззинг-тестирование:

1) Сразу ко всей полиморфной микросервисной АС, в соответствии с бизнес-сценариями или функционалом, который доступен пользователю в соответствии с его ролью в системе. Критерием полноты покрытия считать количество задействованных API-функций и их параметров;
2) Использовать метод оценки временных интервалов ответов микросервисов на мутированные запросы, где начальное состояние данных мутации – полезная нагрузка (payloads) известных или типовых дефектов ИБ;
3) Проводить фаззинг-тестирование по методологии «черного ящика» или «серого ящика», преимущественно, используя модель внешнего нарушителя.

Такой подход оказался эффективным, что следует из нижеописанных результатов.

3. Анализ существующих решений для фаззинг-тестирования

В рамках настоящей статьи был проведен анализ существующих решений и методов, на предмет их применимости к микросервисным структурам.

В статье [21] приводится общий анализ различных решений, таких как: APIFuzzer [22], bBOXRT [23], Dredd [24], EvoMasterBB [25], RESTest [26], RESTler [27], RestTestGen [28], Schemathesis [29] и Tcases [30]. Для оценки эффективности существующих методов и инструментов тестирования API-функций, авторы статьи [21] проводили эмпирические наблюдения, в ходе которых применяли указанные инструменты к разным Web-сервисам. На основании полученных результатов авторы [21] отвечают на вопросы: какая полнота

покрытия у инструментов фаззинга и как много ошибок способны обнаружить рассматриваемые решения. Анализ этой работы показал, что сгенерированные параметры, которые подавались на вход функций исследуемых объектов имели в основном случайный характер. В результате этого, множество запросов не было правильно обработано сервисами. Авторы [21] использовали метод выявления 500 ошибок (по спецификации HTTP 500 ошибка – это 500 Internal Server Error, внутреняя ошибка, чаще всего отказ), которые могут свидетельствовать о наличии дефектов. Проведенные работы касались сервисов, которые функционировали локально и позволяли контролировать свое состояние и оценить покрытие [21]. Эти методы и инструменты могут, безусловно, использоваться в крупных организациях, но в своем составе не содержат функции оценки времени ответов. В работе авторов [21] не рассматривались вопросы авторизации в системах, а зависимости между микросервисами не были учтены. Соединения с сервисами устанавливались прямо и независимо друг от друга. Кроме того, не рассматривается последовательность отправки запросов, в соответствии с бизнес-сценариями и ролью пользователей.

В рамках текущей работы был проведен еще и анализ существующих открытых и коммерческих решений.

Были рассмотрены следующие открытые решения: honggfuzz [31], radamsa [32], AFL [33], LibFuzzer [34], oss-fuzz [35], sulley [36], boofuzz [37], Bfuzz [38], ffuf [39], wfuzz [40], nuclei [41]. В настоящей статье не проводится исследований эффективности этих инструментов в сравнении друг с другом, предполагается, что это будет описано в дальнейших работах, однако, каждый из инструментов был протестирован в рамках данной статьи на различных микросервисных АС. Описание и сравнение инструментов можно найти в статье [42], где описан анализ эффективности решений ffuf [39] и wfuzz [40], а также приводится описание большинства из вышеуказанных решений. Временная оценка ответов для вышеприведенных решений может быть произведена с помощью дополнительных настроек или с помощью написания специального ПО. Так, например, для инструмента nuclei время ответов можно получить с помощью следующих параметров запуска: *-ts, -timestamp*.

Указанные открытые решения можно использовать, как инструменты отправки запросов с полезной нагрузкой, например, используя списки мутированных последовательностей для фаззинга API-функций. Это может потребовать предварительной их генерации. Для генераций мутированных данных применительно к инструменту wfuzz, например, можно воспользоваться инструментом radamsa "Листинг 1":

```
for((i=1; i<100; i++));
do echo '<script>alert(1)</script>' | radamsa 1>> payloads.txt;
done && wfuzz -c -t 50 -w payloads.txt -u
https://service/api/v1/send?q=FUZZ
```

Листинг 1. Пример генерации мутированных данных с помощью radamsa для wfuzz
Listing 1. Mutated data generation using radamsa for wfuzz

Решения [31-40] содержат модули генерации случайных данных «на лету», когда в списках мутированных данных нет необходимости. Однако, простая подача на вход функций случайных данных с помощью генераторов не эффективна [7]. Большое множество генераций не позволяет проводить исследования за разумное время.

Для выполнения задач фаззинга может быть использовано ПО для тестирования на проникновение: Burp Suite [43], OWASP ZAP [44]. Однако, такие решения не обладают полноценными фаззинг-модулями и требуют использовать отдельные большие списки сгенерированных фаззинг-нагрузок или подключать дополнительные расширения. Модули детекции временных интервалов присутствуют в функционале этих инструментов, однако в них нет автоматического временного анализа и выявлять отклонения требуется эмпирически (вручную). Возникает дополнительная задача автоматизации процесса.

Существуют коммерческие решения для DAST-сканирований. В рамках текущей статьи не проводились исследования эффективности различных коммерческих решений по отношению друг к другу и в целом, а только приводится несколько примеров таких решений: PT BlackBox [45], Netsparker [46], appScreener [47], Acunetix [48]. Предполагается, что и они могут проводить фаззинг микросервисов. Вышеперечисленные решения могут быть эффективно встроены в процесс CI/CD [49], как DevSecOps [50] решения. Они могут использоваться на этапе разработки, тестирования или сопровождения, в соответствии с методологией SSDL [3]. Несмотря на то, что эти решения можно применять сразу ко всей микросервисной архитектуре, было выявлено, что при применении различных сканеров динамического анализа не обеспечивается полное покрытие АС проверками, поскольку сигнатурно, в составе таких решений, анализ производится в основном тестовыми запросами для поиска известных дефектов ИБ (CVE – Common Vulnerabilities and Exposures) [51]. Лучше всего они решают задачи анализа соответствия требований настроек систем (анализ настроек безопасности, выявление небезопасных функций), задачи по обнаружению устаревших версий ПО, выявлению слабостей используемых протоколов и обнаружению известных (1-day) дефектов ИБ [51]. Некоторые коммерческие решения имеют функции обнаружения DOS, когда задержки ответов сервера велики. Однако, автоматизированного модуля оценки временных интервалов нигде не содержится, а исследования по методологии «серого ящика» требуют созданий правил сканирования, когда необходимо создавать множество шаблонов и осуществлять их контроль, а это, в свою очередь, требует привлечения дополнительных ресурсов. Как правило, при применении коммерческих решений, отсутствует прозрачность в рамках того, какая именно полезная нагрузка была отправлена микросервису, так как производители предпочитают не разглашать свои технологии. Специалисту ИБ, как правило, будет доступен только отчет или электронная форма после проведения сканирований, которые будут не информативны при выполнения указанных задач в рамках фаззинг-тестирования.

В результате проведенного анализа литературы и апробации различных решений, в рамках текущей статьи делается следующий вывод: на текущий момент не существует универсальных решений для фаззинга сразу всех систем (микросервисов) с автоматизированным модулем оценки временных интервалом, контролем тестовой полезной нагрузки, модулем мутации известных дефектов ИБ. Требуется разработать решение, которое бы позволило выполнять все указанные задачи для фаззинг-тестирований.

4. Высокоуровневое описание объекта исследования

Для проведения исследований был использован тестовый стенд информационной сети, где функционировала полиморфная АС, в составе которой работало множество микросервисов (более 500), обрабатывающие различные типы данных с помощью API-функций. Они работали как с бинарными массивами данных, так и с типизированными, такие как JSON (JavaScript Object Notation) или XML (eXtensible Markup Language). При этом в тестовой среде велись непрерывно разработки ПО различными командами. Для проведения исследований был выбран сетевой протокол прикладного уровня – HTTP, как наиболее популярный. Исследуемая система является распределенной. Микросервисы могут быть связаны интеграциями. В качестве примера, на нижеприведенном рис.1, потоки данных показаны соединительными линиями. Расположение линий, как и подсистем может быть любым.

5. Описание процесса фаззинга

В рамках статьи, фаззинг проводился с помощью разработанного решения для указанных задач, которое состоит из модуля мутации, подстановки и детектирования, а процесс был разделен на несколько этапов.

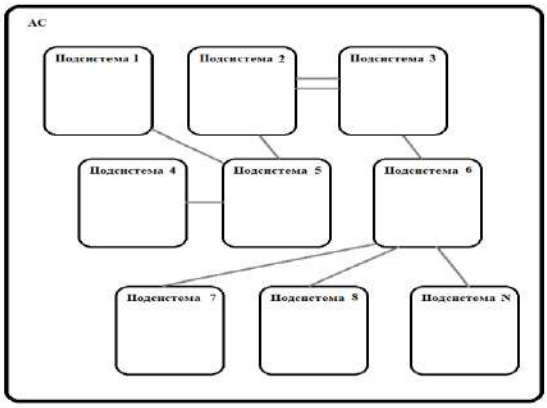


Рис. 1. Высокоуровневая схема полиморфной АС
Fig. 1. High-level scheme of polymorphic system

5.1 Сбор данных об АС и создание шаблонов тестирования

Сбор информации проводится по методологии «черного ящика» или «серого ящика». Результатом этого этапа является набор IP-адресов, DNS (Domain Name System) имен и сформированный список URI (Uniform Resource Identifier) для каждой конечной API функции. Для методологии «серого ящика», кроме того, формируется список всех запросов в рамках АС с параметрами и заголовками (см. Листинг 2).

```
POST /personal/api/v1/personal/setBalance HTTP/1.1
Host: ***
Cookie:
sid=111C5wM2LY4Sfc5LACOTAAAlNLABQtMjE4NzkwNjU1MDc3NjQ2NTk2OQACUzEAAjgz;
Content-Length: 73
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36
Content-type: text/plain
Accept: application/json, text/plain, */*
Connection: close

{"metadata":{"channel":"ib"},"data":{"balanceIsVisible":true},
"balanceSet":7788}


POST /personal/api/v1/personal/getBalance?query=first HTTP/1.1
Host: ***
Cookie: sid=111C5wM2LY4Sfc5LACOTAAAlNLABQtMjE4NzkwNjU1MDcsddsdk2OQACUzEAAjgz;
Content-Length: 93
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36
Content-type: text/plain
Accept: application/json, text/plain, */*
Connection: close

{"metadata":{"channel":"ib"},"data":{"balanceIsVisible":true},
"balanceSet":7788}
```

Листинг 2. Типовые шаблоны HTTP запросов
Listing 2. Typical templates of HTTP requests

Полученные исходные данные сохраняются в соответствующий шаблон в текстовом формате с эталонами всех HTTP-пакетов по спецификации микросервисов, где указываются все параметры в составе пакета для фаззинга. По умолчанию, таковыми являются URI, с разделителем “/”, заголовки протокола HTTP и данные (*data*, параметры и их значения).

5.2 Подготовка типовых тестовых проверок информационной безопасности

На этом этапе создаются списки различных полезных нагрузок для эксплуатации известных дефектов ИБ (CVE), например, такие как команды, выполняемые при работе с базами данных SQL, системные команды Linux, различные инъекции (вредоносные, ненадежные данные для системы, баз данных или кода), ссылки на ресурсы операционной системы в виде системных файлов и другие. Такой подход, как правило, применяется различными DAST-сканерами [43-48]. В итоге, формируются входные параметры в виде списков эталонных данных для тестирования и списка полезных нагрузок для модуля мутации, который является частью решения (см. Листинг 3).

```
%00../../../../../../etc/passwd
%00../../../../../../etc/shadow
-1.0
') or ('x'='x
0 or 1=1
' or 0=0 --
' UNION SELECT
t'exec master..xp_cmdshell 'nslookup localhost'--
%20$(sleep%2050)
<?xml version="1.0" encoding="ISO-8859-1"?><!DOCTYPE foo [<!ELEMENT foo
ANY><!ENTITY xxe SYSTEM "file://c:/boot.ini">]><foo>&xxe;</foo>
{77*88}
```

Листинг 3. Пример списка значений полезных нагрузок для модуля мутации
Listing 3. Payloads examples for mutation

Указанный список полезных нагрузок будет использован в качестве параметра для модуля мутации. В основе работы модуля мутаций используются следующие функции изменения данных:

1) инверсия данных: случайным образом выбираются биты входных данных и их количество, затем над ними производится замена с 0 на 1 и наоборот;
2) сокращение данных или переполнение: сокращается количество переменных для параметров запросов или сокращаются длины передаваемых данных. Для переполнения используется подача на вход данных с длиной, превышающей допустимую или с помощью пошаговой инкременции, где каждое последующее тест-значение увеличивается по длине на 1 бит или 1 байт;
3) преобразуется числовой формат;
4) внесение интервалов: во входные данные добавляются множества пробелов, знаков форматирования, например, переходы на новую строку (“/n”, “%0a”), а также используется техника кодирования – Percent-encoding (добавления знаков % в URI или в других параметрах запросов, например в данных);
5) преобразование форматов: изменяется кодировка или производится сериализация объектов (преобразование объекта (параметра) в поток байтов (битов) для сохранения или передачи в память, базу данных или файл);
6) модификация строковых или числовых значений: строковые значения заменяются на типовые значения для обнаружения дефектов ИБ, при этом, изменяется расположение входных данных случайным образом в запросе. Числовые значения заменяются на отрицательные или на значения с плавающей точкой.
7) расширение данных: производится дублирование данных, повторение параметров запроса, случайное увеличение числа значений для одного и того же параметра;
8) рандомизация данных: на вход параметров запросов подаются случайные последовательности в требуемом формате или с модификациями формата;
9) конкатенация или сдвиг: производится объединение данных или линейный сдвиг на случайную величину.

Типы данных для каждого шаблона микросервисов могут отличатся. Ввиду этого реализована автоматическая подстройка мутатора к требуемому формату данных.

Также, в составе решения, реализован модуль подстановок. В основе модуля лежит алгоритм, который на основании шаблона и полезной нагрузки, генерирует последовательность вызова функций API в составе АС и регламентирует положение сгенерированной последовательности из модуля мутаций, без повторений. Примеры сгенерированных выходных данных для тестирования могут иметь вид, показанный на Листинге 4.

```
{"channel":"0%20'../../../etc/passwd.."}
{"channel":"inform,0+123"}
{"channel":"""""""""""""""}{""}
{"-dl-.echo'1';sleep 9999999":[]}
{"channel":"0%20'../../../etc/passwd.."}
GET /api/v1/service_n?%%%%%%%%sleep%%%2009999%30001
```

Листинг 4. Примеры сгенерированных выходных данных для тестирования
Listing 4. Examples of generated output for testing

Для контроля состояния тестирования отвечает модуль детектирования, который сохраняет полученные результаты фаззинг-тестов, автоматически анализируя следующее:

1) аномальные ответы сервера;
2) обрабатываемые ошибки;
3) интервалы времени между запросом и ответом;
4) статусы ответов;
5) падение (DOS – Denial of Service) подсистемы;
6) наличие типовых дефектов ИБ;
7) сигнатуру ответа.

При проведении фаззинг-тестов какой-либо API-функции, применяется метод вызова связанных цепочек данных, когда выходные мутированные данные снова подаются на вход модуля мутации. Применение такого конечного автомата обеспечивает более полное покрытие [7]. Если в результате какого-либо теста было получено время ответа, которое отличается от предыдущих запросов для исследуемой API-функции, то производится отправка стандартного запроса, без полезных нагрузок, а затем повторяется мутированный запрос, для которого была замечена аномалия. Если отклонение повторяется, то такая нагрузка помечается, как потенциальный дефект ИБ.

Количество мутаций выходных данных задается с помощью параметра глубины мутации (повторений мутации) – *J*. Одна мутация равна одному тесту. Чем больше глубина мутаций и, соответственное им количество тестов, тем большая вероятность обнаружения дефектов. Также можно настроить максимум *J,* чтобы данные параметров не имели слишком случайный характер. Реакция ошибок кода микросервисов для описываемого метода не учитывается из-за ограничений обратной связи. В целом, предлагаемый метод фаззинга можно описать схемой, показанной на рис. 2.

В процессе фаззинга именно модулем детектирования происходит измерение времени ответа для каждого запроса к АС. Собираются все значения временных интервалов для каждой API функции всех подсистем. В ходе анализа такой статистики было замечено, что при отправке к системе некоторых мутированных запросов, содержащих полезную нагрузку, ряд значений времени *T* увеличены по сравнению с другими запросами. Автором статьи была выдвинута гипотеза о том, что такое поведение связано с тем, что микросервисы используют дополнительный ресурс для обработки ответов в виде выполнения системных команд, чтения файлов или выполнения иных математических операций, ввиду чего и возникает временная задержка. Также, автором статьи была выдвинута и другая гипотеза о том, что такое поведение может быть связано с наличием дефектов ИБ в микросервисах.

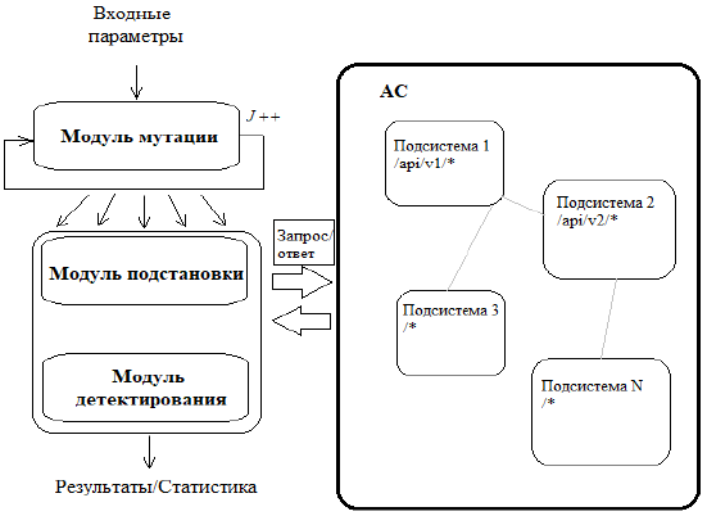


Рис. 2. Высокоуровневая схема процесса фаззинга
Fig. 2. High-level scheme of fuzzing process

Для проверки гипотез в модуле детектирования был реализован следующий алгоритм:

1) для каждой полезной нагрузки в составе передаваемого параметра на API функцию для соответствующей глубины J сохраняется время ответа T системы на запрос;
2) собирается и анализируется статистика задержек запросов/ответов в виде:

$$\Delta TJn = T1 + T2 + T3 + \ldots + Tn\ ;$$

3) если ΔTJn - ΔTJn-1 < 0, то такой результат тестирования в виде сигнатуры HTTP запроса и ответа сохраняется, как гипотетически содержащий дефект.

Вышеприведенная схема была реализована автором в составе программы, написанной на языке Golang, которая не использует в своем составе открытые (Open Source) решения или библиотеки. Это позволило избавиться от зависимостей. Для отправки запросов использовался стандартный пакет – *net/http*. Для достижения целей фаззинг-тестирования в организациях могут использоваться описанные выше открытые, коммерческие решения или могут создаваться собственные инструменты командами разработки, где будут реализованы функции детектирования и анализа времени ответов микросервисов. В рамках описанных задач автором рекомендуется использовать за основу открытое решение nuclei [41], но при этом потребуется разработать модуль анализа полученных результатов с особенностями, применительно к различным организациям и их микросервисам.

6. Анализ полученных результатов и выводы

Результатом работы решения является собранное множество запросов (HTTP-пакетов) и ответов на эти запросы, полученные от АС. Важно отметить, что модуль детектирования не сохраняет одинаковые ответы системы, а только те, которые отличаются от эталонных или те, которые не были обнаружены ранее. Отдельно сохраняются запросы, для которых время ответов было аномальным (большим). Такой подход позволяет сократить размер результата-листинга, который может быть большим в зависимости от глубины *J*. Происходит сравнение каждого нового ответа с каждым сохраненным, пока не будет получен уникальный результат. В качестве параметра сохранения результата, для обеспечения скорости обработки данных, была выбрана в том числе и длина ответа. Такие методы фильтрации результатов обеспечивают минимизацию ложноположительных тестов. В целом результаты проведенных

работ собраны в табл. 1, полученной при тестировании микросервисной АС в тестовой среде (см. раздел 4).

Табл. 1. Результаты проведенных фаззинг-тестов
Table1. The results of the fuzzing tests

Количество микросервисов	Количество API функций	Глубина мутации	Время фаззинг-тестирования	Количество уникальных ответов	Количество уникальных временных задержек
1	5	10	~ 5 минут	1	1
10	93	30	~ 60 минут	2	2
100	1068	20	~ 500 минут	11	13
200	3015	10	~ 3000 минут	37	23

Конечно, приведенные в табл. 1 данные субъективны и значения будут сильно зависеть от информационной архитектуры различных организаций. Поэтому при применении описываемого подхода, требуется эмпирически рассчитать среднее время обработки запросов и разработать критерии анализа уникальных ответов. Иногда, возникают случаи, когда микросервис из-за обновлений, влияния других процессов, высокой нагрузки может выдавать ложноположительные результаты, в виде ошибок (500 статус) или в виде высоких временных задержек на ответы, поэтому все подобные результаты фаззинг-тестирования проходят повторное сканирование. После проведения работы для исследуемой АС, набор гипотетических дефектов анализируется в рамках одного микросервиса, где применяется такой же набор входных параметров на этапе *J*, при которых была обнаружена задержка: проводится направленный повторный одиночный тест для того, чтобы опровергнуть или подтвердить наличие дефектов.

В результате проведения исследования, было получено, что временные задержки – действительно обусловлены тем, что в ряде систем содержатся потенциально опасные дефекты ИБ. Поэтому возникают отказы или время ответа возрастает по сравнению с другими ответами. Несколько таких тестовых примеров, которые действительно являлись на момент исследования дефектами ИБ, приведены в табл. 2. Конечно, по этическим соображениям, в рамках статьи, некоторые данные об объекте исследования были изменены.

Табл. 2. Примеры запросов с полезной нагрузкой, ответов и время ответов
Table 2. Requests examples with payloads, answers and response time

№	Пример HTTP запроса (тест)	Пример ответа на запрос	Время ответа (миллисекунды)
1	GET /aa/settings/?sender=user&pl=web HTTP/2 Host: app.mobile.xxxxxx.com:8899 Content-Type: text/plain; charset=utf-8 Content-Length: 50 {"data":"input", "command": "; /n ping 127.0.0.1"}	HTTP/2 200 OK Server: nginx/1.20.2 Date: Tue, 17 Oct 2023 20:18:30 GMT Content-Type: application/json Content-Length: 91 {"data":"message", "print": "Ошибка! Проверьте входные данные."}	6413 мс
2	POST /api/v2/change/?%20%0a'%2d%2dselect%20*from%20users HTTP/2 Host: app.mobile.xxxxxx.com:8899	HTTP/1.1 500 Internal Server Error Server: Apache/2.4.25	5422 мс
3	POST /s1/trace?start=%7dAAAAAAAA%7b HTTP/2 Host: app.mobile.xxxxxx.com:8899	Нет ответа	-

На рис. 3, рис. 4 и рис. 5 приведен вывод результатов для запросов 1, 2 и 3 из табл. 2, соответственно. Как видно из рисунков, ответы на запросы, которые не содержали дефектов, имеют меньшие величины времени ответа *T*. Также следует заметить, что статус или текст

ответа микросервисов, не всегда является корректным и явно не свидетельствует о наличии в АС какого-либо дефекта ИБ.

Рис. 3. Пример № 1 обнаружения дефекта ИБ среди множества запросов
Fig. 3. Example No. 1 of detecting a defect among many requests

Рис. 4. Пример № 2 обнаружения дефекта ИБ среди множества запросов
Fig. 4. Example No. 2 of detecting a defect among many requests

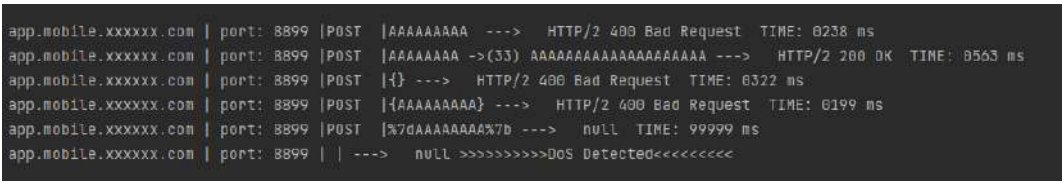


Рис. 5. Пример № 3 обнаружения дефекта ИБ среди множества запросов
Fig. 5. Example No. 3 of detecting a defect among many requests

В результате, было установлено, что для каждого типа дефекта ИБ соответствуют разные ΔT. Например, для дефектов типа DoS – ΔT, как правило, принимает самое большое значение, а самые низкие показатели ΔT обнаруживались для дефектов типа IDOR (Insecure direct object reference).

При применении разработанного решения фаззинг-тестирования для исследуемой микросервисной АС было получено следующее:

- отсутствует большая нагрузка на тестируемые микросервисы;
- эффективность разработки и время не снижается, несмотря на то что на тестовом контуре постоянно проводились работы различными командами;
- обеспечивается высокое покрытие микросервисов, поскольку используется универсальный шаблон, применяемый ко всей АС сразу, где исследуется каждая API функция по спецификации самого сервиса;
- автоматизированно производится анализ ответов сервера и их временных интервалов для каждой сгенерированной последовательности, переданной на вход API функции, что сокращает время при ручном (эмпирическом) анализе;
- производится автоматическая настройка в соответствии с протоколом сервера и типом обрабатываемой информации, что позволяет производить проверки непрерывно, в том числе в рамках процесса CI/CD [49];
- нет привязки к формату данных, а тип данных определяется на основании анализа данных из шаблонов запросов/ответов на всех этапах тестирования, что позволяет не тратить время и ресурсы на запуск инструментов фаззинга применительно к каждому микросервису в отдельности.

7. Заключение

В рамках настоящей статьи было установлено, что различные API в структурах микросервисов, могут содержать различные типы дефектов ИБ и ошибок. Для их обнаружения необходимо проводить фаззинг-тестирование, в соответствии с требованиями регуляторов и рекомендациями стандартов. При проведении фаззинг-тестирований отдельно, для каждого микросервиса могут возникать проблемы, связанные с соблюдением сроков разработки и снижением эффективности в крупных организациях. Описанный в настоящей статье метод фаззинг-тестирования для выявления дефектов ИБ, основанный на оценке временных интервалов ответов различных микросервисов, подаче мутированных данных полезных нагрузок на вход API-функций, при использовании методологий «серого» или «черного ящика» - оказался эффективным. При таком подходе нарушений сроков разработки не выявлено. Также, оказалось эффективным и применение процедуры фаззинга сразу ко всей микросервисной структуре в рамках АС, где обеспечивается полное покрытие всех доступных API-функций. Предлагаемый метод и решение сегодня применяется, как один из вариантов проведения фаззинг-тестирования в крупной организации [52]. Сам метод позволил выявить следующие типы дефектов ИБ:

- RCE (Remote Command Execution/Remote Code Execution);
- LFI/RFI (Remote/Local file inclusion);
- SQL-инъекции;
- DoS (Denial of Service);
- IDOR (Insecure direct object reference);
- XXE (external entity injection);
- ошибки логики и ряд других.

Описанный в статье метод рекомендуется применять в крупных организациях, где используется микросервисная структура, как дополнительное или основное решение по обеспечению информационной безопасности для того, чтобы предотвращать критичные отказы инфраструктуры и финансовые потери.

Список литературы / Reterences

[1]. Ниньо-Мартинес В., Очаран-Эрнандес Х., Лимон К., Перес-Арригата Х. Развертывание микросервисов. Труды Института системного программирования РАН. 2023;35(1):57-72. DOI: 10.15514/ISPRAS-2023-35(1)-4.

[2]. Вальдивия Х., Лора-Гонсалес А., Лимон К., Кортес-Вердин К., Очаран-Эрнандес Х. Паттерны микросервисной архитектуры: многопрофильный обзор литературы. Труды Института системного программирования РАН. 2021;33(1):81-96. DOI: 10.15514/ISPRAS-2021-33(1)-6.

[3]. Umeugo, Wisdom. (2023). Secure software development lifecycle: a case for adoption in software smes. International Journal of Advanced Research in Computer Science. 14. 5-12. 10.26483/ijarcs.v14i1.6949.

[4]. Li J., Li J., Zhao B., Zhang C. Fuzzing: a survey // Cybersecurity, 2018, Vol. 1, No 1, p. 6, DOI: 10.1186/s42400-018-0002-y.

[5]. Методика динамического сканирования приложений. DAST. Available at: https://owasp.org/www-project-devsecops-guideline/latest/02b-Dynamic-Application-Security-Testing, accessed 04.01.2024.

[6]. Шарков И.В., Падарян В.А., Хенкин П.В. Об особенностях фаззинг-тестирования сетевых интерфейсов в условиях отсутствия исходных текстов. Труды Института системного программирования РАН. 2021;33(4):211-226. DOI: 10.15514/ISPRAS-2021-33(4)-15.

[7]. Саргсян С.С., Варданян В.Г., Акопян Д.А., Агабалян А.М., Меграбян М.С., Курмангалеев Ш.Ф., Герасимов А.Ю., Ермаков М.К., Вартанов С.П. Платформа автоматического фаззинга программного интерфейса приложений. Труды Института системного программирования РАН. 2020;32(2):161-173. DOI: 10.15514/ISPRAS-2020-32(2)-13.
[8]. ISA/IEC 62443-4-1. Available at: https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards, accessed 04.01.2024.
[9]. ISO/IEC/IEEE 291119. Available at: https://cdn.standards.iteh.ai/samples/81291/6694557ff8304df8841bb191a00ecc6f/ISO-IEC-IEEE-29119-1-2022.pdf, accessed 04.01.2024.
[10]. ISO 27001. Available at: https://www.iso.org/standard/27001, accessed 04.01.2024.
[11]. ГОСТ Р 56939-2016. Разработка безопасного программного обеспечения. Общие требования. Дата введения 2017-06-01.
[12]. Методический документ. "Методика оценки угроз безопасности информации". Утвержден ФСТЭК России. Москва. 5 февраля 2021 г.
[13]. ГОСТ Р 58143-2018. Информационная технология. Методы и средства обеспечения безопасности. Детализация анализа уязвимостей программного обеспечения в соответствии с ГОСТ Р ИСО/МЭК 15408 и ГОСТ Р ИСО/МЭК 18045. Часть 2. Тестирование проникновения. ОКС 35.020. Дата введения 2018-11-01.
[14]. The Swagger API project. Apache License 2.0. Available at: https://swagger.io, accessed 04.01.2024.
[15]. OpenAPI Specification v3.1.0. Published 15 February 2021. Available at: https://spec.openapis.org/oas/latest.html, accessed 04.01.2024.
[16]. OWASP Top-10. 2024. Available at: https://owasp.org/www-project-top-ten/, accessed 04.01.2024.
[17]. OWASP. Fuzz Vectors. 2024. Available at: https://owasp.org/www-project-web-security-testing-guide/stable/6-Appendix/C-Fuzz_Vectors, accessed 04.01.2024.
[18]. Software Development Life Cycle (SDLC) Methodologies for Information Systems Project Management - Mohammad Ikbal Hossain - IJFMR Volume 5, Issue 5, September-October 2023. DOI: 10.36948/ijfmr.2023.v05i05.6223.
[19]. Payloads All The Things. Web Application Security, Pentest and Red Team Cheatsheet. 2023. Available at: https://swisskyrepo.github.io/Payloads-AllTheThings/, accessed 04.01.2024.
[20]. Docker. Available at: https://www.docker.com/, accessed 04.01.2024.
[21]. Myeongsoo Kim, Qi Xin, Saurabh Sinha, and Alessandro Orso. 2022. Automated Test Generation for REST APIs: No Time to Rest Yet. In Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA '22), July 18–22, 2022, Virtual, South Korea. ACM, New York, NY, USA, 13 pages. DOI: 10.1145/3533767.3534401.
[22]. APIFuzzer. Available at: https://github.com/KissPeter/APIFuzzer, accessed 04.01.2024.
[23]. Laranjeiro, Nuno & Agnelo, João & Bernardino, Jorge. (2021). A Black Box Tool for Robustness Testing of REST Services. IEEE Access. PP. 1-1. DOI: 10.1109/ACCESS.2021.3056505.
[24]. Dredd. Available at: https://github.com/apiaryio/dredd, accessed 04.01.2024.
[25]. Andrea Arcuri. 2020. Automated Black-and White-Box Testing of RESTful APIs With EvoMaster. IEEE Software 38, 3 (2020), 72–78. DOI: https://doi.org/10.1145/3293455.
[26]. Martin-Lopez, Alberto & Segura, Sergio & Ruiz-Cortés, Antonio. (2020). RESTest: Black-Box Constraint-Based Testing of RESTful Web APIs. 459-475. DOI: 10.1007/978-3-030-65310-1_33.
[27]. Vaggelis Atlidakis, Patrice Godefroid, and Marina Polishchuk. 2019. Restler: Stateful rest api fuzzing. In 2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE). IEEE, Montreal, QC, Canada, 748–758. DOI: 10.1109/ICSE.2019.00083.
[28]. Emanuele Viglianisi, Michael Dallago, and Mariano Ceccato. 2020. RestTestGen: automated black-box testing of RESTful APIs. In 2020 IEEE 13th International Conference on Software

Testing, Validation and Verification (ICST). IEEE, 142–152. DOI: 10.1109/ICST46399.2020.00024.
[29]. Zac Hatfield-Dodds and Dmitry Dygalo. 2021. Deriving Semantics-Aware Fuzzers from Web API Schemas. arXiv preprint arXiv:2112.10328 (2021). Available at: https://www.researchgate.net/publication/357202018_Deriving_Semantics-Aware_Fuzzers_from_Web_API_Schemas.
[30]. tcases REST API tool. Available at: https://github.com/Cornutum/tcases/tree/master/tcases-openapi, accessed 02.01.2024.
[31]. honggfuzz. Available at: https://honggfuzz.dev/, accessed 02.01.2024.
[32]. radamsa. Available at: https://gitlab.com/akihe/radamsa, accessed 02.01.2024.
[33]. AFL. Available at: https://github.com/google/AFL, accessed 02.01.2024.
[34]. LibFuzzer. Available at: https://llvm.org/docs/LibFuzzer.html, accessed 02.01.2024.
[35]. oss-fuzz. Available at: https://github.com/google/oss-fuzz, accessed 02.01.2024.
[36]. sulley. Available at: https://github.com/OpenRCE/sulley, accessed 02.01.2024.
[37]. boofuzz. Available at: https://github.com/jtpereyda/boofuzz, accessed 02.01.2024.
[38]. Bfuzz. Available at: https://github.com/RootUp/Bfuzz, accessed 02.01.2024.
[39]. ffuf. Available at: https://github.com/ffuf/ffuf, accessed 02.01.2024.
[40]. wfuzz. Available at: https://github.com/xmendez/wfuzz, accessed 02.01.2024.
[41]. nuclei. Available at: https://github.com/projectdiscovery/nuclei, accessed 02.01.2024.
[42]. Matheos Mattsson 40476. Master Thesis in Computer Engineering. Supervisor: Dragos Truscan. Faculty of Science and Engineering. Åbo Akademi University. 2021. A comparison of FFUF and Wfuzz for fuzz testing web applications. Available at: https://www.doria.fi/bitstream/handle/10024/181265/mattsson_matheos.pdf, accessed 07.01.2024.
[43]. Burp Suite. Available at: https://portswigger.net/burp/pro, accessed 05.01.2024.
[44]. OWASP ZAP. Available at: https://www.zaproxy.org/, accessed 05.01.2024.
[45]. PT BlackBox. Available at: https://www.ptsecurity.com/ru-ru/products/blackbox/, accessed 02.01.2024.
[46]. Netsparker. Available at: https://github.com/netsparker, accessed 03.01.2024.
[47]. appScreener. Available at: https://rt-solar.ru/products/solar_appscreener/, accessed 02.01.2024.
[48]. Acunetix. Available at: https://www.acceron.net/index.php/products/acunetix, accessed 02.01.2024.
[49]. Jeffrey Fairbanks, Akshharaa Tharigonda, Nasir U. Eisty. Analyzing the Effects of CI/CD on Open Source Repositories in GitHub and GitLab. 2023. https://doi.org/10.48550/arXiv.2303.16393.
[50]. Myrbakken, Håvard & Colomo-Palacios, Ricardo. (2017). DevSecOps: A Multivocal Literature Review. 17-29. DOI: 10.1007/978-3-319-67383-7_2.
[51]. База данных общеизвестных уязвимостей информационной безопасности. Available at: https://cve.mitre.org/, accessed 03.01.2024.
[52]. АО Газпромбанк, https://www.gazprombank.ru/, accessed 03.01.2024.

Информация об авторах / Information about authors

Артемий Сергеевич ЮРЬЕВ – исполнительный директор, департамент развития технологий защиты информации, АО Газпромбанк, аспирант ИСП РАН. Научные интересы: информационная безопасность, фаззинг информационных систем, анализ защищенности, тестирование на проникновение, динамическое сканирование, безопасная разработка.

Artemiy Sergeevich YUREV – executive director, Department of Development Information Security Technologies, Gazprombank (JSC), postgraduate student of ISP RAS. Research interests: information security, fuzzing of information systems, security analysis, penetration testing, DAST, SSDL.

DOI: 10.15514/ISPRAS-2024-36(1)-5

Применение энтропии для обнаружения ошибок модулярного кода в системах надежного распределенного хранения

[1,2] *В.А. Кучуков, ORCID: 0000-0002-1839-2765 <vkuchukov@ncfu.ru>*

[1] *Северо-Кавказский центр математических исследований,*
Северо-Кавказский федеральный университет,
355017, Россия, г. Ставрополь, ул. Пушкина, 1.
[2] *Институт системного программирования РАН им. В.П. Иванникова,*
109004, Россия, г. Москва, ул. А. Солженицына, д. 25.

Аннотация. В статье рассмотрена проблема обнаружения и локализации ошибок модулярного кода. Рассмотрено применение энтропии для обнаружения ошибок полиномиальной системы классов вычетов, исправление ошибок в которой осуществляется методом наибольшего правдоподобия. Для системы остаточных классов предложен подход к обнаружению ошибок через энтропию, позволяющий обнаружить ошибки большей кратности, по сравнению с классическим подходом. Для исправления ошибок рассмотрены методы наибольшего правдоподобия и метод проекций. Введенные ограничения на избыточное основание СОК позволило обнаружить не только все одиночные ошибки по рабочим модулям, но также и ряд ошибок по двум основаниям. Предложена система надежного распределенного хранения, позволяющая обнаружить и исправить ошибки, возникающие при приеме данных из облаков.

Ключевые слова: система остаточных классов; полиномиальная система классов вычетов; энтропия; обнаружение ошибок.

Для цитирования: Кучуков В.А. Применение энтропии для обнаружения ошибок модулярного кода в системах надежного распределенного хранения. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 61–72. DOI: 10.15514/ISPRAS–2024–36(1)–5.

Благодарности: Работа выполнена при поддержке Российского научного фонда 19-71-10033, https://rscf.ru/project/19-71-10033/.

Application of Entropy for Modular Code Error Detection in Reliable Distributed Storage Systems

[1,2] *V.A. Kuchukov ORCID: 0000-0002-1839-2765 <vkuchukov@ncfu.ru>*

[1] *North-Caucasus Center for Mathematical Research, North-Caucasus Federal University,*
1, Pushkin st., Stavropol, 355017, Russia.
[2] *Ivannikov Institute for System Programming of the Russian Academy of Sciences,*
25, Alexander Solzhenitsyn st., Moscow, 109004, Russia.

Abstract. The paper considers the problem of error detection and localization of modular code. The polynomial residue number system represents the input number as a set of polynomials over the finite field GF(2^m), which are residues from dividing the original polynomial by a set of irreducible polynomials. The introduction of redundant moduli provides the required corrective capability of the noise-tolerant code. The application of entropy for error detection of a polynomial residue number system, error correction of which is performed by

the maximum likelihood method, is considered. In the residue number system, a number is represented as residues from division by a set of mutually prime numbers. An approach to error detection through entropy is proposed for the residue number system, which allows to detect errors of higher multiplicity compared to the classical approach. The maximum likelihood and projection methods are considered for error correction. The introduced constraints on the control modulo allowed us to detect not only all single errors on working moduli, but also a number of errors on two moduli. A computational experiment was carried out to investigate the corrective abilities for three sets of moduli {3, 5, 7, 8}, {3, 5, 7, 37}, {3, 5, 7, 71}. A reliable distributed storage system is proposed to detect and correct errors that occur when data is ingested from clouds.

Keywords: residue number system; polynomial residue number system; entropy; error detection.

For citation: Kuchukov V.A. Application of entropy for modular code error detection in reliable distributed storage systems. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 61-72 (in Russian). DOI: 10.15514/ISPRAS-2024-36(1)-5.

Acknowledgements. This work was supported by the Russian Science Foundation 19-71-10033, https://rscf.ru/project/19-71-10033/.

1. Введение

Для многих удаленных инфраструктурных объектов, таких как системы контроля на железной дороге или нефтеперерабатывающей промышленности, важно обеспечение надежной работы и гарантированной достоверности получаемых результатов. При этом для каналов связи вероятность однократных ошибок значительно выше вероятности ошибок большей кратности [1-2]. Таким образом, актуальной является проблема использования корректирующих кодов, позволяющих обнаружить и исправить одиночную ошибку. При этом использование корректирующих кодов связано с проблемой, что зачастую информация в них строго делится на информационную и контрольную, при этом контрольные цифры числа не позволяют производить вычисления над избыточными числами [3]. К кодам, в которых информационная и контрольная части числа равноценны, относятся полиномиальная система классов вычетов и система остаточных классов. Для обеспечения возможности определения ошибок рассмотрим понятие энтропии.

Рассмотрим подход к определению «количества информации», рассмотренный в статье [4]. Если в некотором множестве X, состоящем из N элементов, задана переменная x, то энтропия переменной x равна

$$H(x) = \log_2 N.$$

При этом для передачи количества информации I необходимо употреблять $\lceil \log_2 N \rceil$ двоичных символов. Например, для кодирования в двоичной системе счисления чисел от 0 до 5 необходимо $\lceil \log_2 6 \rceil = 3$ двоичных знака.

Если переменные $x_1, x_2, \dots, x_n$ принимают значения из множеств, состоящих из $N_1, N_2, \dots, N_n$ элементов соответственно, то

$$H(x_1, x_2, \dots, x_n) = H(x_1) + H(x_2) + \cdots + H(x_k).$$

Рассмотрим применение данного понятия для обнаружения и исправления ошибок полиномиальной системой классов вычетов и системой остаточных классов.

Далее статья организована следующим образом. В разделе 2 рассмотрена полиномиальная система классов вычетов с возможностью обнаружения и коррекции ошибки на основе энтропии и метода наибольшего правдоподобия. В разделе 3 рассмотрены корректирующие свойства избыточной системы остаточных классов. В разделе 4 рассмотрен вычислительный эксперимент, описывающий особенности обнаружения и исправления ошибок с избыточным модулем специального вида. В разделе 5 рассмотрена модель системы надежного распределенного хранения в системе остаточных классов. В заключении обобщены полученные результаты и даны рекомендации по использованию корректирующих модулярных кодов.

2. Корректирующие свойства полиномиальной системы классов вычетов

Одним из способов обеспечения надежности при хранении информации в облаках является использование при разделении данных схем разделения секрета (СРС), в частности взвешенных СРС [5], в которых данные делятся на доли разного размера, зависящего от надежности хранилища. Исходные данные могут быть восстановлены в случае некорректности одной или нескольких долей.

В качестве схемы разделения секрета может быть взята избыточная полиномиальная система классов вычетов (ПСКВ, PRNS). Возьмем исходные данные A, которые могут быть представлены в виде многочлена $A_P(x)$ над полем $GF(2^m)$ в виде $A_P(x) = \sum_{i=1}^{m-1} f_i \cdot x^i$, где $f_i \in \{0,1\}$ [5].

Избыточная ПСКВ представляет собой набор из n неприводимых многочленов $m_1(x), m_2(x), \ldots, m_n(x)$, где n – общее количество модулей ПСКВ, r – количество избыточных модулей ПСКВ, тогда $k = n - r$ – количество рабочих модулей, при этом d_i – степень многочлена $m_i(x)$. Тогда любой многочлен $A_P(x)$ представляется в виде остатков от деления на модули ПСКВ, т.е. $A_{PRNS} = \big(a_1(x), a_2(x), \ldots, a_n(x)\big)$. При этом $M(x) = \prod_{i=1}^{k} m_i(x)$ – рабочий диапазон $\big[0, M(x)\big)$ со степенью $D = \deg M(x) = \sum_{i=1}^{k} d_i$, а $\widehat{M}(x) = \prod_{i=1}^{n} m_i(x)$ – полный диапазон $\left[0, \widehat{M}(x)\right)$ со степенью $\widehat{D} = \deg \widehat{M}(x) = \sum_{i=1}^{n} d_i$. Если $\deg A_P(x) < D$, то представление $A_P(x)$ единственное.

За счет вводимых избыточных модулей может быть проведено определение ошибки вычисления или передачи данных. Если полученный при восстановлении многочлен удовлетворяет выражению $\deg A_P(x) < D$, то многочлен корректен или содержит более сложную комбинацию ошибок, чем способен распознать избыточный код, если же $\deg A_P(x) \geq D$, то многочлен содержит ошибку. Таким образом, ошибка может быть обнаружена, если многочлен имеет степень большую, либо равную D, но меньшую $\widehat{D}$.

Одним из подходов к восстановлению многочлена из его представления в виде остатков является вариант Китайской теоремы об остатках для многочленов, а именно

$$A_P(x) = \left| \sum_{i=1}^{n} a_i(x) \cdot \left|\widehat{M}_i^{-1}(x)\right|_{m_i(x)} \cdot \widehat{M}_i(x) \right|_{\widehat{M}(x)},$$

где $\widehat{M}_i(x) = \widehat{M}(x)/m_i(x)$ и $\left|\widehat{M}_i^{-1}(x)\right|_{m_i(x)}$ – мультипликативные инверсии, для которых выполняется $\left|\widehat{M}_i^{-1}(x)\right|_{m_i(x)} \cdot \widehat{M}_i(x) \bmod m_i(x) = 1$.

Рассмотрим вопрос избыточности данных. Многочлен, входящий в рабочий диапазон, имеет степень, меньшую D, таким образом можно сказать, что на представление исходного числа требуется D бит. С учетом избыточных модулей степень многочлена в избыточной ПСКВ ограничена $\widehat{D}$. Тогда в случае сбалансированной системы все остатки используют $d_1 = d_2 = \cdots = d_n = d$ бит и избыточность равна

$$R = \frac{\widehat{D}}{D} - 1 = \frac{n \cdot d}{k \cdot d} - 1 = \frac{n-k}{k}.$$

Многочлен, содержащий ошибку может быть представлен в виде $\overline{A}_{PRNS} = A_{PRNS} + E_{PRNS}$, при этом ошибка имеет форму $E(x) = \beta(x)\widehat{M}_I(x)$, где $\beta(x)$ – ненулевой многочлен, $\widehat{M}_I(x) = \prod_{i \in I} m_i(x)$ и I – множество остатков без ошибки [5].

В статье [5] также введено понятие энтропии для полиномиальной системы классов вычетов. Поскольку степень $A_P(x)$ меньше D, то $A_P(x)$ может принимать 2^D различных значения. Тогда, согласно [4], энтропия $A_P(x)$ будет равна

$$H\big(A_P(x)\big) = \log_2 2^D = D = \sum_{i=1}^{k} d_i.$$

Если $i \in [1,n]$ и $a_i(x) = |A_P(x)|_{m_i(x)}$, то энтропия $a_i(x)$ равна d_i:

$$H\big(a_i(x)\big) = d_i.$$

Таким образом, остатки $a_i(x)$ содержат некоторую информацию об $A_P(x)$. Если $H\big(a_i(x)\big) = 0$, то остаток не содержит информации о $A_P(x)$, если же энтропия равна D, то имеется полная информация о $A_P(x)$. Если количество известной информации больше или равно исходной, то выполняется выражение

$$\sum_{i\in I} d_i \geq \sum_{i=1}^{k} d_i.$$

Таким образом, мы можем безошибочно восстановить $A_P(x)$.

В [5] приведена следующая теорема об обнаружении ошибки.

Теорема 1. Если задана полиномиальная система классов вычетов с n модулями $m_1(x)$, $m_2(x)$, …, $m_n(x)$, где r – количество избыточных модулей, и $k = n - r$ – количество рабочих модулей, то для многочлена $A_P(x)$, представленного в виде остатков $\overline{A}_{PRNS} = \big(\overline{a}_1(x), \overline{a}_2(x), \dots, \overline{a}_n(x)\big)$ и множества $\overline{I}$ остатков с ошибкой, ошибка может быть обнаружена, если выполняется условие

$$\sum_{i\in\overline{I}} d_i \leq \sum_{i=1}^{r} d_{k+i}.$$

Таким образом, для обнаружения ошибки используется Теорема 1. Для локализации и исправления ошибки в статье [5] модифицирован метод декодирования с максимальным правдоподобием (MLD) [6]. Для исправления ошибки множество остатков без ошибок должно удовлетворять условию $\sum_{i\in I} d_i > D$. В процессе локализации и исправления ошибок находится множество V возможных кандидатов $A_P(x)$, удовлетворяющих условию $\deg A_P\,(x) < D$. Каждый из возможных $A_P(x)$ обозначается $V_P^l(x)$, т.е:

$$V = \big\{V_P^1(x), V_P^2(x), \dots, V_P^\lambda(x)\big\},$$

где λ – количество кандидатов, попадающих в разрешенный диапазон.

Поскольку в общем случае ПСКВ относится к взвешенным кодам коррекции ошибок, для которых $d_1 \neq d_2 \neq \cdots \neq d_n$, то расстояние Хэмминга H_T^l, которое определяется как количество элементов, по которым различаются два вектора V_{PRNS}^l и $\overline{A}_{PRNS}$, не обеспечивает корректной оценки, поскольку остатки несут разное количество информации об $A_P(x)$.

Для вычисления веса Хэмминга H_W^l кандидата $V_{PRNS} = \big(v_1(x), v_2(x), \dots, v_n(x)\big)$ в случае получения значения с ошибкой $\overline{A}_{PRNS} = \big(\overline{a}_1(x), \overline{a}_2(x), \dots, \overline{a}_n(x)\big)$ в статье [5] предложен алгоритм согласно которому вычисление H_W^l проходит в три этапа: на первом этапе вычисляется вектор Хэмминга $h = (h_1, h_2, \dots, h_n)$, где $h_i = 0$ когда $v_i(x) = \overline{a}_i(x)$, иначе $h_i = 1$. На втором шаге вычисляется обратная величина вектора h, $\overline{h} = \big(\overline{h}_1, \overline{h}_2, \dots, \overline{h}_n\big)$, где $\overline{h}_i$ равно единице, если остаток $a_i(x)$ не содержит ошибки, в противном случае $\overline{h}_i$ равно нулю. Третьим шагом является вычисление величины энтропии H_W^l как поэлементное произведение двух векторов: $\overline{h} = \big(\overline{h}_1, \overline{h}_2, \dots, \overline{h}_n\big)$ и вектора, состоящего из энтропий остатков от деления, $\overline{a}_i(x)$, т.е. $H_W^l = \overline{h} \cdot h_E$, где $h_E = \Big(H\big(\overline{a}_1(x)\big), H\big(\overline{a}_2(x)\big), \dots, H\big(\overline{a}_n(x)\big)\Big) = (d_1, d_2, \dots, d_n)$.

Пример 1. Рассмотрим к качестве полиномиальной системы классов вычетов набор модулей $\{x^2+x+1, x^3+x+1, x^3+x^2+1, x^6+x+1\}$ для которого $h_E=(2,3,3,6)$, рабочий диапазон $M(x)=x^8+x^6+x^5+x^4+x^3+x^2+1$, полный диапазон $\widehat{M}=x^{14}+x^{12}+x^{11}+x^{10}+x^7+x^6+x^2+x+1$. Избыточность данной системы равна $R=\widehat{D}/D-1=14/8-1=3/4$. В статье [5] в примере 4 указано, что кандидаты должны удовлетворять условию $\deg V_P^l(x)<D=\sum_{i=1}^{k} d_i$.

Введем в многочлен $A(x)=x^7+x^5+x^3+x+1=(x+1, x^2+x, x^2, x^5+x^3+x^2+1)$ ошибку по второму модулю $E=(0,1,0,0)$, получим многочлен $\overline{A}_{PRNS}(x)=(x+1, x^2+x+1, x^2, x^5+x^3+x^2+1)=x^{12}+x^8+x^7+x^6+x^5+1$. Поскольку $\deg \overline{A}_{PRNS}(x)>D$, многочлен содержит ошибку.

В соответствии с теоремой 1 могут быть обнаружены все одиночные ошибки, а также двойные ошибки по первому и второму, первому и третьему, второму и третьему модулям.

Тогда кандидатами будут

- $V_1(x)=x^7+x^5+x^3+x+1=(x+1, x^2+x, x^2, x^5+x^3+x^2+1)$ для которого $H_T=1$, $H_W=11$;
- $V_2(x)=x^6+x=(x+1, x^2+x+1, x^2, 1)$ для которого $H_T=1$, $H_W=8$;
- $V_3(x)=x^6+x^5+x^3+x^2+x=(x, x^2+x+1, x^2+x, x^5+x^3+x^2+1)$ для которого $H_T=2$, $H_W=9$.

В соответствии с методом наибольшего правдоподобия максимальный вес Хэмминга $H_W=11$ имеет кандидат $V_1(x)$, который соответствует числу без ошибки.

В общем случае добавление одного избыточного модуля позволяет только обнаружить одиночную ошибку, однако подход с использованием энтропии для полиномиальной системы классов вычетов позволил обнаружить и ряд двойных ошибок, а метод наибольшего правдоподобия позволяет исправить данную ошибку.

Использование многочленов в полях Галуа в полиномиальной системе классов вычетов может усложнить процесс написания вычислительных модулей. Другим подходом к использованию модулярного кода является система остаточных классов, для которой используются числовые значения.

Рассмотрим применение данных методов для системы остаточных классов.

3. Избыточная система остаточных классов для исправления ошибок

Еще одним эффективным представлением чисел при параллельной обработке является система остаточных классов (СОК). Если задан ряд положительных целых чисел $p_1, p_2, \dots, p_n$, называемых модулями или основаниями системы, то под системой остаточных классов понимается система, в которой целое положительное число представляется в виде набора остатков по выбранным основаниям $X=(x_1, x_2, \dots, x_n)$, где $x_i=|X|_{p_i}=X \bmod p_i$ для $i=1, 2, \dots, n$ [3]. Из теории чисел известно, что если модули p_i взаимно простые, то представление числа $X=\{x_1, x_2, \dots, x_n\}$ является единственным. При этом можно выполнять модульные операции сложения, вычитания и умножения с остатками независимо по каждому модулю [7].

Рассмотрим избыточную систему остаточных классов с одним избыточным модулем, т.е. n – общее число модулей, $r=1$ – количество избыточных модулей, $k=n-1$ – количество рабочих модулей. При этом $X<P=p_1 p_2 \dots p_{n-1}$, где P – динамический диапазон представления чисел. Добавление избыточного основания p_n в систему остаточных классов с модулями $\{p_1, p_2, \dots, p_{n-1}\}$ позволяет обнаружить ошибку, а именно, число корректно, если

лежит в диапазоне $[0, P)$, в случае нахождения числа в диапазоне $\left[P, \hat{P}\right)$, где $\hat{P} = P \cdot p_n$ – полный диапазон системы, число содержит ошибку.

Чтобы оценить, в какой диапазон входит число, необходимо найти его позиционную характеристику. Одним из методов получения позиционной характеристики является перевод из СОК в позиционную систему счисления с использованием метода на основе Китайской теоремы об остатках (КТО), по которой число X может быть получено из формулы

$$X = \left| \sum_{i=1}^{n} \hat{P}_i \cdot x_i \cdot \left| \hat{P}_i^{-1} \right|_{p_i} \right|_{\hat{P}}, \quad (1)$$

где $\hat{P}$ – полный динамический диапазон, $\hat{P}_i = \hat{P}/p_i$, $\left| \hat{P}_i^{-1} \right|_{p_i}$ – мультипликативная инверсия $\hat{P}_i$ по модулю p_i [8].

В общем случае добавление одного избыточного основания позволяет обнаружить одиночную ошибку. На основании теоремы 1 введем понятие энтропии для обнаружения ошибок большей кратности.

Энтропия модулей системы остаточных классов отражает количество информации, которое содержится в данном остатке, энтропия модуля равна $H(p_i) = \log_2 p_i$. Тогда энтропию модулей СОК можно записать в виде

$$h_E = \left(H(p_1), H(p_2), \dots, H(p_n)\right).$$

В [3] приведено утверждение, что если среди модулей СОК есть такие малые основания $p_{j,1}$, $p_{j,2}$, ..., $p_{j,t}$, для которых выполняется

$$\prod_{i=1}^{t} p_{j,i} < p_n,$$

то любые искажения в цифрах по нескольким или даже всем этим модулям превращают правильное число в неправильное и, следовательно, во всех случаях наличие искажений может быть обнаружено. Обобщим данное утверждение с помощью энтропии для системы с k рабочими модулями и $r = n - k$ избыточными модулями, где n общее количество модулей СОК. Введем теорему 2.

Теорема 2. Если задана система остаточных классов с n модулями $p_1, p_2, \dots, p_n$, где r – количество избыточных модулей, и $k = n - r$ – количество рабочих модулей, то для числа A, представленного в виде остатков $(a_1, a_2, \dots, a_n)$ и множества $\overline{I}$ остатков с ошибкой, ошибка может быть обнаружена, если выполняется условие

$$\sum_{i \in \overline{I}} H(p_i) \leq \sum_{j=1}^{r} H\left(p_{k+j}\right). \quad (2)$$

Доказательство:

Правильность числа в СОК с избыточным модулем означает, что число входит в рабочий диапазон, т.е. $A < \prod_{i=1}^{k} p_i = P$.

1) Пусть $\overline{I}$ – пустое множество и ошибок нет, тогда условие (2) примет вид $0 \leq \sum_{j=1}^{r} H\left(p_{k+j}\right)$, что выполняется для всех p_{k+j}, $j \in [1, r]$ и число A может быть восстановлено из КТО по формуле (1).

2) Рассмотрим случай с одним избыточным модулем и одиночной ошибкой, тогда условие (2) примет вид $H(p_i) \leq H(p_n)$, откуда из определения энтропии $\log_2 p_i \leq \log_2 p_n$, и $p_i < p_n$. Из определения правильности числа $A < P = \frac{\hat{P}}{p_n}$ и т.к. $p_i < p_n$, то $\frac{\hat{P}}{p_i} > \frac{\hat{P}}{p_n}$ и $A < \frac{\hat{P}}{p_n} < \frac{\hat{P}}{p_i}$. Поскольку число $\overline{A}$ содержит ошибку, то $\overline{a_i} \neq a_i$ и число $\overline{A}$ не может находиться в интервале

$\left[0, \frac{P}{p_i}\right)$, следовательно $\overline{A} > \frac{\hat{P}}{p_i}$ и тогда имеет место выражение $\overline{A} > \frac{\hat{P}}{p_n}$ и число $\overline{A}$ является неправильным.

3) Если возникли ошибки по нескольким основаниям $p_{i_1}, \dots, p_{i_l}$, то условие (2) примет вид $\sum_{\mathrm{j}=1}^{l} H\left(p_{i_j}\right) \le H(p_n)$, откуда $\log_2 p_{i_1} + \dots + \log_2 p_{i_l} \le \log_2 p_n$ и $\prod_{j=1}^{l} p_{i_j} \le p_n$. Рассмотрим произведение $\prod_{j=1}^{l} p_{i_j}$ как единое основание $\overline{p_i}$, получим $\overline{p_i} \le p_n$ и случай сводится к случаю 2, что означает, что любая ошибка по модулям $p_{i_j}, j \in [1, l]$ может быть обнаружена.

4) Если избыточных оснований несколько, $p_{k+1}, \dots, p_{k+r}$, то условие (2) примет вид $\sum_{\mathrm{j}=1}^{l} H\left(p_{i_j}\right) \le \sum_{j=1}^{r} H\left(p_{k+j}\right)$, откуда $\log_2 p_{i_1} + \dots + \log_2 p_{i_l} \le \log_2 p_{k+1} + \dots + \log_2 p_{k+r}$ и $\prod_{j=1}^{l} p_{i_j} \le \prod_{j=1}^{r} p_{k+j}$. Рассмотрим $\prod_{j=1}^{r} p_{k+j}$ как единое основание $\overline{p_n}$, то соблюдается условие $\overline{p_i} \le \overline{p_n}$ и случай сводится к случаю 2.

Таким образом, любая ошибка по модулям $p_{i_j}, j \in [1, l]$ может быть обнаружена.

Накладывая ограничения на избыточные модули СОК можно обнаружить ошибки большей кратности. Введем алгоритм 1 выбора позиций ошибок E, которые могут быть обнаружены. Для обнаружения ошибки нужно проверить комбинации ошибок от однократных до ошибок кратности $n-1$. Количество однократных ошибок равно C_n^1, количество двукратных ошибок C_n^2, и так далее, количество ошибок кратности $\mathrm{n}-1$ равно C_n^{n-1}, где C_n^k – количество сочетаний из n по k. Из бинома Ньютона $(a+b)^n = \sum_{k=0}^{n} C_n^k a^{n-k} b^k$ можно получить, что количество ошибок кратности от 1 до $n-1$ равно $2^n - C_n^0 - C_n^n = \sum_{k=1}^{n-1} C_n^k = 2^n - 2$. $\langle h_E \cdot \mathrm{bin}(i) \rangle$ – скалярное произведение $h_E = \{H(p_1), H(p_2), \dots, H(p_n)\}$ и $\mathrm{bin}(i)$, где $\mathrm{bin}(i)$ – вектор из n нулей и единиц, полученный при переводе i в двоичную систему счисления.

Input: $\{p_1, p_2, \dots, p_n\}$,
$h_E = \{H(p_1), H(p_2), \dots, H(p_n)\}$
Output: E

1. $E = \{\}$
2. $T = \sum_{j=1}^{r} H\left(p_{k+j}\right)$
3. Для i от 1 до $2^n - 2$ выполнять:
   3.1. Если $\langle h_E \cdot \mathrm{bin}(i) \rangle \le T$, то
      3.1.1. Добавить $\mathrm{bin}(i)$ к E
4. Возвратить E

Алгоритм 1. Выбор позиций обнаруживаемых ошибок

Пример 2. Возьмем СОК с модулями $\{3, 5, 7, 37\}$ и одним избыточным модулем, для которых энтропии равны $h_E = (1.58, 2.32, 2.81, 5.21)$. Рассмотрим работу алгоритма 1. $T = 5.21$, тогда для $\mathrm{bin}(1) = \{0,0,0,1\}$ скалярное произведение $\langle h_E \cdot \mathrm{bin}(1) \rangle = H(p_n)$=5.21. Условие (2) выполняется, значит ошибка на позиции $\{0,0,0,1\}$ может быть обнаружена. Из алгоритма 1 выражение (2) выполняется для следующих сочетаний ошибок $\{p_1\}$, $\{p_2\}$, $\{p_3\}$, $\{p_4\}$, $\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_2, p_3\}$, следовательно, ошибки на данных позициях могут быть обнаружены.

Рабочим диапазоном данной СОК является $P = 105$. Возьмем число $X = 10 = (1, 0, 3, 10)$ и введем ошибку по двум основаниям $E = (2, 0, 5, 0)$, получим число $\overline{X} = (0, 0, 1, 10)$. Используя формулу (1), получим $\overline{X} = (0,0,1,10) = 750$ и поскольку $750 > P = 105$, ошибка может быть обнаружена.

Важным приложением выражения (2) является возможность определения позиций обнаруживаемых ошибок, что позволит строить распределенные вычислительные системы с требуемыми корректирующими возможностями.

Очевидно, что количество избыточных модулей и их размеры обеспечивают необходимые возможности к обнаружению ошибок. Так, рассмотренный выше набор позволяет обнаружить ряд двойных ошибок, в то время как набор {3,5,7,8} обнаруживает только одиночные ошибки.

Рассмотрим применение веса Хэмминга и метода наибольшего правдоподобия из статьи [5] для системы остаточных классов. Для формирования кандидатов V_i во всех методах используется сравнение с рабочим диапазоном. Для нахождения кандидатов метода наибольшего правдоподобия выбираются все значения, входящие в рабочий диапазон, в которых изменены значения на позициях, найденных из выражения (2). В дальнейшем для каждого кандидата вычисляется вес Хэмминга и выбирается кандидат с наибольшим весом.

Пример 3. Для рассмотренного выше примера 2 с ошибочным числом $\overline{X} = (0,0,1,10)$ такими кандидатами будут числа $V_1 = (0,0,1,15)$, $V_2 = (1,0,3,10)$, $V_3 = (0,4,0,10)$, которые попадают в рабочий диапазон. Максимальное значение веса Хэмминга равно 7.53 для числа $V_2 = (1,0,3,10)$.

Однако если возникла ошибка $E = (2,0,4,0)$, то для полученного числа $\overline{X} = (0,0,0,10)$ кандидатами будут $V_1 = (0,4,0,10)$, $V_2 = (0,0,0,0)$, $V_3 = (1,0,3,10)$. Максимальное значение веса Хэмминга равно 9.60 для числа $V_1 = (0,4,0,10)$, что не соответствует исходному числу без ошибок.

Таким образом, метод наибольшего правдоподобия некорректно исправляет ошибки. Рассмотрим другой подход к исправлению ошибок, а также проанализируем исправляемые ошибки избыточной системой остаточных классов с одним избыточным основанием.

Другим подходом к локализации ошибки в системе остаточных классов является метод проекций. Проекцией X_i числа $X = (x_1, x_2, \dots, x_n)$ по модулю p_i будет число, полученное вычеркиванием цифры x_i в представлении X. Проекцией $X_{i,j}$ по основаниям p_i и p_j называется число, полученное из X вычеркиванием цифр по основаниям p_i и p_j. Если в упорядоченной системе остаточных классов с одним избыточным основанием проекция X_i числа X по модулю p_i удовлетворяет условию $X_i > \hat{P}/p_n$, то цифра x_i правильная, если возможна лишь одиночная ошибка [3].

При этом для правильного числа X все позиционные значения проекций равны между собой. Данное свойство может быть использовано в том случае, когда ряд проекций попадает в легитимный диапазон, и если среди них есть равные, то данное число и будет исходным, не содержащим ошибки.

Введение только одного избыточного основания в общем случае не позволяет локализовать ошибку.

Пример 4. Рассмотрим коррекцию ошибки методом проекций для СОК с одним избыточным основанием. Для СОК $\{3,5,7,37\}$ с числом $X = (1,0,3,10)$ введем ошибку $E = (0,1,0,0)$ и получим число $\overline{X} = (1,1,3,10)$. Поскольку $\overline{X} = 2341 > P = 105$, то число содержит ошибку. Построим проекции данного числа для позиций, найденных из выражения (2).

Тогда проекции, входящие в рабочий диапазон равны $\overline{X}_2 = (1,3,10) = 10$, $\overline{X}_4 = (1,1,3) = 31$, $\overline{X}_{1,2} = (3,10) = 10$, $\overline{X}_{2,3} = (1,10) = 10$. Поскольку часть проекций равна 10, то исходное число без ошибки равно $X = 10 = (1,0,3,10)$.

В [3] приведена теорема, согласно которой если избыточный модуль удовлетворяет условию $p_n > 2p_{n-1}p_{n-2}$, то ошибка по рабочему модулю может быть исправлена. Однако в данном случае система остаточных классов является несбалансированной, размерность избыточного основания более чем в 2 раза превышает размерность максимального рабочего модуля.

В статье [9] доказано уточнение, что исправление одиночной ошибки по рабочему модулю может быть осуществлено при $p_n > p_{n-1}p_{n-2}$.

4. Вычислительный эксперимент по обнаружению и исправлению ошибок

Рабочий диапазон $P = \prod_{i=1}^{k} p_i$ отражает разрешенные в данной системе числа, в то время как полный диапазон $\hat{P} = \prod_{i=1}^{n} p_i$ отражает все возможные числа в данной системе. При этом условием отсутствия ошибки числа A является выражение $A < P$, отсюда можно сделать вывод, что количество ошибочных значений равно $\hat{P} - P = P(\prod_{i=1}^{r} p_{k+i} - 1)$.

Количество одиночных ошибок по основанию p_i равно $p_i - 1$. Количество двойных ошибок по основаниям p_i и p_j равно $(p_i - 1)(p_j - 1)$. Количество ошибок больше кратности по основаниям p_{i_j} равно $\prod_j (p_{i_j} - 1)$. При этом ряд ошибок будет иметь сложную структуру, попадать в разрешенный диапазон и обнаружить их не будет возможности. В соответствии с выражением (2) можно определить позиции ошибок, которые гарантированно могут быть обнаружены, при этом могут быть обнаружены и дополнительные ошибки большей кратности, но существует вероятность ложного необнаружения ошибок.

Для исследование были взяты три набора модулей СОК с одним избыточным основанием: $\{3, 5, 7, 8\}$, в которой избыточное основание удовлетворяет выражению $p_n > p_{n-1}$; $\{3, 5, 7, 71\}$, в которой избыточное основание удовлетворяет выражению $p_n > 2p_{n-1}p_{n-2}$; $\{3, 5, 7, 37\}$, в котором избыточное основание удовлетворяет выражению $p_n > p_{n-1}p_{n-2}$. В ходе вычислительного эксперимента для всех чисел из рабочего диапазона были добавлены все допустимые ошибки, которые могут быть обнаружены в соответствии с выражением (2) и найдено количество исправляемых и неисправляемых ошибок.

Используя формулу (2) и вычисляя h_E для каждого набора модулей, получим, что

- набор $\{3, 5, 7, 8\}$ может обнаружить ошибки по следующим сочетаниям модулей $\{p_1\}, \{p_2\}, \{p_3\}, \{p_4\}$, всего 19 одиночных ошибок, при этом гарантированно ни одна ошибка не может быть исправлена;
- набор $\{3, 5, 7, 37\}$ может обнаружить ошибки по следующим сочетаниям модулей $\{p_1\}$, $\{p_2\}$, $\{p_3\}$, $\{p_4\}$, $\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_2, p_3\}$, всего 92 ошибки, из них 48 одиночных и 44 двойных ошибки. Если учесть требование надежности избыточного основания, то система может обнаружить 56 ошибок по модулям $\{p_1\}$, $\{p_2\}$, $\{p_3\}$, $\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_2, p_3\}$, 12 одиночных и 44 двойных. В этом случае и метод наибольшего правдоподобия и метод проекций гарантированно исправляют все одиночные ошибки;
- набор $\{3, 5, 7, 71\}$ может обнаружить ошибки по следующим сочетаниям модулей $\{p_1\}$, $\{p_2\}$, $\{p_3\}$, $\{p_4\}$, $\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_2, p_3\}$, всего 126 ошибок, из них 82 одиночных и 44 двойных ошибок. Если учесть требование надежности избыточного основания, то система может обнаружить 56 ошибок по модулям $\{p_1\}$, $\{p_2\}$, $\{p_3\}$, $\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_2, p_3\}$, 12 одиночных и 44 двойных. В этом случае и метод наибольшего правдоподобия и метод проекций гарантированно исправляют все одиночные ошибки.

Таким образом, все ошибки на позициях, удовлетворяющих выражению (2), могут быть обнаружены. Накладывание ограничений на избыточное основание позволяет исправить ошибки по рабочим модулям.

Рассмотрим результаты моделирования обнаружения и исправления ошибок.

Для набора модулей $\{3, 5, 7, 8\}$ на практике возможно обнаружение всех 19 одиночных ошибок, 110 из 128 двойных ошибок, 312 тройных ошибок из 356, 294 из 336 ошибок

кратности 4. Однако кроме случая гарантированного обнаружения одиночных ошибок, ошибки большей кратности могут иметь сложных характер и попадать в разрешенный диапазон, что приведет к ошибкам вычисления. Для случая исправления ошибок гарантированно исправить ошибку не представляется возможным, на практике метод наибольшего правдоподобия, как и метод проекций, в среднем позволяет исправить только 7 из 19 гарантированно обнаруживаемых одиночных ошибок. Если учесть требование надежности избыточного основания, то система может обнаружить ошибки по модулям $\{p_1\}$, $\{p_2\}$, $\{p_3\}$ и исправить 7 одиночных ошибок из 12 для обоих методов.

Для набора модулей $\{3, 5, 7, 37\}$ на практике возможно обнаружение всех 48 одиночных ошибок, 464 из 476 двойных ошибок, 1582 из 1632 тройных ошибок и 1682 из 1728 ошибок по всем 4 основаниям. При этом метод наибольшего правдоподобия позволяет исправить 13 из 48 одиночных и 24 из 44 двойных гарантированно обнаруживаемых ошибок. Если учесть требование надежности избыточного основания, то оба метода исправляют все одиночные ошибки и 24 из 44 двойных ошибок.

Для набора $\{3, 5, 7, 71\}$ на практике возможно обнаружение всех 82 одиночных ошибок, 872 из 884 двойных ошибок, 3083 из 3128 тройных ошибок и 3313 из 3360 ошибок по всем основаниям. При этом метод наибольшего правдоподобия позволяет исправить 34 из 82 одиночных и 37 из 44 двойных гарантированно обнаруживаемых ошибок. Если учесть требование надежности избыточного основания, то оба метода исправляют все одиночные ошибки и 24 из 44 двойных ошибок.

Как видно из результатов эксперимента, для исправления всех одиночных ошибок достаточно выполнения условия $p_n > p_{n-1}p_{n-2}$, что позволяет вдвое сократить размер надежного модуля, снизив тем самым разбалансировку модулей СОК и повысив производительность отказоустойчивой вычислительной системы.

5. Система надежного распределенного хранения в системе остаточных классов

Рассмотрим модель системы надежного распределенного хранения в системе остаточных классов, представленную четырьмя облаками, представленную на рис. 1. Пользовательская часть системы содержит блок преобразования позиционного числа X в систему остаточных классов с модулями $\{p_1, p_2, p_3, p_4\}$ и передачи данных в облака, а также блок приема данных, обнаружения и коррекции ошибки, и перевода из системы остаточных классов в позиционный код. При этом используемое количество облаков равно количеству модулей системы остаточных классов, и они хранят остатки x_i от деления исходного числа X на модули p_i, $x_i = |X|_{p_i}$. Данная система относится к взвешенным схемам разделения данных, при этом данные по модулям с большей энтропией размещаются в наиболее надежных облаках.

Рассмотрим систему, в которой могут возникать только одиночные и двойные ошибки, т.е. данные с одного или двух облаков могут быть потеряны или содержать ошибки. Будем рассматривать только гарантированно обнаруживаемые и исправляемые ошибки.

В этом случае для СОК с модулями $\{3, 5, 7, 8\}$ вероятность обнаружения и исправления однократных и двухкратных ошибок равны соответственно $P_{\text{обн}} = 0{,}129$ и $P_{\text{исп}} = 0$. Для СОК с модулями $\{3, 5, 7, 37\}$ вероятность обнаружения и исправления однократных и двухкратных ошибок равны соответственно $P_{\text{обн}} = 0{,}175$ и $P_{\text{исп}} = 0{,}023$. Для СОК с модулями $\{3, 5, 7, 71\}$ вероятность обнаружения и исправления однократных и двухкратных ошибок равны соответственно $P_{\text{обн}} = 0{,}130$ и $P_{\text{исп}} = 0{,}012$.

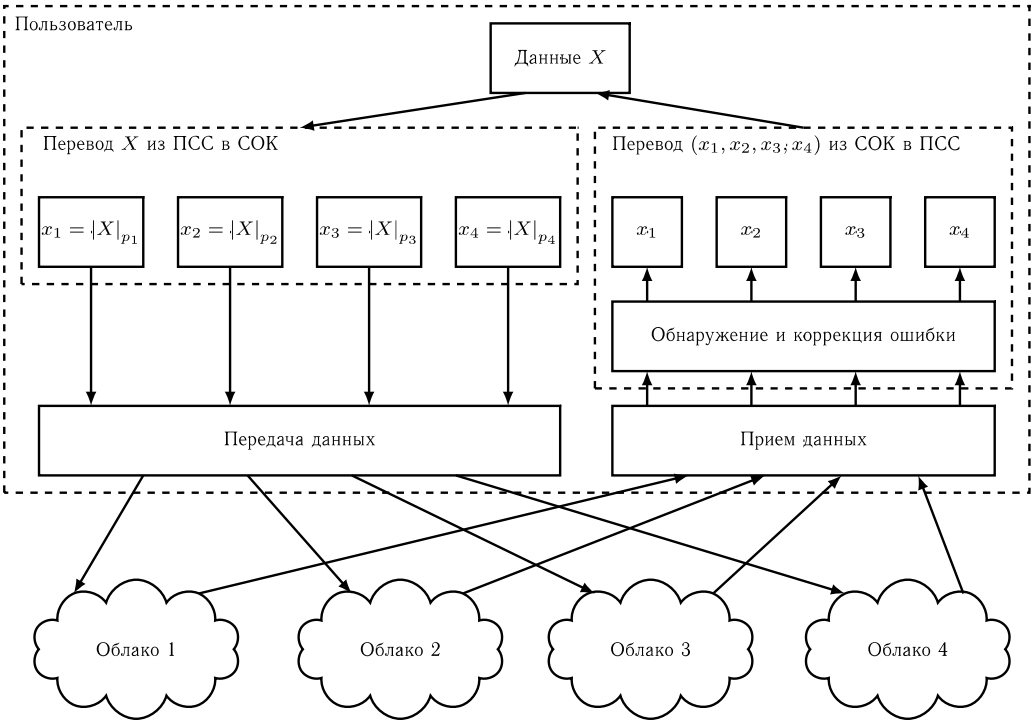


Рис. 1. Структура системы надежного распределенного хранения в системе остаточных классов

6. Заключение

В статье рассмотрена проблема обнаружения и исправления ошибок модулярного кода. Применение энтропийного подхода к обнаружению ошибок и метода наибольшего правдоподобия к локализации ошибок полиномиальной системы классов вычетов позволяет обнаружить и исправить ошибки большей кратности, по сравнению с классическими подходами. В то же время вычисления с многочленами требуют построения новой архитектуры вычислительных систем, что приводит к некоторым трудностям.

Использование энтропийного подхода к системе остаточных классов также позволяет повысить кратность обнаруживаемых ошибок и найти позиции ошибок, которые могут быть обнаружены данной системой. Доказана теорема, позволяющая определить позиции ошибок, которые точно могут быть обнаружены. Однако исправить все эти ошибки ни методом наибольшего правдоподобия, ни методом проекций не удается.

Если $p_n > p_{n-1}p_{n-2}$ и x_n – корректно, то код исправления ошибок позволяет исправить однократные ошибки и обнаружить двукратные ошибки. Вероятность обнаружения ошибки возрастает на 35% по сравнению с традиционными кодами исправления ошибок в модулярном коде.

Таким образом, эффективным для построения систем надежного распределенного хранения в системе остаточных классов является использование избыточной системы остаточных классов, в которой надежный избыточных модуль превышает произведение двух максимальных рабочих модулей. В этом случае система может обнаружить все однократные ошибки, а также ряд двухкратных ошибок по модулям, удовлетворяющим условию теоремы 2. Также система позволяет исправить все одиночные ошибки по рабочим модулям, что позволяет построить отказоустойчивую систему хранения данных.

Список литературы / References

[1]. Обнаружение и исправление ошибок в дискретных устройствах. Под ред. В.С. Толстякова. М.: Сов. радио, 1972. – 288 с. / Detection and correction of errors in discrete devices. Edited by V.S. Tolstyakov. Moscow: Sov. radio, 1972. - 288 p. (in Russian).

[2]. Стахов А.П. Введение в алгоритмическую теорию измерения. Москва: Сов. радио, 1977. – 288 с. / Stakhov, A.P. Introduction to the algorithmic theory of measurement / A.P. Stakhov. - Moscow: Sov. radio, 1977. - 288 p. (in Russian).

[3]. Акушский И.Я., Юдицкий Д.И. Машинная арифметика в остаточных классах. М., Советское радио, 1968, 440 с. / Akushsky I. Ya., Yuditsky D. I. Computer arithmetic in residual classes. Moscow, Soviet Radio, 1968, 440 p. (in Russian).

[4]. Колмогоров А.Н. Три подхода к определению понятия «количество информации», Пробл. передачи информ., 1:1, 1965, с. 3–11 / Kolmogorov, A.N. Three approaches to the definition of the concept “quantity of information”. Probl. Peredachi Inf., 1965, 1, pp. 3–11. (in Russian).

[5]. Tchernykh A. et al. En-AR-PRNS: Entropy-Based Reliability for Configurable and Scalable Distributed Storage Systems //Mathematics. – 2021. – Т. 10. – №. 1. – С. 84.

[6]. Goh, V.T.; Siddiqi, M.U. Multiple error detection and correction based on redundant residue number systems. IEEE Trans.Commun. 2008, 56, 325–330. DOI: 10.1109/TCOMM.2008.050401

[7]. Chang C. H. et al. Residue number systems: A new paradigm to datapath optimization for low-power and high-performance digital signal processing applications //IEEE circuits and systems magazine. – 2015. – Т. 15. – №. 4. – С. 26-44. DOI: 10.1109/MCAS.2015.2484118

[8]. Aremu I. A., Gbolagade K. A. Redundant residue number system based multiple error detection and correction using Chinese remainder theorem (CRT) //Software Engineering. – 2017. – Т. 5. – №. 5. – С. 72-80. DOI: 10.11648/j.se.20170505.12

[9]. Gladkov A. et al. Modified Error Detection and Localization in the Residue Number System // Programming and Computer Software, 2022, Vol. 48, No. 8, pp. 598–605 DOI: 10.1134/S0361768822080126

Информация об авторах / Information about authors

Виктор Андреевич КУЧУКОВ – научный сотрудник отдела технологий программирования Института системного программирования РАН, младший научный сотрудник Северо-Кавказского центра математических исследований Северо-Кавказского федерального университета. Сфера научных интересов: высокопроизводительные вычисления, система остаточных классов, нейронные сети, цифровая обработка сигналов.

Viktor Andreevich KUCHUKOV is a researcher at the Department of Programming Technologies of the Ivannikov Institute for System Programming of the Russian Academy of Sciences, junior researcher at the North Caucasus Center for Mathematical Research of the North-Caucasus Federal University. Research interests: high-performance computing, residue number systems, neural networks, digital signal processing.

DOI: 10.15514/ISPRAS-2024-36(1)-6

The Foundations of Quantum Computing and Their Relation to Software Engineering

[1] *Reyes Juárez-Ramírez*, ORCID: 0000-0002-5825-2433 <*reyesjua@uabc.edu.mx*>
[1] *Christian X. Navarro*, ORCID: 0000-0002-7220-7006 <*cnavarro@uabc.edu.mx*>
[1] *Samantha Jiménez*, ORCID: 0000-0003-0938-7291 <*samantha.jimenez@uabc.edu.mx*>
[2] *Alan Ramírez*, ORCID: 0000-0002-8634-9988 <*alandramireznoriega@uas.edu.mx*>
[3] *Verónica Tapia-Ibarra*, ORCID: 0000-0002-0501-8600 <*veronica.tapia@leon.tecnm.mx*>
[4] *César Guerra-García*, ORCID: 0000-0002-9290-6170 <*cesar.guerra@uaslp.mx*>
[4] *Hector G. Perez-Gonzalez*, ORCID: 0000-0003-3331-2230 <*hectorgerardo@uaslp.mx*>
[5] *Carlos Fernández-y-Fernández*, ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>

[1] *Universidad Autónoma de Baja California, Tijuana, Baja California, México.*
[2] *Universidad Autónoma de Sinaloa, Sinaloa, México.*
[3] *Instituto Tecnológico de León, León, Guanajuato, México.*
[4] *Universidad Autónoma de San Luis Potosí, San Luis Potosí, SLP, México.*
[5] *Universidad Tecnológica de la Mixteca, Huajuapan de León, Oaxaca, México.*

Abstract. The principles of quantum mechanics – superposition, entanglement, measurement, and decoherence – form the foundation of quantum computing. Qubits, which are abstract objects having a mathematical expression to implement the rules of quantum physics, are the fundamental building blocks of computation. Software is a key component of quantum computing, along with quantum hardware. Algorithms make up software, and they are implemented using logic gates and quantum circuits. These qualities make quantum computing a paradigm that non-physicists find difficult to comprehend. It is crucial to incorporate a conceptual framework of the principles upon which quantum computing is founded into this new method of creating software. In this paper, we present a kind of taxonomical view of the fundamental concepts of quantum computing and the derived concepts that integrate the emerging discipline of quantum software engineering. Because the systematic review's main goal is to identify the core ideas behind quantum computing and quantum software, we conducted a quasi-systematic mapping as part of the review process. The findings can serve as a starting point for computer science teachers and students to address the study of this field.

Keywords: Quantum computing; quantum principles; qubits; software engineering; taxonomy.

For citation: Juárez-Ramírez R., Navarro C.X., Jiménez S., Ramírez A., Tapia-Ibarra V., Guerra-García C., Perez-Gonzalez H. G., Fernández-y-Fernández C. The Foundations of Quantum Computing and Their Relation to Software Engineering. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 73-104. DOI: 10.15514/ISPRAS-2024-36(1)-6.

Full text: R. Juárez-Ramírez, C. X. Navarro, Samantha Jiménez, Alan Ramírez, Verónica Tapia-Ibarra, César Guerra-García, Hector G. Perez-Gonzalez, and Carlos Fernández-y-Fernández. A Taxonomic View of the Fundamental Concepts of Quantum Computing – A Software Engineering Perspective. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 682–704. DOI: 10.1134/S0361768823080108.

Acknowledgements. We would like to thank Universidad de Guanajuato, División de Ciencias e Ingeniería, for their support during the sabbatical stage. Also, thanks to Centro de Investigación en Matemáticas for their support in providing access to electronic databases. Finally, we would like to thank Universidad Autónoma de Baja California, for financing this research through the project 3908_300/6/C/63/23 “Computación Cuántica: Implicaciones en la Ingeniería de Software y las Competencias del Ingeniero de Software”, 23ª Convocatoria

Interna de Proyectos de Investigación. Also, thanks to the team Red Mexicana de Ingeniería de Software who participate in this Project and literature review.

Основы квантовых вычислений и их связь с разработкой программного обеспечения

[1] *Рейес Хуарес-Рамирес,* ORCID: 0000-0002-5825-2433 <*reyesjua@uabc.edu.mx*>
[1] *Христиан К. Наварро,* ORCID: 0000-0002-7220-7006 <*cnavarro@uabc.edu.mx*>
[1] *Саманта Хименес,* ORCID: 0000-0003-0938-7291 <*samantha.jimenez@uabc.edu.mx*>
[2] *Алан Рамирес,* ORCID: 0000-0002-8634-9988 <*alandramireznoriega@uas.edu.mx*>
[3] *Вероника Тапия-Ибарра,* ORCID: 0000-0002-0501-8600 <*veronica.tapia@leon.tecnm.mx*>
[4] *Сесар Герра-Гарсия,* ORCID: 0000-0002-9290-6170 <*cesar.guerra@uaslp.mx*>
[4] *Эктор Х. Перес-Гонсалес,* ORCID: 0000-0003-3331-2230 <*hectorgerardo@uaslp.mx*>
[5] *Карлос Фернандес-и-Фернандес,* ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>

[1] *Автономный университет Нижней Калифорнии, Тихуана, Нижняя Калифорния, Мексика.*
[2] *Автономный университет Синалоа, Синалоа, Мексика.*
[3] *Технологический институт Леона, Леон, Гуанахуато, Мексика.*
[4] *Автономный университет Сан-Луис-Потоси, Сан-Луис-Потоси, СЛП, Мексика.*
[5] *Технологический университет Миштека, Уахуапан де Леон, Оахака, Мексика.*

Аннотация. Принципы квантовой механики – суперпозиция, запутанность, измерение и декогерентность – формируют основу квантовых вычислений. Кубиты, которые являются абстрактными объектами, представляемые математическими выражениями, моделирующими законы квантовой физики, являются фундаментальными строительными блоками вычислений. Программное обеспечение является, наряду с квантовым оборудованием, ключевым компонентом квантовых вычислений. Программы строятся на основе алгоритмов, которые реализуются с помощью логических вентилей и квантовых схем. Эти качества делают квантовые вычисления парадигмой, которую трудно понять тем, кто не имеет специального физического образования. Для этой новой парадигмы очень важно выработать концептуальные основы, главные принципы. В статье представлен таксономический взгляд на фундаментальные концепции квантовых вычислений и другие концепции, которые объединяют возникающую дисциплину квантовой программной инженерии. Основной целью систематического обзора является выявление основных идей, лежащих в основе квантовых вычислений и квантового программного обеспечения, в рамках сделанного обзора мы провели квазисистематическое картирование. Результаты могут служить отправной точкой для учителей информатики и студентов при изучении этой предметной области.

Ключевые слова: квантовые вычисления; квантовые законы; кубиты; разработка программного обеспечения; таксономия.

Для цитирования: Хуарес-Рамирес Р., Наварро-Кота Ч.К., Хименес С., Рамирес А., Тапия-Ибарра В., Герра-Гарсия С., Перес-Гонсалес Э.Х., Фернандес-и-Фернандес К. Основы квантовых вычислений и их связь с разработкой программного обеспечения. Труды ИСП РАН, том. 36, вып. 1, 2024. стр. 73-104 (на английском языке). DOI: 10.15514/ISPRAS-2024-36(1)-6.

Полный текст: Хуарес-Рамирес Р., Наварро-Кота Ч.К., Хименес С., Рамирес А., Тапия-Ибарра В., Герра-Гарсия С., Перес-Гонсалес Э.Х., Фернандес-и-Фернандес К. Таксономический взгляд на фундаментальные концепции квантовых вычислений: перспективы разработки программного обеспечения. Programming and Computer Software, 2023, т. 49, №. 8, стр. 682–704 (на английском языке). DOI: 10.1134/S0361768823080108.

Благодарности. Мы благодарны Отделению науки и техники Университета Гуанахуато за предоставление творческого отпуска. Также благодарим Центр математических исследований,

предоставивший доступ к электронным базам данных. Наконец, мы благодарим Автономный университет Нижней Калифорнии за финансирование этого исследования в рамках проектов 3908_300/6/C/63/23 «Квантовые вычисления: последствия для программной инженерии и компетенции инженера-программиста», 23ª «Привлечение учащихся к исследовательским проектам», также команду Мексиканской сети программной инженерии, которая участвует в этом проекте и обзоре литературы.

1. Introduction

Quantum computing involves information processing tasks, which are implemented using quantum mechanical systems [1]. To process, store, and transfer the quantum information set represented by qubits, quantum computers are based on quantum mechanics phenomena such as quantum superposition and entanglement. [2]. Quantum computing can significantly cut both execution time and energy usage when compared to conventional digital computing [3]; it sounds simple and obvious, but is it? In terms of hardware, the construction of quantum systems is still facing challenges [4-5]. In terms of software, it is based on the mathematical concepts of quantum mechanics phenomena [6]; this fact makes its implementation more complicated. In this fashion, the authors of [7] argued that, for academicians and practitioners, there is an emergent nature of quantum computer research and an increasing need for interdisciplinarity to address the identified challenges.

In general terms, quantum computing is a multidisciplinary field that brings together aspects of physics, mathematics, and computer science [5] and uses quantum mechanics to solve complex problems faster than classical computers [8]. A supporting definition appears in [9]: "*Quantum computing is a fascinating new field at the intersection of computer science, mathematics, and physics, which strives to harness some of the uncanny aspects of quantum mechanics to broaden our computational horizons*".

Let us introduce the fundamental concepts involved.

The qubit is an abstract object, with a mathematical expression, so it is a mathematical object with certain specific properties [1]; it is a quantum system with two basic states, $|0\rangle$ and $|1\rangle$ [8] manipulated arbitrarily, which are well distinguishable by physical measurements. In contrast to classic binary digits represented as (0,1), qubit as the most fundamental unit of the quantum information set attains a state that is a superposition of 0 and 1 and is represented as $|0\rangle$ and $|1\rangle$ [10-11] where:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

A pure qubit state is a coherent superposition of the basis states. This means that a single qubit (ψ) is a linear combination of $|0\rangle$ and $|1\rangle$ as follows:

$$|\psi\rangle = |\alpha|^2 + |\beta|^2 \quad (1)$$

This gives rise to new logic gates that make new algorithms possible. A quantum gate is a basic quantum circuit that operates on a small set of qubits [8]. The quantum gates allow the implementation of quantum mechanics principles [1], such as superposition and entanglement. A quantum gate is simply an operator that acts on qubits. Unitary matrices will be used to represent these operators [9]. For example, two essential quantum gates are Hadamard (One-qubit gate) gate and CNOT (Multi-qubit gate) gate [8]. Hadamard gate allows the superposition of a qubit, while the CNOT gate allows the entanglement of multiple qubits [12]. This is their matrix representation:

$$\text{Hadamard} = \text{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$\text{CNOT} = \text{CX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

A quantum program or algorithm is implemented through a quantum circuit, which is formed by a set of quantum gates. A complete quantum program is expressed in a considerable number of circuits.

Based on this explanation, quantum programming can be seen as the process of assembling sequences of instructions, called quantum circuits, which can run on a quantum computer.

As we can see, the quantum programming model is fundamentally different from traditional computer programming. It is observed in the quantum computing labor market that most positions require Ph.D. degrees due to the elevated level of expertise required by quantum principles.

There is a rapidly growing demand for a quantum workforce [13] educated in the basics of quantum computing, particularly in quantum programming [14].

Therefore, it is necessary to provide a more "intuitive" way to think and write quantum algorithms, thereby simplifying the design and implementation of quantum software [11].

Quantum computing education is ramping up, however, now, there are few computer science specialists to teach these topics, so there are few offerings for non-specialists and little information on best practices for training computer science and engineering students [14].

In [15], the authors suggest bridging the gap between computer professionals and non-physicists by offering the conceptual and notational information that segregates quantum computing from conventional computing. In [16], the authors stated that interest in building dedicated quantum information science and engineering (QISE) education programs have greatly expanded in recent years and that there will be a need for a wide variety of expertise and education levels to create a balanced technical workforce like that seen in other professional scientific and engineering fields.

To contribute to the efforts in making quantum computing literacy more understandable, in this paper, we present a taxonomic view of the fundamental concepts involved in quantum computing, as an initial proposal to integrate a body of knowledge (BOK) for quantum software engineering. We focused on the set of concepts (as the first element of a BOK), which are extracted from the literature, emphasizing the quantum mechanics fundamentals, and the related knowledge of mathematics and computer science. To find the fundamental concepts we performed a quasi-systematic mapping, with ingredients of systematic literature review and multivocal literature review to include grey literature.

The rest of the paper is organized as follows. Section 2 contains a background describing how a body of knowledge is integrated, emphasizing the set of fundamental concepts. Section 3 contains related work, describing proposals of taxonomies in the context of quantum computing, and education initiatives for quantum computing as well as skills required for quantum computing. Section 4 contains the methodology used, describing how the systematic mapping was done. Section 5 contains the results, presenting the taxonomical view of the fundamental concepts of quantum computing and the emerging concepts of quantum software engineering. Section 6 contains a discussion. Finally, section 7 presents the conclusions and future work.

2. Background: Body of knowledge

There are various BOK definitions in the literature. In this section, we present some of them.

A body of knowledge is the complete set of concepts, terms, and activities that make up a professional domain, as defined by the relevant learned society or professional association [17]. The body of knowledge is "generally recognized" by practitioners and may be codified in a variety of ways for a variety of different uses [18].

In [19], the following definition is given for a body of knowledge: (1) "Structured knowledge that is used by members of a discipline to guide their practice or work"; and (2) "The prescribed aggregation of knowledge in a particular area an individual is expected to have mastered to be considered or certified as a practitioner".

We are going to work with this idea: "*A body of knowledge is a set of knowledge within a profession or subject area which is agreed as both essential and known*" [17]. A body of knowledge is the accepted ontology for a specific domain. Furthermore, a BOK also is considered "*the systematic collection of activities and outcomes in terms of their values, constructs, models, principles, and instantiations, which arises from continuous discovery and validation work by members of the profession and enables self-reflective growth and reproduction of the profession*" [20].

Integrating a BOK is a challenging task. There are many ways of doing it. The authors of [21] cited some methods: *developing taxonomies*, *engaging communities of practice*, and *constructing framing metaphors*.

As we mentioned before, a BOK is integrated by various elements, however, a basic element is the set of fundamental concepts of the referred discipline, so that, this paper is focused on the set of fundamental concepts of quantum computing.

3. Related work

Quantum Ontologies and Quantum Body of Knowledge. In [22], a review of quantum computing literature is presented. The authors proposed a taxonomy of quantum computing, which is used to map various related studies to identify the research gaps. A main taxonomy is presented emphasizing quantum computing technology, which has two specialized branches: (1) time and gates characteristics and (2) algorithmic characteristics. A kind second level of taxonomy is presented for three branches: (1) Software Applications in Quantum Computing, (2) Quantum Annealing-based Software Components, (3) Quantum Software Life Cycle and Associated Terminologies.

The authors did not describe explicitly the systematic approach used to perform the literature review. Furthermore, the proposed taxonomy is not presented in a hierarchical view.

In [15], well-described definitions of quantum mechanics fundamentals are presented, such as superposition, entanglement, and decoherence, as well as the qubit as the fundamental unit of computation. Also, the authors presented a summary of the most used quantum algorithms, quantum technologies, and software tools. A taxonomy of one level is presented, characterizing the uses cases of quantum computing, having two branches: (1) emotions and sentiment analysis, error correction, quantum internet, quantum materials, cryptography, post-quantum cryptography, drug discovery, genetic programming, and navigation; (2) image processing, cloud computing, weather prediction, energy management, open-source software, transport engineering, machine learning, chemistry, and finance.

In the paper, the conceptual part of quantum mechanics fundamentals is well described, as well as the use cases. However, the taxonomical view is presented only at the level of use cases, not at the level of fundamental and supporting quantum computing concepts.

Teaching quantum computing. Quantum computing is a difficult field for non-experts to understand [23-24]. Additionally, most methods for implementing quantum computing are restricted to literature or software implementation.

The authors of [25] propose a Bloch sphere interactive system to visualize quantum computing simulation. This report describes a variety of programming assignments that can be used to teach quantum computing in a practical manner, experiencing most of the software development phases.

In [26], the authors present the results of two semesters of a new undergraduate course on Quantum Computing for Fundamental Sciences and Engineering students. The course was taught employing

a blended learning approach, with a combination of synchronous classes, asynchronous video lessons, and projects using IBM's Qiskit framework.

In both cases [25-26], the authors did not present a complete set of fundamental concepts of quantum computing supported by quantum mechanics.

Skills required for quantum software development. The rapid rise of interest and investments in quantum information science and engineering has led to increasing demand for a quantum-trained workforce [27]. Recent assessments of the needs of the quantum industry [16, 28] identified quantum software engineering and applications development as essential skills required for certain types of industry jobs [27]. Surveys of the university programs offering Master-level education in QISE [16] show that quantum programming is frequently included as either a standalone course or one of the topics in an introductory course [27].

Motivation from related work. This related work gives us a general view of efforts that are focused on trying to organize the knowledge around quantum computing; it is a motivation for us. Especially its limitations detected in terms of a taxonomical view of the fundamental concepts of quantum computing, encourage us to perform the current research work presented in this paper.

4. Methodology

To perform this review, we considered the recommendations for a systematic mapping (SM) from [29-31]. (The review performed is based on SM). Also, we considered recommendations for systematic literature reviews (SLR) from [32-37], especially for establishing selection criteria.

In terms of approaches for carrying out the review and extending the reach of the search, both methods are complementary. Sometimes additional sources are well accepted to enrich the coverture of the review [36], including manual and less structured searches of the Internet and other sources [38] and grey literature as is recommended in [39].

In [32] a comparison is presented, emphasizing the difference in breadth and depth of SM and SLR: *In a systematic mapping study, more articles can be considered as they do not have to be evaluated in such detail*. Therefore, a larger field can be structured. This can be reflected in the search string and inclusion criteria [29]. The search string can be more generic, and the inclusion criteria can be less restrictive [31].

Because the review's goal is to locate the fundamental ideas of quantum computing and quantum software engineering rather than to thoroughly examine the state of the art of each field or the solutions offered to the problems or difficulties that these disciplines face, this ability of SM allows us to perform a systematic mapping.

We decided to follow the recommendations of SLR, especially in terms of the first level of selection criteria, without quality assessment criteria.

There are several formal reports about quantum computing, however, a structured search may not include previous work with significant contributions to our objective. In an unstructured search, we identified work with significant contributions not included in the formal search, so we decided to review these references.

4.1 Definition of Research Questions (Outcome: Research Scope)

Software is an essential element for computation [40]. Quantum software applications are getting popular because the power of quantum computing facilitates the application of this paradigm to solve complex problems in any field of science and the real world [41-44]. Quantum software plays a critical role in exploiting the full potential of quantum computing systems [11]. Quantum applications require the use of a completely different kind of computer and algorithms, which have the potential to solve tasks that we do not even dare dream of today [45]. Considering the significance of software in the quantum world, we state the main objective of the literature review.

The objective of the literature review: To look for what is the impact of quantum computing on software. How do the fundamental concepts of quantum computing and its implementation participate in quantum software development? Furthermore, we intend to organize those fundamental concepts in a kind of taxonomical way for a better understanding of the multidisciplinary approach of quantum computing.

Research questions: Four research questions were formulated:

RQ1: What is the definition of Quantum Computing?

RQ2: What are the fundamental concepts of Quantum Computing?

RQ3: What are the supporting concepts for implementing Quantum Computing?

RQ4: What are the new Software Engineering concepts emerging from Quantum Computing?

Searching phrases: We decided to formulate a search phrase that includes quantum computing as the main topic and its relationship with software engineering.

The search phrase is the following:

"Quantum Computing" AND "Software Engineering"

Databases considered: We considered looking for papers in three databases: ACM, IEEE Xplore, and ScienceDirect.

Developing review protocol: In a session with the participation of two researchers, we defined a kind of general protocol, considering the following aspects: What type/source of papers to consider? What parts of the papers should be revised? How many reviewers are going to review the same set of papers? Defining the filters for selecting the papers, and formats for gathering information.

4.2 Conduct Search for Primary Studies (Outcome: All Papers)

Identifying the relevant research: We tried with a user account from a Mexican institution, with the proper privileges to access the advanced search section on the website of databases and to obtain the corresponding source files of the papers. The result of the search is shown in Table 1. The three databases provide more than 100 items from the search phrase.

Table 1. Results from the search

Database	Number of papers (ítems) found
ACM (AC)	201
ScienceDirect (SD)	199
IEEE Xplore (XP)	140

4.3 Screening of Papers for Inclusion and Exclusion (Outcome: Relevant Papers)

Selecting the primary studies: We defined a structure for identifying the resulting documents: demographic aspects, inclusion criteria, and exclusion criteria.

Demographics: This attribute identifies the nature of the document.

D1: Type of document (Research paper, communication paper, white paper). We decided to select research papers, unless there may be other types of work with significant contribution.

D2: Origin of the document (Conference, Journal, Book, other.) We decided to select papers from journals and conferences, unless there may be other types of work with significant contributions.

D3: Language – (Results that are written in English/Spanish). We decided to consider only work written in English.

D4: Accessibility – Full texts are accessible by means of institutional accounts.

INCLUSION CRITERIA: We established inclusion criteria on two levels, in the form of filters.

Level 1: Header of the paper

F1: **The title of the paper.** Contains one significant word of a phrase (e.g., quantum, software), one part of the search phrase, or two parts.

F2: **The keyword section.** Contains one significant word of a phrase (e.g., quantum, software), one part of the search phrase, or two parts.

F3: **Abstract section.** Contains an established relationship between "quantum computing" and "software engineering," for example:

- There are concepts related to one or both areas ("quantum computing," "software engineering"), explicitly cited.
- A relationship between "quantum computing" and "software engineering" is established.

Note: F3 was the filter with the highest acceptance value because we noticed that in some cases, the title and keywords section did not contain the expected elements, however, the abstract gave signs that the paper contains contribution.

Level 2: Body of the paper

B1: Results that introduce and describe concepts of QC and/or quantum software engineering.

B2: Results that discuss Quantum Computing and some aspects of software engineering, or vice versa.

EXCLUSION CRITERIA: Exclusion criteria are oriented to removing items that we consider do not provide relevant information to the research or do not contain complete information or are not available.

EX01: Remove the duplicates found in the databases.

EX02: Remove items that are not research papers. (Unless there may be other types of work with a significant contribution.)

EX03: Remove papers if only the abstract but not the full text is available.

EX04: Remove results not written in English.

EX05: Results that do not introduce and describe the concepts of quantum computing and/or quantum software engineering.

EX06: Results that do not discuss the association between quantum computing and/or quantum software engineering.

Results of selection: We applied the exclusion criteria EX01, EX02, EX03, and EX04, as well as the filters of level 1, having the selection expressed in Table 2. Appendix A labels the papers consecutively numbered as S1, S2, …Sn.

Table 2. Results from the selection: Filters Level 1 and exclusion criteria

Database	Papers selected
IEEE Xplore	87
ACM	43
ScienceDirect	13

As we mentioned earlier, we considered all the papers in this selection stage, trying to consider a wider spectrum of literature to detect the fundamental concepts. Furthermore, we include additional sources to complete the report.

4.4. Keywording of Abstracts (Outcome: Classification Scheme)

According to the objective of the research, we created four categories for classifying the selected papers: (1) Quantum computing basics and tendencies, (2) Combination of software engineering and quantum computing, (3) Education in quantum computing, and (4) Skills for quantum computing.

Category 1 is expected to include papers containing descriptions of quantum computing fundamentals and topics related to tendencies in research in quantum computing and quantum technology development.

Category 2 is expected to include papers that emphasize the connection between quantum computing and software engineering, describing advances in the quantum-oriented approach of software engineering concepts.

Category 3 is expected to include papers that explicitly contain proposals or studies for quantum computing or quantum software engineering education.

Category 4 is expected to include papers that explicitly emphasize the required skills and competencies for quantum computing and related topics, such as quantum hardware construction and quantum programming.

4.5. Data Extraction and Mapping of Studies (Outcome: Systematic Map)

In Table 3 we present the grouping of papers in terms of the four categories indicated.

As we can see, Category 1 has the highest frequency of papers, followed by Category 2. As we mentioned in the introductory sections, education in quantum computing is not much attended (Category 3), nor the skills required for quantum computing (Category 4).

The results from the analysis of the mapping are presented in the next section, answering the research questions.

5. Results

In this section we present the results, answering the research questions. For some questions we considered the papers resulting from the search; for other questions, we considered complimentary literature which provides significant contributions to our research project.

RQ1: What is the definition of Quantum Computing?

These definitions were extracted from the reviewed literature and supporting literature. We chose those papers with definitions [46-50] that involve several aspects of quantum computing. Three definitions are presented next.

Definition 1 [46]: "Quantum Computing is a paradigm that intersects computer science, mathematics, and physics. Unlike other computing fields, quantum computing uses the law of quantum mechanics with the goal of achieving high computation efficiency." The focus of Quantum Computing is the issue of storing, handling and transmitting data stored in quantum mechanical systems. This data mode is therefore referred to as quantum information known as Qubit. Mathematically, a qubit may be denoted with the help of a vector $|\psi\rangle$ in the two-dimensional complex vector area which has a related inner product, such that $|\psi\rangle \epsilon$ H2 [47].

Definition 2: Quantum computing is referred to as "the field of science which directly uses quantum mechanical phenomena like superposition and entanglement to perform operations on data" [48]. Quantum computing studies quantum computers with the quantum mechanics' phenomenon of superposition, entanglement, tunneling, and annealing to solve problems that cannot be solved in the life span of human beings [49].

Definition 3: Quantum computing works with abstract units named quantum bits, so the basic storage unit is the quantum bit (Qubit) [50]. Quantum bits, or qubits, are considered like quantum particles. The manipulation of qubits by control devices is at the core of a quantum computer's processing power.

RQ2: What are the fundamental concepts of Quantum Computing?

To answer this question, we present a kind of taxonomy at distinct levels.

For this part of our research, we implemented three steps of the process to build a taxonomy: (1) Determine requirements, (2) identify the concepts, and (3) develop a first version of the taxonomy. The remaining steps are proposed as part of the future work. We considered the following criteria: (a) Terms should be unambiguous and clear, yet not too wordy and long; (b) hierarchical relationships between concepts are of type generic-specific, whole-part, instance; and (c) the structure, for the main concepts' components, depth is about three or four levels, except for entities representing processes or subfields.

The first level integrates the four main supporting disciplines of quantum computing, which are: physics, computer science, mathematics, and electronics. See Fig. 1.

In the second level of taxonomy, we include subfields involved. See Fig. 2-4.

Quantum mechanics is the subfield of physics, which provides the essential principles that quantum computing is based on, and it is strongly related to mathematics. Four subfields of mathematics give support to quantum computing (see Fig. 2): basic logic, linear algebra, numbers theory, and probability. From linear algebra, vectors, matrixes, and operations with them are the core of qubits representation and processing, at the level of qubits and quantum gates. From numbers theory, complex numbers are used to try with the coefficients of the terms of superposed states of qubits.

Table 3. Grouping of papers into categories

Category	Related papers
1.Quantum computing basics and tendencies	S4, S8, S9, S10, S12, S13, S20, S21, S22, S27, S28, S29, S31, S33, S34, S37, S38, S41, S42, S43, S44, S46, S47, S48, S51, S52, S53, S54, S55, S56, S57, S58, S61, S62, S65, S67, S68, S69, S70, S71, S72, S73, S74, S75, S76, S77, S78, S79, S80, S81, S83, S84, S85, S86, S87, S88, S89, S90, S92, S92, S93, S94, S95, S98, S99, S101, S102, S103, S104, S105, S106, S107, S108, S109, S110, S111, S112, S113, S114, S115, S116, S117, S118, S119, S122, S123, S124, S125, S126, S127, S128, S129, S130, S131, S132, S133, S134, S135, S136, S137, S138, S139, S140
2. Software engineering and quantum computing	S1, S2, S5, S6, S7, S9, S14, S15, S17, S19, S19, S23, S24, S26, S30, S32, S35, S36, S39, S40, S45, S49, S50, S56, S59, S60, S63, S64, S66, S82, S91, S96, S97, S100, S120, S121
3. Education in quantum computing	S16, S25, S55
4. Skills for quantum computing	S3

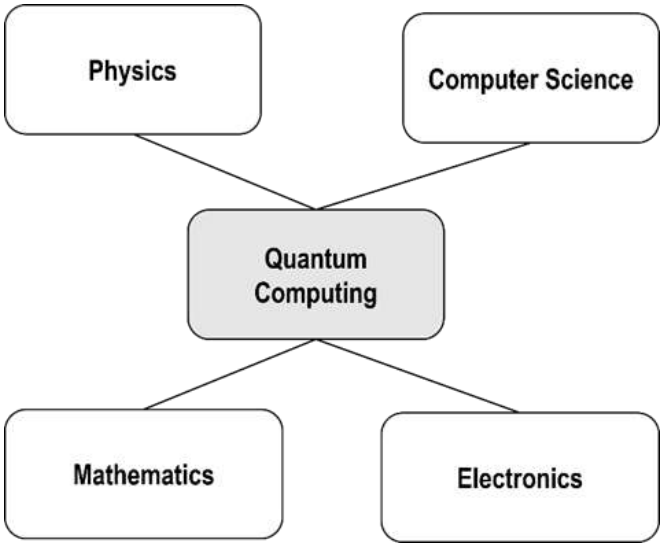


Fig. 1. Disciplines involved in quantum computing

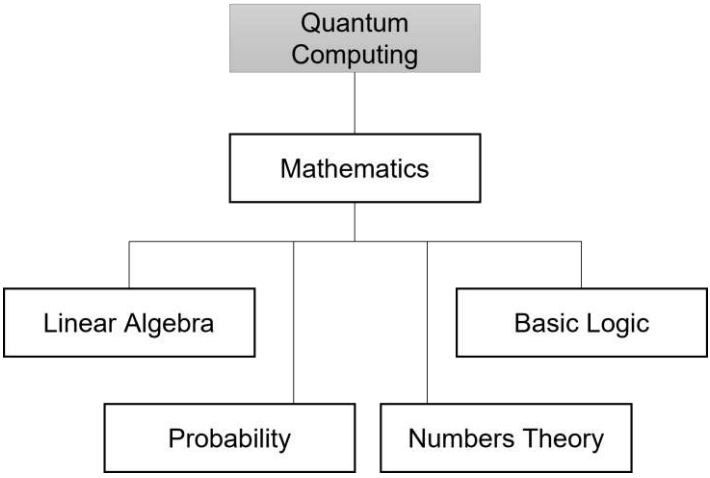


Fig. 2. Areas of mathematics involved in quantum computing

In the case of computer science (see Fig. 3), for the practical implementations of quantum computation, these concepts are involved: computer architecture, programming languages, programming fundamentals, and algorithms. Computer architecture concepts and elements are needed to configure the real expression quantum computation on hardware.

Programming language elements such as syntax, semantics, and capabilities are especially important for the human conceptualization of quantum computing and to indicate computations to do by quantum computers.

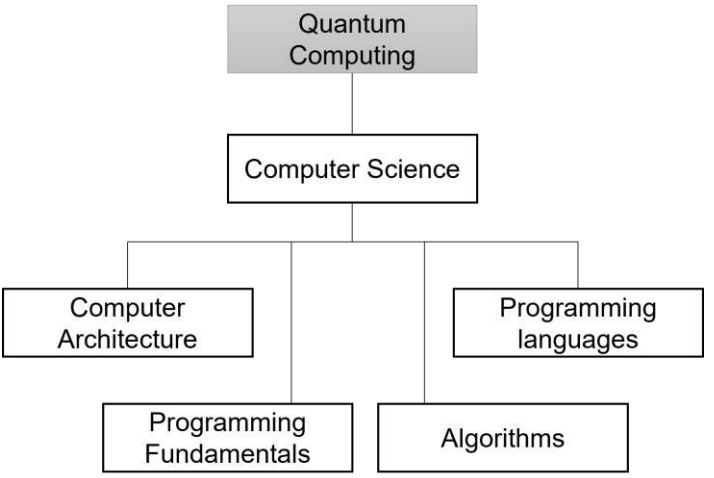


Fig. 3. Areas of computer science involved in quantum computing

Programming fundamentals are the core elements for indicating the logic and the flow of computations, processing information, as well as for visioning the target solution. Algorithms in quantum computing are important tools because they encompass quantum logic and qubits transformations needed to perform quantum computation.

In the case of basic electronics, it gives support for figuring out the practical implementation of quantum computing. Quantum logic gates and circuits allow to implement complete quantum programming and algorithms (see Fig. 4).

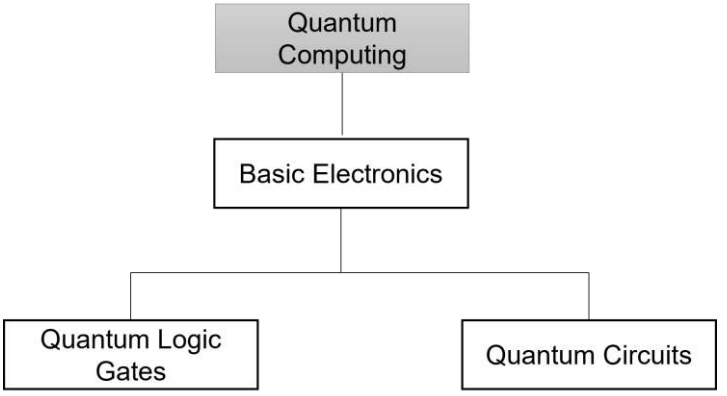


Fig. 4. Areas of electronics involved in quantum computing.

In a third level of taxonomy, we introduce the main concepts of quantum mechanics: Superposition, entanglement, uncertainty principle, probability principle, measurement, decoherence, and non-locality. See Fig. 5.

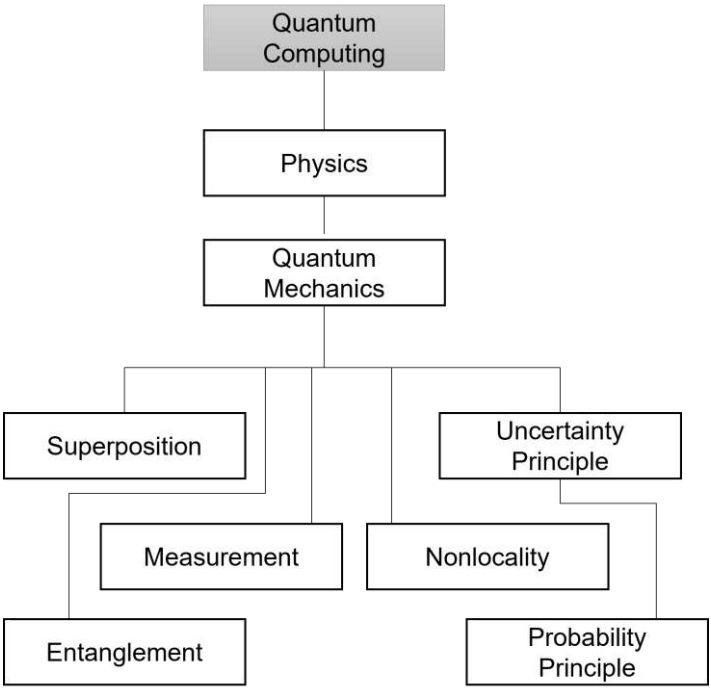


Figure 5. Concepts of quantum mechanics involved in quantum computing.

Superposition: In the quantum mechanics context, the superposition principle is the idea that a system (or an atomic particle) is in all possible states at the same time, until it is measured [51]; after measurement, it then falls to one of the basis states that form the superposition, thus destroying the original configuration. Such phenomenon is applied to the qubits, so that, a qubit can take, in addition to the basic states 0 and 1, a set of states resulting from the linear combination of the basic states [8, 52].

Entanglement: This is a state where two or more particles are generated so that the quantum state of a single particle cannot be destroyed independently. No matter how far the particles are, they will reflect the same quantum state [49]. Any change at one end in the quantum state will be responsible for the change in the other particle. In the case of qubits, entanglement is the ability of qubits to correlate their state with other qubits.

The properties of the two qubits in an entangled state are linked to each other such that by looking (i.e., measuring) one of them, will reveal the other qubit, even when they are at physically large separations [52]. Two or more individually independent quantum objects are said to be entangled when: a) their behavior is random individually, but at the same time, and b) it is strongly correlated despite each object being independent of the other. A multiqubit state that cannot be expressed as a list of the individual constituent qubits is entangled.

Uncertainty principle of Quantum Mechanics: It was formulated in 1926 by Werner Heisenberg, this principle states that an electron, or any other particle, can never have its exact position known, or even specified. The uncertainty principle states that the position and the momentum of an elementary particle (EP) cannot be simultaneously determined with certainty [53]. Quantum systems are so small; it is impossible to measure all properties of a Quantum system without disturbing it [54-55]. As a result, there is no way of accurately predicting all the properties of a particle in a Quantum System.

Probability principle [56-58]: A first general principle in quantum mechanics is that the probability that a particle will arrive at x position when let out at the source s can be represented quantitatively by the absolute square of a complex number called a probability amplitude—in this case, the "amplitude that a particle from a starting point s will arrive at position x."

In quantum mechanics, particles do not have classical properties like "position" or "momentum," rather, there is a wave function that assigns a (complex) number, called the "amplitude," to each measurement outcome. The Born Rule is then quite simple: *it says that the probability of obtaining any possible measurement's outcome is equal to the square of the corresponding amplitude*. (The wave function is just the set of all the amplitudes).

Born Rule: Probability (x) = |amplitude(x)|2

In quantum mechanics, a probability amplitude is a complex number used for describing the behavior of systems. The modulus squared of this quantity represents a probability density. Probability amplitudes provide a relationship between the quantum state vector of a system and the results of observations of that system, a link was first proposed by Max Born, in 1926. The interpretation of values of a wave function as the probability amplitude is a pillar of the Copenhagen interpretation of quantum mechanics.

Measurement principle: In quantum physics, measurement is the testing or manipulation of a physical system to yield a numerical result [59]. The predictions that quantum physics makes are in general probabilistic. For each measurement that can be defined, the probability distribution over the outcomes of that measurement can be computed from the density operator.

Decoherence: Quantum decoherence is the loss of quantum coherence [8], the process in which a system's behavior changes from that which can be explained by quantum mechanics to that which can be explained by classical mechanics. Decoherence can be viewed as the loss of information from a system into the environment (often modeled as a heat bath) [60] since every system is loosely coupled with the energetic state of its surroundings.

Non-locality: In [61], it is indicated that non-locality is the most characteristic feature of quantum mechanics, but recent research suggests the possible existence of non-local correlations stronger than those predicted by theory. This raises the question of whether nature is in fact more non-local than expected from quantum theory or, alternatively, whether there could be a yet undiscovered principle limiting the strength of non-local correlations.

RQ3: What are the supporting concepts for implementing Quantum Computing?

Quantum computing as an engineering discipline is still in its infancy [62-63], and only some practical prototypes have been announced. Quantum hardware is going to be implemented, it is a clear objective [64]; the principal manufacturers are creating different technologies.

By today, there are efforts in developing quantum technology applications with practical realizations using photons, atoms, and electrons [65]. In [22], it is presented a list of major hardware candidates for industrial quantum computer and their properties: Qubit Technologies, Trapped Ion Qubits, Super-conducting Qubits, Silicon Qubits, Photonic Qubits, and Topological Qubits.

By this time, there are simulators running on personal computers, which allows the user to run quantum programs. The current implementations of quantum computing are based on quantum gates and quantum algorithms.

Quantum gates are quantum logic objects, and they have a mathematical expression (see examples of Hadamard and CNOT gates in previous sections).

Based on this, we consider as supporting concepts the following: quantum gates, quantum circuits, mathematical expressions, and quantum algorithms. In Fig. 6 we present a kind of quantum gates taxonomy.

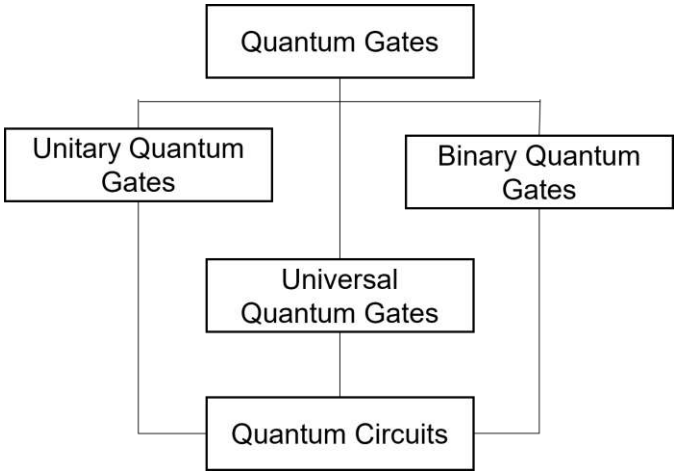


Fig. 6. Concepts of electronics involved in quantum computing

RQ4: What are the new Software Engineering concepts emerging from Quantum Computing?

The connection between quantum computing and software engineering is clearly deduced, software is an essential part of quantum computing, and developing software is the main objective of software engineering. In this section, we present the main concepts detected about software engineering, which emerged from the context of quantum computing.

We present eight elements of quantum software engineering (see Fig. 7, 8): (1) quantum software life cycle, (2) quantum software processes, (3) quantum software modeling, (4) quantum software implementation, (5) quantum software quality assurance, (6) quantum software programming, (7) quantum software tools, and (8) quantum software maintenance. Quantum software modeling is divided into quantum software analysis and quantum software design. Next, we will describe some of them.

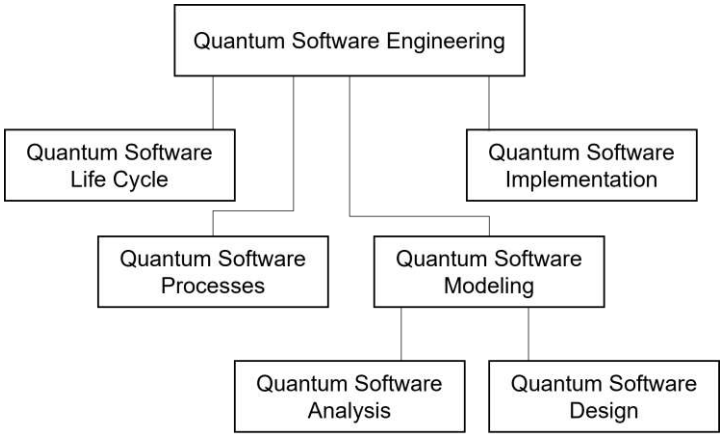


Figure 7. Areas of Quantum Software Engineering. Part 1

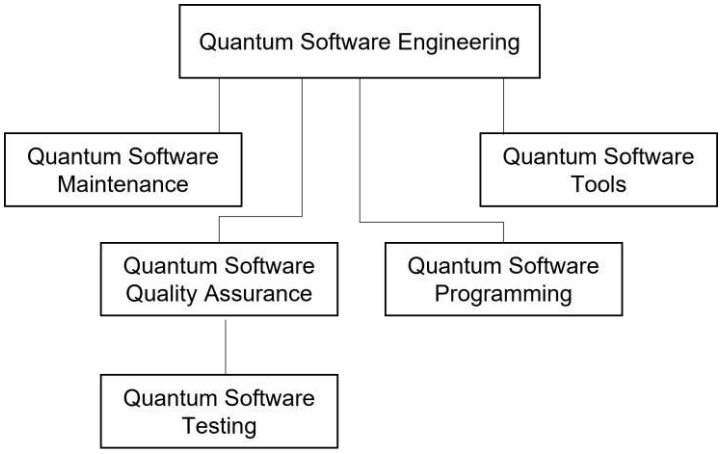


Figure 8. Areas of Quantum Software Engineering. Part 2

Quantum software lifecycle. There are proposals for the lifecycle of quantum software, such as [11, 44, 64]. They propose a kind of traditional main phase: requirements, analysis, design, implementation, testing, and maintenance. Other proposals include specific subphases according to the quantum nature. A generic proposal is presented in [52], which describes the phases according to their quantum nature. The lifecycle for quantum software is divided into ten phases (See Fig. 9, 10).

The principal activities in each phase are cited below:

1. *Quantum-Classical Splitting*: Problem separation in classical and quantum parts. E.g., manual by experts, decision support based on patterns.
2. Hardware-independent Implementation: Quantum circuit & classical software artifacts, testing & verification of circuits.
3. Quantum Circuit Enrichment: Data preparation/initialization, oracle expansion.
4. Hardware-independent Optimization: Removal of unnecessary gates or qubits, based on cost functions (e.g., circuit depth, accuracy).
5. *Quantum Hardware Selection*: Analysis of quantum circuits, selection of suitable hardware (e.g., based on metrics or benchmarks).
6. *Readout-Error Mitigation Preparation*: Analysis of selected hardware, and determination of the error model (e.g., calculation of the correction matrix).
7. *Compilation & Hardware dependent Optimization*: Optimizations based on hardware characteristics, compilation to machine instructions.
8. *Integration*: Deployment of classical software artifacts, and provisioning of quantum resources.
9. *Execution*: On heterogeneous hardware, multiple iterations for variational algorithms, readout-error mitigation.
10. *Result Analysis*: Result verification, return to the user or next iteration for improvement.

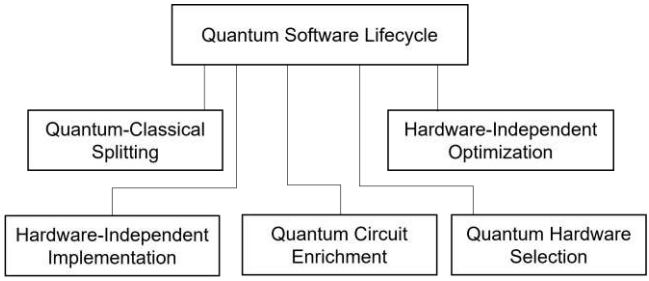


Fig. 9. Stages of the Quantum Lifecycle. Part 1

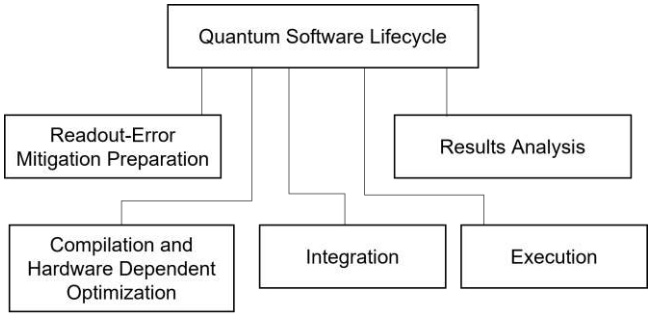


Fig. 10. Stages of the Quantum Life Cycle. Part 2

Quantum software modeling. The design is one of the most cited in the literature, this is because both quantum circuits and quantum algorithms are the basis of quantum software, so this is a kind of low-level design. However, from the software engineering point of view, the design phase is required to be supported by techniques like those for developing classical software, in this case, it is needed to create techniques and tools for quantum requirements analysis and quantum modeling. UML is a representative tool to support software analysis and design, and it can be used as a starting point.

There are a few proposals about modeling techniques for quantum software. Some observations are stated to establish the basis to create supporting tools for the design phase [66]. The central difference between quantum and classical computation is in how it achieves its goals. Quantum computers have access to quantum algorithms [67], and quantum data structures [68], that are unavailable to classical computers—hence their performance advantage. Algorithms and data structures are, however, implementation details. Algorithms are an essential design choice while programming in the small. However, they are not completely ignored in large-scale software architectural design. For instance, UML diagrams seldom portray algorithms and data structures beyond a very high-level design perspective.

In classic Software Engineering, at the software modeling phase, there are in total 14 types of UML diagrams, split into two categories [69]: structure and behavior diagrams.

In [66], the authors propose to create an extension of UML, called Q-UML, which follows the guiding principles behind any quantum software modeling language: Quantum classes, quantum elements (quantum variables, quantum operations), quantum supremacy, and quantum aggregation). These elements make a difference with respect to classical software modeling.

In [69] the author suggests that of the fourteen diagram types in UML, the most widely used (and hence important) diagrams are: use case, class, object, state machine, sequence, and activity diagrams. Based on this, the author presents a proposal for Q-UML diagrams, as we can see in Fig. 11. The Q-UML diagrams contain similar elements to UML diagrams, distinguishing the quantum

elements with highlights such as bold text and double-lines to portray quantum information textually and pictorially.

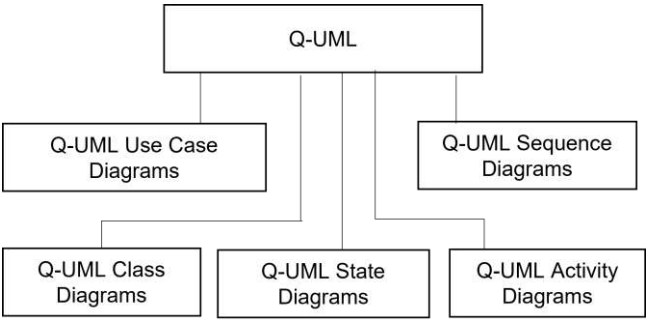


Fig. 11. Q-UML basic diagrams

Quantum programming languages: Quantum programming is the process of assembling sequences of instructions, called quantum circuits, which can run on a quantum computer. Quantum programming languages help express quantum algorithms using high-level constructs [70]. Quantum programming languages are used for controlling existing physical devices, for estimating the execution costs of quantum algorithms on future devices, for teaching quantum computing concepts, or for verifying quantum algorithms and their implementations [71].

In [72] a classification of quantum programming languages is presented (see Fig. 12), which is considered as a starting point for studying quantum programming. This taxonomy has two branches, *programming languages*, and *programming tools*.

Quantum software testing. By nature, quantum software is error susceptible. Quantum physics properties as superposition mean that quantum computers deliver probabilistic measures when classical observations are made on qubits; that is when a qubit in a superposition state is collapsed into a classical value, it takes a given value with a given probability [73]. Some selected publications address quantum computing validation from a probabilistic perspective from circuit and software levels. The behavior of quantum circuits is inherently probabilistic, so while the goal of traditional testing has always been to detect the presence of faults, probabilistic testing aims to estimate fault probability.

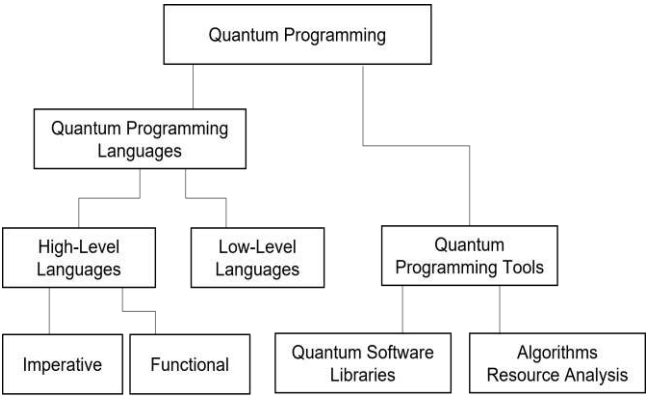


Fig. 12. Quantum programming branches

The testing phase is one of the most covered by hardware producers and scientists. In Fig. 13 we show a classification of types of testing techniques for quantum software. It was extracted from [63, 74-84], which treat specific techniques and tools for debugging and testing.

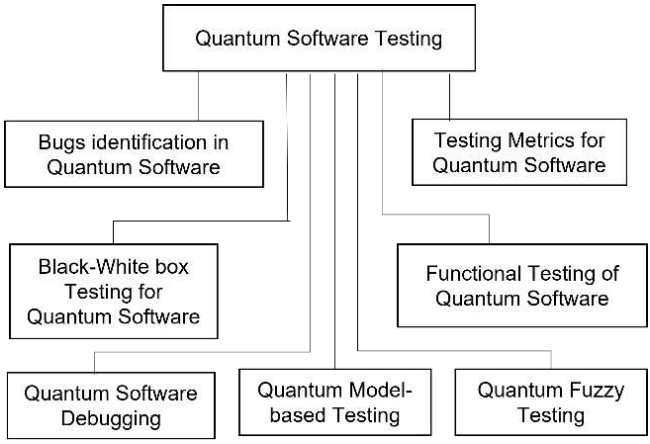


Fig. 13. Quantum programming branches

As we can see, the classification includes bug detection and debugging as part of the quality assurance. White-Black box testing and functional testing are like classical software testing applied to quantum software. Specialized testing is included, such as fuzz testing [80, 84], which is an automated software testing method that injects invalid, malformed, or unexpected inputs into a system to reveal software defects and vulnerabilities. Quantum noise provides an effective built-in fuzzing capability that is centered around the actual answer to a computation. Specialized techniques are introduced for testing quantum software, such as search-based [79], mutation-based [24, 76, 82], and property-based [24, 78].

6. Discussion

Considering the results presented in the previous section we can describe the following facts.

6.1. Tendencies in the coverture of quantum fundamental concepts

- In this sample of literature review, it is seen that there are more sources on quantum computing essentials and tendencies (Category 1). Most research has been done on quantum algorithms, hardware prototyping, and error mitigation.
- Even when these publications attend essential aspects of quantum computing, some of them do not include a full description of the quantum mechanics principles that support quantum computing, commonly they include references to earlier publications and books.
- Few publications exist talking about quantum computing education, as well as few formal publications exist talking about the skills required for practicing quantum computing (Categories 3, 4).

6.2. Tendencies in the coverture quantum software development

- Due to the nature of quantum computing and its exposure to errors, debugging and evaluating such errors has been a problem to address since the beginning of quantum computing practical implementations. Then, more progress is registered for quantum software testing.

- Modeling quantum software shows advances in modeling circuits and quantum algorithms; it is a kind of inherent design required. However, high-level modeling is still in its infancy.
- There is a proliferation in quantum programming languages. At the high-level category, there are several languages based on instructions that implement the quantum mechanics principles, such as superposition, entanglement, and measurement.

6.3. Benefits of the presented results of this review

- The related work cited in this paper presents significant effort in literature review and taxonomical proposals of specific aspects of quantum computing, especially at the level of main aspects, technologies, and use cases. However, the fundamental concepts are presented in a descriptive way, not in a taxonomical view.
- The taxonomical view presented in our paper will allow us to present the knowledge better organized. Also, it represents a starting point to document the principal part of a body of knowledge, the fundamental concepts.

7. Conclusions and future work

A new paradigm of computation known as quantum computing is founded on the ideas of quantum theory, which is concerned with contemporary physics, particularly quantum mechanics. The main goal of quantum mechanics is to describe how matter and energy behave at the atomic and subatomic scales. Quantum computing makes use of quantum phenomena, such as superposition and entanglement to perform data operations. Those principles are becoming popular due to their importance, such as entanglement [85] which is particularly useful in information security issues.

The multi-disciplinarity of quantum computing, especially the quantum mechanics principles and the mathematical expression of qubits and their treatment, make this paradigm difficult to understand by non-physics experts [23-24].

To reduce the complication of understanding and mastering quantum computing, and for addressing the study of this field and the implementation of quantum software, practitioners (academicians and industry professionals) require a kind of taxonomic view of the fundamental concepts of quantum computing and the supporting ones.

In this paper, we presented a quasi-systematic mapping, performed to:

1. Identify the fundamental concepts involved in quantum computing and quantum software engineering.
2. Build a taxonomy to include the main concepts of both disciplines.
3. Identify a set of fundamental concepts to integrate the first element of a body of knowledge.

Software engineering education has implications for the software industry in emerging countries [86] and developed countries. Education and training in software development require considering both hard and soft skills [87]; this is the same for quantum software development.

The proposed taxonomy represents an initial proposal of the body of knowledge of quantum computing, especially for software development, so it can contribute to education in quantum computing and quantum software engineering.

For future work, we identify the next actions:

1. To complete the analysis of the results from all databases recommended for any literature review, and to find how the literature covers the fundamental concepts of both disciplines.
2. To formalize the integration of the taxonomy, with a kind of theoretic validation or with judges from experts.

3. To assess the usefulness of the first levels of the taxonomy, in different scenarios such as guiding curricula creation and course design.
4. To continue with the literature review to gather the remaining elements of a body of knowledge, such as practices and methods.

References

[1]. M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, 10th edition. Cambridge, United Kingdom: Cambridge University Press, 2016.
[2]. A. A. Khan et al., “Agile Practices for Quantum Software Development: Practitioners Perspectives,” ArXiv, vol. abs/2210.09825, 2022.
[3]. A. J. et al., “Quantum Algorithm Implementations for Beginners,” Apr. 2018, doi: 10.1145/3517340.
[4]. J. J. García-Ripoll, “Specialty Grand Challenge: Quantum engineering,” Frontiers in Quantum Science and Technology, vol. 1, Sep. 2022, doi: 10.3389/frqst.2022.1029525.
[5]. S. T. Marella and H. S. K. Parisa, “Introduction to Quantum Computing,” in Quantum Computing and Communications, Y. Zhao, Ed., Rijeka: IntechOpen, 2020, p. Ch. 5. doi: 10.5772/intechopen.94103.
[6]. M. Weingärtner and T. Weingärtner, “Quantum Tic-Tac-Toe - learning the concepts of quantum mechanics in a playful way,” Computers and Education Open, vol. 4, p. 100125, 2023, doi: https://doi.org/10.1016/j.caeo.2023.100125.
[7]. U. Awan, L. Hannola, A. Tandon, R. K. Goyal, and A. Dhir, “Quantum computing challenges in the software industry. A fuzzy AHP-based approach,” Inf Softw Technol, vol. 147, p. 106896, 2022, doi: https://doi.org/10.1016/j.infsof.2022.106896.
[8]. T. G. Wong, Introduction to Classical and Quantum Computing. Omaha, Nebraska: Rooted Grove, 2022.
[9]. N. S. Yanofsky and M. A. Mannucci, Quantum computing for computer scientists. New York, NY: Cambridge University Press, 2008.
[10]. E. Rieffel and W. Polak, QUANTUM COMPUTING A Gentle Introduction. Cambridge, Massachusetts: The MIT Press, 2011.
[11]. J. Zhao, Quantum Software Engineering: Landscapes and Horizons. 2020.
[12]. P. Zhao, J. Zhao, and L. Ma, “Identifying Bug Patterns in Quantum Programs,” in 2021 IEEE/ACM 2nd International Workshop on Quantum Software Engineering (Q-SE), 2021, pp. 16–21. doi: 10.1109/Q-SE52541.2021.00011.
[13]. P. P. Angara, U. Stege, A. MacLean, H. A. Müller, and T. Markham, “Teaching Quantum Computing to High-School-Aged Youth: A Hands-On Approach,” IEEE Transactions on Quantum Engineering, vol. 3, pp. 1–15, 2022, doi: 10.1109/TQE.2021.3127503.
[14]. M. Mykhailova and K. M. Svore, “Teaching Quantum Computing through a Practical Software-Driven Approach: Experience Report,” in Proceedings of the 51st ACM Technical Symposium on Computer Science Education, in SIGCSE ’20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1019–1025. doi: 10.1145/3328778.3366952.
[15]. J. Singh and K. S. Bhangu, “Contemporary Quantum Computing Use Cases: Taxonomy, Review and Challenges,” Archives of Computational Methods in Engineering, vol. 30, no. 1, pp. 615–638, 2023, doi: 10.1007/s11831-022-09809-5.
[16]. C. D. Aiello et al., “Achieving a quantum smart workforce,” Quantum Sci Technol, vol. 6, no. 3, p. 030501, Jul. 2021, doi: 10.1088/2058-9565/abfa64.
[17]. G. R. Oliver, Foundations of the assumed business operations and strategy body of knowledge (BOSBOK) : an outline of shareable knowledge. Sidney: Sydney University Press, 2012.
[18]. P. Bourque and R. E. Fairley, “Guide to the Software Engineering Body of Knowledge, Version 3.0,” 2014.
[19]. T. I. Ören, “Toward the Body of Knowledge of Modeling and Simulation,” in Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2005, 2005.
[20]. G. Romme, The Quest for Professionalism: The Case of Management and Entrepreneurship. Oxford: Oxford University Press, 2016.
[21]. H. Hart and C. Baehr, “Sustainable Practices for Developing a Body of Knowledge,” Tech Commun, vol. 60, no. 4, pp. 259–266, 2013, [Online]. Available: https://www.jstor.org/stable/26464355
[22]. S. S. Gill et al., “Quantum Computing: A Taxonomy, Systematic Review and Future Directions,” Sep. 2020, doi: 10.48550/arxiv.2010.15559.

[23]. L. S. Barbosa, "Software Engineering for 'Quantum Advantage,'" in Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops, in ICSEW'20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 427–429. doi: 10.1145/3387940.3392184.

[24]. G. Pontolillo and M. R. Mousavi, "A Multi-Lingual Benchmark for Property-Based Testing of Quantum Programs," in 2022 IEEE/ACM 3rd International Workshop on Quantum Software Engineering (Q-SE), 2022, pp. 1–7. doi: 10.1145/3528230.3528395.

[25]. Y.-P. Liao, Y.-L. Cheng, Y.-T. Zhang, H.-X. Wu, and R.-C. Lu, "The interactive system of Bloch sphere for quantum computing education," in 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), 2022, pp. 718–723. doi: 10.1109/QCE53715.2022.00097.

[26]. G. P. Temporão, T. B. S. Guerreiro, P. S. C. Ripper, and A. M. B. Pavani, "Teaching Quantum Computing without prerequisites: a case study," in 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), 2022, pp. 673–676. doi: 10.1109/QCE53715.2022.00090.

[27]. M. Mykhailova, "Developing Programming Assignments for Teaching Quantum Computing and Quantum Programming," in 2022 IEEE International Conference on Quantum Computing and Engineering (QCE), 2022, pp. 688–692. doi: 10.1109/QCE53715.2022.00092.

[28]. C. Hughes, D. Finke, D.-A. German, C. Merzbacher, P. M. Vora, and H. J. Lewandowski, "Assessing the Needs of the Quantum Industry," Aug. 2021, doi: 10.48550/arxiv.2109.03601.

[29]. K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic Mapping Studies in Software Engineering," pp. 1–10, 2008, doi: 10.14236/ewic/EASE2008.8.

[30]. B. Barn, S. Barat, and T. Clark, "Conducting Systematic Literature Reviews and Systematic Mapping Studies," in Proceedings of the 10th Innovations in Software Engineering Conference, in ISEC '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 212–213. doi: 10.1145/3021460.3021489.

[31]. K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," Inf Softw Technol, vol. 64, pp. 1–18, 2015, doi: https://doi.org/10.1016/j.infsof.2015.03.007.

[32]. B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," Technical report, EBSE Technical Report EBSE-2007-01, 2007. [Online]. Available: https://www.cs.auckland.ac.nz/~norsaremah/2007%20Guidelines%20for%20performing%20SLR%20in%20SE%20v2.3.pdf

[33]. B. A. Kitchenham, "Systematic reviews," in 10th International Symposium on Software Metrics, 2004. Proceedings., 2004, pp. xii–xii. doi: 10.1109/METRIC.2004.1357885.

[34]. J. Hannay, D. Sjøberg, and T. Dybå, "A Systematic Review of Theory Use in Software Engineering Experiments," Software Engineering, IEEE Transactions on, vol. 33, pp. 87–107, Mar. 2007, doi: 10.1109/TSE.2007.12.

[35]. G. Tebes, D. Peppino, P. Becker, and L. Olsina, Enhancing the Process Specification for Systematic Literature Reviews. 2019. doi: 10.13140/RG.2.2.14262.96321/1.

[36]. P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," Journal of Systems and Software, vol. 80, no. 4, pp. 571–583, 2007, doi: https://doi.org/10.1016/j.jss.2006.07.009.

[37]. B. Kitchenham, E. Mendes, and G. Travassos, "Cross versus Within-Company Cost Estimation Studies: A Systematic Review," Software Engineering, IEEE Transactions on, vol. 33, pp. 316–329, Jun. 2007, doi: 10.1109/TSE.2007.1001.

[38]. M. R. W. Hiebl, "Sample Selection in Systematic Literature Reviews of Management Research," Organ Res Methods, vol. 26, no. 2, pp. 229–261, Jan. 2021, doi: 10.1177/1094428120986851.

[39]. V. Garousi, M. Felderer, and M. V Mäntylä, "Guidelines for including grey literature and conducting multivocal literature reviews in software engineering," Inf Softw Technol, vol. 106, pp. 101–121, 2019, doi: https://doi.org/10.1016/j.infsof.2018.09.006.

[40]. A. Sarkar, "Automated Quantum Software Engineering: why? what? how?," Dec. 2022, doi: 10.48550/arxiv.2212.00619.

[41]. B. Weder, J. Barzen, F. Leymann, and D. Vietz, "Quantum Software Development Lifecycle," Jun. 2021, doi: 10.48550/arxiv.2106.09323.

[42]. A. A. Khan et al., "Software Architecture for Quantum Computing Systems -- A Systematic Review," Feb. 2022, Accessed: Mar. 18, 2023. [Online]. Available: http://arxiv.org/abs/2202.05505

[43]. M. De Stefano, F. Pecorelli, D. Di Nucci, F. Palomba, and A. De Lucia, “Software Engineering for Quantum Programming: How Far Are We?,” Mar. 2022, doi: 10.48550/arxiv.2203.16969.
[44]. M. A. Serrano, J. A. Cruz-Lemus, R. Perez-Castillo, and M. Piattini, “Quantum Software Components and Platforms: Overview and Quality Assessment,” ACM Comput. Surv., vol. 55, no. 8, Dec. 2022, doi: 10.1145/3548679.
[45]. M. Piattini and J. M. Murillo, “Quantum Software Engineering Landscape and Challenges,” in Quantum Software Engineering, M. A. Serrano, R. Pérez-Castillo, and M. Piattini, Eds., Cham: Springer International Publishing, 2022, pp. 25–38. doi: 10.1007/978-3-031-05324-5_2.
[46]. M. Openja, M. M. Morovati, L. An, F. Khomh, and M. Abidi, “Technical debts and faults in open-source quantum software systems: An empirical study,” Journal of Systems and Software, vol. 193, p. 111458, 2022, doi: https://doi.org/10.1016/j.jss.2022.111458.
[47]. S. Singh, M. T. Pandian, A. K. Aggarwal, S. P. Awasthi, H. Bhardwaj, and J. Pruthi, “Quantum learning theory: A classical perspective for quantum image,” Mater Today Proc, vol. 80, pp. 2786–2793, 2023, doi: https://doi.org/10.1016/j.matpr.2021.07.039.
[48]. G. Arun and V. Mishra, “A review on quantum computing and communication,” in 2014 2nd International Conference on Emerging Technology Trends in Electronics, Communication and Networking, 2014, pp. 1–5. doi: 10.1109/ET2ECN.2014.7044953.
[49]. J. Singh and M. Singh, “Evolution in Quantum Computing,” in 2016 International Conference System Modeling & Advancement in Research Trends (SMART), 2016, pp. 267–270. doi: 10.1109/SYSMART.2016.7894533.
[50]. S. Aralikatti, “Quantum Computing: Challenges and Opportunities,” in 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2021, pp. 1–4. doi: 10.1109/ICECCT52121.2021.9616647.
[51]. M. Dickson, “NON-RELATIVISTIC QUANTUM MECHANICS,” in Philosophy of Physics, J. Butterfield and J. Earman, Eds., Amsterdam: North-Holland, 2007, pp. 275–415. doi: https://doi.org/10.1016/B978-044451560-5/50007-5.
[52]. B. Sodhi and R. Kapur, “Quantum Computing Platforms: Assessing the Impact on Quality Attributes and SDLC Activities,” in 2021 IEEE 18th International Conference on Software Architecture (ICSA), 2021, pp. 80–91. doi: 10.1109/ICSA51549.2021.00016.
[53]. J. Bub, “Quantum Mechanics as a Principle Theory,” Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics, vol. 31, no. 1, pp. 75–94, 2000, doi: https://doi.org/10.1016/S1355-2198(99)00032-5.
[54]. J. Hilgevoord, “The uncertainty principle for energy and time,” Am J Phys, vol. 64, no. 12, pp. 1451–1456, Dec. 1996, doi: 10.1119/1.18410.
[55]. J. Hilgevoord, “The uncertainty principle for energy and time. II,” Am J Phys, vol. 66, no. 5, pp. 396–402, May 1998, doi: 10.1119/1.18880.
[56]. E. Sakai, “On the principles of quantum mechanics,” May 2004, doi: 10.48550/arxiv.quant-ph/0405069.
[57]. D. J. Velleman, “Probability and quantum mechanics,” Am J Phys, vol. 66, no. 11, pp. 967–969, Nov. 1998, doi: 10.1119/1.19007.
[58]. W. B. Hodge, S. V Migirditch, and W. C. Kerr, “Electron spin and probability current density in quantum mechanics,” Am J Phys, vol. 82, no. 7, pp. 681–690, Jun. 2014, doi: 10.1119/1.4868094.
[59]. L. Masanes, T. D. Galley, and M. P. Müller, “The measurement postulates of quantum mechanics are operationally redundant,” Nat Commun, vol. 10, no. 1, p. 1361, 2019, doi: 10.1038/s41467-019-09348-x.
[60]. D. Bacon, “Decoherence, Control, and Symmetry in Quantum Computers,” May 2003, Accessed: Mar. 20, 2023. [Online]. Available: https://arxiv.org/abs/quant-ph/0305025
[61]. S. Popescu, “Nonlocality beyond quantum mechanics,” Nat Phys, vol. 10, no. 4, pp. 264–270, 2014, doi: 10.1038/nphys2916.
[62]. N. C. Jones et al., “Layered Architecture for Quantum Computing,” Phys Rev X, vol. 2, no. 3, p. 31007, Jul. 2012, doi: 10.1103/PhysRevX.2.031007.
[63]. D. Fortunato, J. CAMPOS, and R. ABREU, “Mutation Testing of Quantum Programs: A Case Study With Qiskit,” IEEE Transactions on Quantum Engineering, vol. 3, pp. 1–17, 2022, doi: 10.1109/TQE.2022.3195061.
[64]. R. Van Meter and M. Oskin, “Architectural implications of quantum computing technologies,” ACM Journal on Emerging Technologies in Computing Systems (JETC), vol. 2, pp. 31–63, 2006.

[65]. A. G. Jordan, "Frontiers of research and future directions in information and communication technology," Technol Soc, vol. 30, no. 3, pp. 388–396, 2008, doi: https://doi.org/10.1016/j.techsoc.2008.05.002.
[66]. C. A. Perez-Delgado and H. G. Perez-Gonzalez, "Towards a Quantum Software Modeling Language," Jun. 2020, Accessed: Mar. 18, 2023. [Online]. Available: https://arxiv.org/abs/2006.16690
[67]. P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 124–134, 1994.
[68]. L. Zhao, C. A. Pérez-Delgado, and J. F. Fitzsimons, "Fast graph operations in quantum computation," Phys Rev A (Coll Park), vol. 93, no. 3, p. 032314, Mar. 2016, doi: 10.1103/PhysRevA.93.032314.
[69]. C. A. Pérez-Delgado, "A Quantum Software Modeling Language," in Quantum Software Engineering, M. A. Serrano, R. Pérez-Castillo, and M. Piattini, Eds., Cham: Springer International Publishing, 2022, pp. 103–119. doi: 10.1007/978-3-031-05324-5_6.
[70]. J. A. Miszczak, High-level Structures in Quantum Computing. Springer Cham, 2012.
[71]. B. Heim et al., "Quantum programming languages," Nature Reviews Physics, vol. 2, no. 12, pp. 709–722, 2020, doi: 10.1038/s42254-020-00245-7.
[72]. O. Ayoade, P. Rivas, and J. Orduz, "Artificial Intelligence Computing at the Quantum Level," Data (Basel), vol. 7, no. 3, p. 28, Feb. 2022, doi: 10.3390/data7030028.
[73]. A. García de la Barrera, I. García-Rodríguez de Guzmán, M. Polo, and M. Piattini, "Quantum software testing: State of the art," Journal of Software: Evolution and Process, vol. n/a, no. n/a, p. e2419, Dec. 2021, doi: https://doi.org/10.1002/smr.2419.
[74]. M. Paltenghi, "Cross-Platform Testing of Quantum Computing Platforms," in 2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2022, pp. 269–271. doi: 10.1145/3510454.3517061.
[75]. N. Costa, J. P. Fernandes, and R. Abreu, "Asserting the Correctness of Shor Implementations Using Metamorphic Testing," in Proceedings of the 1st International Workshop on Quantum Programming for Software Engineering, in QP4SE 2022. New York, NY, USA: Association for Computing Machinery, 2022, pp. 32–36. doi: 10.1145/3549036.3562062.
[76]. D. Fortunato, J. Campos, and R. Abreu, "QMutPy: A Mutation Testing Tool for Quantum Algorithms and Applications in Qiskit," in Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis, in ISSTA 2022. New York, NY, USA: Association for Computing Machinery, 2022, pp. 797–800. doi: 10.1145/3533767.3543296.
[77]. J. Wang, Q. Zhang, G. H. Xu, and M. Kim, "QDiff: Differential Testing of Quantum Software Stacks," in 2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE), 2021, pp. 692–704. doi: 10.1109/ASE51524.2021.9678792.
[78]. S. Honarvar, M. R. Mousavi, and R. Nagarajan, "Property-Based Testing of Quantum Programs in Q#," in Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops, in ICSEW'20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 430–435. doi: 10.1145/3387940.3391459.
[79]. X. Wang, P. Arcaini, T. Yue, and S. Ali, "QuSBT: Search-Based Testing of Quantum Programs," in Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Companion Proceedings, in ICSE '22. New York, NY, USA: Association for Computing Machinery, 2022, pp. 173–177. doi: 10.1145/3510454.3516839.
[80]. M. Trinca, J. F. Ferreira, and R. Abreu, "A Preliminary Study on Generating Well-Formed Q# Quantum Programs for Fuzz Testing," in 2022 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), 2022, pp. 118–121. doi: 10.1109/ICSTW55395.2022.00033.
[81]. J. Campos and A. Souto, "QBugs: A Collection of Reproducible Bugs in Quantum Algorithms and a Supporting Infrastructure to Enable Controlled Quantum Software Testing and Debugging Experiments," in 2021 IEEE/ACM 2nd International Workshop on Quantum Software Engineering (Q-SE), 2021, pp. 28–32. doi: 10.1109/Q-SE52541.2021.00013.
[82]. D. Fortunato, J. Campos, and R. Abreu, "Mutation Testing of Quantum Programs Written in QISKit," in 2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 2022, pp. 358–359. doi: 10.1145/3510454.3528649.
[83]. D. Zhu, S. Wang, J. Han, R. Wei, J. Wu, and L. Song, "Research on method for adding new friends to instant messaging system based on SIP in quantum communication network," in 2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2015, pp. 467–469. doi: 10.1109/ICSESS.2015.7339098.

[84]. J. Wang et al., “QuanFuzz: Fuzz Testing of Quantum Program,” ArXiv, vol. abs/1810.10310, 2018.

[85]. V. V Kornyak, “Dynamic Simulation of Quantum Entanglement in Finite Quantum Mechanics: A Computer Algebra Approach,” Programming and Computer Software, vol. 47, no. 2, pp. 124–132, 2021, doi: 10.1134/S0361768821020067.

[86]. L. Vives, K. Melendez, and A. Dávila, “ISO/IEC 29110 and Software Engineering Education: A Systematic Mapping Study,” Programming and Computer Software, vol. 48, no. 8, pp. 745–755, 2022, doi: 10.1134/S0361768822080229.

[87]. R. Juárez-Ramírez et al., “How COVID-19 Pandemic affects Software Developers’ Wellbeing, and the Necessity to strengthen Soft Skills,” Programming and Computer Software, vol. 48, no. 8, pp. 614–631, 2022, doi: 10.1134/S0361768822080047.

.

APPENDIX A. Selected papers after applying filters of level 1 and exclusion criteria

ID_SR: Source Reference extracted from the search, consecutively numbered.
ID_REF: Reference number from the list of References on the paper.

Note: A complete list of references in APA format for the papers in this Appendix can be found in the file "PCS2023_Appendix A_v6.11.pdf", at:
https://drive.google.com/file/d/1qQvusv9-Jm9iSvYAdVEtABRQXoxbvFmM/view?usp=share_link

Table 4. Papers selected from ScienceDirect

ID_SR	ID_REF	PAPER TITLE
S1	[43]	Software engineering for quantum programming: How far are we?
S2	[7]	Quantum computing challenges in the software industry. A fuzzy AHP-based approach
S3	-	Assessing attitudes towards evidence-based software engineering in a government agency
S4	-	A Comprehensive but not Complicated Survey on Quantum Computing
S5	-	Engineering the development of quantum programs: Application to the Boolean satisfiability problem
S6	-	Software modernization to embrace quantum technology
S7	[46]	Technical debts and faults in open-source quantum software systems: An empirical study
S8	-	Automated data validation: An industrial experience report
S9	[47]	Quantum learning theory: A classical perspective for quantum image
S10	-	Rapid solution of logical equivalence problems by quantum computation algorithm
S11	-	Comparing the performance of quantum-inspired evolutionary algorithms for the solution of software requirements selection problem
S12	[65]	Frontiers of research and future directions in information and communication technology
S13	-	East-West paths to unconventional computing

Table 5. Papers selected from ACM

ID_SR	ID_REF	PAPER TITLE
S14	-	Quantum computing for software engineering: prospects
S15	-	Quantum Computing: A New Software Engineering Golden Age
S16	-	Software Engineering Education of Classical Computing vs. Quantum Computing: A Competency-Centric Approach
S17	-	When software engineering meets quantum computing
S18	-	Quantum computing: synergies and opportunities
S19	[23]	Software engineering for 'quantum advantage'
S20	[50]	Quantum computing: challenges and opportunities
S21	-	Making Quantum Computing Open: Lessons from Open Source Projects
S22	-	Towards Higher-Level Abstractions for Quantum Computing
S23	[74]	Cross-platform testing of quantum computing platforms
S24	-	AI4ASE: Quantum artificial intelligence for automotive software engineering
S25	[14]	Teaching Quantum Computing through a Practical Software-driven Approach: Experience Report
S26	-	A Quantum Algorithm for Software Engineering Search
S27	-	Assessing the quantum-computing landscape
S28	-	Bugs in Quantum computing platforms: an empirical study
S29	-	Towards Quantum-algorithms-as-a-service
S30	-	Embracing iterations in Quantum software: a vision

S31	-	Automatic generation of test circuits for the verification of Quantum deterministic algorithms
S32	[75]	Asserting the correctness of Shor implementations using metamorphic testing
S33	-	The Quantum software lifecycle
S34	-	A Backend-agnostic, Quantum-classical Framework for Simulations of Chemistry in C++
S35	[66]	Towards a Quantum Software Modeling Language
S36	-	Quantum Software: Model-driven or Search-driven? A Q-SE 2021 Workshop Report
S37	-	Quantum Annealing-Based Software Components: An Experimental Case Study with SAT Solving
S38	-	Hybrid quantum-classical problem solving in the NISQ era
S39	-	An empirical study on the current adoption of quantum programming
S40	-	Using Quantum computers to speed up dynamic testing of software
S41	-	Quantum optimization for fast CAN bus intrusion detection
S42	-	About a criterion of successfully executing a circuit in the NISQ era: what wd $\ll 1/\epsilon$ eff really means
S43	-	Towards practical quantum applications using hybrid problem solving techniques
S44	[76]	QMutPy: a mutation testing tool for Quantum algorithms and applications in Qiskit
S45	[77]	QDiff: differential testing of quantum software stacks
S46	[3]	Quantum Algorithm Implementations for Beginners
S47	-	Programming quantum computers using 3-D puzzles, coffee cups, and doughnuts
S48	-	Progress and Prospects of Quantum Algorithms
S49	[78]	Property-based Testing of Quantum Programs in Q#
S50	[79]	QuSBT: search-based testing of quantum programs
S51	-	Quito: a coverage-guided test generator for quantum programs
S52	-	Silq2Qiskit - Developing a quantum language source-to-source translator
S53	-	Time-optimal Qubit mapping
S54	-	Quantum Hoare Logic with Classical Variables
S55	-	QuaFL: a typed DSL for quantum programming
S56	-	Invariants of quantum programs: characterizations and generation

Table 6. Papers selected from IEEE Xplore

ID_SR	ID_REF	PAPER TITLE
S55	[27]	Developing Programming Assignments for Teaching Quantum Computing and Quantum Programming
S56	-	Understanding Quantum Software Engineering Challenges An Empirical Study on Stack Exchange Forums and GitHub Issues
S57	-	Poster: Automatically Solving NP-Complete Problems on a Quantum Computer
S58	-	Evolution of Quantum Computing: A Systematic Survey on the Use of Quantum Computing Tools
S59	-	Towards Model-Driven Quantum Software Engineering
S60	-	Modelling Quantum Circuits with UML
S61	-	Quantum Computing: State-of-Art and Challenges
S62	-	1-2-3 Reproducibility for Quantum Software Experiments
S63	[80]	A Preliminary Study on Generating Well-Formed Q# Quantum Programs for Fuzz Testing
S64	[52]	Quantum Computing Platforms: Assessing the Impact on Quality Attributes and SDLC Activities
S65	-	A Tool For Debugging Quantum Circuits
S66	-	Quantum Software Models: The Density Matrix for Classical and Quantum Software Systems Design
S67	-	Quantum-Inspired Immune Memory Algorithm for Self-Structuring Antenna Optimization

S68	-	An Improved Multi-Objective Quantum Genetic Algorithm Based on Cellular Automaton
S69	-	Improved Quantum-Inspired Evolutionary Algorithm and Its Application to 3-SAT Problems
S70	-	Quantum Approximation on Some Classes of Multivarite Functions
S71	-	A Formal Derivation of Grover's Quantum Search Algorithm
S72	-	Explaining the Implicit Parallelism of Genetic Algorithm and Computational Complexity by Quantum Theory
S73	-	Quantum Machine Learning for Software Supply Chain Attacks: How Far Can We Go?
S74	-	Quantum phase estimation-based algorithms for machine learning
S75	-	A Review of Quantum Cybersecurity: Threats, Risks and Opportunities
S76	-	Metamorphic Testing of Oracle Quantum Programs
S77	-	Advance quantum based binary neural network learning algorithm
S78	-	An implementation of compact genetic algorithm on a quantum computer
S79	-	Quantum genetic algorithm optimized BP neural network for high-resolution remote sensing image classification
S80	-	Real-Coded Quantum Evolutionary Algorithm Based on Immune Theory for Multi-modal Optimization Problems
S81	[81]	QBugs: A Collection of Reproducible Bugs in Quantum Algorithms and a Supporting Infrastructure to Enable Controlled Quantum Software Testing and Debugging Experiments
S82	-	Some Size and Structure Metrics for Quantum Software
S83	-	Studies on the Quality Assessment of PRNG for Q-Trits Quantum Cryptography Protocols
S84	-	Accelerator circuits for quantum simulation
S85	-	Generalised Quantum Tree Search
S86	-	Multilevel Minimum Cross Entropy Threshold Selection Based on Quantum Particle Swarm Optimization
S87	-	Quantum network based on multiparty quantum secret sharing
S88	-	Quantum Determined Key Distribution Scheme Using Quantum Teleportation
S89	-	Learning quantum operator by quantum adiabatic computation
S90	[82]	Mutation Testing of Quantum Programs Written in QISKit
S91	-	Quantum Software Engineering Supremacy in Intelligent Robotics
S92	-	Compiling Quantamorphisms for the IBM Q Experience
S92	-	Is Your quantum Program Bug-Free?
S93	-	A Quantum Algorithm for Ray Casting using an Orthographic Camera
S94	-	Insight into the operation of NTRU and a comparative study of NTRU, RSA and ECC public key cryptosystems
S95	-	Visualizing time dependent semantics: an application to quantum algorithms
S96	-	Software requirements selection using Quantum-inspired Elitist Multi-objective Evolutionary algorithm
S97	-	Software requirements selection using Quantum-inspired Multi-objective Differential Evolution Algorithm
S98	-	Experimental Implementation of Discrete Time Quantum Walk with the IBM Qiskit Library
S99	-	Queueing theory study of round robin versus priority dynamic quantum time round robin scheduling algorithms
S100	-	Quantum Software as a Service Through a Quantum API Gateway
S101	[63]	Mutation Testing of Quantum Programs: A Case Study with Qiskit
S102	-	An Empirical Study of Optimizers for Quantum Machine Learning
S103	-	Performance Evaluation of TLS 1.3 Handshake on Resource-Constrained Devices Using NIST's Third Round Post-Quantum Key Encapsulation Mechanisms and Digital Signatures

S104	-	Investigating Quantum Cause-Effect Graphs
S105	-	Quantum homomorphic signature using coherent states
S106	-	Approaches to Overpower Proof-of-Work Blockchains Despite Minority
S107	-	Fake News Detection: An Application of Quantum K-Nearest Neighbors
S108	-	Nitride Semiconductor Quantum Dots - Mathematical Models of the Electronic Spectrum and Methods for its Simulation
S109	-	On quantum methods for machine learning problems, part I: Quantum tools
S110	-	On quantum methods for machine learning problems, part II: Quantum classification algorithms
S111	-	A Flowchart Language for Quantum Programming
S112	-	Training of Process Neural Networks Based on Improved Quantum Genetic Algorithm
S113	-	A novel quantum behaved Particle Swarm optimization algorithm with chaotic search for image alignment
S114	-	Quantum-Behaved Particle Swarm Optimization with Normal Cloud Mutation Operator
S115	[83]	Research on method for adding new friends to instant messaging system based on SIP in quantum communication network
S116	-	Quantum Annealing Approach for the Optimal Real-time Traffic Control using QUBO
S117	-	Optimizing Quantum Programs Against Decoherence: Delaying Qubits into Quantum Superposition
S118	-	An Efficient Controlled Quantum Secure Direct Communication Protocol via GHZ-like States
S119	-	Bugs4Q: A Benchmark of Real Bugs for Quantum Programs
S120	-	Hybrid Quantum Applications Need Two Orchestrations in Superposition: A Software Architecture Perspective
S121	-	Reverse Engineering of Hamiltonian Expressions from D-Wave programs
S122	-	A simulator-based performance analysis of multilevel feedback queue scheduling
S123	-	Remarks on concavity of the auxiliary function appearing in quantum reliability function
S124	-	DQRA: Deep Quantum Routing Agent for Entanglement Routing in Quantum Networks
S125	-	A practical quantum public-key encryption model
S126	-	Quantum-Inspired Evolutionary Algorithms applied to numerical optimization problems
S127	[12]	Identifying Bug Patterns in Quantum Programs
S128	-	Performance Comparison of Population-Based Quantum-Inspired Evolutionary Algorithms
S129	-	A Hybrid Adaptive Quantum Behaved Particle Swarm Optimization Algorithm Based Multilevel Thresholding for Image Segmentation
S130	-	GCQW: A Quantum Walk Model for Predicting Missing Links of Complex Networks
S131	-	A Video Steganography Scheme Based on Post-Quantum Cryptography
S132	[24]	A Multi-Lingual Benchmark for Property-Based Testing of Quantum Programs
S133	-	Enhancing data and privacy security in mobile cloud computing through quantum cryptography
S134	-	Quantum Information Security Protocols and Quantum Coding Theory
S135	-	Improved Security of SDN based on Hybrid Quantum Key Distribution Protocol
S136	-	Synthesizing hybrid quantum circuits without ancilla qudits
S137	-	Convergence analysis on a class of quantum-inspired evolutionary algorithms
S138	-	Quantum Control Mechanisms of Superdense Coding
S139	-	Quantum particle swarm optimization for multiobjective combined economic emission dispatch problem using cubic criterion function

S140	-	A hybrid Quantum-Inspired Evolutionary Algorithm for open vehicle routing problem

Информация об авторах / Information about authors

Рейес ХУАРЕС-РАМИРЕС имеет степень PhD по программированию, профессор Автономного университета Нижней Калифорнии (Мексика) с 2002 года. Эксперт в области программной инженерии, в настоящее время является Президентом Мексиканской сети программной инженерии – мексиканской профессиональной ассоциации, продвигающей научные и образовательные инициативы в области программной инженерии. Сфера научных интересов: программная инженерия, человеко-машинное взаимодействие. В последнее время занялся изучением основ квантовых вычислений. Участник Мексиканской национальной исследовательской системы.

Reyes JUÁREZ-RAMÍREZ – PhD in Computer Science, professor at the Universidad Autónoma de Baja California since 2002. Software Engineering expert; he currently is the president of the Red Mexicana de Ingeniería de Software (“Mexican Network of Software Engineering”), a specialized association that addresses research and education initiatives in software engineering in México. His research areas are software engineering, human-computer interaction, and currently starting with quantum computing. He is part of the National Researchers System in Mexico.

Чристиан Ксавьер НАВАРРО-КОТА имеет степень PhD по программированию от испанского Университета Кастилии в Ла-Манче. В настоящее время работает профессором Автономного университета Нижней Калифорнии в городе Энсенада, штат Нижняя Калифорния, Мексика. Его научные интересы включают образовательные технологии, мобильные и повсеместные вычисления, человеко-машинное взаимодействие, исследования опыта взаимодействия.

Christian Xavier NAVARRO-COTA – PhD in Computer Science from the Universidad de Castilla La Mancha, Spain. Currently, he serves as a professor at Universidad Autónoma de Baja California in Ensenada, Baja California, México His research interests include educational technology, mobile and ubiquitous computing, human-computer interaction, and user experience (UX).

Саманта ХИМЕНЕС имеет основное образование в области вычислительной техники, степень магистра в области обработки данных и степень PhD по программированию. Она является профессором в мексиканском Автономном университете Нижней Калифорнии, город Валье-де-лас-Пальмас, Нижняя Калифорния, Мексика, а также приглашенным преподавателем в Университете Глобальных знаний, Сан-Диего, участвует в мексиканской Национальной системе поддержки исследователей. Сфера её научных интересов: человеко-машинное взаимодействие, диалоговые системы, аффективные вычисления и образовательные системы.

Samantha JIMÉNEZ – PhD in Computer Science and a master’s in data science, complemented by a solid foundation in Computational Systems Engineering. She is a professor at the Universidad Autónoma de Baja California located in Valle de las Palmas, Baja California, México, and a dedicated educator at San Diego Global Knowledge University. Her research interests are human-computer interaction, dialogue systems, affective computing, and educational systems. She is part of the National Researchers System in Mexico.

Алан Давид РАМИРЕС-НОРЬЕГА получил степень магистра прикладных вычислений в Автономном университете Синалоа и степень PhD по программированию в Автономном университете Нижней Калифорнии. Он является штатным профессором и исследователем в Университете Синалоа, членом мексиканской Национальной системы поддержки

исследователей уровня 1 в области IX (междисциплинарные исследования). Имеет несколько публикаций в известных журналах (JCR, SCOPUS), выступал на национальных и международных конференциях по темам, связанным с интеллектуальными системами обучения, разработкой программного обеспечения и добычей данных, последняя из которых является основной областью интересов. Кроме того, руководил подготовкой бакалавров, магистров и докторов по различным направлениям и специальностям.

Alan David RAMÍREZ-NORIEGA – Obtained his master's degree in applied computing from the Autonomous University of Sinaloa and his Ph.D. in computer science from the Universidad Autónoma de Baja California. He is a Full-Time Professor and Researcher at the Facultad de Ingeniería Mochis at the Universidad Autónoma de Sinaloa. He is currently a member of the National System of Researchers level 1 in area IX (Interdisciplinary). He has several publications in high-impact journals (JCR, SCOPUS) and national and international conferences on topics related to Intelligent Tutoring Systems, Software Engineering and Data Mining, the latter being the main areas of interest. In addition, he has participated in various directions and synodalities of Bachelor's, Master's, and Doctorate theses.

Ма Вероника ТАПИЯ-ИБАРРА – инженер, основное образование – вычислительные инженерные системы, завершившееся получением степени магистра. Работает профессором в Технологическом институте Леона, Гуанахуато, Мексика, где преподает курсы по вычислительной технике и программированию.

Ma Veronica TAPIA-IBARRA – She is an engineer with a solid foundation in Computational Systems Engineering, complemented by master 's degree studies. She is a professor at the TecNM-Instituto Tecnológico de León located in León, Guanajuato, México, teaching hardware and programming courses.

Сесар Артуро ГЕРРА-ГАРСИЯ – имеет ученую степень PhD Университета Кастилии, Ла-Манча, Испания. Профессор Автономного университета Сан-Луис-Потоси, член мексиканской Национальной системы поддержки исследователей. Сфера научных интересов: инженерия требований, программная инженерия, качество данных и информации, научное волонтерство и информационная безопасность.

César Arturo GUERRA GARCÍA – PhD in computer science, graduated from Universidad de Castilla-La Mancha, Spain. He is a professor at the Universidad Autónoma de San Luis Potosí. His research interests are requirements engineering, software engineering, data and information quality, citizen science and informatic security. He is part of the National Researchers System in México.

Эктор Херардо ПЕРЕС-ГОНСАЛЕС – штатный профессор-исследователь Автономного университета Сан-Луис-Потоси (Мексика), имеет ученую степень доктора компьютерных наук. Автор научных статей и глав в книгах по автоматизации проектирования программного обеспечения и человеко-машинного взаимодействия, выступал с научными докладами на международных конференциях в США, Канаде, Великобритании, Португалии и в Сингапуре, участвует в мексиканской Национальной системе поддержки исследователей. Область научных интересов: проектирование программного обеспечения, преподавание методов разработки программного обеспечения, обработка цифровых изображений, разработка программного обеспечения для квантовых компьютеров.

Hector Gerardo PEREZ-GONZALEZ – Full-time research professor at Universidad Autónoma de San Luis Potosi, Mexico. PhD in Computer Science from the University of Colorado in 2003. Author of research articles and book chapters on Automatic Software Design and Human-Computer Interaction. He has been a speaker at international conferences in the USA, Canada, UK, Portugal,

and Singapore. His research areas are software design, computer science education, and quantum software engineering. He is a member of the National Researchers System in Mexico.

Карлос Альберто ФЕРНАНДЕС-И-ФЕРНАНДЕС имеет степень PhD университета Шеффилда по программированию, эксперт в области программирования. В настоящее время возглавляет Институт вычислений в Технологическом университете в мексиканском регионе Миштека, координирует магистерские программы по прикладным аспектам вычислительных технологий. Сфера научных интересов: визуальное моделирование, гибкие технологии разработки и формальные спецификации программного обеспечения.

Carlos Alberto FERNÁNDEZ-Y-FERNÁNDEZ – Software Engineering expert with a Ph.D from the University of Sheffield. He currently leads the Institute of Computing at Universidad Tecnológica de la Mixteca and coordinates the Master's program in Applied Computing Technologies. His research interests include visual modeling, agile methods, and formal software specification.

DOI: 10.15514/ISPRAS-2024-36(1)-7

Research Trends and Recommendations for Future Microservices Research

[1] *Z. Stojanov*, ORCID: 0000-0001-6930-5337 <*zeljko.stojanov@uns.ac.rs*>
[2] *I. Hristoski*, ORCID: 0000-0002-9291-7873 <*ilija.hristoski@uklo.edu.mk*>
[1] *J. Stojanov*, ORCID: 0000-0002-2237-8026 <*jelena.stojanov@uns.ac.rs*>
[1] *A. Stojkov*, ORCID: 0000-0002-1078-7068 <*aleksandra.stojkov@tfzr.rs*>
[1] *University of Novi Sad, Technical Faculty "Mihajlo Pupin" Zrenjanin, BB, Djure Djakovica st., Zrenjanin, 23000, Serbia.*
[2] *University "St. Kliment Ohridski" - Bitola, Faculty of Economics - Prilep, 143, Prilepski Braniteli, Prilep, 7500, North Macedonia.*

Abstract. Microservices are the most promising direction for developing heterogeneous distributed software systems capable of adapting to dynamic changes in business and technology. In addition to the development of new software systems, the migration from legacy monolithic systems to microservice architectures is also a prominent aspect of microservices use. These trends resulted in an increasing number of primary and secondary studies on microservices, stressing the need for systematization of research at a higher level. The objective of this study is to comprehensively analyze secondary studies in the field of microservices with objectives to inquire about publishing trends, research trends, domains of implementation, and future research directions. The study follows the guidelines for conducting a systematic literature review, which resulted in the findings derived from 44 secondary studies. The study findings are structured to address the proposed research objectives. Recommendations for further literature reviews relate to the improvement of quality assessment of selected studies to increase the validity of findings, a more detailed review of human and organizational factors through the microservices life cycle, the use of social science qualitative methods for more detailed analysis of selected studies, and inclusion of gray literature that will bring the real opinions and experiences of experts from industry.

Keywords: microservices; tertiary study; systematic literature review; research trends; recommendations.

For citation: Stojanov Z., Hristoski I., Stojanov J., Stojkov A. Research Trends and Recommendations for Future Microservices Research. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 105-130. DOI: 10.15514/ISPRAS-2024-36(1)-7.

Full text: Stojanov Z., Hristoski I., Stojanov J., Stojkov A. A Tertiary Study on Microservices: Research Trends and Recommendations. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 796–821. DOI: 10.1134/S0361768823080200.

Направления будущих исследований и рекомендации по развитию микросервисной архитектуры

[1] *Ж. Стоянов*, ORCID: 0000-0001-6930-5337 <*zeljko.stojanov@uns.ac.rs*>
[2] *И. Христоский*, ORCID: 0000-0002-9291-7873 <*ilija.hristoski@uklo.edu.mk*>
[1] *Е. Стоянова*, ORCID: 0000-0002-2237-8026 <*jelena.stojanov@uns.ac.rs*>
[1] *А. Стойкова*, ORCID: 0000-0002-1078-7068 <*aleksandra.stojkov@tfzr.rs*>

[1] *Технический факультет "Михайло Пупин" университета Нови-Сада, Сербия, 23000, Зренянин, ул. Джуре Джаковича, ББ.*
[2] *Университет Св. Климента Охридского - Битола, Экономический факультет - Прилеп, Северная Македония, 7500, Прилеп, ул. Защитники Прилепа, 143.*

Аннотация. Микросервисы являются наиболее перспективным направлением для разработки разнородных распределенных программных систем, способных адаптироваться к динамическим изменениям бизнеса и технологий. В дополнение к разработке новых программных систем, переход от устаревших монолитных систем к микросервисным архитектурам также является важным аспектом использования микросервисов. Эти тенденции привели к увеличению числа первичных и вторичных исследований микросервисов, что подчеркивает необходимость систематизации исследований на более высоком уровне. Целью настоящего исследования является всесторонний анализ вторичных исследований в области микросервисов, который поможет выявить тенденции в направленности публикаций, исследований, уточнить области использования полученных результатов и перспективы будущих исследований. Представленное исследование следует рекомендациям по проведению систематического обзора литературы, в процессе его проведения были выявлены результаты 44 вторичных исследований. Эти результаты структурированы в соответствии с сформулированными авторами целями. Рекомендации для дальнейших обзоров литературы касаются улучшения оценки качества отдельных исследований для повышения достоверности результатов, повышения детализации обзоров человеческих и организационных факторов через жизненный цикл микросервисов, использования качественных методов социальных наук для более подробного анализа отдельных исследований, и включения в оборот литературы, обычно остающейся вне области внимания коммерческих и академических журналов, но содержащей реальные мнения и опыт промышленных экспертов.

Ключевые слова: микросервисы; третичное исследование; систематический обзор литературы; тенденции исследований; рекомендации.

Для цитирования: Стоянов Ж., Христоский И., Стоянова Е., Стойкова А. Направления будущих исследований и рекомендации по развитию микросервисной архитектуры. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 105–130 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–7..

Полный текст: Стоянов Ж., Христоский И., Стоянова Е., Стойкова А. Третичное исследование микросервисов: направления исследований и рекомендации. Programming and Computer Software, 2023, т. 49, № 8, стр. 796–821 (на английском языке). DOI: 10.1134/S0361768823080200.

1. Introduction

Microservices have recently emerged as popular and widely used architectural model for cloud-based applications, representing a new trend in developing distributed software systems [1-2]. As small and independent services, they offer improved performance and support for continuous delivery [3]. Microservices based applications are in many cases built by breaking up monolithic applications, which assumes considering factors such as the number of objects owned by a service, the level of responsibility, and the team distribution [4]. In the beginning, microservices were adopted by large companies like Amazon, LinkedIn, and Netflix, and later by other companies [5-6], leading to an increasing trend in using microservices for developing cloud-based applications. The adoption of microservices in developing or reengineering software systems includes a new organizational and business culture in software organizations [7]. Adoption of DevOps in software

companies is crucial for better integration of microservices-based applications throughout the system life cycle [8], leading to improvement their competitiveness [9].

Microservices-based applications consist of multiple components that collectively form the entire system. Each component performs a single task, with its boundaries shielding it from external knowledge, while the processed results can be shared and accessed by other microservices [2]. A system structure is stable even when upgrades or extensions are necessary. With microservices, clients can be confident that any changes or growth in their business will be implemented into software. Microservices show better performance than monolithic architectures, particularly in terms of meeting business requirements, ensuring systems reliability, enhancing maintainability, and bolstering infrastructure resilience [2]. Although microservices require a larger number of teams and greater effort, the long-term benefits make the investment worthwhile [10]. Migration of monolithic or legacy systems to service-oriented architectures is a common trend in contemporary software systems [11], particularly to microservice architectures, resulting with improved system performance [12-13]. Use of design patterns results in improvement of development practices and better fulfillment of various architectural quality attributes [14].

Microservices-based systems consist of individual microservices, each independently performing a specific functionality. Consequently, if one microservice fails, the entire system remains unaffected. The principle of *Autonomy* is responsible for this behavior, while other key principles are [10]: *Resilience* – ensuring that the application can continue providing services even if a specific microservice encounters failure; *Transparency* – exposing the necessary details and providing documentation for each microservice; *Automation* – employing tools that enhance the efficiency, reliability, and scalability of the microservices' building and maintenance processes; and *Alignment* – relating to harmonizing different microservices within the system.

Systematic Literature Reviews (SLRs) [15] and Systematic Mapping Studies (SMSs) [16] have recently been adopted by software engineering research community, for systematizing and analyzing the evidence on the practice and leading to Evidence-Based Software Engineering (EBSE) [17]. Review of academic literature, commonly referred to as "white literature", has recently been supplemented with "gray literature" sources such as blog posts, white papers, industrial magazines, and videos, introducing Multivocal Literature Reviews (MLRs) [18].

Based on the above discussion, the objectives of this study are: (1) to present the current publishing trends of secondary studies research, (2) to determine topics inquired in secondary studies, (3) to inquire in which domains are microservices commonly implemented, and (4) to present identified future research directions. A SLR based on the guidelines proposed in [15, 17] was performed, resulting in 44 secondary studies that were used for drawing research findings and recommendations for further research.

This paper is structured as follows. The second section presents related work on tertiary studies related to microservices. The third section outlines the research methods employed in the study, while the fourth section presents the research findings. Recommendations for future reviews are discussed in the fifth section. The last section contains conclusions.

2. Related work

Tertiary studies have been recently used in software engineering for reviewing secondary studies and conducting meta-analyses on specific research topics. Some of the tertiary studies relate to DevOps [19], architecting systems of systems [20], cloud computing [21], agile software development [22], variability in software product lines [23], or testing artifact quality [24].

Two tertiary studies on microservices were identified: "Research on Microservice Architecture: A Tertiary Study" by Liu et al. [25], and the second study titled "Microservice Architecture: A Tertiary Study" by Costa et al. [26]. Table 1 presents information on the period covered and the number of secondary studies included in identified tertiary studies. The review types encompass Systematic

Literature Reviews (SLRs), Systematic Mapping Studies (SMSs), Systematic Grey Literature Reviews (SGLRs), and Multivocal Literature Reviews (MLRs).

Table 1. Tertiary studies on microservices

Study reference	Time span	SLRs	SMSs	MLRs	SGLRs
Liu et al. (2022) [25]	2016-2021	17	20	0	0
Costa et al. (2020) [26]	2016-2019	5	14	2	1

Liu et al. [25] conducted a SLR and identified 37 secondary studies on microservices published in the period from 2016 to 2021. The authors formulated two research questions: (RQ1) What are the common topics addressed in secondary studies related to microservices architecture (MSA), and what are their findings? (RQ2) What are the potential areas for new research in the field of MSA? Quality of the secondary studies was assessed based on the DARE quality criteria [17].

Costa et al. [26] conducted a SLR and identified 22 secondary studies on microservices published in the period from 2016 to 2019. The original study was written in Portuguese, which required the translation of methodological issues and results into English. The authors addressed the following research questions: (RQ1) Which secondary studies have been published in the field of microservices? (RQ2) What research topics on microservices have been investigated? (RQ3) What emerging patterns have been identified? (RQ4) What solutions and support tools have been utilized to facilitate the development and operation of microservices architecture? (RQ5) In which areas, particularly in the industry, are microservices being applied? (RQ6) Which topics exhibit gaps and require further exploration in future microservices research? The quality of secondary studies was assessed by using the DARE quality criteria [17].

3. Research methods

This study is based on the guidelines for conducting SLRs proposed in [15, 17]. The research process contains the following main phases: (1) planning the review, (2) conducting the review, and (3) reporting the findings.

3.1 Planning the review

Justification of the need for a tertiary study, determining research questions, selecting digital databases, and defining the studies search and selection process with clearly stated keywords for searching, inclusion/exclusion criteria, and quality assessment criteria are described in this section.

3.1.1 Need to conduct a tertiary study on microservices

In the last decade, research on microservices has gained popularity by the researchers, resulting in an increasing number of empirical studies and leading to the execution of systematic reviews. These reviews were performed as SLRs [15], SMSs [16], and even MLRs [18].

During the search of literature on microservices, two tertiary studies on microservices were identified: a study titled “Research on Microservice Architecture: A Tertiary Study” by Liu et al. [25], and a study titled “Microservice Architecture: A Tertiary Study” by Costa et al. [26]. Insights into these two studies revealed a consistent increase in the number of secondary studies over the years, and it can be expected that this trend will continue in 2023 and beyond, highlighting the need for new reviews of recent secondary research.

3.1.2 Research questions

The following research questions (RQs) are proposed:

RQ1: What are the publishing trends observed in secondary studies?

RQ2: What are the predominant topics investigated in secondary studies?

RQ3: In which domains are microservices commonly implemented?

RQ4: What future research directions have been identified?

3.1.3 Search and selection process of secondary studies

Proposed research questions were used for selecting keywords for searching for secondary studies. Keywords were put into two groups: the first group includes the keywords “microservices” and “microservices architecture”, while the second group comprises the keywords “SLR”, “Systematic literature review”, “SMS” and “Systematic mapping study”. The following search strings were constructed for searching for secondary studies:

[1]: (“microservices architecture” OR “microservices”) AND (“SLR” OR “Systematic literature review”)

[2]: (“microservices architecture” OR “microservices”) AND (“SMS” OR “Systematic mapping study”)

The process of searching and selecting studies contains the following phases (Ph#No):

- Ph#1: Searching digital libraries using constructed search strings.
- Ph#2: Selecting specific studies based on their title, abstract, and keywords. This phase also involves removing duplicates (in case a study appears in multiple databases) and selecting the most recent version of the study (if there are multiple versions by the same authors).
- Ph#3: Using snowball search method [27] for finding additional studies and minimize the possibility of omitting relevant secondary studies.
- Ph#4: Applying inclusion/exclusion criteria to studies that passed phases Ph#2 and Ph#3.
- Ph#5: Conducting a detailed reading and analysis of the studies that passed Ph#4.

The digital libraries used for searching secondary studies are ACM Digital Library, IEEE Xplore, ScienceDirect, Springer, Wiley Online Library, and MDPI. These libraries were selected because they publish a majority of the leading journals and conference proceedings in the field.

Filtering of the studies identified during the search of digital libraries and snowball search was based on the inclusion and exclusion criteria. *Inclusion criteria* are: (I1) A study reviews relevant studies on microservices, (I2) A study follows guidelines for conducting SLR or SMS, (I3) A study answers research questions in the domain of microservices. *Exclusion criteria* are: (E1) Full text of a study is not available, (E2) A study is not peer-reviewed, (E3) The study is less than 6 pages, (E4) A study is not written in English, (E5) A review study that includes gray literature.

All selected secondary studies were evaluated against these inclusion and exclusion criteria, and if a secondary study failed to meet even one criterion, it was excluded from further analysis.

3.1.4 Quality assessment of secondary studies

The secondary studies were evaluated for quality based on guidelines proposed in [17]. The primary objective of the quality assessment was to identify and exclude low-quality studies from the detailed analysis and synthesis of review findings. Quality assessment was based on a three-point scale with values 1 (Yes), 0.5 (Partly), and 0 (No). This scale was based on the five questions (Q2-Q6) proposed in [17], while an additional question concerning the use of the review methodology (Q1) was added. The quality assessment questions are:

Q1: Is the review methodology clearly stated and appropriate?

Q2: Are the review's inclusion and exclusion criteria described and appropriate?

Q3: Is the literature search likely to have covered all relevant studies?

Q4: Did the reviewers assess the quality/validity of the included studies?

Q5: Were basic studies adequately described?

Q6: Were the extracted data from included studies synthesized in the findings?

Based on the authors’ agreement, studies with an average quality score of less than 0.5 will be excluded from the detailed analysis.

3.2 Conducting the review

The search for secondary studies was performed in January 2023. The first phase resulted in the identification of 821 papers. The details of the search conducted in digital libraries are presented in Table 2.

Table 2. Total number of papers obtained through search in digital libraries

Library	Number of search results for search string [1]	Number of search results for search string [2]	Totally for search strings [1] and [2]
ACM Digital Library	68	90	158
IEEE Xplore	13	14	27
ScienceDirect	127	85	212
Springer	144	183	327
Wiley Online Library	31	58	89
MDPI	5	3	8
TOTAL	388	433	821

The phased process of selecting secondary studies is presented in Fig. 1, while the selection process of SLRs and SMSs throughout the phases is presented in Table 3.

Filtering in Ph#2 and Ph#3 resulted in the selection of 57 secondary studies. After implementing inclusion/exclusion criteria on each of the 57 studies, 44 were selected for further analysis and quality assessment. Three studies were excluded based on the E3 exclusion criterion, one study based on the E1 exclusion criterion, and nine studies based on the E5 exclusion criterion.

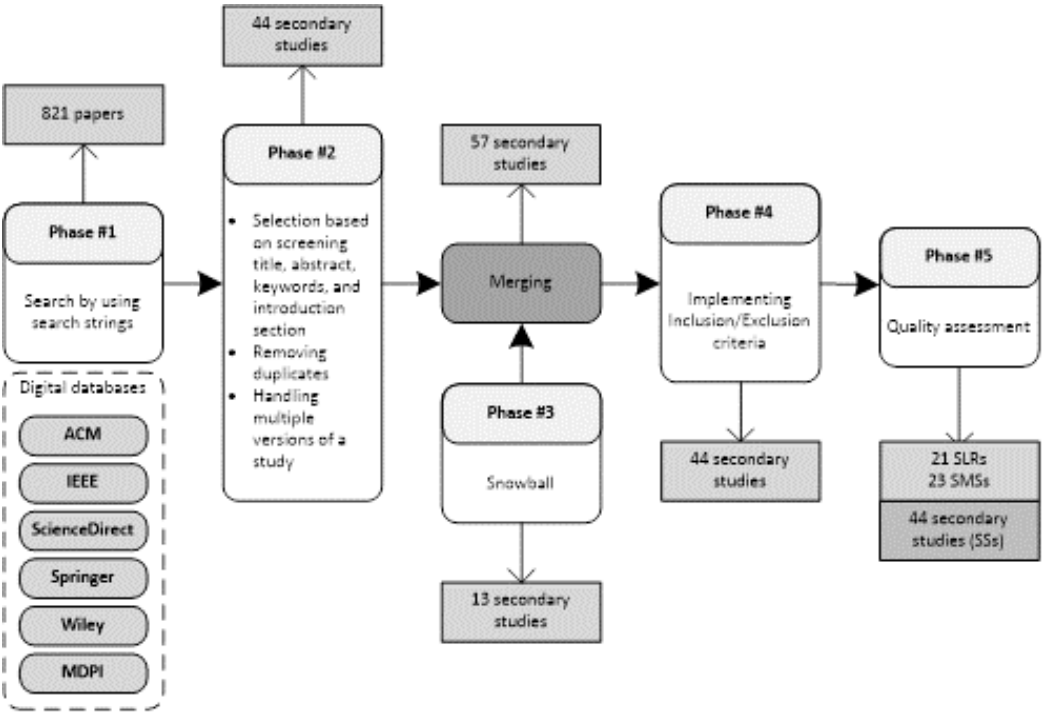


Fig. 1. The phased process of selecting secondary studies

Table 3. Details of the phased process for selecting secondary studies

	SLRs	SMSs	MVRs	Totally SSs
Ph#1: Selected studies after checking titles, abstracts, and keywords	19	23	14	56
Ph#2: Selected studies after removing duplicates	16	19	9	44
Ph#3: Snowball search for additional studies	6	7	0	13
Merging digital libraries and snowball search results	22	26	9	57
Ph#4: Selected studies after implementing Inclusion/Exclusion criteria	21	23	0	44
Ph#5: A final set of secondary studies after quality assessment	21	23	0	44

3.2.1 Quality assessment

The first, third, and fourth authors assessed the quality of secondary studies, while the second author reviewed the grades and calculated the average quality scores for all studies. Each evaluator assigned a mark from the three-point scale (0.0, 0.5, or 1.0) to each study for all quality assessment criteria. The average quality score for each study was then calculated as the mean of all average values assigned by each evaluator. Table 4 presents the average scores for the quality assessment of the selected studies. It is evident that all studies exceeded the minimum required quality threshold for further analysis (overall quality score greater than 0.50).

Table 4. Average scores for quality assessment of the selected secondary studies

ID	Year	Type	Q1	Q2	Q3	Q4	Q5	Q6	AVG
SS01	2020	SLR	0.67	0.67	0.83	0.33	0.83	0.83	0.69
SS02	2019	SLR	1.00	1.00	1.00	0.83	1.00	1.00	0.97
SS03	2019	SLR	0.83	1.00	1.00	0.83	0.83	0.83	0.89
SS04	2020	SLR	0.67	1.00	1.00	0.33	0.83	0.67	0.75
SS05	2018	SLR	0.50	0.50	0.67	0.33	0.83	0.83	0.61
SS06	2021	SLR	1.00	1.00	1.00	1.00	0.83	1.00	0.97
SS07	2021	SLR	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SS08	2018	SLR	0.50	0.67	0.67	0.50	1.00	1.00	0.72
SS09	2019	SLR	0.50	0.50	0.83	0.67	0.83	0.83	0.69
SS10	2020	SLR	1.00	1.00	1.00	0.33	1.00	1.00	0.89
SS11	2022	SLR	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SS12	2022	SLR	0.50	0.33	1.00	0.17	0.83	0.83	0.61
SS13	2022	SLR	1.00	1.00	0.83	0.00	0.83	0.83	0.75
SS14	2022	SLR	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SS15	2022	SLR	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SS16	2022	SLR	0.83	1.00	1.00	0.83	1.00	1.00	0.94
SS17	2021	SLR	1.00	1.00	0.83	0.83	1.00	1.00	0.94
SS18	2021	SLR	0.67	1.00	1.00	0.50	1.00	1.00	0.86
SS19	2018	SLR	1.00	1.00	0.67	0.83	1.00	0.83	0.89
SS20	2021	SLR	0.67	1.00	0.83	0.33	1.00	0.83	0.78
SS21	2021	SLR	0.67	1.00	1.00	0.00	0.83	0.83	0.72
SS22	2022	SMS	1.00	1.00	0.83	0.00	1.00	0.83	0.78
SS23	2017	SMS	1.00	0.83	1.00	0.00	0.67	0.83	0.72
SS24	2021	SMS	0.83	1.00	1.00	0.00	1.00	1.00	0.81
SS25	2020	SMS	1.00	1.00	1.00	0.00	0.67	1.00	0.78
SS26	2021	SMS	1.00	1.00	0.83	0.00	0.67	0.83	0.72
SS27	2017	SMS	1.00	1.00	1.00	0.00	0.67	1.00	0.78
SS28	2019	SMS	1.00	1.00	1.00	0.67	1.00	0.83	0.92
SS29	2016	SMS	1.00	1.00	0.83	0.17	1.00	1.00	0.83
SS30	2022	SMS	1.00	1.00	1.00	0.00	1.00	0.83	0.81
SS31	2022	SMS	0.83	0.83	1.00	0.75	0.50	0.83	0.79
SS32	2021	SMS	0.83	1.00	1.00	0.00	1.00	0.67	0.75
SS33	2020	SMS	1.00	0.83	1.00	0.33	1.00	1.00	0.86
SS34	2019	SMS	1.00	1.00	1.00	0.00	1.00	1.00	0.83

SS35	2021	SMS	1.00	1.00	1.00	1.00	1.00	0.83	0.97
SS36	2017	SMS	0.50	0.50	0.83	0.00	0.67	0.83	0.56
SS37	2019	SMS	1.00	1.00	1.00	0.00	0.83	0.83	0.78
SS38	2022	SMS	1.00	0.83	1.00	0.00	1.00	0.67	0.75
SS39	2021	SMS	0.83	1.00	1.00	0.00	1.00	1.00	0.81
SS40	2019	SMS	0.83	1.00	1.00	0.00	1.00	1.00	0.81
SS41	2019	SMS	0.83	1.00	1.00	0.00	0.17	1.00	0.67
SS42	2019	SMS	0.50	0.33	0.83	0.33	0.50	0.67	0.53
SS43	2016	SMS	1.00	1.00	1.00	0.17	1.00	1.00	0.86
SS44	2023	SMS	1.00	1.00	1.00	0.00	0.33	1.00	0.72
Average scores for all SSs			0.86	0.91	0.94	0.37	0.87	0.90	0.81
Average scores SLRs			0.81	0.89	0.91	0.60	0.93	0.91	0.84
Average scores SMSs			0.91	0.92	0.96	0.15	0.81	0.89	0.77

Based on the quality assessment results, the minimum score achieved was 0.53 (for study SS42), while the average quality score across all studies was 0.81. All 44 secondary studies passed quality analysis and were selected for in-depth analysis. Selected secondary studies (SSs) are listed in Appendix A.

3.2.2 Data extraction

The template presented in Table 5 is used for extracting data on secondary studies, encompassing general information about each study's publication, data relevant for quality assessment, and specific data relevant to each research question. The extracted data was organized in an Excel spreadsheet.

Table 5. Data extraction template

ID	Explanation	Use
D1	Study ID	Demography, RQ1
D2	Title	Demography, RQ1
D3	Year	Demography, RQ1
D4	Study type (SLR, SMS)	Demography, RQ1
D5	Venue type (conference, journal, book chapter)	Demography, RQ1
D6	Sample size (number of primary studies)	Demography, RQ1
D7	Used research methodology description	Quality assessment Q1
D8	Inclusion/Exclusion criteria	Quality assessment Q2
D9	Coverage of relevant studies	Quality assessment Q3
D10	Quality assessment questions	Quality assessment Q4
D11	Method for describing selected studies	Quality assessment Q5
D12	Data extraction methods and tools	Quality assessment Q6
D13	Research questions	Research topics, RQ2
D14	Research topics	Research topics, RQ2
D15	Technical implementation areas	Application area, RQ3
D16	Future research directions	Future research direction, RQ4

4. Research findings

The findings were derived from the data extracted from 44 selected secondary studies, 23 of them are SMSs, while the remaining 21 are SLRs (refer to Table 3). The findings will be organized and presented in alignment with the research questions.

4.1 Publishing trends for secondary studies (RQ1)

The selected secondary studies span the publication period from 2016 to 2023. Fig. 2 presents the publication trends of secondary studies based on the publishing venue.

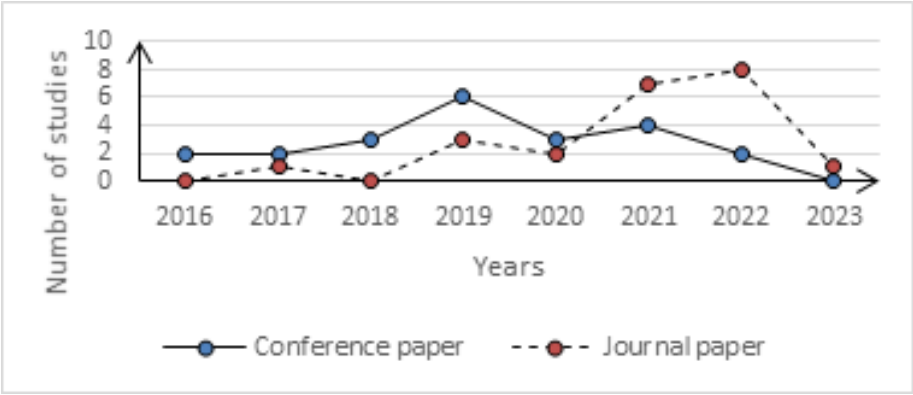


Fig. 2. Secondary studies publication trends by venue

It is obvious that the total number of studies has consistently grown over the years, and there has been a notable shift from primarily conference proceedings to journals. Out of the 44 studies, 22 were published in conference proceedings, and another 22 were published in journals.

4.2 Topics inquired in secondary studies (RQ2)

The identification of topic areas in the selected secondary studies is based on a comprehensive examination of each study. The extracted data corresponds to values in columns D13 (research questions) and D14 (research topics) in Table 5. A general overview of the topics investigated in the secondary studies is depicted in Fig. 3. It is worth noting that while most studies have a primary focus on a specific topic, they also touch upon other related topics.

4.2.1 Architecture

Architectural design is essential for the development of microservice-based software systems because it encompasses both the technical design of system functionalities and non-functional requirements, which are often referred to as quality attributes. Well-designed architecture is important for efficient development, operation, and maintenance of software systems. The main topics related to microservices architecture in the selected secondary studies include Analysis, Granularity, Patterns, Presentation, and Quality attributes (refer to Fig. 3). A more comprehensive examination of architecture topics, including identified subtopics and the distribution of secondary studies that mention them, is provided in Table 6.

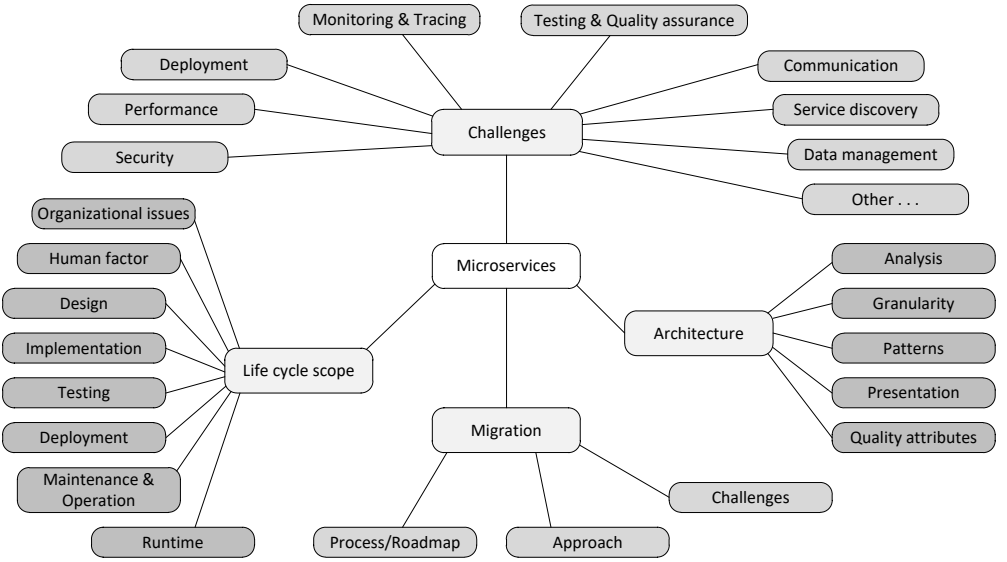


Fig. 3. General overview of microservices research topics

Analysis. The analysis of software architecture is crucial for understanding both new software systems and those that require migration to microservice architectures. It focuses on the following aspects:

(1) *Methods* - various methods can be distinguished, including Static Analysis, Dynamic Analysis, Combined Dynamic and Static Analysis, Model-Based Analysis, Graph-Based Analysis, and Pattern-Based Analysis;

(2) *Tools* - specific tools tailored to each analysis method, facilitating automated work; and

(3) *Challenges* – these encompass architectural analysis, software architecture reconstruction, technical debt analysis, quality attribute analysis, and fault analysis.

Granularity. Granularity refers to the size of individual microservices within a software system based on the microservices architecture. It plays a crucial role in determining the functioning of the system and its quality attributes, including performance, maintainability, data storage, and scalability. Determining the optimal granularity involves finding the right balance between the level of functionality encapsulated within each microservice and the need for modularity, maintainability, and scalability. Insights from the selected studies on granularity reveal the following main subtopics:

(1) *Methods or approaches* for defining granularity;

(2) *Metrics* used for evaluating granularity; and

(3) *Quality attributes* affected by granularity.

Table 6. Topics and subtopics related to microservices architecture

Topic	**Subtopic**	**Secondary studies**
Analysis	Methods	SS09, SS27, SS35, SS39, SS44
	Tools	SS30, SS39
	Challenges	SS30, SS39
Granularity	Approach	SS04, SS09, SS17, SS24
	Quality attributes	SS17
	Metrics	SS17
Patterns	Design	SS03, SS05, SS19, SS27, SS33, SS34, SS37, SS43

	Composition	SS02, SS09, SS19, SS33, SS43
	Communication	SS03, SS05, SS06, SS19, SS33, SS37
	Deployment	SS03, SS06, SS37, SS19, SS33, SS37
	Data storage	SS03, SS19, SS33, SS37
	Antipattern	SS01, SS44
Presentation	Languages	SS22, SS27, SS29, SS33
	Diagrams	SS27, SS29
	Visualization	SS30
Quality attributes	Reliability	SS03, SS08, SS17, SS27, SS33, SS34
	Security	SS03, SS07, SS08, SS17, SS27, SS29, SS33, SS40
	Compatibility	SS03, SS27, SS33, SS34, SS40
	Maintainability	SS03, SS08, SS17, SS27, SS29, SS33, SS34
	Performance	SS03, SS07, SS17, SS19, SS27, SS29, SS33, SS34, SS40
	Portability	SS03, SS27, SS33, SS34
	Testability	SS07, SS33
	Availability	SS07, SS08, SS17, SS33
	Monitorability	SS07, SS33
	Scalability	SS07, SS08, SS17, SS19, SS27, SS29, SS33, SS34, SS40
	Modularity	SS17, SS29
	Other …	SS29, SS33, SS40

Patterns. The identification and categorization of specific challenges and their corresponding solutions during software development, operation, and maintenance contribute to the recognition of recurring scenarios, commonly referred to as patterns. Incorporating patterns into the software life cycle enables developers to find reliable solutions to common problems, enhances communication among team members and with clients, and aids in meeting quality requirements. The following categories of patterns are identified:

(1) *Design* – patterns used for structuring and organizing microservices efficiently (API gateway, publish/subscribe, circuit breaker, proxy, and load balancer)

(2) *Composition* – patterns related to composing different microservices in a software system (semantic annotation, best-fitting, and workload-based approaches);

(3) *Communication* – patterns related to communication between multiple microservices (synchronous communication, publish/subscribe communication, combination of HTTP and message queue, communication using message-oriented middleware, asynchronous communication, point-to-point communication, and communication using binary protocols);

(4) *Deployment* – patterns related to the deployment or distribution of microservices to multiple resources for operational use (serverless deployment, service instances per VM, and service instances per container); and (5) *Data storage* – patterns oriented towards improving performances of data management systems (database-per-service pattern, the database cluster pattern, and the shared database server pattern). In addition, antipatterns refer to design and implementation choices that result in inadequate/poor software system design, leading to issues during operation and maintenance.

Presentation. Efficient design and utilization of microservice-based systems require the presentation of software architecture through various approaches that depict the system's structure and behavior. The secondary studies identified the following subtopics:

(1) *Languages* – specific languages for describing the architecture, such as RAML, YAML, Jolie, or various pseudocodes;

(2) *Diagrams* – visual diagrams, ranging from informal drawings to specialized and sophisticated diagrams such as Component/Container, Process/Behavior, Sequence, Execution Timeline, Deployment, Class, Use Case, Type Graph, Instance Graph, and Dependency Graph; and

(3) *Visualization* – dynamic analysis techniques, supported with specialized tools, for analyzing or recovering software architecture.

Quality attributes. Quality characteristics of software systems are typically reflected in the form of quality attributes, which are associated with non-functional requirements. While there are numerous quality attributes, it is often challenging to satisfy all of them simultaneously. In such cases, trade-offs must be made when considering quality attributes. The most common quality attributes mentioned in many studies are Reliability, Security, Compatibility, Maintainability, Performance, Portability, Testability, Availability, Monitorability, Scalability, and Modularity. These attributes have been extensively discussed in various sources and are relevant throughout the software life cycle, from design to operational use and maintenance. The selected secondary studies also mention additional quality attributes, such as Modifiability, Usability, Deployability, Flexibility, Reusability, Manageability, Independence, Traceability, Complexity, Load balancing, and Organizational alignment. To effectively address quality attributes, appropriate metrics need to be proposed, such as time, complexity, number of requests, or number of affected files. These metrics facilitate continuous improvements in quality attributes over time.

4.2.2 Life cycle scope

Microservice-based systems undergo various lifecycle phases that bring forth unique challenges, necessitating the utilization of specific methods, approaches, and tools. The primary topics identified in the selected secondary studies pertaining to the life cycle scope of microservices encompass Design, Implementation, Testing, Deployment, Maintenance and Operation, Runtime, Organizational issues, and the Human factor (refer to Fig. 3). A review of life cycle scope topics, subtopics and the distribution of secondary studies mentioning them is given in Table 7.

Table 7. Topics and subtopics related to the life cycle scope of microservices

Topic	Subtopic	Secondary studies
Design	Identification strategies	SS02, SS04
	Domain Driven Design	SS04, SS39
	Representations	SS04, SS34, SS36, SS39
	Design for failure	SS08, SS36
Implementation	Technology stack	SS08, SS36, SS37
	Supporting systems	SS08, SS34, SS36, SS37
	Services interfaces	SS08, SS34
Testing	Approaches	SS14, SS23, SS25, SS36, SS37, SS40
	Tools	SS25, SS37, SS40
Deployment	Platform	SS08, SS32, SS34, SS37
	Monitoring	SS08, SS14, SS23, SS32, SS34, SS36, SS39
	Approaches	SS06, SS23, SS32, SS37

Maintenance & Operation	Load balancing	SS11, SS34
	Fault diagnosis	SS11, SS23, SS36, SS37, SS39
	Autoscaling	SS11, SS12, SS21, SS34, SS37
	Anomaly detection	SS13, SS37
	Resource Scheduling	SS13, SS33
	Analysis	SS36, SS39
Runtime	Virtualization	SS08, SS43
	Discovery	SS14, SS43
	Control	SS08, SS34, SS39
	Verification and Validation	SS08, SS21
	Visualization	SS30, SS39, SS44
Organizational issue	DevOps	SS08, SS19, SS25, SS32, SS33, SS34, SS37, SS43
	Continuous processes	SS08, SS25, SS43
Human factor	Roles	SS04, SS17
	Skills	SS04, SS38

Design. The design phase of the lifecycle is crucial for achieving the desired system structure and fulfilling the proposed quality characteristics. The following subtopics are identified in the secondary studies:

(1) *Identification strategies* –focuses on the identification of services during the design of complex systems;

(2) *Domain-Driven Design* – relates to the use of principles, patterns, and domain-specific knowledge during system design;

(3) *Representations* –the use of various methods and tools for representing microservice system being developed; and

(4) *Design for failure* – relates to design principles and methods that enable the design of systems with increased fault tolerance, self-healing capabilities, and variability characteristics.

Implementation. The implementation phase utilizes the products and decisions from the design phase to create microservices and integrate them into a system. The following subtopics are identified:

(1) *Technology stack* –the use of various languages (formal, scripting, object-oriented), interaction models for communication flow, and protocols for data exchange (e.g., REST/HTTP, RPC-alike, message queues);

(2) *Supporting systems* –focuses on data storage systems for distributed microservices (e.g., SQL, graph-oriented, document-oriented) and systems for service discovery in a dynamic environment; and

(3) *Service interfaces* – concerns the specification of contracts for microservices communication.

Testing. The complex nature and dynamic behavior of microservice-based systems present several challenges in their testing. The following subtopics are identified:

(1) *Approaches* –encompasses various testing approaches employed during development, ranging from unit testing to integration testing (continuous testing as part of DevOps and continuous engineering practices, testing of microservices and system performance, testing during migration, and model-based testing); and

(2) *Tools* – relates to tools utilized in the testing process, with a preference for automated testing. It includes libraries and frameworks that enable specific types of tests.

Deployment. A deployment practice encompasses activities, methods, and tools necessary for the establishment of heterogeneous microservices to meet the requirements of contemporary businesses. Automated and continuous development and deployment processes are essential in ensuring the reliable and scalable delivery and operation of microservice-based systems. The following subtopics are identified:

(1) *Platform* – pertains to the selection of a hosting system for running microservices;

(2) *Monitoring* – relates to the activities performed to prevent or respond to failures or changes in the environment; and

(3) *Approaches* – addresses the various ways and strategies for facilitating the utilization of microservices-based systems.

Maintenance & Operation. The primary focus of maintenance and operation activities in the software life cycle is to ensure the usability and operability of the software. The following subtopics are identified:

(1) *Load balancing* – pertains to the coordination and management of a large number of service requests in systems with heterogeneous and distributed microservices;

(2) *Fault diagnosis* – involves improving the quality and efficiency of software operation by detecting faults (monitoring and localization of faults, identifying fault types, and fault modeling);

(3) *Autoscaling* – relates to the adjustment of system resources to meet changing needs and growing requirements (resource allocation, prediction and scheduling methods);

(4) *Anomaly detection* – focuses on identifying critical behaviors or abnormal states in application performance;

(5) *Resource scheduling* – involves the dynamic adjustments of system resources in response to the overall system state and workload.; and (6) *Analysis* – covers methods and tools for analyzing the states and behaviors of microservice-based systems during operational use.

Runtime. Analyzing the architecture, functioning, and performance of microservices-based systems requires extracting information from both static and dynamic sources during runtime, which is crucial due to the changes in structure and communication. The following subtopics are identified:

(1) *Virtualization* – pertains to different levels of platform abstraction, isolation, and sharing;

(2) *Discovery* – relates to identifying and finding appropriate services based on workload, scalability, and service quality considerations;

(3) *Control* – involves managing execution at both the local level of individual microservices and the system level as a whole;

(4) *Verification and Validation* – focuses on assessing the quality of microservices during runtime; and

(5) *Visualization* – relates to visually representing the microservices architecture during runtime, which covers techniques, tools, and types of information to be presented.

Organizational issue. Software development, operation, and maintenance take place within specific organizational contexts, including software development organizations and client organizations. These contexts have their own processes, procedures, challenges, and cultures that impact the technical and technological aspects of software processes. The following subtopics are identified:

(1) *DevOps* –encompasses the cultural and practical aspects of organizing stakeholders involved in the *development* and operation; and

(2) *Continuous Delivery, Integration, and Deployment* – focuses on continuous activities that facilitate a seamless and smooth transition between life cycle phases in microservices-based systems.

Human factor. Even though the literature primarily emphasizes technical and technological aspects, it is important to recognize that all activities in the software systems' life cycle are carried out and supervised by people. The following subtopics are identified:

(1) *Roles* – pertains to the various roles that individuals assume in the microservice life cycle; and

(2) *Skills* – focuses on the technical and soft skills that are necessary for individuals in different roles. It encompasses the specific knowledge, expertise, and abilities required to effectively perform their tasks.

4.2.3 Migration

One of the primary challenges with monolithic legacy software systems is the need for subsequent modifications to keep them operational and useful for end users. Frequent modifications can increase software complexity, reduce performance, and make maintenance challenging. A common solution is migrating software systems to microservices architectures. The main topics related to migration to microservices are Approach, Process/Roadmap, and Challenges (refer to Fig. 3). A more detailed review of migration topics, including identified subtopics and the distribution of secondary studies that mention them, is presented in Table 8.

Approach. In practice, different types of legacy systems require varying approaches for modernization and migration to microservice architectures. The identified subtopics are:

(1) *Strategy* – pertains to the overall strategy chosen in a migration project, such as clustering, candidate identification based on quality attributes, data-driven approaches, or bottom-up approaches;

(2) *Decomposition method* – focuses on the selection of the analysis method used to decompose the legacy system and identify microservices;

(3) *Unit level* – involves selecting the most suitable level of software artifacts during the decomposition of the old system and migration to microservices (business functions, database tables, classes, use cases); and

(4) *Evolution* – relates to supporting the scalability and maintainability features throughout the migration process.

Table 8. Topics and subtopics related to migration to microservices

Topic	Subtopic	Secondary studies
Approach	Strategy	SS24, SS13, SS26, SS38
	Decomposition method	SS09, SS24, SS13, SS26
	Unit level	SS09, SS24, SS38
	Evolution	SS09, SS24, SS38
Process/Roadmap	Definition	SS24, SS26, SS38
	Input information	SS24, SS26, SS38
	Output information	SS24, SS26
	Success factors	SS38
	Motivation	SS24
Challenges	Technical	SS24, SS26, SS38
	Organizational	SS26, SS38
	Knowledge and skills	SS04, SS26, SS38

Process/Roadmap. Every migration project follows a process or roadmap that guides the organization of activities and determines the input and output information. The identified subtopics are:

(1) *Definition* – pertains to the selection or proposal of different processes, guidelines, and roadmaps, supported by specific tools;

(2) *Input information* – focuses on identifying the required input information for the migration process;

(3) *Output information* – relates to the information produced upon completing the migration process (e.g., microservices candidates, communication approaches);

(4) *Success factors* – encompasses the factors that influence the successful execution and completion of the migration process; and

(5) *Motivation* – explores the motivations or driving factors (technical, operational, or organizational) behind organizing a migration project.

Challenges. The migration of existing systems to a new microservice-based architecture is a challenging project that poses various obstacles for organizations, teams, and individuals. The identified subtopics are:

(1) *Technical* – focuses on the introduction of new technologies and the selection of the most suitable tools for the migration process;

(2) *Organizational* – pertains to organizational changes within an IT company that undertakes a migration process; and

(3) *Knowledge and skills* – emphasizes the importance of selecting team members with the appropriate knowledge and a combination of technical and non-technical skills necessary for the successful implementation of the migration project.

4.2.4 Challenges

The adoption of microservices in industrial practice presents numerous challenges for both practitioners and researchers, given the inherent complexity and heterogeneity of microservice-based systems. Fig. 3 illustrates eight challenges that have been mentioned in at least three secondary studies, with additional challenges categorized under the shape "Other ...". A comprehensive list of 21 identified challenges, along with the corresponding secondary studies that reference them, is presented in Table 9. By understanding and tackling these challenges, practitioners, and researchers can make significant strides in overcoming the obstacles inherent in microservice adoption.

Regardless of the topics covered, all studies identified specific challenges and proposed corresponding solutions. Among the challenges identified, Security emerges as the most crucial, with 13 occurrences across the studies. These discussions encompass a wide range of topics, needs, and scenarios, with proposed taxonomies or frameworks to address security issues. Following closely, Communication is mentioned in seven studies as the second most frequently cited challenge. Communication challenges may arise from remote calls, during replication of services or data, or service discovery. The third most frequently mentioned challenge pertains to Testing and quality assurance of microservice-based systems. This encompasses various specific challenges, including faster test feedback, automated testing, intercommunication testing, granularity testing, runtime testing, integration testing, and performance testing.

Table 9. Challenges related to microservices

Challenge	Secondary studies	Frequency
Security	SS14, SS15, SS16, S18, SS23, SS27, SS28, SS29, SS33, SS34, SS35, SS36, SS42	13
Communication	SS06, SS14, SS18, SS23, SS27, SS29, SS34	7

Testing & Quality assurance	SS14, SS23, SS25, SS27, SS34, SS36	6
Performance	SS14, SS23, SS29, SS33, SS36	5
Deployment	SS06, SS23, SS29, SS32, SS36	5
Monitoring & Tracing	SS14, SS23, SS29, SS32, SS36	5
Service discovery	SS14, SS23, SS29	3
Data management	SS14, SS27, SS34	3
Scalability	SS21, SS36	2
Migration	SS26, SS38	2
Complexity	SS27, SS34	2
Composition	SS27, SS34	2
Decomposition	SS14	1
Orchestration	SS14	1
Modeling	SS23	1
Context awareness	SS23	1
Integration	SS29	1
Fault tolerance	SS29	1
Publishing	SS32	1
Upgrading	SS32	1
Availability	SS36	1

4.3 Technical implementation and integration (RQ3)

The dynamics and the increasing need for integrating heterogeneous and complex systems in various sectors such as healthcare, industry, and transportation necessitate the adoption of service-oriented architectures, especially microservices-based architectures, to facilitate distributed processing capabilities and data integration. The identified topics are Service type, Service domain, and Industry adoption (refer to Fig. 3). Table 10 provides a detailed overview of the identified related subtopics and the distribution of secondary studies that mention them.

Table 10. Topics and subtopics related to technical implementation and integration of microservices.

Topic	**Subtopic**	**Secondary studies**
Service type	Functional	SS05, SS34
	Infrastructure	SS05, SS34
Domain	Smart systems	SS05, SS09, SS31, SS43
	Fog applications	SS42
	Big Data	SS20
	Blockchain	SS31
	Enterprise	SS31, SS33, SS43
Industry adoption	Readiness level	SS27, SS34
	Industry involvement	SS27, SS34
	Tools and system support	SS27, SS34
	Evaluation & Benchmarking	SS27, SS31, SS34

Service type. Microservices are deployed within complex and heterogeneous systems, requiring sophisticated infrastructures for hosting and execution. Apart from the functional services that cater

to user needs, it is crucial to deploy services that support the infrastructure essential for the proper functioning of microservices. Subtopics in this context are:

(1) *Functional services* – are responsible for providing functionalities to users, enabling them to perform tasks or access features; and

(2) *Infrastructure services* – non-functional requirements, infrastructure, and service monitoring, as well as maintenance tasks that are not directly related to user functionalities.

Domain. The secondary studies have identified the following subtopics:

(1) *Smart systems* – focuses on the implementation of microservices for technical integration in smart systems, such as smart cities, smart transportation, and IoT applications;

(2) *Fog applications* – explores the use of microservices in fog computing applications;

(3) *Big Data* – examines the utilization of microservices in Big Data applications;

(4) *Blockchain* – investigates the combination of microservices with blockchain technologies; and

(5) *Enterprise* – focuses on the use of microservices in various business domains, such as healthcare, online commerce, supply chain management, financial systems, and telecommunications.

Industry adoption. The majority of studies report the widespread acceptance and adoption of microservices in various business scenarios, establishing them as a prominent software development approach in the software and IT industry. The identified subtopics are:

(1) *Readines*s l*evel* – pertains to the maturity level of specific methods, tools, and technologies intended for implementation in industrial projects;

(2) *Industry involvement* – explores the degree of engagement and participation of industry experts in research projects;

(3) *Tools and system support* – examines the utilization of specialized tools and systems that support the development and operation of microservice-based systems; and

(4) *Evaluation & Benchmarking* – emphasizes the use of evaluation and benchmarking tools to assess the quality and effectiveness of designed microservice-based architectures.

4.4 Directions for future research (RQ4)

Potential future research directions have been identified by analyzing the discussion sections and concluding remarks of each examined secondary study. Therefore, we searched within the secondary studies using keywords such as ‘future’ in combination with the words ‘research’, ‘work’, and ‘direction’, as well as the keywords ‘further’, ‘challenge’, and ‘gap’.

Notably, the majority of authors indicate future research directions in two main ways:

(a) by highlighting and elaborating on potential research gaps, open challenges, and issues encountered during their analysis, and using them as a basis to propose broader research directions; and

(b) by proposing specific research directions that are closely related to the research methodology employed in their study.

In the coming years, the research focus will center on addressing the following issues:

- Exploring various types of microservice architectures, their structure, and design aspects. This involves expanding the existing knowledge base in a systematic, structured, and consistent manner, (a) by including both theoretical and practical learning and exploration, (b) by investigating the application of microservices in specific domains or real-world

scenarios, and (c) by considering organizational factors and addressing human-related issues in microservice architectures.

- To ensure the quality of microservice architectures, it is crucial to assess the associated quality attributes, while considering their complex interactions and trade-offs. By evaluating and addressing these attributes, researchers and practitioners can optimize the performance, robustness, and overall effectiveness of microservice architectures. This comprehensive assessment can help in developing resilient and adaptable systems.
- Focusing on the industrial adoption of microservices architecture, which involves several crucial phases, including design, implementation, validation, operation, deployment, maintenance, and testing of microservice architectural designs in practice. By emphasizing these essential phases, organizations can effectively adopt microservice architectures and reap the benefits of scalability, flexibility, and maintainability offered by this architectural style.
- Standardizing microservice architecture, interfaces, and related aspects such as load balancing, fault detection, and autoscaling.
- Designing fault-tolerant and event-driven/asynchronous microservices, particularly for smart systems, fog applications, and IoT applications.
- Transitioning from specific solutions and their related validation to more general solutions through fundamental research, reusable practices, and lessons learned.
- Addressing the complexities of the migration process from monolithic applications to microservice-based architectures systematically, as well as tackling challenges related to microservice identification, granulation, and proper design.
- Conducting additional systematic literature reviews that consider gray literature to compare findings and challenges identified in both white and gray literature, extending existing secondary studies to include the latest knowledge supplements, or exploring additional databases for comprehensive coverage, and enhancing literature review approaches by improving data extraction and synthesis methods, validity, and quality assessment of primary studies.

All these research directions aim to advance our understanding of microservices architecture research and facilitate the development of the best practices, standardized approaches, and improved methodologies in this rapidly evolving field.

The analysis of the directions for future research reveals that they are numerous and diverse. This can be attributed to the relative novelty of microservice architectures, and their unique, yet relatively unexplored nature, characterized by heterogeneity, decentralization, and independence. The breadth of future research directions signifies that the field of microservice analysis remains open to innovations and methodologies.

This study provides a valuable roadmap for researchers, highlighting areas that require further exploration. It also serves as a guide for practitioners, enabling them to assess the progress made thus far and determine which tools and approaches are suitable for practical implementation.

5. Recommendations for future literature reviews

This section provides comprehensive recommendations for further literature reviews in the field of microservices, encompassing primary, secondary, and tertiary studies. These recommendations are drawn directly from the research findings of this study and aimed at enhancing existing research and enriching the knowledge base on microservices. Recommendations relate to quality assessment of selected studies, microservices architecture, life cycle issues, migration, technical implementation in different domains, and further research directions.

Quality assessment of selected studies. According to guidelines for conducting literature reviews, quality assessment of included studies is typically considered a mandatory component of SLRs, whereas it may not be required for SMSs. Consequently, the quality assessment question Q4 received a low score of 0.37. Therefore, the primary recommendation for addressing this issue is to assess the quality of all the studies selected for inclusion in the review and to establish a minimum value threshold for study inclusion to maintain rigorous standards throughout the review process.

Architecture. Research about microservices architecture exhibits a significant level of research maturity, particularly in the areas of architecture analysis, pattern utilization, and the adoption of various methods for architectural representation. Moreover, considerable attention has been given to exploring quality attributes and their associated trade-offs. However, the research findings indicate that there is a need to assess quality attributes and metrics pertaining to granularity, runtime architecture visualization, as well as specific quality attributes such as context awareness, integrability, fault tolerance, upgrading, and availability.

Life cycle issues. Attention to life cycle issues, particularly the utilization of contemporary approaches like DevOps and continuous software engineering practices (continuous integration, continuous delivery, continuous deployment, and monitoring) increased in recent years. However, human factor issues, domain-driven design, and specific maintenance and operation concerns need more attention in further research.

Migration. The migration of legacy systems, typically characterized by monolithic architectures, to microservice architectures is a challenging research topic. Numerous methods and tools have been proposed and evaluated in real-world settings, forming a substantial knowledge base for practitioners and researchers. Based on the study findings, it is evident that motivation and success factors have not been sufficiently addressed in the existing research requiring further studies.

Technical implementation in different domains. Limited research has been documented in research studies and literature reviews concerning the technical implementation of microservices across various domains. Most of the research focuses on the initial development of microservice-based systems, with limited evaluations conducted in real-world settings. In addition, projects carried out by academic institutions often lack involvement of industry experts. There is an evident need to conduct implementation studies within real settings across different domains and to engage industry experts to gain a more comprehensive understanding of microservice-based system implementations.

Methodological issues for further research. Recommendations in the methodological areas are:

(1) Integration of gray literature aimed for complementing the insights obtained from white literature, as well as for increasing understanding of the industry experts' perspectives and the current state of practice;

(2) Utilization of data extraction and synthesis methods from social sciences that enable identification of patterns in practice and the development of theories necessary for building and extending a knowledge base in this area of software engineering.

6. Conclusions

This tertiary study focuses on publication trends, research topics, domains of implementation, and future research challenges in the domain of microservices. Conducted systematic literature review resulted in the selection of 44 secondary studies that are used for deriving findings. Based on the emerging findings, recommendations for further literature reviews are discussed. The main contributions of this study are detailed and structured recommendations for future literature reviews, which include improvement of quality assessment of analyzed studies, more detailed analysis of architecture quality attributes, analysis of implementation in various domains of business and human life, exploration of human factors and organizational issues, and addressing maintenance and operation challenges. In addition, the inclusion of domain experts in the preparation and implementation of these literature reviews is recommended for increasing the accuracy and validity

of the findings. And finally, the creation of multidisciplinary teams with experts from social sciences (e.g., sociologists, psychologists, economists) will enable a more comprehensive approach to the analysis of human and organizational factors at different stages of the microservice life cycle, resulting in more comprehensive and reliable literature reviews.

From the methodological standpoint, it is recommended to use qualitative social science methods to obtain more structured findings and methodologically grounded data analysis of unstructured text in analyzed studies. This research recommendation will lead to the development of theories about the practice, increasing the knowledge base in this area of software engineering.

References

[1]. Cerny T., Donahoo M. J., Pechanec J. Disambiguation and Comparison of SOA, Microservices and Self-Contained Systems. In Proc. of the International Conference on Research in Adaptive and Convergent Systems, 2017. pp. 228-235. doi: 10.1145/3129676.3129682.

[2]. Newman S. Building Microservices: Designing Fine-Grained Systems. O'Reilly Media, Sebastopol, CA, USA, 2021.

[3]. Dragoni N., Giallorenzo S., Lafuente A. L., Mazzara M., Montesi F., Mustafin R., Safina L. Microservices: Yesterday, Today, and Tomorrow. In Mazzara M. and Meyer B. (eds.) Present and Ulterior Software Engineering. Springer International Publishing, Cham, Switzerland, 2017. pp. 195-216. doi: 10.1007/978-3-319-67425-4_12.

[4]. Alaasam A. B., Radchenko G., Tchernykh A. Refactoring the Monolith Workflow into Independent Micro-Workflows to Support Stream Processing. Programming and Computer Software, 47(8), 2021, pp. 591–600. doi: 10.1134/S0361768821080077.

[5]. Baškarada S., Nguyen V., Koronios A. Architecting Microservices: Practical Opportunities and Challenges. Journal of Computer Information Systems, 60(5), 2018, pp. 428-436. doi: 10.1080/08874417.2018.1520056.

[6]. Niño-Martínez V. M., Ocharán-Hernández J. O., Limón X., Pérez-Arriaga J. C. A Microservice Deployment Guide. Programming and Computer Software, 48(8), 2022, pp. 632-645. doi: 10.1134/S0361768822080151.

[7]. Larrucea X., Santamaria I., Colomo-Palacios R., Ebert C. Microservices. IEEE Software, 35(3), 2018, pp. 96–100. doi: 10.1109/MS.2018.2141030.

[8]. Ebert C., Gallardo G., Hernantes J., Serrano N. DevOps. IEEE Software, 33(3), 2016, pp. 94–100. doi: 10.1109/MS.2016.68.

[9]. Bosch J. Continuous Software Engineering. Springer International Publishing, Cham, Switzerland, 2014.

[10]. Bruce M., Pereira P. A. Microservices in Action. Manning Publications, Shelter Island, NY, USA, 2019.

[11]. Abdellatif M., Shatnawi A., Mili H., Moha N., El Boussaidi G., Hecht G., Privat J., Guéhéneuc Y-G. A taxonomy of service identification approaches for legacy software systems modernization. Journal of Systems and Software, 173, 2021, pp. 110868. doi: 10.1016/j.jss.2020.110868.

[12]. Henry A., Ridene Y. Migrating to Microservices. In Bucchiarone A., Dragoni N., Dustdar S., Lago P., Mazzara M., Rivera V., Sadovykh A. (eds.) Microservices: Science and Engineering. Springer International Publishing, Cham, Switzerland, 2020, pp. 45-72. doi: 10.1007/978-3-030-31646-4_3.

[13]. Stojkov A., Stojanov Z. Review of methods for migrating software systems to microservices architecture. Journal of engineering management and competitiveness, 11(2), 2021, pp. 152-162.

[14]. Valdivia J. A., Lora-González A., Limón X., Cortes-Verdin K., Ocharán-Hernández J. O. Patterns related to microservice architecture: a multivocal literature review. Programming and Computer Software, 46(8), 2020, pp. 594–608. doi: 10.1134/S0361768820080253.

[15]. Kitchenham B. Procedures for Performing Systematic Review. Technical report TR/SE-0401. Software Engineering Group. Department of Computer Science. Keele University, Keele, UK, 2004.

[16]. Petersen K., Feldt R., Mujtaba S., Mattsson M. Systematic Mapping Studies in Software Engineering. In Proc. of the 12th International Conference on Evaluation and Assessment in Software Engineering, 2008. pp. 68-77.

[17]. Kitchenham B., Budgen D., Brereton P. Evidence-based software engineering and systematic reviews. CRC Press, Boca Raton, FL, USA, 2016.

[18]. Garousi V., Felderer M., Mäntylä M. V. Guidelines for including grey literature and conducting multivocal literature reviews in software engineering. Information and Software Technology, 106, 2019, pp. 101-121. doi: 10.1016/j.infsof.2018.09.006.

[19]. Arvanitou E.M., Ampatzoglou A., Bibi S., Chatzigeorgiou A., Deligiannis I. Applying and Researching DevOps: A Tertiary Study. IEEE Access, 10, 2022, pp. 61585-61600. doi: 10.1109/ACCESS.2022.3171803.

[20]. Cadavid H., Andrikopoulos V., Avgeriou P. Architecting systems of systems: A tertiary study. Information and Software Technology, 118, 2020, pp. 106202. doi: 10.1016/j.infsof.2019.106202.

[21]. Delavari V., Shaban E., Janssen M., Hassanzadeh A. Thematic mapping of cloud computing based on a systematic review: a tertiary study. Journal of Enterprise Information Management, 33, 2019, pp. 161-190. doi: 10.1108/JEIM-02-2019-0034.

[22]. Hoda R., Salleh N., Grundy J., Tee H. M. Systematic literature reviews in agile software development: A tertiary study. Information and Software Technology, 85, 2017, pp. 60-70. doi: 10.1016/j.infsof.2017.01.007.

[23]. Raatikainen M., Tiihonen J., Männistö T. Software product lines and variability modeling: A tertiary study. Journal of Systems and Software, 149, 2019, pp. 485-510. doi: 10.1016/j.jss.2018.12.027.

[24]. Tran H. K. V., Unterkalmsteiner M., Börstler J., bin Ali N. Assessing test artifact quality-A tertiary study. Information and Software Technology, 139, 2021, pp. 106620. doi: 10.1016/j.infsof.2021.106620.

[25]. Liu X., Li S., Zhang H., Zhong C., Wang Y., Waseem M., Babar M. A. Research on Microservice Architecture: A Tertiary Study. SSRN Electronic Journal, 2022, p. 23. doi: 10.2139/ssrn.4204345.

[26]. Costa D. I. C., Filho E. P. e S., da Silva R. F., Quaresma Gama T. D. de C., Cortés M. I. Microservice Architecture: A Tertiary Study. In Proc. of the 14th Brazilian Symposium on Software Components, Architectures, and Reuse, 2020. pp. 61-70. doi: 10.1145/3425269.3425277.

[27]. Wohlin C. Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. In Proc. of the 18th International Conference on Evaluation and Assessment in Software Engineering, 2014. pp. 38:1-38:10. doi:10.1145/2601248.2601268.

APPENDIX A. List of secondary studies

[SS01] Tighilt, R., Abdellatif, M., Moha, N., Mili, H., Boussaidi, G. E., Privat, J., & Guéhéneuc, Y. G. (2020). On the study of microservices antipatterns: A catalog proposal. In Proceedings of the European Conference on Pattern Languages of Programs 2020 (pp. 1-13). https://doi.org/10.1145/3424771.3424812.

[SS02] Chávez, K., Cedillo, P., Espinoza, M., & Saquicela, V. (2019). A systematic literature review on composition of microservices through the use of semantic annotations: solutions and techniques. In 2019 International Conference on Information Systems and Computer Science (INCISCOS) (pp. 311-318). https://doi.org/10.1109/INCISCOS49368.2019.00056.

[SS03] Valdivia, J. A., Limón, X., & Cortes-Verdin, K. (2019). Quality attributes in patterns related to microservice architecture: a Systematic Literature Review. In 2019 7th International Conference in Software Engineering Research and Innovation (CONISOFT) (pp. 181-190). https://doi.org/10.1109/10.1109/CONISOFT.2019.00034.

[SS04] Schmidt, R. A., & Thiry, M. (2020, June). Microservices identification strategies: A review focused on Model Driven Engineering and Domain Driven Design approaches. In 2020 15th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). https://doi.org/10.23919/CISTI49556.2020.9141150.

[SS05] Prasetyo, Y. A. & Suhardi (2018). Microservice Platform for Smart City: Concepts, Services and Technology. In 2018 International Conference on Information Technology Systems and Innovation (ICITSI) (pp. 358-363). https://doi.org/10.1109/ICITSI.2018.8695927.

[SS06] Karabey Aksakalli, I., Çelik, T., Can, A. B., & Tekinerdogan, B. (2021). Deployment and communication patterns in microservice architectures: A systematic literature review. Journal of Systems and Software, 180, 111014. https://doi.org/10.1016/j.jss.2021.111014.

[SS07] Li, S., Zhang, H., Jia, Z., Zhong, C., Zhang, C., Shan, Z., ... & Babar, M. A. (2021). Understanding and addressing quality attributes of microservices architecture: A Systematic literature review. Information and software technology, 131, 106449. https://doi.org/10.1016/j.infsof.2020.106449.

[SS08] Garriga, M. (2017). Towards a taxonomy of microservices architectures. In International conference on software engineering and formal methods (pp. 203-218). https://doi.org/10.1007/978-3-319-74781-1_15.

[SS09] Fritzsch, J., Bogner, J., Zimmermann, A., & Wagner, S. (2019). From monolith to microservices: A classification of refactoring approaches. In International Workshop on Software Engineering Aspects

of Continuous Development and New Paradigms of Software Production and Deployment (pp. 128-141). https://doi.org/10.1007/978-3-030-06019-0_10.

[SS10] Razzaq, A. (2020). A systematic review on software architectures for IoT systems and future direction to the adoption of microservices architecture. SN Computer Science, 1(6), 1-30. https://doi.org/10.1007/s42979020-00359-w.

[SS11] Wang, L., Jiang, Y. X., Wang, Z., Huo, Q. E., Dai, J., Xie, S. L., ... & Jiang, Z. P. (2022). The operation and maintenance governance of microservices architecture systems: A systematic literature review. Journal of Software: Evolution and Process, e2433. https://doi.org/0.1002/smr.2433.

[SS12] Jawaddi, S. N. A., Johari, M. H., & Ismail, A. (2022). A review of microservices autoscaling with formal verification perspective. Software: Practice and Experience, 52(11), 2476-2495. https://doi.org/10.1002/spe.3135.

[SS13] Nguyen, H. X., Zhu, S., & Liu, M. (2022). A Survey on Graph Neural Networks for Microservice-Based Cloud Applications. Sensors, 22(23), 9492. https://doi.org/10.3390/s22239492.

[SS14] Söylemez, M., Tekinerdogan, B., & Kolukısa Tarhan, A. (2022). Challenges and Solution Directions of Microservice Architectures: A Systematic Literature Review. Applied Sciences, 12(11), 5507. https://doi.org/10.3390/app12115507.

[SS15] de Almeida, M. G., & Canedo, E. D. (2022). Authentication and Authorization in Microservices Architecture: A Systematic Literature Review. Applied Sciences, 12(6), 3023. https://doi.org/10.3390/app12063023.

[SS16] Berardi, D., Giallorenzo, S., Mauro, J., Melis, A., Montesi, F., & Prandini, M. (2022). Microservice security: a systematic literature review. PeerJ Computer Science, 8, e779. https://doi.org/10.7717/peerj-cs.779.

[SS17] Vera-Rivera, F. H., Gaona, C., & Astudillo, H. (2021). Defining and measuring microservice granularity – a literature overview. PeerJ Computer Science, 7, e695. https://doi.org/10.7717/peerj-cs.695.

[SS18] Leines-Vite, L., Pérez-Arriaga, J. C., & Limón, X. (2021). Confidentiality and Integrity Mechanisms for Microservices Communication. International Journal of Network Security & Its Applications, 13(6), 85-103. https://doi.org/10.5121/ijnsa.2021.13607.

[SS19] Osses, F., Márquez, G., & Astudillo, H. (2018). An exploratory study of academic architectural tactics and patterns in microservices: A systematic literature review. Avances en Ingenieria de Software a Nivel Iberoamericano, CIbSE, 2018, 71-84.

[SS20] Staegemann, D., Volk, M., Shakir, A., Lautenschläger, E., & Turowski, K. (2021). Examining the Interplay Between Big Data and Microservices–A Bibliometric Review. Complex Systems Informatics and Modeling Quarterly, (27), 87-118. https://doi.org/10.7250/csimq.2021-27.04.

[SS21] Nunes, J. P. K., Bianchi, T., Iwasaki, A. Y., & Nakagawa, E. Y. (2021). State of the art on microservices autoscaling: An overview. Anais do XLVIII Seminário Integrado de Software e Hardware, 30-38. https://doi.org/10.5753/semish.2021.15804.

[SS22] Lelovic, L., Mathews, M., Elsayed, A., Cerny, T., Frajtak, K., Tisnovsky, P., & Taibi, D. (2022). Architectural languages in the microservice era: a systematic mapping study. In Proceedings of the Conference on Research in Adaptive and Convergent Systems (pp. 39-46). https://doi.org/10.1145/3538641.3561486.

[SS23] Cerny, T., Donahoo, M. J., & Trnka, M. (2018). Contextual understanding of microservice architecture: current and future directions. ACM SIGAPP Applied Computing Review, 17(4), 29-45. https://doi.org/10.1145/3183628.3183631.

[SS24] Wolfart, D., Assunção, W. K., da Silva, I. F., Domingos, D. C., Schmeing, E., Villaca, G. L. D., & Paza, D. D. N. (2021). Modernizing legacy systems with microservices: A roadmap. In Evaluation and Assessment in Software Engineering (pp. 149-159). https://doi.org/10.1145/3463274.3463334.

[SS25] Waseem, M., Liang, P., Márquez, G., & Di Salle, A. (2020). Testing microservices architecture-based applications: A systematic mapping study. In 2020 27th Asia-Pacific Software Engineering Conference (APSEC) (pp. 119-128). https://doi.org/10.1109/APSEC51365.2020.00020.

[SS26] Velepucha, V., & Flores, P. (2021). Monoliths to microservices-Migration Problems and Challenges: A SMS. In 2021 Second International Conference on Information Systems and Software Technologies (ICI2ST) (pp. 135-142). https://doi.org/10.1109/ICI2ST51859.2021.00027.

[SS27] Di Francesco, P., Malavolta, I., & Lago, P. (2017). Research on architecting microservices: Trends, focus, and potential for industrial adoption. In 2017 IEEE International Conference on Software Architecture (ICSA) (pp. 21-30). https://doi.org/10.1109/ICSA.2017.24.

[SS28] Pereira-Vale, A., Márquez, G., Astudillo, H., & Fernandez, E. B. (2019). Security mechanisms used in microservices-based systems: A systematic mapping. In 2019 XLV Latin American Computing Conference (CLEI) (pp. 01-10). https://doi.org/10.1109/CLEI47609.2019.235060.

[SS29] Alshuqayran, N., Ali, N., & Evans, R. (2016). A systematic mapping study in microservice architecture. Proceedings-2016 IEEE 9th International Conference on Service-Oriented Computing and Applications, SOCA 2016 pp. 44–51. https://doi.org/10.1109/SOCA.2016.15.

[SS30] Gortney, M. E., Harris, P. E., Cerny, T., Al Maruf, A., Bures, M., Taibi, D., & Tisnovsky, P. (2022). Visualizing Microservice Architecture in the Dynamic Perspective: A Systematic Mapping Study. IEEE Access, 10, 119999-120012. https://doi.org/10.1109/ACCESS.2022.3221130.

[SS31] Santos, R., Soares, P., Rodrigues, E., Maia, P. H. M., & Silveira, A. (2022). How Blockchain and Microservices are Being Used Together: a Systematic Mapping Study. In 2022 IEEE/ACM 5th International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB) (pp. 39-46). https://doi.org/10.1145/3528226.3528371.

[SS32] Niño-Martínez, V. M., Ocharán-Hernández, J. O., Limón, X., & Pérez-Arriaga, J. C. (2021, October). Microservices Deployment: A Systematic Mapping Study. In 2021 9th International Conference in Software Engineering Research and Innovation (CONISOFT) (pp. 24-33). https://doi.org/10.1109/CONISOFT52520.2021.00016.

[SS33] Waseem, M., Liang, P., & Shahin, M. (2020). A systematic mapping study on microservices architecture in DevOps. Journal of Systems and Software, 170, 110798. https://doi.org/10.1016/j.jss.2020.110798.

[SS34] Di Francesco, P., Lago, P., & Malavolta, I. (2019). Architecting with microservices: A systematic mapping study. Journal of Systems and Software, 150, 77-97. https://doi.org/10.1016/j.jss.2019.01.001.

[SS35] Hannousse, A., & Yahiouche, S. (2021). Securing microservices and microservice architectures: A systematic mapping study. Computer Science Review, 41, 100415. https://doi.org/10.1016/j.cosrev.2021.100415.

[SS36] Vural, H., Koyuncu, M., & Guney, S. (2017). A systematic literature review on microservices. In International Conference on Computational Science and Its Applications (pp. 203-217). https://doi.org/10.1007/978-3-319-62407-5_14.

[SS37] Taibi, D., Lenarduzzi, V., & Pahl, C. (2019). Continuous architecting with microservices and DevOps: A systematic mapping study. In International Conference on Cloud Computing and Services Science (pp. 126-151). https://doi.org/10.1007/978-3-030-29193-8_7.

[SS38] Razzaq, A., & Ghayyur, S. A. (2022). A systematic mapping study: The new age of software architecture from monolithic to microservice architecture—awareness and challenges. Computer Applications in Engineering Education. https://doi.org/10.1002/cae.22586.

[SS39] Bushong, V., Abdelfattah, A. S., Maruf, A. A., Das, D., Lehman, A., Jaroszewski, E., ... & Bures, M. (2021). On microservice analysis and architecture evolution: A systematic mapping study. Applied Sciences, 11(17), 7856. https://doi.org/10.3390/app11177856.

[SS40] Ghani, I., Wan-Kadir, W. M., Mustafa, A., & Babir, M. I. (2019). Microservice testing approaches: A systematic literature review. International Journal of Integrated Engineering, 11(8), 65-80. https://doi.org/10.30880/ijie.2019.11.08.008.

[SS41] Velepucha, V., Flores, P., & Torres, J. (2019). Migration of monolithic applications towards microservices under the vision of the information hiding principle: a systematic mapping study. In The International Conference on Advances in Emerging Trends and Technologies (pp. 90-100). https://doi.org/10.1007/978-3030-32022-5_9.

[SS42] Yu, D., Jin, Y., Zhang, Y., & Zheng, X. (2019). A survey on security issues in services communication of Microservices-enabled fog applications. Concurrency and Computation: Practice and Experience, 31(22), e4436. https://doi.org/10.1002/cpe.4436.

[SS43] Pahl, C., & Jamshidi, P. (2016). Microservices: A Systematic Mapping Study. In the 6th International Conference on Cloud Computing and Services Science (CLOSER 2016) CLOSER (1), 137-146.

[SS44] Parker, G., Kim, S., Al Maruf, A., Cerny, T., Frajtak, K., Tisnovsky, P., & Taibi, D. (2023). Visualizing Anti-Patterns in Microservices at Runtime: A Systematic Mapping Study. IEEE Access, 11, 4434-4442. https://doi.org/10.1109/ACCESS.2023.3236165.

Информация об авторах / Information about authors

Желько СТОЯНОВ имеет PhD степень в области компьютерных наук и информатики. Профессор Нови-Садского университета, технический факультет “Михайло Пупин”, Сербия. Научные интересы: разработка программного обеспечения, жизненный цикл программного обеспечения, бизнес-информатика, качественные исследования, управление знаниями, человеческие аспекты инженерии.

Zeljko STOJANOV – PhD degree in Computer Science and Informatics. Full professor at the University of Novi Sad, Technical Faculty “Mihajlo Pupin,” Serbia. Research interests: software engineering, software life cycle, business informatics, qualitative research, knowledge management, human aspects of engineering.

Илья ХРИСТОСКИЙ имеет степень PhD в области технических наук. Профессор университета “Св. Климент Охридский” в Битоле, кафедра бизнес-информатики на экономическом факультете в Прилепе, Северная Македония. Область научных интересов: моделирование и симуляция, стохастические сети Петри, системная динамика, системы баз данных, бизнес-информатика, управление информацией и знаниями, финансовая эконометрика.

Ilija HRISTOSKI – PhD degree in Technical Sciences. Full professor at the University “St. Kliment Ohridski” in Bitola, Department of Business Informatics at the Faculty of Economics in Prilep, North Macedonia. Research interests: modelling and simulation, stochastic Petri Nets, system dynamics, database systems, business informatics, information and knowledge management, financial econometrics.

Елена СТОЯНОВА имеет степень PhD в области математических наук. Доцент Нови-Садского университета, технический факультет “Михайло Пупин”, Сербия. Область научных интересов: прикладная математика, дифференциальная геометрия, математическое образование.

Jelena STOJANOV – PhD degree in Mathematics. Associate professor at the University of Novi Sad, Technical Faculty “Mihajlo Pupin,” Serbia. Research interests: applied mathematics, differential geometry, mathematics education.

Александра СТОЙКОВА – имеет степень магистра информационных технологий. Аспирант и ассистент преподавателя на техническом факультете “Михайло Пупин” Нови-Садского университета, Сербия. Область научных интересов: разработка программного обеспечения, информационные системы, базы данных, реинжиниринг программных систем.

Aleksandra STOJKOV – MsC degree in Information Technology. PhD student and teaching assistant at the University of Novi Sad, Technical Faculty “Mihajlo Pupin,” Serbia. Research interests: software engineering, information systems, databases, reengineering of software systems.

DOI: 10.15514/ISPRAS-2024-36(1)-8

Deep Learning for Non-functional Requirements: A Convolutional Neural Network Approach

[1] *S. E. Martinez García,* ORCID: 0009-0006-3907-8384 <*estefmartinezgarcia@gmail.com*>
[2] *C. A. Fernández-y-Fernández,* ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>
[2] *E. G. Ramos Pérez,* ORCID: 0000-0001-8337-4195 <*erik@mixteco.utm.mx*>

[1] *División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, Huajuapan de León, Oax., México.*
[2] *Instituto de Computación, Universidad Tecnológica de la Mixteca, Huajuapan de León, Oax. México.*

Abstract. The Requirements Engineering (ER) phase plays a critical role in software development, as any shortcomings during this stage can lead to project failure. Analysts rely on Requirements Specification (RS) to define a comprehensive list of quality requirements. The process of requirements classification, within RS, involves assigning each requirement to its respective class, presenting analysts with the challenge of accurate categorization. This research focuses on enhancing the classification of non-functional requirements (NFR) using a Convolutional Neural Network (CNN). The study also emphasizes the significance of preprocessing techniques, the implementation of sampling strategies, and the incorporation of pre-trained word embeddings such as Fasttext, Glove, and Word2vec. Evaluation of the proposed approach is performed using metrics like Recall, Precision, and F1, resulting in an average performance improvement of up to 30% compared to related work. Additionally, the model is assessed concerning its utilization of pre-trained word embeddings through ANOVA analysis, providing valuable insights into its effectiveness. This study aims to demonstrate the utility of CNNs and pre-trained word embeddings in the classification of NFRs, offering valuable contributions to the field of Requirements Engineering and enhancing the overall software development process.

Keywords: deep learning; non-functional requirements; convolutional neural network; requirements engineering.

For citation: Martinez-García S. E., Fernández-y-Fernández C. A., Ramos-Pérez E. G. Deep Learning for Non-functional Requirements: A Convolutional Neural Network Approach. Trudy ISP RAN/Proc. ISP RAS, vol. 36, issue 1, 2024. pp. 131-142. DOI: 10.15514/ISPRAS-2024-36(1)-8.

Full text: Martinez García S. E., Fernández-y-Fernández C. A., Ramos Pérez E. G. Classification of Non-Functional Requirements Using Convolutional Neural Networks. Programming and Computer Software, 2023, Vol. 49, No. 8, pp. 705–711. DOI: 10.1134/S0361768823080133.

Глубокое обучение при выработке нефункциональных требований: подход на основе сверточных нейронных сетей

[1] *С. Э. Мартинес Гарсия*, ORCID: 0009-0006-3907-8384 <*estefmartinezgarcia@gmail.com*>
[2] *К. А. Фернандес-и-Фернандес*, ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>
[2] *Э. Г. Рамос Перес*, ORCID: 0000-0001-8337-4195 <*erik@mixteco.utm.mx*>

[1] Аспирантура Технологического университета Миштека,
Уахуапан де Леон, Оахака, Мексика.
[2] *Институт вычислений Технологического университета Миштека,*
Уахуапан де Леон, Оахака, Мексика.

Аннотация. Фаза разработки требований (ER) играет решающую роль в разработке программного обеспечения, поскольку любые недостатки на этом этапе могут привести к провалу проекта. Аналитики полагаются на спецификацию требований (RS) для определения полного списка требований к качеству. Процесс классификации требований в рамках RS включает отнесение каждого требования к соответствующему классу, что ставит перед аналитиками задачу точной классификации. Данное исследование направлено на улучшение качества классификации нефункциональных требований (NFR) на основе применения сверточной нейронной сети (CNN). В исследовании также подчеркивается важность методов предварительной обработки, реализации стратегий выборки и включения предварительно обученных векторных представлений слов, таких как Fasttext, Glove и Word2vec. Оценка предлагаемого подхода выполняется с использованием таких метрик, как Recall, Precision и F1, что приводит к среднему улучшению производительности до 30% по сравнению с другими подходами. Кроме того, модель оценивается в отношении использования предварительно обученных векторных представлений слов с помощью анализа ANOVA, предоставляя ценную информацию о ее эффективности. Это исследование направлено на то, чтобы продемонстрировать полезность CNN и предварительно обученных векторных представлений слов в классификации NFR, предлагая ценный вклад в области инженерии требований и улучшая общий процесс разработки программного обеспечения.

Ключевые слова: глубокое обучение; нефункциональные требования; сверточная нейронная сеть; инженерия требований.

Для цитирования: Мартинес-Гарсия С. Э., Фернандес-и-Фернандес К. А., Рамос-Перес Э. Г. Глубокое обучение при выработке нефункциональных требований: подход на основе сверточных нейронных сетей. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 131–142 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–8.

Полный текст: Мартинес Гарсия С. Э., Фернандес-и-Фернандес К. А., Рамос Перес Э. Г. Классификация нефункциональных требований на основе сверточных нейронных сетей. Programming and Computer Software, 2023, т. 49, № 8, стр. 705–711 (на английском языке). DOI: 10.1134/S0361768823080133.

1. Introduction

During the initial phases of the software development life cycle, regardless of the model that is intended to be followed, the requirements phase is declared as a key piece to achieving a successful development [1-3, 8, 31]. If the requirements are not discovered and defined correctly in this early phase, failures arise during development, which promotes that the final delivery is that of incomplete software, that is to say, that it does not do what it should do, adding to that the Established times are not met and are extended, which will cause previously estimated costs to rise [2, 24].

For this reason, this phase is considered vital since the correct execution of the activities will prevent failure of software development [2, 5, 27, 30]. To combat this problem, analysts have used Requirements Engineering (RE) [4, 17, 23], which is characterized by producing a list of quality requirements as a final result. When carrying out the classification of requirements, there are difficulties of interpretation and identification [13, 18] (an inherent characteristic of natural language

[9, 25, 28]) to determine to which class each of these belong, since there are functional (FR) and non-functional (NFR) requirements, of which the latter contain subclasses. In addition to this difficulty, the extensive list of requirements is also presented, which can number in the thousands, so it would be a job that takes too much time and effort [12, 20, 26].

In this research machine learning techniques are used to apply them in the RE in the Requirements Specification (RS) activity, specifically on the classification of requirements. Said activity consists of identifying the class of requirement to which it belongs or simply differentiating between a requirement and information [13-14]. In particular, it will focus on NFRs, since they are frequently discriminated against because they are considered of little or no importance for software development, as well as the lack of knowledge to identify them [6-7, 11, 16].

2. Background

2.1 Data set

The NFR quality attributes data set, also known as the PROMISE [29] corpus, is a compilation of requirements specifications for 15 software projects developed by students at DePaul University as a term project for a course in Requirements Engineering; The language of the content is in English. The data set consists of 326 NFRs and 358 FR requirements. The NFR dataset lends itself, for purposes of this research, to the multi-label classification of various types of NFR requirements.

2.2 Input format for classifier

A classifier expects the data to be in the form of a list of strings of requirements (referred to as examples) in the form of a vector of one-hot words *one-hot* and an attached list of vectors representing the associated requirement class.

2.3 Sampling Strategies

The objective of the sampling strategies is to avoid the imbalance in the distribution of the class that the datasets constantly present, this imbalance causes the automatic learning algorithms to have low performance in the minority class; since the cost of misclassifying it is usually much higher than the cost of other misclassifications [10, 19, 32]. Therefore, when selecting the Promise data set, it was observed that the distribution of the set is unbalanced, so it is appropriate to use this strategy.

2.4 Evaluation metrics for classifier performance

To evaluate the model, the same metrics used by [10, 15, 33] will be taken as a reference, since in addition to establishing the improvement of the work done, they measure the performance of the model with respect to the correct predictions it makes. They are briefly shown below:

- **Accuracy**. It is the total percentage of cases classified correctly.
- **Recall**. It is the number of data correctly identified as positive out of the total number of true positives.
- **Score F1**. It can be interpreted as a weighted average of precision and *recall*, where an F1 score reaches its best value at 1 and its worst score at 0.
- **Precision**. Accuracy is the ratio of correct predictions to the total number of predicted correct predictions.

2.5 Related work

The aim of this section is to explore, identify and improve the CNN preprocessing and configuration bases proposed by [10, 15, 33]. The key point for the classification of NFR is that its nature is multiclass.

- In [33] does not use the *Promise* dataset, and performs binary classification. In their research, they do not show which configuration was used, and they do not mention any type of validation used, in addition, the authors mention that because the data set they used contains little information, the requirements classification obtained an accuracy of 73.
- [15], perform multiclass classification, if you have the *Promise* data set, but use all 12 classes, that is, both Functional and Non-functional Requirements. He used the embedded fastText model. He got 80 %, using cross- validation with parameters of $k = 10$ and applying the optimizer *AdamOptimizer*.
- [10], performs multiclass classification, also occupies *Promise* and implements experimentation on NFR, especially on 9 classes, since the other two that belong to this type of requirements have few examples, which that prevents the use of sampling strategies; therefore, at least two examples are required to carry it out. It does not refer to what type of strategy obtained the results it presents, so for the purposes of this paper both random oversampling (ROS) and random undersampling (RUS) were tested, however, in this section only the optimal result, which was ROS, is presented, to see the results of the experimentation with RUS see the section in the index. He used the embedded fastText and Word2Vec models. The accuracy result was 80.4% with the *Word2Vec* vectorization method, this was the highest compared to the pre-trained Fasttext matrix and a random weights matrix.

3. Experimentation and results

The experimentation seeks to determine the influence of the text preprocessing, the vectorization of the data, the ROS sampling strategy, the implementation of the pre-trained embedded matrices of Word2Vec, fastText, and Glove in the embedded layer of the CNN, and the hyper parameterization of its subsequent layers. The key point for the classification of NFR is that its nature is multiclass, so it was determined, based on previous experiments, to build models combining the improvements made in each previous experiment and, if possible, adapt them to this model. rating with CNN to try to increase rating metrics. So, the points to consider are the following:

- **Preprocessing**. Whether or not to include pre-cleaning of the Promise dataset, as well as applying lemmatization to words. It is also tested with 3 types of vectorizers: *TF-IDF*, *Tokenizer*, and *CountVectorizer*.
- **Hyperparameterization in the data partition**. Implementation of the sampling strategy ROS proposal.
- **CNN architecture**. Starting with the base architecture including or not the weights of the pre-trained matrices.

3.1 Results and comparison

The results shown below are presented gradually; that is, it indicates the way in which by adding the proposed techniques to a base CNN architecture, the classification performance improves.

Now we show the concentration of results obtained from experimentation with a progressive integration of proposals to improve the classification of NFR using NFR. First, in Table 1, where an average of *Recall* of 0.11, *Precision* of 0.01 and *F1* of 0.02 was achieved for the vectorizer test. TF-IDF; On the other hand, with the Tokenizer vectorizer, an average of 0.44 was obtained for *Recall*, 0.46 for *Precision* and 0.44 for the metric *F1*. On the other hand, with the CountVectorizer

vectorizer, it can be seen that the training of the network with that tool gained a *Recall* with 0.59, *Precision* with 0.53, and *F1* with 0.54. The second part of the experimentation is integrating the previous cleaning and lemmatization, in addition to making the hyper parameterization in the data partition; therefore, the averages resulting from this stage are those shown in Table 2. An increase in *accuracy* is observed, however, when directing attention to the average of the metrics, it is examined that when applying these proposals, a notable improvement is found for the case of the vectorizer *CountVectorizer*, where emphRecall has a 0.70, *Precision* with a 0.69 and *F1* with a 0.67. Therefore, the classifier has shown a particular behavior on *CountVectorizer*, both in this test and in the previous one. Finally, one more proposal is added, which is to train the model with the weights of the pre- trained matrix *Word2vec*, since it was a common denominator among the proposals of [10, 15, 33]. The averages obtained can be seen in Table 3, *Recall* with 0.72, *Precision* with 0.74, and *F1* with 0.72; Therefore, an increase in the averages could be observed using said weight matrix. Therefore, when observing the effect of each of these proposals implemented on the classifier, it was possible to determine that for the following experiments, it is convenient to use CountVectorizer since with the 2 vectorizers it can be seen that they do not promote an improvement with respect to the processing of the data. In addition to continuing to use the proposals based on what has been observed in the experiments already carried out.

Table 1. Base model results

Test	Vectorizer	Average		
		Recall	Precision	F1
1	TF-IDF	0.11	0.01	0.02
2	Tokenizer	0.44	0.46	0.44
3	CountVectorizer	0.59	0.53	0.54

Table 2. Base model results with integration of two proposals

Test	Vectorizer	Average		
		Recall	Precision	F1
1	TF-IDF	0.11	0.02	0.04
2	Tokenizer	0.41	0.45	0.39
3	CountVectorizer	0.70	0.69	0.67

Table 3. Base model results with integration of three proposals

Test	Vectorizer	Average		
		Recall	Precision	F1
1	TF-IDF	0.11	0.02	0.04
2	Tokenizer	0.42	0.41	0.40
3	CountVectorizer	0.72	0.74	0.72

CNN with the implementation of pre-trained matrices and ROS sampling strategy: For this experiment, a *Dropout* layer was added, after the embedded layer, as well as another set of a convolution followed by a *MaxPooling* layer.

Table 4 shows the results of the metrics obtained by implementing each of the different pre-trained embedded matrices. The text preprocessing, the ROS sampling strategy, and the new architecture were used for this training that was carried out with 100 epochs. However, the data vectorization was the important factor in improving the results. For this, the *CountVectorizer* library was used, which in addition to obtaining properties such as eliminating *stopwords*, calculating the frequency of words among others, also has the *ngram range* argument. This argument determines the lower and upper bound of the range of n values for different words, called *n-grams* [21-22]. Hence, this argument was essential and was considered to obtain a vectorization of words that helps to make

sense of each of the requirements according to their context. The *ngram range* that allowed the results shown in Table 4 to be obtained was (1, 4) for fastText and Word2Vec and (1, 2) for Glove. It is important to highlight that tests were carried out without *ngram range* and with ranges of (1,1), (1,2), (1,3), and (1,4), for each experimentation with the embedded matrices selected. Therefore, 5 tests were carried out for each script.

Making the comparison in Table 5, it can be seen how the base proposal with 100 epochs reflected a notable increase with respect to the optimal results obtained by [10] with 140 epochs. To determine if there really is an improvement with respect to the initial configurations, it has been proposed to carry out a statistical analysis in the following section.

Table 4. CNN results with 100 epochs for NFR classification using 3 types of embedded arrays

Embedded matrix	Average		
	Recall	Precision	F1
Word2vec	0.88	0.90	0.88
FastText	0.83	0.85	0.83
Glove	0.82	0.79	0.79

3.2 Evaluation of the proposals implemented to the CNN model with cross-validation k-fold by analysis of variance

To evaluate the model implemented for the CNN with the different 3 pre-trained matrices used, the analysis of variance (ANOVA) was performed. Taking the averages of *F1* resulting from cross-validation training *k-fold* with *k*=10, since said metric represents the average between *Recall* and *Precision*. The goodness of *F1* is useful when there is an unequal distribution in the classes, this being the case of the data set being used, for this reason, it was decided to perform ANOVA on that metric.

Table 5. Results of the metrics obtained from the optimal preprocessing and architecture of [10] vs. results of the preprocessing and base architecture proposed in this paper

(a) Results obtained by [10] for 140 epochs

Embedded matrix	Average		
	Recall	Precision	F1
Random	0.66	0.66	0.66
Word2vec	0.75	0.79	0.77
FastText	0.73	0.76	0.76

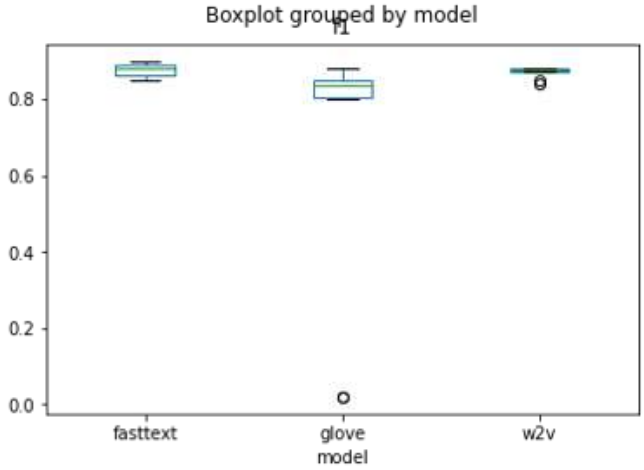
(b) Results obtained in this paper for 100 epochs as initial run for base architecture

Embedded matrix	Average		
	Recall	Precision	F1
Glove	0.82	0.79	0.79
Word2vec	0.88	0.90	0.88
FastText	0.83	0.85	0.83

To begin the ANOVA calculation, the average F1 metric results of each display obtained from cross-validated training for Word2vec, Glove, and FastText were collected. An analysis of the distribution of said data was carried out, where it can be seen with the naked eye in Fig. 1a that for the model with Glove atypical values are reflected, for example, containing an average of 0.02 (see Fig. 1a) for *fold* number 6 and 10, on the other hand, Word2vec presents outliers to the average, in contrast to FastText which contains no outliers.

Now, as already mentioned, the purpose of this proposal was to evaluate the difference between each model and the proposals implemented with respect to the metric *F1*. The overall mean of the metric

F1, as shown in Fig. 2a, was 0.81 for 30 samples (N) and its confidence interval (*CI*, for its acronym in English) at 95% was (0.72,0.89). The specific means for the groups by model with *N=10* were the following (see Fig. 2b): FastText with a mean of 0.88 and a CI at 95% of (0.86,0.88), for Glove's case obtained an average of 0.67 with a 95% CI of (0.45,0.90), on the other hand, Word2vec returned an average of 0.87 and a 95% CI of (0.86,0.88).



(a) Boxplot of the averages of the F1 metrics belonging to the models trained with the 3 embedded matrices

	kf1	kf2	kf3	kf4	kf5	kf6	kf7	kf8	kf9	kf10
W2vec	0.88	0.88	0.87	0.88	0.88	**0.85**	0.87	0.88	0.87	**0.84**
Glove	0.80	0.88	0.85	0.81	0.86	**0.02**	0.85	0.83	0.84	**0.02**
FastText	**0.90**	0.89	0.89	0.85	0.87	0.86	0.85	0.88	0.88	**0.90**

(b) Table of averages of the F1 metric for models trained with pre-trained matrices

Fig. 1: Table of averages of the F1 metric for the trained models and distribution diagram of said data

Variable	N	Average	SD	SE	95% Conf	Interval
F1	30	0.81	0.21	0.04	0.72	0.89

(a) Summary table for total samples against F1 metric averages

Variable	Average	SD	SE	95% Conf	Interval
V1	0.57	0.05	0.01	0.54	0.60
V2	0.70	0.04	0.01	0.67	0.72
V3	0.88	0.02	0.01	0.86	0.89

(b) Table of F1 averages of individual models

Fig. 2: Tables of general and particular averages of the models analyzed with respect to F1

The standard significance value of = 0.05 was taken as a reference, however, when calculating ANOVA for the samples presented, it can be seen in Table 6 that there is a statistically significant difference between the models presented because it was obtained a p=1.148591e-16. Therefore, with the observed data, there is sufficient evidence to assume a significant difference between the models exposed in this evaluation. It is worth mentioning that the assumptions of the test were verified using the Kruskall-Wallis non-parametric test. In addition, the honestly significant difference test of *Tukey (Tukey's HSD)* was performed, which is used to test the differences between the means of the sample in terms of significance, testing the differences by peers.

Table 6: ANOVA result for the models with respect to F1

	degrees of freedom	Sum of Squares	average of squares	F	P
Models	2.0	0.48	0.24	191.17	1.14e-16
Residual	27.0	0.03	0.001		

4. Conclusions and future work

NLP, a subset of AI, has been great allies in solving text classification problems. However, the solution to these problems is generally reserved for large-scale problems with large volumes of data samples, so working with databases with few examples suggests low results with respect to classification performance. This problem has been present when trying to optimize the classification of software requirements since during the analysis of these they are embodied in a document that is frequently represented by sentences with little text or information. Furthermore, requirements data sets only contain hundreds to thousands of documents, which is orders of magnitude less in volume than is typically considered necessary for deep learning. In addition, taking into account that there are different classes of requirements, especially the NFRs that are obtained from the Promise data set, which makes the classification task difficult to obtain desired results. That is why in this work, an investigation was carried out to determine which were the best strategies used in the state of the art that led to obtaining acceptable results in each of the investigations, in order to unify them and observe if said strategies together reflected a performance efficiency for the classification of NFRs using convolutional neural networks. In principle, it was possible to observe, for experimentation, how to vectorize the data set with the embedding of words, apply methods of Random over sampling strategies, hyper parameterize the configuration when performing the data partition, an increase in the average of the metrics was reflected Recall, Precision and F1 against the state of the art, since unlike [10] up to 30% was obtained in the average increase of the mentioned metrics. This is the first guideline to apply them to CNN. When implementing these proposals to the CNN architecture, as well as performing its hyper parameterization, it was decided to test with an embedded layer with or without weights, thus showing the importance of using pre-trained matrices that allow improvement in terms of the classification of text. To determine if there really was an improvement in the classifier, the ANOVA analysis was performed, which revealed a p-value of 0.05, therefore, according to the standard significance of, if there is a significant improvement between the models presented. Hence, it can be said that the application of the proposals for the classification of NFR with CNN resulted in the improvement of the performance of the classifier with respect to the state of the art.

The future work that is planned to be carried out in the first instance is to search for data sets with more examples to observe the performance behavior of the classifier, as well as to experiment with recurrent neural networks such as LSTM. Also, due to the challenge represented by carrying out the training with k-folds and the CNN base architecture on hardware in which the memory did not support the execution, it is planned to look for an institution that can provide some resources for research on this topic.

References

[1]. José Alfonso Aguilar, Anibal Zaldívar-Colado, Carolina Tripp-Barba, Roberto Espinosa, Sanjay Misra, Carlos Eduardo Zurita: A Survey About the Impact of Requirements Engineering Practice in Small-Sized Software Factories in Sinaloa, Mexico. International Conference on Computational Science and Its Applications, 331–340. Springer, 2018.

[2]. Alla Sujatha, Pilar Pazos, Rolando DelAguila: The Impact of Requirements Management Documentation on Software Project Outcomes in Health Care. IIE Annual Conference. Proceedings, 1419–1423. Institute of Industrial and Systems Engineers (IISE), 2017.

[3]. S. Almeyda, A. Davila: Process Improvement in Software Requirements Engineering: A Systematic Mapping Study. Programming and Computer Software, 48(8):513–533, 2022.
[4]. Aybuke Aurum, Claes Wohlin: A value- based approach in requirements engineering: explaining some of the fundamental concepts. International Working Conference on Requirements Engineering: Foundation for Software Quality, 109–115. Springer, 2007.
[5]. Christoph Becker, Stefanie Betz, Ruzanna Chitchyan, Leticia Duboc, Steve M. Easterbrook, Birgit Penzenstadler, Norbet Seyff, Colin C. Venters: Requirements: The key to sustainability. IEEE Software, vol.33:56–65, 2016.
[6]. Agustin Casamayor, Daniela Godoy, Marcelo R. Campo: Identification of non-functional requirements in textual specifications: A semi-supervised learning approach. Information and Software Technology, vol.52:436–445, 2010.
[7]. Lawrence Chung, Brian A. Nixon, Eric Yu, John Mylopoulos: Non-functional requirements in software engineering. Springer Science & Business Media, 2012.
[8]. J. R. Aguilar Cisneros, C. A. Fernandez-y-Fernandez, Genaro de la Rosa Garcia, A. Leon: Automotive Post-Collision Control Software System: Requirements and Verification. Programming and Computer Software, 47(8):735–745, 2021.
[9]. F. Dalpiaz, A. Ferrari, X. Franch, C. Palomares: Natural Language Processing for Requirements Engineering: The Best Is Yet to Come. IEEE Software, vol.35:115–119, 2018, ISSN 0740-7459.
[10]. Vivian Lin Fong: Software Requirements Classification Using Word Embeddings and Convolutional Neural Networks. Cal Poly, 2018.
[11]. Martin Glinz: On non-functional requirements. Requirements Engineering Conference, 2007. RE'07. 15th IEEE International, 21–26. IEEE, 2007.
[12]. Margaret Hamill, Katerina Goseva-Popstojanova: Common trends in software fault and failure data. IEEE Transactions on Software Engineering, vol.35:484– 496, 2009.
[13]. Ishrar Hussain, Olga Ormandjieva, Leila Kosseim: LASR: A tool for large scale annotation of software requirements. 2012 Second IEEE International Workshop on Empirical Requirements Engineering (EmpiRE), 57–60. IEEE, 2012.
[14]. IEEE Computer Society, Software Engineering Standards Committee, IEEE SA Standards Board: IEEE Recommended Practice for Software Requirements Specifications. 1998.
[15]. Raul Navarro Almanza, Reyes Juárez-Ramirez, Guillermo Licea: Towards Supporting Software Engineering Using Deep Learning: A Case of Software Requirements Classification. 2017 5th International Conference in Software Engineering Research and Innovation (CONISOFT), 116–120. IEEE, 2017.
[16]. Mohamad Kassab: Non-functional requirements: modeling and assessment. VDM Verlag, 2009.
[17]. Marjo Kauppinen, Juha Savolainen, Tomi Mannisto: Requirements engineering as a driver for innovations. 15th IEEE International Requirements Engineering Conference (RE 2007), 15–20. IEEE, 2007.
[18]. Youngjoong Ko, Sooyong Park, Jungyun Seo, Soonhwang Choi: Using classification techniques for informal requirements in the requirements analysis-supporting system. Information and Software Technology, vol.49:1128–1140, 2007.
[19]. Zijad Kurtanovic, Walid Maalej: Automatically classifying functional and non-functional requirements using supervised machine learning. Requirements Engineering Conference (RE), 2017 IEEE 25th International, 490–495. IEEE, 2017.
[20]. Timo O. A. Lehtinen, Mika Mäntylä, Jari Vanhanen, Juha Itkonen, Casper Lassenius: Perceived causes of software project failures–An analysis of their relationships. Information and Software Technology, vol.56:623–643, 2014.
[21]. Tomas Mikolov: Statistical language models based on neural networks. Presentation at Google, Mountain View, 2nd April, vol.80, 2012.
[22]. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, Jeffrey Dean: Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 3111–3119, 2013.
[23]. Nan Niu, Sjaak Brinkkemper, Xavier Franch, Jari Partanen, Juha Savolainen: Requirements engineering and continuous deployment. IEEE software, vol.35(2):86–90, 2018.
[24]. Carla Pacheco, Ivan Garcia, Miryam Reyes: Requirements elicitation techniques: a systematic literature review based on the maturity of the techniques. IET Software, vol.12:365–378, 2018.

[25]. J. Manuel Pérez-Verdejo, Angel Juan Sanchez Garcia, Jorge Octavio Ocharán-Hernández, Efrén Mezura-Montes, Karen Cortés-Verdin: Requirements and GitHub Issues: An Automated Approach for Quality Requirements Classification. Programming and Computer Software, 47:704–721, 2021.
[26]. Abderahman Rashwan, Olga Ormandjieva, Rene Witte: Ontology-based classification of non-functional requirements in software specifications: a new corpus and SVM-based classifier. Computer Software and Applications Conference (COMPSAC), 2013 IEEE 37th Annual, 381–386. IEEE, 2013.
[27]. Patrick Rempel, Patrick Máder: Preventing defects: The impact of requirements traceability completeness on software quality. IEEE Transactions on Software Engineering, vol.43:777–797, 2017.
[28]. Kevin Ryan: The role of natural language in requirements engineering. Proceedings of the IEEE International Symposium on Requirements Engineering, 240– 242. IEEE, 1993.
[29]. J. Sayyad Shirabad, Tim J. Menzies: The PROMISE Repository of Software Engineering Databases, 2005. http://promise.site.uottawa.ca/SERepository.
[30]. Belal Shanyour, Abdallah Qusef: Global Software Development and its Impact on Software Quality. 2018 Fifth International Symposium on Innovation in Information and Communication Technology (ISIICT), 1–6. IEEE, 2018.
[31]. Ian Sommerville: Software engineering, 9th Edition. Pearson, 2011.
[32]. Jason Van Hulse, Taghi M. Khoshgoftaar, Amri Napolitano: Experimental perspectives on learning from imbalanced data. Proceedings of the 24th international conference on Machine learning, 935–942, 2007.
[33]. Jonas Winkler, Andreas Vogelsang: Automatic classification of requirements based on convolutional neural networks. 2016 IEEE 24th International Requirements Engineering Conference Workshops (REW), 39–45. IEEE, 2016.

Информация об авторах / Information about authors

Сандра Эстефания МАРТИНЕС ГАРСИЯ имеет магистерскую степень Технологического университета Миштека (2021 год) в области прикладных вычислительных технологий и Технологического института Оахака в области вычислительной техники (2019 год). В настоящее время она является специалистом по управлению данными и техническим руководителем мексиканского отделения компании Bosch. Ее области интересов включают искусственный интеллект, разработку программного обеспечения, управление разработкой требований, автоматизацию процессов, бизнес-аналитику, аналитику данных и науку о данных.

Sandra Estefanía MARTINEZ GARCIA obtained a master's degree in Applied Computing Technologies from the Universidad Tecnológica de la Mixteca in 2021 and Computer engineering from Instituto Tecnológico Del Valle de Oaxaca in 2019. She is currently a data scientist and technical lead at Bosch Mexico. Her areas of interest include Artificial Intelligence, Software Engineering, Requirements Engineering, Process Automation, Business Analytics, Data Analytics and Data Science.

Карлос Альберто ФЕРНАНДЕС-И-ФЕРНАНДЕС имеет степень PhD университета Шеффилда по программированию, эксперт в области программирования. В настоящее время возглавляет Институт вычислений в Технологическом университете в мексиканском регионе Миштека, координирует магистерские программы по прикладным аспектам вычислительных технологий. Сфера научных интересов: визуальное моделирование, гибкие технологии разработки и формальные спецификации программного обеспечения.

Carlos Alberto FERNÁNDEZ-Y-FERNÁNDEZ – Software Engineering expert with a Ph.D from the University of Sheffield. He currently leads the Institute of Computing at Universidad Tecnológica de la Mixteca and coordinates the Master's program in Applied Computing Technologies. His research interests include visual modeling, agile methods, and formal software specification.

Эрик Г. РАМОС ПЕРЕС – Профессор-исследователь Технологического университета Миштека. Имеет степени магистра Технологическом университете Ла-Микстека в области прикладных вычислительных технологий (2016 год) и по вычислительной технике (2001 год).

Является координатором Виртуального университета. Его области научных интересов включают машинное обучение и автономную навигацию беспилотных аппаратов.

Erik G. RAMOS PÉREZ – Professor-Researcher at the Technological University of La Mixteca. He obtained a master's degree in Applied Computing Technologies and Computer Engineering from the Technological University of La Mixteca in 2016 and 2001 respectively. He is coordinator of the Virtual University. His areas of research interest include Machine Learning and Autonomous Navigation with drones.

DOI: 10.15514/ISPRAS-2024-36(1)-9

Quantitative and Qualitative Approaches of User Engagement on Facebook* Fan Page

[1] *P. Velazquez-Solis*, ORCID: 0000-0003-2095-9414 <*paola.velazquez@uabc.edu.mx*>
[2] *J. E. Ibarra-Esquer*, ORCID: 0000-0003-2636-5051 <*jorge.ibarra@uabc.edu.mx*>
[2] *M. Astorga-Vargas*, ORCID: 0000-0002-9998-3777 <*angelicaastorga@uabc.edu.mx*>
[1] *B. L. Flores-Rios*, ORCID: 0000-0002-2502-6595 <*brenda.flores@uabc.edu.mx*>
[1] *M. Carrillo-Beltrán*, ORCID: 0000-0002-0035-2298 <*monica@uabc.edu.mx*>
[3] *I. A. García Pacheco*, ORCID: 0000-0002-7594-6410 <*ivan@mixteco.utm.mx*>

[1]*Universidad Autónoma de Baja California (UABC), Instituto de Ingeniería, Baja California, México.*
[2]*Universidad Autónoma de Baja California (UABC), Facultad de Ingeniería, Baja California, México.*
[3]*Universidad Tecnológica de la Mixteca, División de Estudios de Posgrado, Oaxaca, México.*

Abstract. User Engagement is a metric that represents a part of the user experience characterized by attributes of reactions, visibility and user interactivity with others. Quantitative and qualitative analysis were used to establish a new method for calculating User Engagement in Facebook* fan pages focused in dissemination of scientific content, news, and events. We focused on social media processes based on Spearman correlation coefficients and categorization of publications by format type and source of content. Variations in Engagement for individual posts were explained by a multiple linear regression model defined using the number of clicks and the reach of posts with an accuracy of up to 91% (R^2). The User Engagement increases preferably when it is presented in photo format of an original content creation.

Keywords: regression model; Facebook*; Spearman correlation coefficients; user engagement.

For citation: Velazquez-Solis P., Ibarra-Esquer J. E., Astorga-Vargas M., Flores-Rios B. L., Carrillo-Beltrán M., García-Pacheco I. A. Quantitative and qualitative approaches of User Engagement on Facebook* fan page. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 143-156. DOI: 10.15514/ISPRAS-2024-36(1)-9.

Full text: Velazquez-Solis P., Ibarra-Esquer J. E., Astorga-Vargas M., Flores-Rios B. L., Carrillo-Beltrán M., García Pacheco I. A. A Regression Model for Analysis of User Engagement on Facebook* Fan Page for Scientific Dissemination Content. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 832–841. DOI: 10.1134/S036176882308025X.

Acknowledgments. This research was supported by Universidad Autónoma de Baja California through the PhD grant by Consejo Nacional de Ciencia y Tecnología 668484 for the first author.

Количественные и качественные подходы к изучению уровня вовлеченности пользователей Facebook*

[1] *П. Веласкес-Солис*, ORCID: 0000-0003-2095-9414 <*paola.velazquez@uabc.edu.mx*>
[2] *Х. Е. Ибарра-Эскер*, ORCID: 0000-0003-2636-5051 <*jorge.ibarra@uabc.edu.mx*>
[2] *М. Асторга-Варгас*, ORCID: 0000-0002-9998-3777 <*angelicaastorga@uabc.edu.mx*>
[1] *Б. Л. Флорес-Риос*, ORCID: 0000-0002-2502-6595 <*brenda.flores@uabc.edu.mx*>
[1] *М. Карийо-Бельтран*, ORCID: 0000-0002-0035-2298 <*monica@uabc.edu.mx*>
[3] *И. А. Гарсия Пачеко*, ORCID: 0000-0002-7594-6410 <*ivan@mixteco.utm.mx*>

[1]*Автономный университет Нижней Калифорнии, Инженерный институт, Нижняя Калифорния, Мексика.*
[2] *Автономный университет Нижней Калифорнии, Факультет инженерии, Нижняя Калифорния, Мексика.*
[3]*Аспирантура технологического университета Миштека, Оахака, Мексика.*

Аннотация. Вовлеченность пользователей – это показатель, демонстрирующий ту часть пользовательского опыта, которая характеризуется атрибутами реакции, видимости и взаимодействия пользователя с партнерами. Для создания нового метода расчета вовлеченности пользователей на фан-страницах Facebook*, ориентированных на распространение научного контента, новостей и событий использовался количественный и качественный анализ. Авторы были сосредоточены на процессах в социальных сетях, основанных на коэффициентах корреляции Спирмена и категоризации публикаций по типу формата и по источнику контента. Различия в привлекательности для отдельных постов были объяснены с помощью модели множественной линейной регрессии и подсчета количества кликов и уровня доступности постов с точностью до 91% (R2). Размещение в сети оригинального контента и фотографий существенно сказывается на росте пользовательской вовлеченности.

Ключевые слова: регрессионная модель; мессенджер Facebook*; коэффициент ранговой корреляции Спирмена; вовлеченность пользователя.

Для цитирования: Веласкес-Солис П., Ибарра-Эскер Х. Е., Асторга-Варгас М., Флорес-Риос Б. Л., Карийо-Бельтран М., Гарсия-Пачеко И. А. Количественные и качественные подходы к изучению уровня вовлеченности пользователей Facebook*. Труды ИСП РАН, том. 36, вып. 1, 2024. стр. 143-156 (на английском языке). DOI: 10.15514/ISPRAS-2024-36(1)-9.

Полный текст: Веласкес-Солис П., Ибарра-Эскер Х. Е., Асторга-Варгас М., Флорес-Риос Б. Л., Карийо-Бельтран М., Гарсия-Пачеко И. А. Регрессионная модель для анализа интереса пользователей к научному контенту, размещенному на странице Facebook*. Programming and Computer Software, 2023, т. 49, № 8, стр. 832–841 (на английском языке). DOI: 10.1134/S036176882308025X.

Благодарности. Настоящее исследование было поддержано Университетом Нижней Калифорнии грантом Национального совета по науке и технике № 668484 первому автору на подготовку диссертации PhD.

1. Introduction

Social networks have been used as a tool that stimulates interaction, shaping new forms in which people communicate, make decisions, socialize, collaborate, and learn [1-2]. In social networks it is features provide unique and interesting conditions for investigating the interaction of multiple individuals and the incorporation of organizations in user's self-expression [3-4]. There are research papers on social networks, which have used Cloud Computing, broadband networks, Big Data Analysis [5], text analysis [6] and smart devices like internet of people does [7].

Interactivity of social networks, particularly Facebook*, can be used to study the impacts of users' increase of activity towards a specific brand [3, 8], products [9], services [10] and benefiting from

new opportunities [11]. On this way, Facebook* measures interactions by means of Engagement and other metrics [9].

Engagement is obtained from the behavior shown by users through their activity on social networks [2-3, 12]. This is a quantitative metric of the bidirectional interaction between organizations and users of social networks [9]. Through the data collected from Facebook* and the metrics used, a post (publication) can be evaluated [9, 13], but the User Engagement and Engagement Rate formulas, provided by Facebook*, calculate Engagement, but do not reveal the differences between participation levels [14].

The research objectives are to analyze how metrics on Table 1 are correlated with the calculation of User Engagement. Then, using those metrics as independent variables to create a regression model to interpret User Engagement in Facebook* fan pages with an interest on dissemination of scientific information and activities.

Table 1. Facebook Quantitative Metrics*

Metrics	**Description**
Reach	The number of people who viewed the post at least once. Does not include people who saw your story when another user shared it.
Impressions	The number of people who viewed the content in News Feed, no need to interact with the post.
Clicks	The number of clicks on posts that led to destinations or experiences on or off Facebook*.
Reactions	The total number of reactions on the post. This includes "like", "love", "haha", "wow", "sad", and "angry".
Comments	The total number of user comments on the post.
Shared	The total number of times users shared the post. This includes in your profile or as a private message.

In this study, the emphasis is on the fact of incorporating social networking processes for the dissemination of scientific content, as a mutually beneficial collaboration between universities and research centers with society, that makes research information and knowledge useful outside the academic community, helping them to establish specific interest groups. To this end, the following hypotheses were established:

- H1 User Engagement changes as a function of content type
- H2 User Engagement is larger for original content.

*2. User engagement calculation on Facebook**

When logging into to Facebook*, is possible to manage fan pages or groups that enable other types of communication and strategies aimed mainly at organizations, institutions, and social or commercial ventures [2, 15].

Some investigations about Facebook*, on fan page or groups, found posting types and topics have a significant effect on Engagement. In [16] authors suggest Poisson regression models to be deployed to analyze the collected data and assess the effect of brand posts characteristics related to online Engagement. The data Facebook* regression analysis made by [13] indicated that Engagement is positively affected by posting visual content (photos), negatively affected by evening posting, while Post frequency displays no statistically significant effect on Engagement.

The authors in [17] and [18] performed linear and multiple regression models to understand the relationship between cultural values and users' engagement with Facebook* advertising and found that attitude to Facebook* advertising, subjective norms and perceived herd behavior positively

determines the Engagement. In the work of [19], qualitative research was conducted on Facebook* users and their activity on the social network. The authors in [20] presented a study with mixed qualitative and quantitative approach, where they integrate the opinion of experts and subsequently proceed to analyze the data statistically, implementing the calculation and analysis of engagement based on metrics from Facebook*. The User Engagement analysis [8, 13, 17, 21-22] and its prediction continue to be an area of opportunity to identify the acceptance of content on Social networks [23-24].

Having an index from the variables implemented by different formulas and models supports the identification of an integrated measure of Engagement, underscoring the importance of having the parameters normalized [12]. There exist diverse formulas and models for calculating User Engagement based on data obtained from Facebook* fan pages or groups (Table 2).

In Facebook*, User Engagement is calculated as the sum of all interactions in the post (equation 1).

$$\text{User Engagement} = N_R + N_c + N_S \quad (1)$$

where:

- N_R, Number of reactions on the publication.
- N_C, Number of comments on the publication.
- N_S, Number of times the publication has been shared.

Considering the proposals of the related work, a User Engagement regression model is proposed to measure the effect of each metrics associated to a Facebook* fan page focused on topics of scientific dissemination.

*Table 2. Formulas to calculate User Engagement on Facebook**

N_L = number of likes N_R = Number of reactions

N_C = Number of comments N_S = Number of times the publication has been shared N_F = number followers

Authors	Proposed Formula
Bonsón & Ratkai, 2013 [25]	$\frac{N_L + N_S + N_C}{\frac{N_{Message\ \ fanpage/group}}{N_F}}$
Niciporuc, 2014 [26]	$\frac{N_L + N_S + N_C}{N_F}$
Oviedo-García et al.,2014 [9]	$\frac{\frac{N_L + N_S + N_C}{N_F}}{\frac{X_{j\ (Interactivity)}}{\mu_{Scope\ fanpage}}}$
Herrera-Torres (2017) [27]	$\frac{N_R + N_S + N_C}{N_F}$
Ge & Grezel, 2017 [28] Eriksson et al., 2019 [12]	$N_R + 5N_S + 10N_C$
Vadivu & Neelamalar (2015) Peruta & Shields, 2017 [29]	$\frac{N_R + N_S + N_C}{N_{likes\ fanpage}} \text{x } 100$
Phuntusil & Limpiyakorn, 2017 [30]	$\frac{N_R + 5N_c + 10N_s}{N_{Likes\ fanpage^{0.8}}} \text{x } 10^4$
Ballesteros-Herencia, 2018 [31]	$\frac{N_R + N_c + N_s}{Scope\ post} \text{x } 100$
Jayasingh Sudarsan, 2019 [17]	$\frac{N_R + 5N_c + 10N_s}{N_{Likes\ fanpage^{0.8}}} \text{ x } 100$

Martínez-Sala & Segarra-Saavedra, 2020 [22]	$\frac{N_R + N_c + N_s}{N_{Post} \times N_{Fans}} \text{x } 100$

In this way, the following questions addressed this research are:

(RQ1) What is the best regression model to explain User Engagement on Facebook*?

(RQ2) Which content type generates a higher percentage of User Engagement?

(RQ3) What is the correlation between the different variables related to Facebook*?

3. Research methodology

The methodology used is an adaptation of the phases proposed by [24] and extended with the tools and roles of [2].

3.1. Data Preparation

During data preparation phase a Facebook* fan page (FP) for thematic elements around disseminating scientific content was selected as a study case (https://www.Facebook*.com/mujeresinvestigacion). This fan page is part of a scientific dissemination program of Instituto de Ingeniería of the Universidad Autónoma de Baja California, Mexico. Data were obtained in the form of a CVS file from Facebook* Insights, containing interaction details for 166 posts from the year 2022. Five different categories of posts were identified in Table 3 (link, event, photo, podcast, video).

Additionally in Table 4, a categorization was created, corresponding to the origin of the material presented in the posts:

1) external scientific fan page,
2) external universities FP,
3) our university FP,
4) study case, and
5) posts related to the activities of an external dissemination event in which some members of the extension program participated.

To carry out the data preprocessing process, the Anaconda and RStudio tools were used.

3.2. Engagement Interpretation

To examine the relationships between Facebook* metrics with the User Engagement calculation, an analysis was carried out using the importance of the extracted metrics and the results of Spearman correlation coefficients [18]. The preprocessed Facebook* data allowed us to obtain proposals for a regression model with a better fit for Engagement.

4. Data analysis and findings

4.1 Analysis of the variables

To answer to H1 and H2, Tables 3 and 4 present a detailed characterization of posts. The categorical variables allowed an analysis of the means and clearly present the frequencies of each identified element. For User Engagement, equation 1 was used, where the highest value was obtained from photo-type posts (120, $\mu = 25$), followed by video-type posts (13, $\mu = 17$) in which at least three of these posts were above average (H1). A first approach to the behavior of the data can be seen in Fig. 1 with User Engagement by type of post.

The highest concentration of elements on the fan page corresponds to photos (72% of the content), and coincides with the element that has the highest User Engagement. Regarding the original content of the fan page (Table 4), they have been separated into fan page study case and Dissemination event. For these cases, User Engagement was higher for posts related to the dissemination event (μ

= 166) than for regular posts of the fan page (μ = 16.5). Responding to H2, it is observed in the data that for both categories of original content, the engagement was higher than in the others. But in content generated for a specific objective (Category 5 in Table 4), the Engagement mean was even higher.

Table 3. Characterization of the posts by type of content

Categories	Frequency	Percentage	μ Engagement Formula
Link	16	9.63	12
Event	14	8.43	16
Photo	120	72.28	25
Podcast	3	1.80	18
Video	13	7.83	17
Total	166	100	

Table 4. Characterization of the post by source

Categories	Percentage	μ Engagement Formula
1. External FP	15	13
2. External Universities FP	9	10.5
3. Our University FP	27	12.5
4 Study case*	16	16.5
5. Dissemination event*	33	166

* Original Content, FP = Fan page

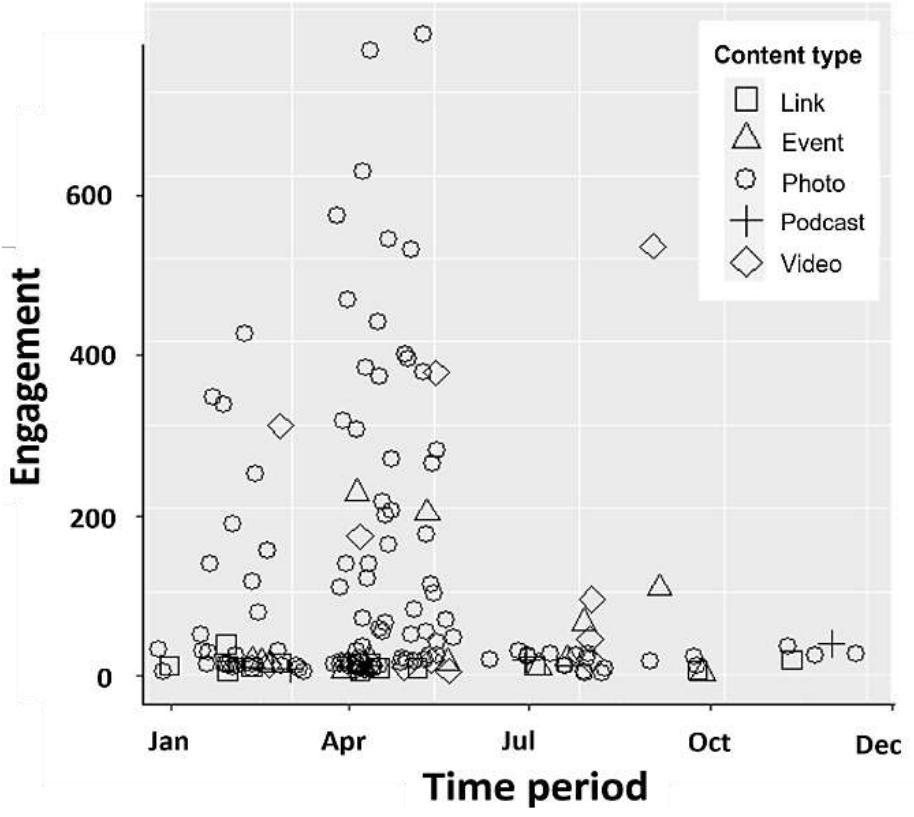


Fig. 1. Graph with the content type shared by the fan page over a span of one year

Fig. 2 allowed us to observe the Coefficient correlations between the different metrics extracted from the data set. This helped to verify the relationship between Reach and Impressions as causal variables, so it was decided to rule out the use of the impressions variable when generating the regression model.

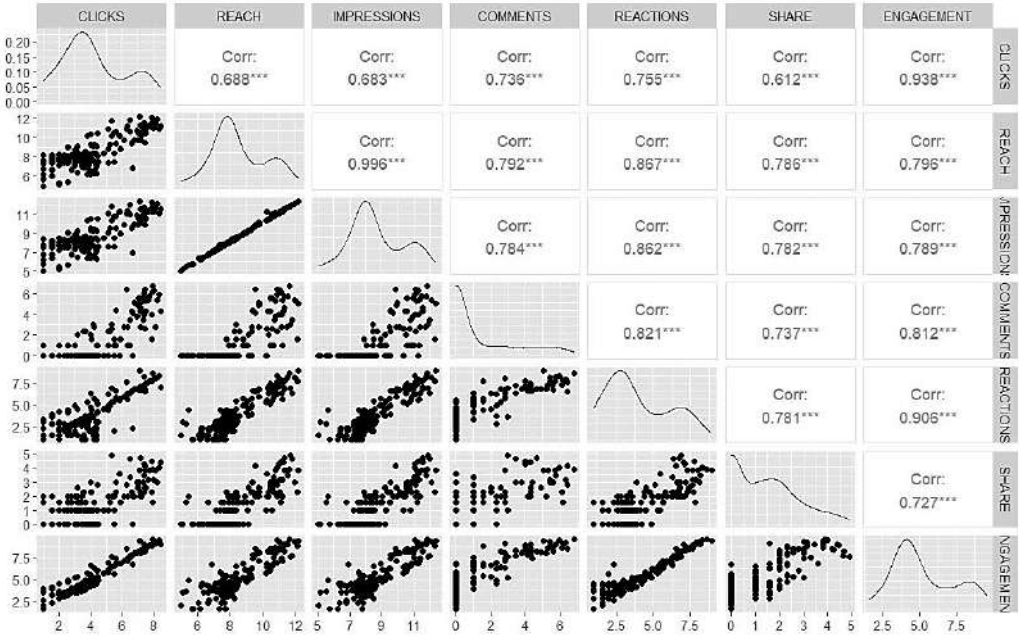


Fig. 2. Spearman correlation coefficients calculation

4.2 Analysis of regression models

Based on the data set obtained for the behavior of the FP, and the categorical variables, several multiple regression models were explored (Table 5), with the purpose of obtaining an adjusted model that integrates most of the variables available for the calculation of User Engagement, omitting the reactions, comments and number of times it is shared. This in order to avoid multicollinearity [8].

First, we validated the variables using the values of the Spearman correlation coefficients (Fig. 2), looking for variables that maintained a positive correlation, considered between medium and strong in their combination. It can be seen that there is a linear relationship between Clicks and Reactions (rs = 0.755). Hence it is inferred that with each click there is a probability of obtaining a reaction, unlike the other variables implemented to calculate Engagement on Facebook*, like Comments (rs = 0.736) and Shares (rs = 0.612). One of the highest values is found in the Engagement and Reactions variables (rs = 0.906), with the understanding that there is a relation between the variables. So, it was considered important to explore the relationship between clicks and the formula that Facebook* uses to calculate Engagement (rs = 0.938).

We proceeded to give an interpretation of the determination coefficients obtained as a result of the three regression models (Table 5). First, we validated the variables using the values of the Spearman correlation coefficients (Fig. 2), looking for variables that maintained a positive correlation, considered between medium and strong in their combination. It can be seen that there is a linear relationship between Clicks and Reactions (rs = 0.755).

Hence it is inferred that with each click there is a probability of obtaining a reaction, unlike the other variables implemented to calculate Engagement on Facebook*, like Comments (rs = 0.736) and Shares (rs = 0.612). One of the highest values is found in the Engagement and Reactions variables

(rs = 0.906), with the understanding that there is a relation between the variables. So, it was considered important to explore the relationship between clicks and the formula that Facebook* uses to calculate Engagement (rs = 0.938). We proceeded to give an interpretation of the determination coefficients obtained as a result of the three regression models (Table 5).

Table 5. Regression models selected

Dependent variable	Independent variables	R^2	Error
User Engagement	Reach Type of content Source of content	0.679	61.52
User Engagement	Type of content Source of content	0.28	57.52
User Engagement	Clicks Reach	0.91	40.46

Throughout several iterations of adjustment, the best regression model with two independent variables was obtained. In the first one, two categorical variables (Type and source) were used together with Reach, obtaining an acceptable prediction of R2 of 0.679. The addition of categorical variables Source and Types did not show a significant difference in the models, so it was decided to implement them separately. The combination of the type and source categorical variables represents only an R2 at 0.28 of prediction. These experiments showed a relationship between the interaction on the contents between the number of Clicks and the Reach, visualized in Fig. 2.

Considering the result of the correlation of the variables available in the sample, it was detected that the sample had a linear relationship between Clicks and Reach with reactions. The selected model (equation 2) is more suitable because it has a lower margin of error and a 91% representativeness of the variables. Therefore, the selected regression model was model three in Table 5:

$$\text{User Engagement} = 1.7\ \text{Clicks} - 0.05\ \text{Reach} \quad (2)$$

The regression model defined (equation 2) obtained a coefficient of multiple determination (R^2) of 0.91 of influence of the independent variables. We worked with a p-value (2.2e-16) of less than 0.05. This means that the User Engagement regression model for a post that has not reached anyone or received clicks would be zero, obtained by moving the origin to the means of the predictors. The 1.7 coefficient of the clicks is associated with the opportunity of obtaining reactions during the interaction. While the 0.05 coefficient of the reach represents the decrement in Engagement for each click received by the post. It means that a person who is reached by the post, but who does not interact (clicks) causes a decrease in User Engagement. Meanwhile with the User Engagement calculation by the formula that implements reactions, comments and shares (Equation 1), the User Engagement had a value of 47. This is considered highly representative and allows predicting User Engagement in future posts based on clicks per post and people reached. To identify additional information, clustering was performed. From a test with the elbow method, four will be extended as the ideal number of clusters. Fig. 3 considers that the Engagement is made up of the interactions (Reactions, Comments, Shares) and the scope belongs to the total number of people who viewed on the publication.

Therefore, each of the dots represents an individual publication. The first group 1 (Red) is associated with the fact that they are publications that have less Interactivity and, as a consequence, there is less reach. In group 2 (Purple), there are behaviors close to group 1, but there is a publication that had a high Interaction rate in relation to the others, keeping the scope below one thousand, showing the main characteristic of this group. It had moderate interactions, but with a low reach. Group 3 (Green) is made up of those publications that had a reach of more than one thousand and less than

three thousand, regardless of the value of interactions they got. And finally, group 4 (Blue) has those posts that had a greater reach and are associated with higher Engagement.

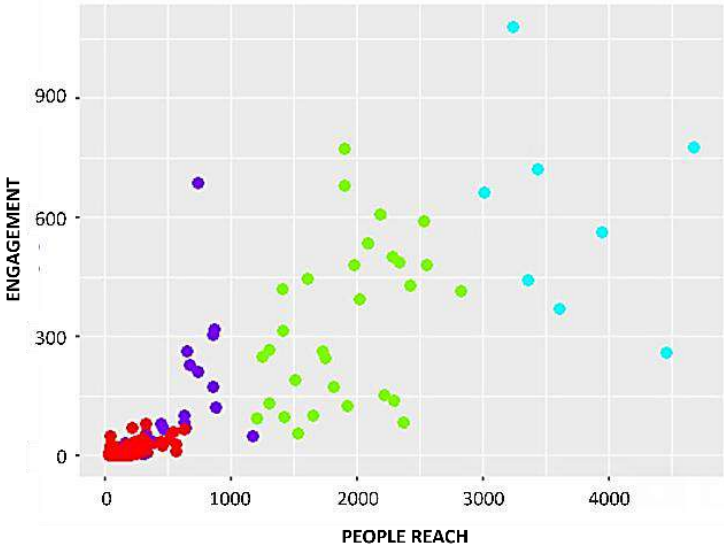


Fig. 3. Clustering by engagement and people reached

5. Discussion

In the qualitative analysis of unstructured data, relationships and groups are extracted that allowed us to take advantage not only of the descriptions obtained but also of the relationships. In this way, qualitative analysis supports the identification of the various types of relationships within data interactivity in the Facebook* social network. A study was carried out to create a regression model of the User Engagement percentage of a sample of posts about scientific dissemination on a Facebook* FP. According to RQ1, the multiple linear regression model with the highest precision was defined using only the number of Clicks and the Reach of the posts (equation 2). The calculation of User Engagement through a regression model has the advantage of adapting to the frequent changes that Facebook* introduces in the establishment of its variables (i.e., number of clicks, scope, impressions, number of comments, number of reactions, number of times a post is shared), which can influence the results as independent variables, unlike the Facebook* proposal based on count values. The regression models generated in this research provide an example of the dynamism that exists in the data generated within social networks, particularly Facebook*.

On the other hand, according to RQ2 using this same model, User Engagement increases preferably for posts with original content and with photos. From the perspective of a Facebook* user, the crucial factors to increase the levels of User Engagement are the interactivity of the users with the posts, in relation to the type of format they have. Finally, RQ3 allowed us to examine the relationships that exist between the metrics extracted from Facebook* involved in the regression model. Using Spearman correlation coefficients [18] provided a more qualitative interpretation of the determination coefficients obtained as a result, compared to proposals made with Pearson's correlation coefficients [29] and Poisson [16].

In the analysis and visualization of the variables of each publication, clustering was also applied as a data mining technique, demonstrating that self-produced content provides a higher Engagement.

6. Conclusions

The scientific community considers the analysis of social network data as a multidisciplinary research area that enables the creation, extension, and adaptation of methods [32] and data analysis models which empirically provide quantitative results using statistical methods [2]. For this reason, statistical analysis methods and qualitative analysis were used to establish a new method for calculating User Engagement on a Facebook* fan page for scientific dissemination content. The results of this research were the analysis of the posts, as well as a model using multiple linear regression with an accuracy of up to 91% (R2). According to the correlation values, it was shown that the number of clicks is the most effective metric compared to the usual interactivity metrics such as the number of comments, number of reactions, and number of times a post is shared.

For the roles involved in the management of social networks (Community manager or social media manager), the results of this research can be used as an idea of how to increase the level of User Engagement in posts by understanding the users interested in science topics. The output of the regression model can be used as a method to select the most fruitful or attractive post structures. Together with other Facebook* metrics, such as follower growth and new likes, they help to understand User Engagement and add precision by predicting the specific needs of each fan pages in the future.

The objective was to analyze the participation of the followers and the User Engagement in the content of the posts of a fan page with scientific dissemination topics. As future work, Disengagement (Negative Engagement) can be studied as a key consequence and its effect on the interactivity with the contents, as well as to determine whether the motivations are the same for the different categories of content types. In addition, the characteristics of the model obtained retain the scalability quality attribute and are adapted to other fan pages that are dedicated to dissemination or diffusion, validating its application and impact on decision-making.

References

[1]. F. Froment, A. J. García-González, and J. Cabero, “The relationship of Twitter with teacher credibility and motivation in university students,” Comunicar, vol. 30, no. 71, pp. 1–12, 2022, doi: 10.3916/C71-2022-10.

[2]. P. E. Velazquez-Solis, B. L. Flores-Rios, M. A. Astorga-Vargas, J. E. Ibarra-Esquer, F. F. González-Navarro, and R. A. Aguilar Vera, “Analysis of scientific dissemination posts on Facebook from a social media approach,” in CISTI’2022 - 17a Conferencia Ibérica de Sistemas y Tecnologías de Información, In press, Ed., Madrid, España, 2022.

[3]. C. R. Hollenbeck and A. M. Kaikati, “Consumers’ use of brands to reflect their actual and ideal selves on Facebook,” International Journal of Research in Marketing, vol. 29, no. 4, pp. 395–405, 2012, doi: 10.1016/j.ijresmar.2012.06.002.

[4]. J. A. Choi and K. Lim, “Identifying machine learning techniques for classification of target advertising,” ICT Express, vol. 6, no. 3. Korean Institute of Communications Information Sciences, pp. 175–180, Sep. 01, 2020. doi: 10.1016/j.icte.2020.04.012.

[5]. R. Massobrio, S. Nesmachnow, A. Tchernykh, A. Avetisyan, and G. Radchenko, “Towards a Cloud Computing Paradigm for Big Data Analysis in Smart Cities,” Programming and Computer Software, vol. 44, no. 3, pp. 181–189, May 2018, doi: 10.1134/S0361768818030052.

[6]. D. Y. Turdakov et al., “Texterra: A framework for text analysis,” Programming and Computer Software, vol. 40, no. 5, pp. 288–295, Sep. 2014, doi: 10.1134/S0361768814050090.

[7]. V. Chang, “A proposed social network analysis platform for big data analytics,” Technol Forecast Soc Change, vol. 130, no. November 2017, pp. 57–68, 2018, doi: 10.1016/j.techfore.2017.11.002.

[8]. M. Sharma, P. Sahai, and V. K. Singh, “Engaging social media influencers credibility on purchase behaviour through lens of brand engagement,” Int J Health Sci (Qassim), no. May, pp. 11288–11298, 2022, doi: 10.53730/ijhs.v6ns2.8030.

[9]. M. Á. Oviedo-García, M. Muñoz-Expósito, M. Castellanos-Verdugo, and M. Sancho-Mejías, “Metric proposal for customer engagement in Facebook,” Journal of Research in Interactive Marketing, vol. 8, no. 4, pp. 327–344, 2014, doi: 10.1108/JRIM-05-2014-0028.

[10]. L. K. Kaye, "Exploring the 'socialness' of social media," Computers in Human Behavior Reports, vol. 3, p. 100083, Jan. 2021, doi: 10.1016/j.chbr.2021.100083.
[11]. F. Poecze and C. Strauss, "Social capital on social media—concepts, measurement techniques and trends in operationalization," Information (Switzerland), vol. 11, no. 11, pp. 1–16, 2020, doi: 10.3390/info11110515.
[12]. N. Eriksson, A. Sjöberg, C.-J. Rosenbröijer, and A. Fagerstrøm, "Consumer brand post engagement on Facebook and Instagram-Consumer brand post engagement on Facebook and Instagram-A study of three interior design brands A study of three interior design brands," in International Conference on Electronic Business (ICEB), Dec. 2019, pp. 116–124.
[13]. M. M. Mariani, M. Mura, and M. Di Felice, "The determinants of Facebook social engagement for national tourism organizations' Facebook pages: A quantitative approach," Journal of Destination Marketing and Management, vol. 8, pp. 312–325, Jun. 2018, doi: 10.1016/j.jdmm.2017.06.003.
[14]. B. Mazza and A. Palermo, "Social media content for business and user engagement on Facebook," Journal for Communication Studies, vol. 11, no. 1, pp. 49–73, 2018.
[15]. J. Brito, W. Laaser, and E. Adrián Toloza, "El uso de redes sociales por parte de las universidades a nivel institucional. Un estudio comparativo.," RED: Revista de Educación a Distancia, no. 32. pp. 6–38, 2012.
[16]. A. Dash, "Influence of Content Type Over Online Engagement on Facebook Brand Pages of SMEs," SEDME (Small Enterprises Development, Management & Extension Journal): A worldwide window on MSME Studies, vol. 46, no. 4, pp. 264–272, Dec. 2019, doi: 10.1177/0970846419894744.
[17]. S. Jayasingh, "Consumer brand engagement in social networking sites and its effect on brand loyalty," Cogent Business and Management, vol. 6, no. 1, Jan. 2019, doi: 10.1080/23311975.2019.1698793.
[18]. K. Sharma and E. E. Lulandala, "Facebook Ad Engagement: A Cross-cultural Analysis," Global Business Review, 2021, doi: 10.1177/09721509211007115.
[19]. D. Franz, H. E. Marsh, J. I. Chen, and A. R. Teo, "Using facebook for qualitative research: A brief primer," J Med Internet Res, vol. 21, no. 8, pp. 1–12, 2019, doi: 10.2196/13544.
[20]. F. Egaña, C. Pezoa-Fuentes, and L. Roco, "Article the use of digital social networks and engagement in Chilean wine industry," Journal of Theoretical and Applied Electronic Commerce Research, vol. 16, no. 5, pp. 1248–1265, 2021, doi: 10.3390/jtaer16050070.
[21]. H. Shahbaznezhad, R. Dolan, and M. Rashidirad, "The Role of Social Media Content Format and Platform in Users' Engagement Behavior," Journal of Interactive Marketing, vol. 53, pp. 47–65, Feb. 2021, doi: 10.1016/j.intmar.2020.05.001.
[22]. A. M. Martínez-Sala and J. Sagarra-Saavedra, "Engagement y disengagement online, factores clave en las estrategias de comunicación turística 2.0.," in Tendencias de la Comunicación para el Turismo, 2021, pp. 149–183.
[23]. Y. H. Hu and K. Chen, "Predicting hotel review helpfulness: The impact of review visibility, and interaction between hotel stars and review ratings," Int J Inf Manage, vol. 36, no. 6, pp. 929–944, 2016, doi: 10.1016/j.ijinfomgt.2016.06.003.
[24]. K. R. Purba, D. Asirvatham, and R. K. Murugesan, "An analysis and prediction model of outsiders percentage as a new popularity metric on Instagram," ICT Express, vol. 6, no. 3, pp. 243–248, Sep. 2020, doi: 10.1016/j.icte.2020.07.001.
[25]. E. Bonsón and M. Ratkai, "A set of metrics to assess stakeholder engagement and social legitimacy on a corporate Facebook page," Online Information Review, vol. 37, no. 5, pp. 787–803, 2013, doi: 10.1108/OIR-03-2012-0054.
[26]. T. Niciporuc, "Comparative analysis of the engagement rate on Facebook and Google Plus social networks," in Proceedings of international academic conferences, 2014.
[27]. L. Herrera-Torres, F. Pérez-Tur, J. García-Fernández, and J. Fernández-Gavira, "El uso de las redes sociales y el engagement de los clubes de la Liga Endesa ACB," Cuadernos de psicología del deporte, vol. 17, no. 3, p. `175-182, 2017.
[28]. J. Ge and U. Gretzel, "The Role of Humour in Driving Customer Engagement," in Information and Communication Technologies in Tourism 2017, Springer, 2017, pp. 461–474. doi: 10.1007/978-3-319-51168-9.
[29]. A. Peruta and A. B. Shields, "Social media in higher education: understanding how colleges and universities use Facebook," Journal of Marketing for Higher Education, vol. 27, no. 1, pp. 131–143, 2017, doi: 10.1080/08841241.2016.1212451.

[30]. N. Phuntusil and Y. Limpiyakorn, "Predicting engaging content for increasing organic reach on facebook," in Lecture Notes in Electrical Engineering, Springer Verlag, 2017, pp. 637–644. doi: 10.1007/978-981-10-4154-9_73.

[31]. C. A. Ballesteros Herencia, "El índice de engagement en redes sociales, una medición emergente en la Comunicación académica y organizacional," RAZÓN Y PALABRA Primera Revista Electrónica en Iberoamérica Especializada en Comunicación, vol. 22, no. 3_102, pp. 96–124, 2018.

[32]. A. O. Savelev et al., "The high-level overview of social media content search engine," in IOP Conference Series: Materials Science and Engineering, IOP Publishing Ltd, Jan. 2021. doi: 10.1088/1757-899X/1019/1/012097.

Информация об авторах / Information about authors

Паола Эльвира ВЕЛАСКЕС-СОЛИС – аспирантка Инженерного института Автономного университета Нижней Калифорнии. Ее научные интересы включают исследования социальных сетей, интеллектуальный анализ данных, анализ данных, она также консультирует частные и государственные компании, занимается преподаванием, проводя занятия в местных университетах.

Paola Elvira VELAZQUEZ-SOLIS is a student of Ph.D. at the Universidad Autónoma de Baja California in the Engineering Institute. Her research interests include social media, data mining, data Analysis and she is also consulting in private and public sector companies and dedicated to teaching, giving classes in local universities.

Хорхе Едуардо ИБАРРА-ЭСКЕР – профессор факультета Инженерии Автономного университета Нижней Калифорнии. Сфера научных интересов: интернет вещей и интеллектуальный анализ данных. Он также принимал участие в исследовательских проектах, связанных с улучшением процессов обучения студентов в области вычислительной техники.

Jorge Eduardo IBARRA-ESQUER is a professor at the Universidad Autónoma de Baja California in the Faculty of Engineering. His research interests include the Internet of Things and Data Mining and has taken part in research projects related to improving the learning processes for students in the Computer Engineering major.

Мария Анжелика АСТОРГА-ВАРГАС – профессор по совершенствованию процессов программного обеспечения факультета Инженерии Автономного университета Нижней Калифорнии. Ее научные интересы – разработка программного обеспечения, в частности предназначенного для улучшения процессов программного обеспечения в небольших компаниях и повышения эффективности команд разработчиков программного обеспечения.

María Angélica ASTORGA-VARGAS is a professor of software process improvement at the Universidad Autónoma de Baja California in the Faculty of Engineering. Her research interests are Software Engineering with a focus in Software Process Improvement in small companies, and the effectiveness in software development teams.

Бренда Летиция ФЛОРЕС-РИОС ведет исследования в области управления знаниями и совершенствования процессов создания программного обеспечения в Инженерном институте Автономного университета Нижней Калифорнии. Сфера ее научных интересов включает в себя совершенствование процессов разработки программного обеспечения и его качества, ведущихся на небольших предприятиях, исследования влияния инженерии знаний на программную инженерию.

Brenda Leticia FLORES-RIOS is a researcher of knowledge management and software process improvement at the Universidad Autónoma de Baja California in the Engineering Institute. Her

research interests are Software Process Improvement and quality in small enterprises and the impact of Knowledge Engineering on Software Engineering.

Моника КАРИЙО-БЕЛЬТРАН ведет исследования пищевых биотехнологий в Инженерном институте Автономного университета Нижней Калифорнии. Сфера научных интересов: инновационная экосистема, в которой новые технологии применяются для разработки, сенсорной оценки и маркетинга более здоровых и безопасных продуктов питания и напитков с реальными преимуществами и ощутимыми достоинствами, с фокусированием внимания на питании людей, устойчивости и здоровье планеты.

Monica CARRILLO-BELTRÁN is a researcher of Food Biotechnology at the Universidad Autónoma de Baja California in the Engineering Institute. Her research interests include innovation ecosystem where emerging technologies are applied for the development, sensory evaluation and marketing of healthier and safer foods and beverages with real benefits and recognizable added values, focusing on human nutrition, sustainability and planetary health.

Иван Антонио ГАРСИЯ ПАЧЕКО – профессор Технологического университета Миштека. Сфера научных интересов: управление проектами программного обеспечения и разработка проектов по улучшению процессов создания программного обеспечения в мексиканской индустрии программного обеспечения.

Ivan Antonio GARCÍA PACHECO – profesor at the Universidad Tecnológica de la Mixteca. His research interests include software project management and software process improvement projects in the Mexican software industry.

*Запрещен в РФ, принадлежит Meta

DOI: 10.15514/ISPRAS-2024-36(1)-10

From Interaction Data to Personalized Learning: Mining User-Object Interactions in Intelligent Environments

J.G. Hernández-Calderón, ORCID: 0000-0002-8014-8819 <*guillermohernandez02@uv.mx*>
E.I. Benítez-Guerrero, ORCID: 0000-0001-5844-4198 <*edbenitez@uv.mx*>
J.R. Rojano-Cáceres, ORCID: 0000-0002-3878-4571 <*rrojano@uv.mx*>
C. Mezura-Godoy, ORCID: 0000-0002-5386-107X <*cmezura@uv.mx*>

Facultad de Estadística e Informática, Universidad Veracruzana
Av. Xalapa S/N, Col. Obrero Campesina, C.P. 91020,
Xalapa, Veracruz México.

Abstract. The aim of this work is to contribute to the personalization of intelligent learning environments by analyzing user-object interaction data to identify On-Task and Off-Task behaviors. This is accomplished by monitoring and analyzing users' interactions while performing academic activities with a tangible-intangible hybrid system in a university intelligent environment configuration. With the proposal of a framework and the Orange Data Mining tool and the Neural Network, Random Forest, Naive Bayes, and Tree classification models, training and testing was carried out with the user-object interaction records of the 13 students (11 for training and two for testing) to identify representative sequences of behavior from user-object interaction records. The two models that had the best results, despite the small number of data, were the Neural Network and Naive Bayes. Although a more significant amount of data is necessary to perform a classification adequately, the process allowed exemplifying this process so that it can later be fully incorporated into an intelligent educational system to contribute to build personalized environments.

Keywords: intelligent learning environments; user-behavior identification; user-object interaction data; data mining.

For citation: Hernández-Calderón J.G., Benítez-Guerrero E., Rojano-Cáceres J.R., Mezura-Godoy C. From Interaction Data to Personalized Learning: Mining User-Object Interactions in Intelligent Environments. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 157-174. DOI: 10.15514/ISPRAS-2024-36(1)-10.

Full text: Hernández-Calderón J.G., Benítez-Guerrero E., Rojano R., Mezura-Godoy C. From Interaction Data to Personalized Learning: Mining User-Object Interaction Data for Student Modeling in Intelligent Learning Environments. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 657–670. DOI: 10.1134/S036176882308008X.

Acknowledgements. The first author gratefully acknowledges CONAHCYT for scholarship No. 421557 for graduate studies.

От простого взаимодействия к персонализированному обучению: исследование взаимодействий пользователя с объектом в интеллектуальном окружении

Х.Г. Эрнандес-Кальдерон, ORCID: 0000-0002-8014-8819 <*guillermohernandez02@uv.mx*>
Э.И. Бенитес-Герреро, ORCID: 0000-0001-5844-4198 <*edbenitez@uv.mx*>
Х.Р. Рохано-Касерес, ORCID: 0000-0002-3878-4571 <*rrojano@uv.mx*>
К. Месура-Годой, ORCID: 0000-0002-5386-107X <*cmezura@uv.mx*>

Факультет статистики и информатики Университета Веракруса,
Мексика, 91020, Веракрус, Халапа, проспект Халапа, квартал Обреро-Кампесина.

Abstract. Цель этой работы состоит в развитии степени персонализации интеллектуальных учебных сред путем анализа данных взаимодействия пользователя с объектом, чтобы определить его поведение при решении задачи и в перерывах в работе (On-Task/Off-Task). Это достигается путем мониторинга и анализа взаимодействия пользователей при выполнении учебных заданий с гибридной системой, адаптированной для интеллектуальной университетской среды. Предложенная конфигурация системы, дополненная инструментарием добычи данных Orange и классификаторами на основе моделей нейронной сети, случайного леса, наивного классификатора Байеса и дерева классификации было проведено обучение взаимодействиям с объектами и тестирование 13 студентов (11 для обучения и два для тестирования), что позволило выявить представительные последовательности действий. Несмотря на небольшое количество данных, удалось понять, что наилучшие результаты показали две модели – нейронная сеть и наивный классификатор Байеса. Хотя для адекватного выполнения классификации необходим более значительный объем данных, проведенный опыт позволил лучше понять процесс. Впоследствии его можно будет полностью включить в интеллектуальную образовательную систему, что позволит внести вклад в создание персонализированных сред.

Ключевые слова: интеллектуальное окружение обучения; идентификация поведения пользователя; данные по взаимодействию пользователя с объектом; извлечение данных.

Для цитирования: Эрнандес-Кальдерон Х.Г., Бенитес-Герреро Э.И., Рохано-Касерес Х.Р., Месура-Годой К. От простого взаимодействия к персонализированному обучению: исследование взаимодействий пользователя с объектом в интеллектуальном окружении. Труды ИСП РАН, том. 36, вып. 1, 2024. стр. 157-174 (на английском языке). DOI: 10.15514/ISPRAS-2024-6(1)-10.

Полный текст: Эрнандес-Кальдерон Х.Г., Бенитес-Герреро Э.И., Рохано Р., Месура-Годой К. От простого взаимодействия к персонализированному обучению: извлечение данных о взаимодействии пользователя с объектом для моделирования поведения студентов в интеллектуальном окружении. Programming and Computer Software, 2023, т. 49, № 8, стр. 657–670 (на английском языке). DOI: 10.1134/S036176882308008X.

Благоларности. Первый автор благодарит Национальный совет Мексики по науке и технике (CONAHCYT) за стипендию № 421557 для обучения в аспирантуре.

1. Introduction

Human behavior refers to how an individual responds to various stimuli in their environment, physically and mentally. Computationally, it is possible to determine user behavior from observation and comparative analysis, and it is conceivable to classify their behavior by monitoring the development of a person's activities. In an intelligent environment, behavior patterns can be obtained from analyzing user interactions data [1]. For a computer system, modeling the behavior of users and understanding signs of the patterns it identifies involves various tasks, the first of which is to monitor and collect data related to user interactions, and then analyze the detected behavior signals considering the context.

Generally, behaviors are detected initially by identifying movements or events that happen within the environment. These are low-level and do require temporal or contextual knowledge to be placed

[2]. In intelligent environments, this information is collected by sensors when any movement or interaction is detected; then, the set of registered events and their space-time relationships give rise to activities.

There exist different challenges to building behavior-aware intelligent environments. An example of these challenges is the development of robust approaches for recognizing activities [3]. Another situation to face is that when detecting human activities, they are complex to model and dynamic because they constantly change and evolve with the user [2].

One approach considers that a critical factor for interpreting human behavior relates to the moment the actions are performed and their duration [2]. Moreover, concurrent activities (executed at the same time) [4] and sequential activities can be recognized [5]. In ubiquitous computing applications, advances in activity recognition have allowed passing from low-level recognition to identifying daily life activities.

This paper presents an approach for identifying user behavior through the analysis of user-object interactions. This is addressed in the context of a case study related to students solving academic activities using a Tangible-Intangible hybrid system that allows detecting and collecting user-object interaction data. The main contribution of this paper is the analysis of user-object interaction data to find relevant information related to user behavior while performing an academic activity in a technologically enhanced environment.

The remainder of this paper is organized as follows. Section 2 presents background information related to behavior representation, behavior detection, and behavior analysis. Section 3 describes a framework for analyzing user-object interaction data, that encompasses 3 phases: 1) Extraction and processing, 2) Transformation, and 3) Pattern recognition and evaluation; further explained in Sections 4, 5, and 6 respectively. Section 7 presents the discussion and limitations. Finally, Section 8 concludes this paper.

2. Background

Human behavior refers to how an individual responds to various stimuli in their environment, physically and mentally. Computationally, it is possible to determine user behavior from observation and comparative analysis, and it is conceivable to classify their behavior by monitoring the development of a person's activities. In an intelligent environment, behavior patterns can be obtained from analyzing user interactions data [1]. For a computer system, modeling the behavior of users and understanding signs of the patterns it identifies involves various tasks, the first of which is to monitor and collect data related to user interactions, and then analyze the detected behavior signals considering the context.

This section describes the characteristics of the styles used in this document.

2.1 Behavior representation

In the educational field, different research projects have defined student behavior from different perspectives and classifications, the most general being On-Task and Off-Task behaviors [6-7].

On-Task behavior is defined as the activity performed by the student that complies with the instructions given by the teacher for the task or lesson; for example, listening to the teacher's instructions, writing, reading a text, talking to the teacher, and looking up words in the dictionary. In addition, it defines Off-Task behavior as any activity performed by the student that is not directly associated with the task in the way that it was instructed, for example, leaving the classroom, excessive erasing, drawing on the desk, or looking at a point in space [6]. Another definition of On-Task behavior refers to the student's attention during class time. While Off-Task behavior, as student inattention during class time as self-distractions, interactions with their peers, environmental distractions, or others that include things or actions that are not classifiable into previous categories [7].

In addition, it is possible to find that behaviors studied and classified in educational settings include affective states such as boredom, commitment, and concentration [8], mental states such as flow experience [9], and dedication or absorption during tasks [10-11].

2.2 Behavior detection

Different research projects have studied human-object interaction detection. One example is the improvement of the accuracy while detecting interactions between persons and objects using computer vision and a graph neural network [12], or a graph model-based algorithm [13]; an application for smart glasses that assists workers in an industrial site recognizing human-object interactions [14]; as a contribution in the visual understanding field [15-16], or to help to solve the problem of missing human behavior objects [17]. It can be observed that behavior detection is related to the technologies available in the intelligent environment configuration, the sensors installed in the background, or can be related to the data sources, such as the case of images, videos, interactions with software systems [18], cameras for collecting images, microphones to record sounds within the classroom [19], and the dialogue between student and teacher and wearable devices [20] through which it is possible to identify tasks and student behavior in a classroom or digital learning environments as intelligent tutors. These data provide information that identifies individual and group behaviors [21]. Although observation methods for identifying student behaviors predominate in the educational field, there are efforts to identify them from the interaction data that the student has with a computer program, as observed in the work of [6, 8, 22, 23].

However, there is no defined record structure to store user interaction data in intelligent educational environments. Even though works such as [13, 24-31] use sensor data to address the situations of interest of each job, no defined structure is observed that they share among themselves.

2.3 Behavior analysis

Several approaches are observed to analyze data generated by users in intelligent environment settings. These include visual representations and analysis routines that allow the teacher to track the performance of one or several students to track their progress [27, 32-33]; comparison, and classification. Analysis of the accelerometer data to identify activities of daily living using a Gaussian mixture model and Gaussian mixture regression [34] to classify the data at runtime. Also, the use of multi-agent systems to analyze user behavior [35] and logistic regression models to predict, at runtime, behaviors of lack of commitment or gaming the system behavior.

Different research projects have used various techniques for data analysis to identify user behavior. In intelligent environment settings, these techniques include but are not limited to the Hidden Markov Model (HMM), probabilistic hierarchical models of human behavior using HMM, conditional random fields, and dynamic Bayesian networks. Other works, such as those by [36] and [37], consider artificial neural networks for monitoring and predicting activities of daily living; On the other hand, [38] proposes a hybrid inference approach to detect abnormal user behavior, and [39] uses sequential pattern mining. Also, for the analysis of human behavior, [40] has proposed using a temporal structure or a set of actions over time with T-patterns, broadly used by [41] and [42] or sequential patterns through the GSP algorithm, proposed by Skirant [43]. This algorithm allows the discovery of sequential patterns with a minimum support specified by the user, where the support of a pattern is the number of data sequences containing that pattern. It makes multiple passes over the data [43].

3. Framework for analyzing user-object interaction data

This section describes a proposal, based on [44-45], for a framework to analyze user-object interaction data for student modeling in intelligent learning environments. It consists of three phases: data acquisition and storage, transformation, and analysis.

Extraction and processing. The main objective of this phase is to collect and store user-object interaction data in an intelligent learning environment setting from a Tangible User Interface, the TanQuery prototype [46]. In TanQuery, there are three different areas where student interactions take place and are identified: the assignment area, the work area, and the result area. In the assignment area, the user assigns a value to the tokens he will use, depending on the token type and the information available in the database. In the work area, the user places the tokens to build a query tree and the system can identify the objects that compose it (root, left child, right child, or attributes); this distribution of objects is evaluated, and relational algebra expressions are constructed. The relational algebra expression (and the equivalent SQL expression) as well as the results of its evaluation are displayed in the result area. Fig. 1 shows the configuration of the TanQuery prototype.

During the activity, the student interacts with different tokens (objects) and generates different records of user-object interactions. This user-object interaction data is stored in a local database with a structure based on the physical layer of the intelligent desktop conceptual model [44] and considers the identity of each object visible to the system, the type of object (relation, attribute, operator, among others), the position (x, y), and the degrees used to calculate the location of the object in the application domain. Also, this record considers the domain information in the root, right child, left child, parent, and attributes, each for each component of the query tree; it also considers the date and interaction time.

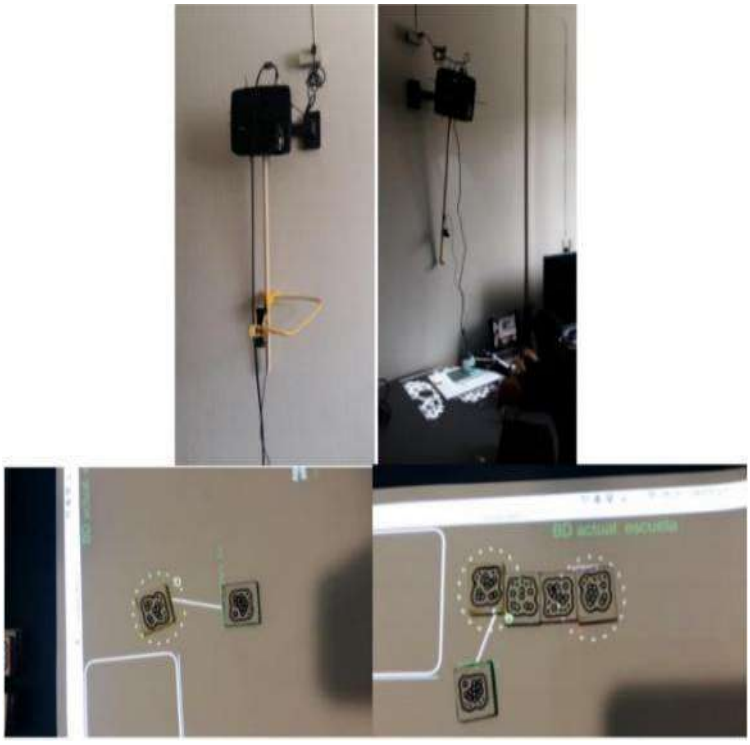


Fig. 1. Implementation of TanQuery prototype

Transformation. This phase transforms the interaction data into user behaviors considering an interval. This phase considers a model of the student's behavior during the activity allowing, based on the presence or absence of activities in an observation period, to identify whether the student has On-Task or Off-Task behavior. Once the behavior is defined, the same behaviors in contiguous intervals generate behavior sequences with different durations. Furthermore, during this phase, the data is preprocessed as input for the GSP algorithm.

Pattern recognition and evaluation. With a sequential algorithm, a process to identify the student behavioral patterns is carried out to identify representative behavioral sequences of each performance quadrant and train a classifier to condense the sequences of student behavior so that the intelligent environment can readily apprehend similarities and dissimilarities.

4. Data extraction and processing

For this study, a group of 16 students (12 male, 4 female) with similar ages (Max: 24 y/o, Min: 20 y/o, Average: 21 y/o), and basic relational database knowledge, all the participants were enrolled in the fifth semester of undergraduate studies. Also, and a teacher (Male with 14 years of experience) from the educational program related to information technology participated voluntarily (using a non-probabilistic sampling scheme) from a university in the southeast of Mexico, they were enrolled in the Databases subject. The participants were asked to participate in an open call, in which the evaluation process and the instruments to be used are explained. Based on the Belmont Report [47] on interaction with human beings, they were asked to request signing a letter as a requirement of consent. The study sample included participants whose reported age was 20 years or older, who had the same credit advancement in the Bachelor of Computational Technologies from the same class, and who were enrolled in the educational experience databases.

All participants interacted individually to solve relational algebra exercises using the TanQuery prototype, while a system stores the interaction data in a database table with the user-object interaction record structure. The connection to the database management system and the insertion in the online records are made each time the user interacts with the objects within the intelligent environment. It is sought that these are always available so that any component of the system and the environment can use them if required. The organization of these data is according to the user who generates them and the session in which they execute the activities. Table 1 illustrates an example of user-object interaction records.

Table 1. Example of User-Object interaction records

Name	Example	Example	Example	Example
ID	8	20	24	45
Type	Relation	Attribute	Attribute	Operator
X position	865	1080	1171	993
Y position	692	864	937	795
Value	STUDENT_INF	NOM_EST	EMAIL_EST	PROYECTION
Rotating degrees	0	358	355	86
Action	updateTUIObj	updateTUIObj	updateTUIObj	updateTUIObj
Root	PROYECTION	PROYECTION	PROYECTION	PROYECTION
Right child	-	-	-	-
Left child	-	-	-	8
Parent node	45	45	20	-
Attributes	-	24	-	20
Date	06/04/17	06/04/17	06/04/17	06/04/17
Time	12:52:57	12:52:57	12:52:57	12:52:57

5. Transformation

Once the dataset is created, the system finds the ObservedBehavior, Interval, StartTime, and EndTime columns. The possible intervals are obtained considering the analysis interval. For this work, an interval of 7 seconds is configured. Based on the interaction record and the number of

possible intervals, an iterative process is carried out that obtains the records generated by the student to determine if he manifests an On-Task behavior or an Off-Task behavior.

After the analysis is finished, is possible to know the observed behavior, the analysis interval, and the moment of the student's interaction in which the interval begins and ends; an example is shown in Table 2.

Subsequently, the user-object interaction data for each student is recovered to identify the On-Task and Off-Task behaviors manifested during the activity to find sequences of student behavior. Also, the characteristics of the activity and the student's result are stored manually in the database to calculate the student's performance in the activity.

To execute the analysis, the behavior sequence structure IS modified by calculating the duration, in intervals, of the behavior and assigning the prefix ONT for On-Task behavior and OFFT for Off-Task behavior. For example, the behavior sequence of student 1, where he manifests the On-Task behavior from intervals 9 to 14, was represented as ONT6, and the Off-Task behavior during interval 15 was represented as OFFT1. To identify the student's performance, the number of hits obtained by the student is recovered. The total time in the activity is calculated by subtracting the time of the last record and the time of the first record. From the activity information, the Minimum Expected Grade (MEG), defined as the minimum score to pass the activity, and the maximum time to develop the activity are recovered (both defined by the professor); these values serve as parameters of the defineStudentTaskPerformance function [44].

Table 2. Identification of student behavior intervals

Observed behavior	Interval	Start time	endTime
On-Task	1	06/04/2017 12:48:07	06/04/2017 12:48:37
On-Task	2	06/04/2017 12:48:37	06/04/2017 12:49:07
On-Task	3	06/04/2017 12:49:07	06/04/2017 12:49:37
On-Task	4	06/04/2017 12:49:37	06/04/2017 12:50:07
On-Task	5	06/04/2017 12:50:07	06/04/2017 12:50:37
On-Task	6	06/04/2017 12:50:37	06/04/2017 12:51:07
Off-Task	7	06/04/2017 12:51:07	06/04/2017 12:51:37
On-Task	8	06/04/2017 12:51:37	06/04/2017 12:52:07
On-Task	9	06/04/2017 12:52:07	06/04/2017 12:52:37
On-Task	10	06/04/2017 12:52:37	06/04/2017 12:53:07
On-Task	11	06/04/2017 12:53:07	06/04/2017 12:53:37
On-Task	12	06/04/2017 12:53:37	06/04/2017 12:54:07
On-Task	13	06/04/2017 12:54:07	06/04/2017 12:54:37

The student performance quadrant scheme allows to classify the students considering the activity's results and time. Each quadrant represents the confluence between the time developed in the activity and the result obtained according to the following:

- Quadrant A: the grade obtained by the student is greater than the MEG, and the time it took to complete the task could be more optimal.
- Quadrant B: the grade obtained by the student is greater than the MEG, and the time it took her to complete the task is greater than optimal.
- Quadrant C: the grade obtained by the student is below the MEG, and the time it took her to complete the task could be more optimal.
- Quadrant D: the grade obtained by the student is below the MEG, and the time it took her to complete the task is greater than optimal.

6. Pattern recognition and evaluation

The objective of this stage is to calculate a classification model from the data related to the student's behaviors. Since the sequences have different lengths for each student (see Fig. 2), it was necessary to characterize all the sequences in the same way. One possible way was to characterize each sequence by the number of On-Task and Off-Task behaviors present in the sequence; however, with this way of doing it, the temporary nature of the sequences was lost. As an alternative, we decided to characterize a sequence by subsequences, which we call motifs. Thus, all the possible motifs were calculated from all the sequences of students' behaviors and, later, each sequence was characterized in terms of the presence or absence of a motif in it, thus representing all the sequences in a homogeneous way. In the end, each student is represented by characterizing their sequence in terms of motifs plus the performance quadrant to which they belong. The data represented in this way were analyzed using machine learning techniques to arrive at the classification model. Section 6.1 details the characterization of the sequences in terms of motifs and section 6.2 explains the analysis using machine learning techniques.

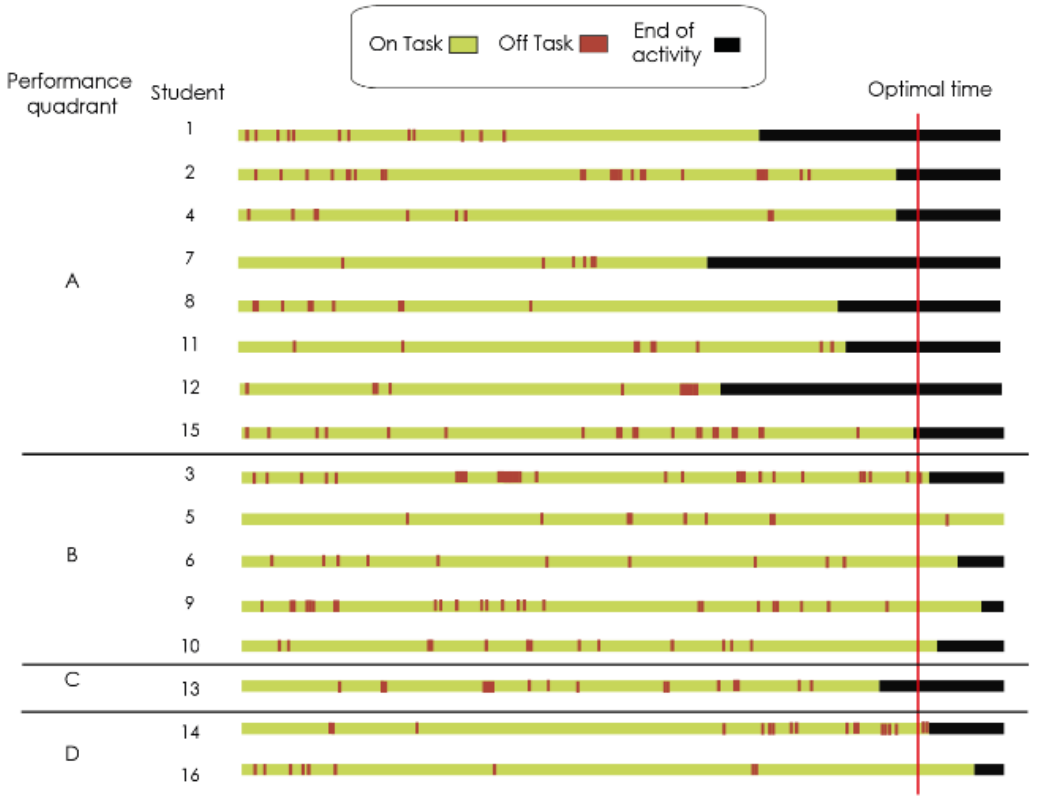


Fig.2. Participant's behavioral sequences

6.1 Characterization of sequences

Once having represented the behavioral sequences considering the performance quadrant and student behavior, the behavioral sequences were ordered alphabetically to assign them a numerical value, shown in Table 3, and they were substituted in the sequences of each participant. The set of behavior sequences for each quadrant was sent as a parameter to the GSP algorithm, and the

sequential patterns and support for each one was obtained. The complete number sequences of quadrants A, B, C, and D students were 213.

Table 3. Example sequence of student 1's behavior representation (Rep)

Participant	Interval		Behavior	Rep..	idSec
	Begin	End			
Student1	1	2	On-Task	ONT2	22
Student1	3	4	Off-Task	OFFT2	3
Student1	5	6	On-Task	ONT2	22
Student1	7	8	Off-Task	OFFT2	5
Student1	9	14	On-Task	ONT6	51
Student1	15	15	Off-Task	OFFT1	1
Student1	16	18	On-Task	ONT3	30
Student1	19	20	Off-Task	OFFT2	3
Student1	21	37	On-Task	ONT17	19
Student1	38	38	Off-Task	OFFT1	1
Student1	39	41	On-Task	ONT3	30

The Minimum Support Value (MinSup) represented as a percentage of occurrence (support) of the transactions containing all the pattern items in the GSP algorithm was set to 50%. For quadrant A, 56 sequences were found with support equal to or greater than four and equal to or less than eight. In quadrant B, 74 sequences with support equal to or greater than three and equal to or less than five were found. In performance quadrants C and D, no sequences with support equal to 50% were found. However, 83 sequences with a support value equal to two were found. It is observed that the patterns with a higher support number contain fewer behavioral sequences, and those with a lower support number are made up of more complex sequences.

An example of the behavioral patterns of quadrant A is shown in Table 4. It can be noted that the shortest patterns contain only one behavior sequence, and the longest contains five. The highest support found in the patterns is eight. The patterns with this support manifest the OFFT1 sequence, which is the presence of the Off-Task behavior during a seven-second interval at different times of the activity, but no more than three times during the entire activity. Sequences with a support of 6 involve the behavior Off-Task with a longer duration (OFFT2) or the presence of OFFT1 four times during the entire activity. Behavior patterns with less support (minimum four) are also observed that involve the occurrence of sequences of behavior Off-Task for two or more periods of seven seconds (OFFT2, OFFT3) and the presence of behaviors On Task and Off-Task at time intervals greater than 21 seconds (ONT3, ONT4). Considering the scope of application, this is interpreted as students in this quadrant spending very little time outside of the activity and combining short periods of inactivity with periods of activity of intermediate duration. Most of the time, they interact and say they are doing the activity continuously. In other words, they spend the most time doing something.

Table 4. Example of behavioral patterns found in quad-A

Sequence	Support
OFFT1,	8.0
OFFT1, OFFT2 ,	6.0
OFFT1, OFFT1, ONT3,	5.0
ONT2, OFFT2, OFFT1, OFFT1,	4.0

Table 5 summarizes the results of measuring the student's performance and identifying the behaviors, showing the occurrence of behaviors (On-Task and Off-Task) manifested by each student during the interaction session grouped by performance quadrant.

In quadrant B, the shortest patterns contain one behavior sequence, while the longest contains ten. The highest support found is five for the pattern with a single sequence of behavior and three for the

pattern with the most extended sequence. The patterns with longer sequences and support three are shown below. In them, it is observed that the OFFT1 behavior sequence predominates, and it is observed in the pattern with a length of eight that this Off Task behavior sequences occur after the student manifests an On-Task behavior for 28 seconds. (ONT4). It is interpreted that the students of quadrant B, during the activity, manifest a more significant amount of dead time. When they return to do it, they do so for more extended periods. Compared to the students in performance quadrant A, the students in performance quadrant B interrupt a more significant number of times, for short periods, performing the activity. After that, they begin to perform it for a longer time. In other words, they tend to manifest periods of inactivity more frequently, followed by periods of activity periodically, causing them to take longer to perform. However, they could have answered the activity exercises correctly.

Table 5. Student behaviors and performance quadrant

Participant	Performance Quadrant	On-Task	Off-Task
Student1	A	183	14
Student2	A	220	30
Student3	B	227	34
Student4	A	241	9
Student5	B	284	3
Student6	B	262	10
Student7	A	171	6
Student8	A	215	12
Student9	B	253	27
Student10	B	250	14
Student11	A	218	11
Student12	A	170	13
Student13	C	224	18
Student14	D	245	18
Student15	A	234	23
Student16	D	269	9

In quadrant C, no patterns with support more significant than three were found because only one student was assigned in this quadrant based on his performance in the activity. It is necessary to have more students be able to identify sequences of behavior in common.

In quadrant D, all the patterns found have support two and range in length from one to eight behavior sequences. The most representative sequence in most patterns is ONT9, which means that the student manifested On Task behavior for 63 consecutive seconds during the activity. It is essential to mention that students from quadrants A and B do not manifest this sequence. In other words, students in the F quadrant spend more time in short periods of inactivity than doing the task and occasionally tend to resume the activity for long periods.

An analysis of the sub-sequences found from quadrants A and B was performed to allow the system to identify the performance quadrant based on the sequence of behaviors manifested during the activity. For this sequential analysis, the authors considered the works of [48-49].

The procedure for the analysis of each student sequence is as follows:

- The sub-sequence identifier algorithm finds all quadrants' sub-sequences in the sequence.
- Each sub-sequence occurrence is stored, a matrix is generated, and this matrix is sent to the classifier.

The sequences of the students are those identified when the atomic behaviors manifested during the study session were coupled. The sub-sequences used for this exercise are those found by the GSP algorithm from performance quadrants A and B. The sub-sequences of quadrants C and D weren't

considered due to the small number of students who, according to the calculation of their performance, have 1 and 2 students, respectively.

Table 6 presents an example of sub-sequence organization considering the motif approach. Subsequently, the sub-sequences found were counted for each student (13 students, eight from quadrant A and five from quadrant B). The students from C and D quadrants were not considered because their GSP patterns were not representative due to the small number of students belonging to those quadrants. Subsequently, these were organized, considering the structure of the motifs of the article [48] and the analysis process [48-49], resulting in the information being organized as presented in Table 7. A total of 128 sub-sequences correspond to the performance quadrants A and B after removing two repeated sequences in both performance quadrants.

Table 6. Example of sub-sequences and motif organization

Sub-sequence	Motif
OFFT1	motif1
OFFT2	motif2
OFFT3	motif3
ONT2	motif4
…	
OFFT1, OFFT1, OFFT1, OFFT1, OFFT2	motif53
OFFT1, OFFT1, OFFT1, ONT3, OFFT1	motif54
OFFT1	motif55
OFFT2	motif56

Table 7. Example of motif(m) occurrence in each student (St) sequence

St	m1	m2	m3	m4	m8	m10	m13
1	6	4	0	2	15	20	12
2	8	3	2	2	28	7	3
3	14	1	1	1	91	3	10
4	5	2	0	1	10	3	5
5	3	0	0	0	3	0	0
6	10	0	0	0	45	0	0
7	4	1	0	1	6	0	0
10	8	3	0	1	28	12	7
11	5	0	2	0	10	0	0
12	3	1	0	0	3	2	0
15	9	4	2	1	36	6	6

6.2 Analysis of sub-sequences of student behavior

Once the occurrence of each behavior sub-sequence in each sequence of the 13 students was recovered, two students were randomly drawn. The remaining 11 were used to train the classifier using the Orange Data Mining tool, version 3.30. 1. The data on sub-sequence occurrence from the two students separated from the original group were used to assess whether the classification was adequate. The results of the different classification models in the Orange Data Mining tool are shown in Table 8. There we can see that of the four selected models (Neural Network, Random Forest, Naive Bayes, Tree), the two models that had the best results, despite the small data, were the Neural Network and Naive Bayes.

Various feature elimination methods were used to know the most representative sub-sequences of quadrants A and B. The authors used ReliefF and Information Gain as scoring methods. The results are shown in Table 9.

One of the approaches to identify the performance quadrant while the student is performing the activity is through decision trees, defined by [44] as "a prediction model that, given a set of data,

diagrams of logical constructions are made, which serve to represent and categorize a series of conditions that occur successively, for the resolution of a problem".

Table 8. Evaluation results

Model	AUC	CA	F1	Precision	Recall
Tree	0.214	0.455	0.398	0.354	0.455
Random Forest	0.143	0.273	0.285	0.315	0.273
Neural Network	0.500	0.636	0.636	0.636	0.636
Naive Bayes	0.500	0.636	0.636	0.636	0.636

Table 9. Best-scored sub-sequences using the Scoring Methods (SM): ReliefF (RF) and Information Gain (IG)

SM	Id	Sub-sequence	Value
RF	58	ONT20	0.192
RF	67	ONT10, ONT4	0.178
RF	80	ONT4, ONT10, OFFT1	0.163
IG	58	ONT20	0.468
IG	67	ONT10, ONT4	0.468
IG	80	ONT4, ONT10, OFFT1	0.370

Finally, to identify in an intelligent educational environment, at runtime, the student's performance quadrant based on the interaction with objects, it is necessary to find a set of rules that, incorporated into the ambient intelligence system, allow personalized support without waiting to finish the activity. A decision tree and the data set of the 11 previously selected students were used to find them.

Fig. 3 shows the resulting decision tree, from which the following rules are identified:

- If (motif1 > 9) = QuadrantedperformanceB.
- If (motif1 ≤ 9)&(motif58 ≤ 0) = QuadrantedperformanceA.
- If (motif1 ≤ 9)&(motif58 > 0) = QuadrantedperformanceB.

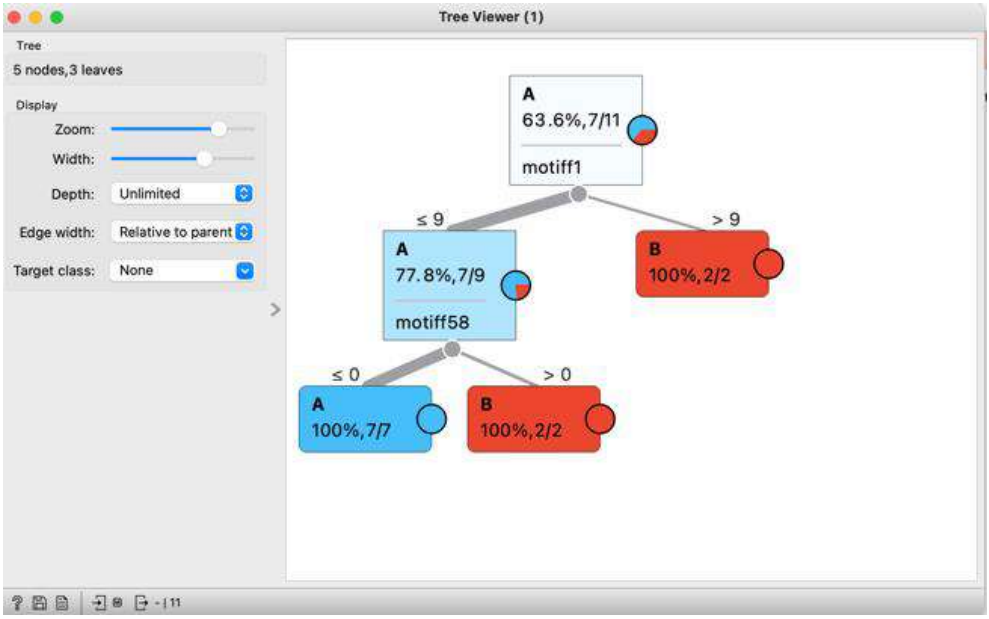


Fig. 3. Decision tree

The rules obtained from the decision tree are likely used so that, during the activity, the intelligent environment can generate personalized interventions that improve student performance. For example, if it is identified from the interaction records (that the student generates during the activity) that the student may result in low performance, the environment would intervene to help him improve his final performance in a personalized way.

7. Discussion

Including personalization mechanisms in an intelligent educational environment will allow the environment or educational systems to execute prompt interventions by identifying students in at-risk situations and aiding based on teaching-learning experiences [50].

To achieve this, it is necessary to consider contextual information about the student (academic and personal) to identify the student's learning needs, why they need to learn it, and the most appropriate way to learn it. With this, it is possible to design a personalized learning path for each student that, through a non-intrusive follow-up supported by technology, would allow the identification of deviations in those paths and, therefore, students in risk situations.

Knowing the student's behaviors while performing the activity would allow the intelligent educational environment to improve student performance by helping them move from a C or D quadrant, which performs below the desired level, to an A or B. An intelligent educational environment would provide support materials or interventions on the activity's topics.

The process described in this work can be helpful to incorporate personalization mechanisms into educational environments and to enhance customization services or applications like affective feedback systems [51], identifying the competency level of students [52], monitoring students while writing academic tasks [53], and learning monitoring and customization in mobile learning platforms [54]. To analyze data from user interaction with objects or with educational software and provide customized learning materials for each student considering their strengths, needs, skills, interests, and academic performance.

8. Conclusions and future work

The proposal addressed in this work contributes to identifying user behaviors through monitoring the user-object interactions manifested when performing academic activities and monitoring through a non-intrusive observation of the activities carried out in intelligent educational environments. Mainly, when addressing the case study, it is intended to contribute to the improvement of teaching-learning processes by supporting its users by identifying their behaviors and associating them with the performance of the academic task they are performing to identify needs presented by students or to identify ways to improve their performance in a university intelligent learning environment setting.

It was necessary to analyze recent works on intelligent environments, ubiquitous computing, and ambient intelligence that address identifying user behaviors and those applied in the educational field. This topic is partially covered in the educational field, so it was in our interest to work on it.

During the data-collection process, light conditions changed from natural to artificial to avoid problems related to the detection of markers. Also, a local database server was set up to avoid misconnection and data-storage problems.

The test results showed that it is possible to measure the student's performance quadrant and identify her performance by analyzing her interactions with objects. Also, a higher concentration of students is observed in the A and B performance quadrants, and fewer students in the C and D quadrants. However, this is an unexpected result due to the balance of the quadrants for the classification of students and subsequent discovery of patterns; it is a positive result that indicates that the students, for the most part, perform well in terms of having a higher-than-expected result in the activity. Most students exhibited On Task behavior for most of the activity; it is possible to interpret this as a

greater involvement of the participants in the task. Nevertheless, it is possible to develop further studies in a controlled environment where the student is in a more familiar environment for him and with more freedom.

Although it is not possible to generalize the results from such a limited sample, observable indications and differences are obtained in the development of the activity of the students in each of the quadrants. Students in quadrants A and C show more behavior Off Task at the beginning and end of the activity, while students in quadrant B show this behavior throughout the development of the activity. In the two students in performance quadrant D, the development of the activity is different.

Concerning the use of observed behavior patterns, identifying the occurrence of sub-sequences in each behavior sequence of the students in quadrants A and B was carried out due to having a more significant number of students. Using the Orange Data Mining tool and the classification models Neural Network, Random Forest, Naive Bayes, and Tree, training, and testing were done with the data of the 13 students (11 for training and two for testing).

The number of participants for this work is small and it is not possible to generalize results. Due to this very limited number of data, the performance of the classifiers is low and can be improved by increasing the sample. Therefore, the results obtained are not conclusive, so more experiments with a larger number of participants will be needed to obtain more reliable results.

Future work considers the evaluation with a larger number of participants, also the incorporation of the analysis process into a system to automatically identify student behavior and performance during the activity to provide customized assistance during the learning process.

References

[1]. G. Virone et al., Behavioral Patterns of Older Adults in Assisted Living, in IEEE Transactions on Information Technology in Biomedicine, vol. 12, no. 3, pp. 387-398, May 2008, doi: 10.1109/TITB.2007.904157.

[2]. Pantic Maja, et al. Human computing and machine understanding of human behavior: A survey. Proceedings of the 8th international conference on Multimodal Interfaces. 2006.

[3]. Favela Jesus. Activity, Behavior, and Context: The ABC of Pervasive Healthcare Research. Intelligent Environments (Workshops). 2012.

[4]. Nef Tobias et al. Evaluation of three state-of-the-art classifiers for recognition of activities of daily living from smart home ambient data. Sensors 15.5 (2012): 11725-11740.

[5]. Vinh La The, et al. Semi-Markov conditional random fields for accelerometer-based activity recognition. Applied Intelligence 35 (2011): 226-241.

[6]. Moore Dennis W. et al. Increasing on-task behavior in students in a regular classroom: Effectiveness of a self-management procedure using a tactile prompt. Journal of Behavioral Education 22 (2013): 302-311.

[7]. Godwin Karrie E. et al. Off-task behavior in elementary school children. Learning and Instruction 44 (2016): 128-143.

[8]. Doddannara Lakshmi S. et al. Exploring the relationships between design, students' affective states, and disengaged behaviors within an ITS. Artificial Intelligence in Education: 16th International Conference, AIED 2013, Memphis, TN, USA, July 9-13, 2013. Proceedings 16. Springer Berlin Heidelberg, 2013.

[9]. Wang Li-Chun, Ming-Puu Chen. The effects of game strategy and preference-matching on flow experience and programming performance in game-based learning. Innovations in Education and Teaching International 47.1 (2010): 39-52.

[10]. Schaufeli Wilmar B., Arnold Bakker. UWES Utrecht work engagement scale preliminary manual. Occupational Health Psychology Unit (2003).

[11]. Bumbacco C., Scharfe E. (2023). Why attachment matters: first-year post-secondary students' experience of burnout, dIsengagement, And drop-out. Journal of College Student Retention: Research, Theory & Practice, 24(4), 988-1001.

[12]. Zhang J., Mohd Yunos Z, Haron H. Interactivity Recognition Graph Neural Network (IR-GNN) Model for Improving Human–Object Interaction Detection. Electronics. 2023; 12(2):470. https://doi.org/10.3390/electronics12020470

[13]. Qing Ye, Xiuju Xu, Human-object interaction detection based on graph model, Proc. SPIE 12610, Third International Conference on Artificial Intelligence and Computer Engineering (ICAICE 2022), 126100A (28 April 2023); https://doi.org/10.1117/12.2671248
[14]. Mazzamuto Michele, et al. A wearable device application for human-object interactions detection. International Conference on Computer Vision Theory and Applications (VISAPP). 2023.
[15]. Luo Tianlun, et al. From detection to understanding: A survey on representation learning for human-object interaction. Neurocomputing (2023): 126243.
[16]. Kim, Sanghyun, Deunsol Jung, Minsu Cho. Relational Context Learning for Human-Object Interaction Detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.
[17]. Ye Qing, et al. Human object interaction detection based on feature optimization and key human-object enhancement. Journal of Visual Communication and Image Representation 93 (2023): 103824.
[18]. Zidianakis Emmanouil, et al. Building a sensory infrastructure to support interaction and monitoring in ambient intelligence environments. Distributed, Ambient, and Pervasive Interactions: Second International Conference, DAPI 2014, Held as Part of HCI Interational 2014, Heraklion, Crete, Greece, June 22-27, 2014. Proceedings 2. Springer International Publishing, 2014.
[19]. D'Mello Sidney K. et al. Multimodal capture of teacher-student interactions for automated dialogic analysis in live classrooms. Proceedings of the 2015 ACM on international conference on multimodal interaction. 2015.
[20]. Belapurkar Neha, Sagar Shelke, Baris Aksanli. The case for ambient sensing for human activity detection. Proceedings of the 8th International Conference on the Internet of Things. 2018.
[21]. Jayarajah Kasthuri et al. Need accurate user behaviour? pay attention to groups!. Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing. 2015.
[22]. Baker Ryan Shaun et al. Off-task behavior in the cognitive tutor classroom: When students "Game the system". Proceedings of the SIGCHI conference on Human factors in computing systems. 2004.
[23]. Baker Ryan. Modeling and understanding students' off-task behavior in intelligent tutoring systems. Proceedings of the SIGCHI conference on Human factors in computing systems. 2007.
[24]. Leonidis Asterios et al. An intelligent task assignment and personalization system for students' online collaboration. Universal Access in Human-Computer Interaction. Applications and Services: 6th International Conference, UAHCI 2011, Held as Part of HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings, Part IV 6. Springer Berlin Heidelberg, 2011.
[25]. Sanish Rai, Xiaolin Hu. Behavior pattern detection for data assimilation in agent-based simulation of smart environments. 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). Vol. 2. IEEE, 2013.
[26]. Wang Qunbo, Wenjun Wu, Yuxing Qi. A Learning Analytic Model for Smart Classroom. Web and Big Data: APWeb-WAIM 2018 International Workshops: MWDA, BAH, KGMA, DMMOOC, DS, Macau, China, July 23–25, 2018, Revised Selected Papers 2. Springer International Publishing, 2018.
[27]. Korozi Maria et al. LECTOR: towards reengaging students in the educational process inside smart classrooms. Intelligent Human Computer Interaction: 9th International Conference, IHCI 2017, Evry, France, December 11-13, 2017, Proceedings 9. Springer International Publishing, 2017.
[28]. White Ryen W., Ahmed Hassan Awadallah, Robert Sim. Task completion detection: A study in the context of intelligent systems. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019.
[29]. Montebello Matthew. Assisting education through real-time learner analytics. 2018 IEEE Frontiers in Education Conference (FIE). IEEE, 2018.
[30]. Matsui Kanae, Tatsuhiko Kasai, Keiya Sakai. Challenges for data collecting of teacher and student'behavior in different types of class using video and wearable device. 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE, 2019.
[31]. Prabono Aria Ghora, Seok-Lyong Lee, Bernardo Nugroho Yahya. Context-based similarity measure on human behavior pattern analysis. Soft Computing 23 (2019): 5455-5467.
[32]. Biswas Gautam, Brian Sulcer. Visual exploratory data analysis methods to characterize student progress in intelligent learning environments. 2010 International Conference on Technology for Education. IEEE, 2010.
[33]. G. Mathioudakis et al. Amiria: Real-time teacher assistance tool for an ambient intelligence classroom. Proceedings of the Fifth International Conference on Mobile, Hybrid, and Online Learning (eLmL 2013). 2013.

[34]. Barbara Bruno et al. Analysis of human behavior recognition algorithms based on acceleration data. 2013 IEEE International Conference on Robotics and Automation. IEEE, 2013.
[35]. Valérian Guivarch et al. Hybrid system to analyze user's behaviour. 2016 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2016.
[36]. Louis Atallah, Guang-Zhong Yang. The use of pervasive sensing for behaviour profiling – a survey. Pervasive and mobile computing 5.5 (2009): 447-464.
[37]. Zacharoula Papamitsiou, Eirini Karapistoli, Anastasios A. Economides. Applying classification techniques on temporal trace data for shaping student behavior models. Proceedings of the sixth international conference on learning analytics & knowledge. 2016.
[38]. Valeria Soto-Mendoza, et al. Detecting abnormal behaviours of institutionalized older adults through a hybrid-inference approach. Pervasive and Mobile Computing 40 (2017): 708-723.
[39]. Andreas D. Lattner, et al. Sequential pattern mining for situation and behavior prediction in simulated robotic soccer. RoboCup 2005: Robot Soccer World Cup IX 9. Springer Berlin Heidelberg, 2006.
[40]. Magnus S. Magnusson. Discovering hidden time patterns in behavior: T-patterns and their detection. Behavior research methods, instruments, & computers 32.1 (2000): 93-110.
[41]. Maurizio Casarrubea et al. T-pattern analysis for the study of temporal structure of animal and human behavior: a comprehensive review. Journal of neuroscience methods 239 (2015): 34-46.
[42]. Carlos Santoyo et al. Observational analysis of the organization of on-task behavior in the classroom using complementary data analyses. Anales de Psicología, 2017, vol. 33, num. 3, p. 497-514 (2017).
[43]. Ramakrishnan Srikant, Rakesh Agrawal. Mining sequential patterns: Generalizations and performance improvements. Advances in Database Technology—EDBT'96: 5th International Conference on Extending Database Technology Avignon, France, March 25–29, 1996 Proceedings 5. Springer Berlin Heidelberg, 1996.
[44]. Hernández-Calderón, José Guillermo et al. A System to Match Behaviors and Performance of Learners From User-Object Interactions: Model and Architecture. International Journal of Information Technologies and Systems Approach (IJITSA) 12.2 (2019): 82-103.
[45]. Magdalena Cantabella et al. Analysis of student behavior in learning management systems through a Big Data framework. Future Generation Computer Systems 90 (2019): 262-272.
[46]. José Antonio Xohua-Chacón, Edgard Iván Benítez-Guerrero, Carmen Mezura-Godoy. A tangible system for learning relational algebra. Revista Colombiana de Computación 19.1 (2018): 39-55.
[47]. United States. National Commission for the Protection of Human Subjects of Biomedical, and Behavioral Research. The Belmont report: ethical principles and guidelines for the protection of human subjects of research. Vol. 1. Department of Health, Education, and Welfare, National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research, 1978.
[48]. Krisztian Buza, Lars Schmidt-Thieme. Motif-based classification of time series with bayesian networks and svms. Advances in Data Analysis, Data Handling and Business Intelligence: Proceedings of the 32nd Annual Conference of the Gesellschaft für Klassifikation eV, Joint Conference with the British Classification Society (BCS) and the Dutch/Flemish Classification Society (VOC), Helmut-Schmidt-University, Hamburg, July 16-18, 2008. Springer Berlin Heidelberg, 2010.
[49]. Jia-Min Ren, Jyh-Shing Roger Jang. Time-constrained sequential pattern discovery for music genre classification. 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2011.
[50]. Vanessa Alejandra Camacho Vázquez. Detección de emociones negativas en las redes sociales., Ph.D. Thesis, 2017.
[51]. Samantha Jiménez, R. Juárez-Ramírez, Víctor H Castillo, Alan Ramírez Noriega. The Role of Personality in Motivation to use an Affective Feedback System. Program Comput Soft 47, 793–802 (2021). https://doi.org/10.1134/S0361768821080156.
[52]. J. R. Aguilar-Cisneros, R. Valerdi, B. P. Sullivan. Students' Systems Thinking Competency Level Detection through Software Cost Estimation Concept Modeling. Program Comput Soft 48, 499–512 (2022). https://doi.org/10.1134/S0361768822080060.
[53]. Kochegurova E. A., Martynova Y. A. Aspects of Continuous User Identification Based on Free Texts and Hidden Monitoring. Program Comput Soft 46, 12–24 (2020). https://doi.org/10.1134/S036176882001003X.
[54]. H. Del Ángel-Flores, E. López-Domínguez, Y. Hernández-Velázquez et al. Usability Evaluation of a Mobile Learning Platform Focused on Learning Monitoring and Customization based on a Laboratory Study. Program Comput Soft 48, 583–597 (2022). https://doi.org/10.1134/S0361768822080102.

Информация об авторах / Information about authors

Хосе- Гильермо ЭРНАНДЕС-КАЛЬДЕРОН имеет степень PhD мексиканского Университета Веракруса в области компьютерных наук. Профессор факультета статистики и информатики этого университета. Сфера научных интересов: человеко-машинное взаимодействие, искусственный интеллект, совместные вычисления и визуализация.

José-Guillermo HERNÁNDEZ-CALDERÓN – Ph. D. in Computer Science from the University of Veracruz in México. Professor at the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Research interests: Human Computer Interaction, Artificial Intelligence, Collaborative Computing, and Visualization.

Эдгар Иван БЕНИТЕС-ГЕРРЕРО – доктор компьютерных наук Гренобльского университета во Франции. Профессор факультета статистики и информатики Университета Веракруса в Мексике. Научные интересы: взаимодействие человека с компьютером, искусственный интеллект, совместные вычисления, управление данными и визуализация.

Edgard Ivan BENÍTEZ-GUERRERO – Ph. D. in Computer Science from the University of Grenoble in France. Professor at the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Research interests: Human Computer Interaction, Artificial Intelligence, Collaborative Computing, Data Management and Visualization.

Хосе Рафаэль РОХАНО-КАСЕРЕС имеет степень PhD Технологического института фундаментальных исследований Доктор наук в области компьютерных наук в Институте технических наук, кампус Куэрнавака, Монтеррей. Также имеет степень магистра по искусственному интеллекту Университета Веракрузана. В настоящее время штатный профессор факультета статистики и информатики университета Веракрузана. Сфера научных интересов: семантическая глобальная информационная сеть, инструменты для совместной работы и изучение проблем нетрудоспособных людей.

José Rafael ROJANO-CÁCERES – PhD in computer science from the Instituto Tecnológico de Estudios Superiores Monterrey campus Cuernavaca. Master in Artificial Intelligence from the Universidad Veracruzana. Currently Full Time Professor at the Faculty of Statistics and Informatics of the Universidad Veracruzana. His research areas include semantic web, collaborative tools, and disability.

Кармен МЕСУРА-ГОДОЙ получила степень доктора компьютерных наук в Университете Савойи во Франции. Профессор факультета статистики и информатики Университета Веракруса в Мексике. Основные исследовательские интересы: человеко-машинное взаимодействие, совместные вычисления, визуализация и мультиагентные системы.

Carmen MEZURA-GODOY – PhD in Computer Science from the University of Savoie in France. Professor at the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Main research interests: Human-Computer Interaction, User eXperience-UX, Computer Support Collaborative Work, Visualization and Multiagent Systems.

DOI: 10.15514/ISPRAS-2024-36(1)-11

A Systematic Literature Review on Vision-Based Human Event Recognition in Smart Classrooms: Identifying Significant Events and Their Applications

M. L. Córdoba-Tlaxcalteco, ORCID: 0009-0001-8150-8097 <*marcordoba@uv.mx*>
E. Benítez-Guerrero, ORCID: 0000-0001-5844-4198 <*edbenitez@uv.mx*>

Faculty of Statistics and Informatics, University of Veracruz
Xalapa, Veracruz, Mexico

Abstract. The field of vision-based human event recognition in smart environments has emerged as a thriving and successful discipline, with extensive efforts in research and development driving notable progress. This progress has not only yielded valuable insights but also practical applications across various domains. Within this context, human actions, activities, interactions, and behaviors are all considered as events of interest in smart environments. However, when focusing on smart classrooms, a lack of unified consensus on the definition of "human event" poses a significant challenge for educators, researchers, and developers. This lack of agreement hinders their ability to precisely identify and classify specific situations that are relevant to the educational context. To address this challenge, the aim of this paper is to conduct a systematic literature review of significant events, with a particular emphasis on their applications in assistive technology. The review encompasses a comprehensive analysis of 227 published documents spanning from 2012 to 2022. It delves into key algorithms, methodologies, and applications of vision-based event recognition in smart environments. As a primary outcome, the review identifies the most significant events, categorizing them according to single-person behavior, multiple-person interactions, or object-person interactions, examining their practical applications within the educational context. The paper concludes with a discussion on the relevance and practicality of vision-based human event recognition in smart classrooms, especially in the post-COVID era.

Keywords: human event recognition, smart classroom, computer vision, artificial intelligence, educational technology.

For citation: Córdoba-Tlaxcalteco M.L., Benítez-Guerrero E. A Systematic Literature Review on Vision-Based Human Event Recognition in Smart Classrooms: Identifying Significant Events and Their Applications. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 175-198. DOI: 10.15514/ISPRAS-2024-36(1)-11.

Full text: M. L. Córdoba-Tlaxcalteco and E. Benítez-Guerrero. Human Event Recognition in Smart Classrooms Using Computer Vision: A Systematic Literature Review. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 625–642. DOI: 10.1134/S0361768823080066.

Систематический обзор литературы по визуальному распознаванию событий с людьми: выявление значимых событий и их применение

М.Л. Кордова-Тлакскальтеко, ORCID: 0009-0001-8150-8097 <*marcordoba@uv.mx*>
Э. Бенитес-Герреро, ORCID: 0000-0001-5844-4198 <*edbenitez@uv.mx*>
Факультет статистики и информатики университета Веракруса, Халапа, Веракрус, Мексико.

Аннотация. Область распознавания человеческих событий на основе видения в интеллектуальных средах стала процветающей и успешной дисциплиной, а обширные усилия в области исследований и разработок привели к заметному прогрессу. Этот прогресс не только дал ценную информацию, но также открыл возможность практических применений в различных областях. В этом контексте действия человека, действия, взаимодействия и поведение рассматриваются как события, представляющие интерес в интеллектуальных средах. Однако при сосредоточении внимания на умных классах отсутствие общепризнанного определения «человеческого события» создает серьезную проблему для педагогов, исследователей и разработчиков. Это отсутствие согласия препятствует их способности точно определять и классифицировать конкретные ситуации, имеющие отношение к образовательному контексту. Чтобы решить эту проблему, авторы поставили цель провести систематический обзор литературы о значительных событиях, уделяя особое внимание их применению в вспомогательных технологиях. Обзор включает в себя всесторонний анализ 227 опубликованных документов, охватывающих период с 2012 по 2022 год. Он углубляется в ключевые алгоритмы, методологии и приложения распознавания событий на основе видения в интеллектуальных средах. В качестве основного результата обзор определяет наиболее значимые события, классифицируя их в соответствии с поведением одного человека, взаимодействиями между несколькими людьми или взаимодействиями между объектом и человеком, изучая их практическое применение в образовательном контексте. Документ завершается обсуждением актуальности и практичности распознавания человеческих событий на основе видения в умных классах, особенно в эпоху после COVID.

Ключевые слова: распознавание событий с людьми; умный класс; компьютерное зрение; искусственный интеллект; образовательные технологии.

Для цитирования: Кордова-Тлакскальтеко М. Л., Бенитес-Герреро Э. Систематический обзор литературы по визуальному распознаванию событий с людьми: выявление значимых событий и их применение. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 175–198 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–11.

Полный текст: М.Л. Кордова-Тлакскальтеко, Э. Бенитес-Герреро. Распознавание событий с людьми в умных классах на основе машинного зрения: систематический обзор литературы. *Programming and Computer Software*, 2023, т. 49, № 8, стр. 625–642 (на английском языке). DOI: 10.1134/S0361768823080066.

1. Introduction

Human Event Recognition (HER) in Smart Classrooms (SC) involves using computer techniques to identify some human actions, activities, interactions, and behaviors within educational spaces equipped with data acquisition and processing infrastructure. Specifically, video data obtained from cameras in smart classrooms is of particular interest for interpreting educational scenes. This technology enables the detection, learning, recognition, and prediction of learners' and teachers' actions, allowing the system to assess and assist them accordingly [1]. This topic has proven beneficial for classroom management (e.g., automated attendance tracking), learning and teaching support (e.g., detecting social interactions and collaborative learning), and enhancing students' academic performance (e.g., identifying action patterns related to academic achievement) [2][4][5].

Previous reviews have addressed video-based HER (see Table 1). For instance, reference [9] provides an overview of recent vision-based techniques for recognizing human behaviors and

surveillance systems. In [10], deep learning methods with automatic feature extraction for vision-based human event recognition are reviewed. The authors of [11] present a comprehensive review of approaches to recognizing and representing human actions through visual data. Reference [12] presents a state-of-the-art review on recognizing suspicious behaviors in surveillance videos, including six different systems. Finally, reference [13] describes major video datasets for human event recognition. It's worth noting that while these works are important as they share common underlying techniques, none of them specifically focuses on SCs. In [243], a conceptual account of SC evolution and its relationship with AI and emerging educational technologies is provided.

This paper aims to analyze the state of the art in vision-based recognition of human events in smart classrooms, with a specific focus on identifying the most significant events. It seeks to provide educators, researchers, and educational technology developers with a comprehensive overview of the topic while also highlighting the lack of consensus on what events are considered the most significant in this context. To achieve this, the paper presents a systematic literature review of published works in the last 10 years. The review covers key concepts and methodologies, drawing from the analysis of 227 documents, and identifies relevant events and their applications in educational settings. By doing so, it aims to address research gaps and identify opportunities for further exploration in this evolving field

The paper is organized as follows. Section 2 provides background information on HER and SCs. Section 3 outlines the systematic review method. Section 4 presents the review's results, including a list of relevant events with references and brief descriptions. Finally, Section 5 concludes the paper.

Table 1. List of similar reviews in the literature

2004	[53]
2005	[54]
2008	[55]
2009	[7]
2010	[56]
2011	[58]
2012	[59], [60], [61]
2013	[62], [63], [64]
2014	[57], [65], [66]
2015	[67], [68], [69]
2016	[70], [71]
2017	[72], [73], [74], [75]
2018	[12], [11]
2019	[10], [77], [78], [79], [80], [81], [82], [83], [84], [85]
2020	[9]

2. Background

This section is organized as follows. First, the discussion focuses on HER and SCs. Next, Computer Vision methods for object extraction are presented. Finally, event understanding from video scenes is examined.

2.1 SC and HER in the Context of Educational Technology

The origins of Smart Classroom (SC) and Human Event Recognition (HER) can be traced back to the late 20th century when computers and the Internet were introduced in educational settings in the

1980s. The 1990s saw the emergence of Computer Supported Collaborative Learning (CSCL), which linked education and computational technology in collaborative settings. The early 2000s witnessed the growth of e-learning, online education, and the Internet of Things (IoT), leading to the establishment of educational spaces with high technological content. Concurrently, advancements in Machine Learning (ML) and Artificial Intelligence (AI) enabled robust and precise object detection and classification based on data, making it applicable in complex situations.

Since 2010, the widespread use of smartphones, mobile devices, and cloud computing has facilitated data collection, storage, processing, and sharing, giving rise to the concept of Smart Environments (SE). The COVID-19 pandemic in 2020 further emphasized the need for technologically assisted educational services in modern society. However, the complexity of the current educational setting remains a significant challenge for HER in SCs [6].

According to the taxonomy in [244], there are four types of SCs: Basic, Interactive, Collaborative, and Immersive. The majority of the reviewed research presented in this article corresponds to Basic SCs, equipped with multimedia equipment and a computer connected to the Internet. In this context, Computer Vision proves advantageous in easily obtaining SC data compared to the use of biometric, ambient, or wearable sensors. Moreover, much of the reviewed research focuses on the psychological, social, or behavioral dimensions of educational experiences. Current trends aim to combine multimodal data acquisition in IoT with data fusion in AI to cater to face-to-face, online, or hybrid educational modalities [85]. References from [231] to [242] review hybrid and sensor-based approaches to HER.

2.2 Computer Vision methods for object extraction

In video data analysis, the first step involves detecting features known as objects, which can be things, people, or combinations of both (see fig. 1). Object Detection (OD), Object Classification (OC), and Object Tracking (OT) are the common processes for extracting features or objects.

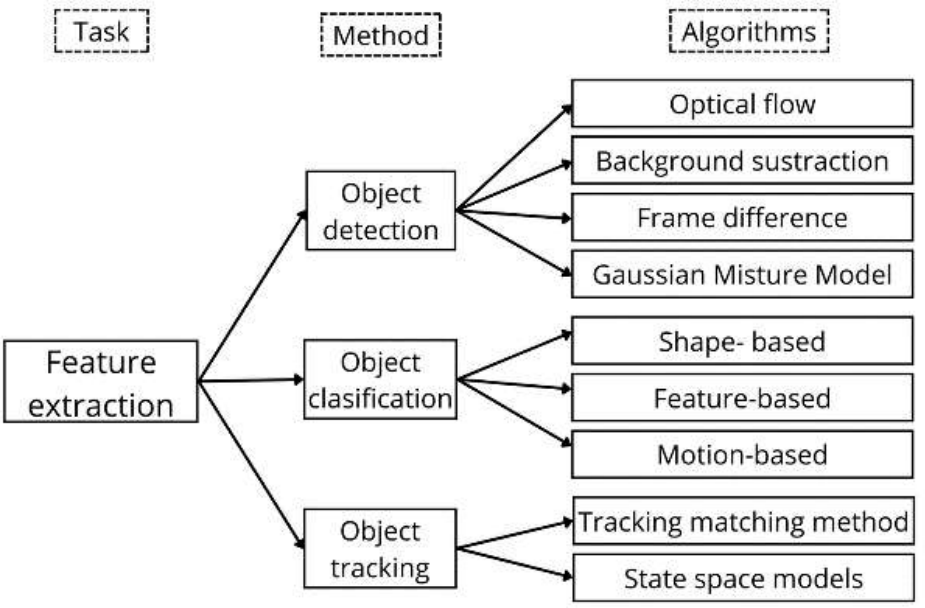


Fig. 1. Schematic representation of methods and algorithms for feature extraction with information from [246], created by the authors

The techniques for feature extraction include Optical Flow, Background Subtraction, Frame Difference, and Gaussian Mixture Model. Optical flow is described in [14], background subtraction in [15], and frame difference in [16]. Gaussian Mixture Model is used to estimate probability distributions, such as Gaussian distribution [17] or a mixture of Gaussian distribution [18], with fast estimation algorithms shown in [19] or [20]. For a more detailed discussion, refer to [246], upon which this section is primarily based.

OC (Object Classification) is the next step, involving shape-based, motion-based, and feature-based methods. Shape-based OC uses geometric properties like height/width ratio, perimeter, and area [21], useful for human figure classification [13, 22-23]. Motion-based classification relies on distinguishing objects based on their motion characteristics, recognizing human movements like walking or running [24-25]. Feature-based classification uses specific frame elements, such as skin color [26], which can also be combined with other descriptors [27].

The final stage is OT (Object Tracking), which creates a track of each object by capturing their locations over time [28]. Tracking Matching Methods find correspondence between object detections in different frames [29-30]. Another category, State Space Models, estimates object state (position, velocity, etc.) using a motion model corrected by incomplete measurements [32], with complete measurements obtained through OD algorithms [32].

2.3 Event understanding from video

Event understanding in video scenes involves interpreting elements based on known context (see fig. 2). It can be data-driven, using supervised and unsupervised machine learning methods like decision trees, KNN, SVMs, HMMs, and Bayesian networks [41]. Unsupervised learning constructs recognition models from unlabeled data using density estimation or clustering methods [35], including graphical models and eigen-decomposition [33-34].

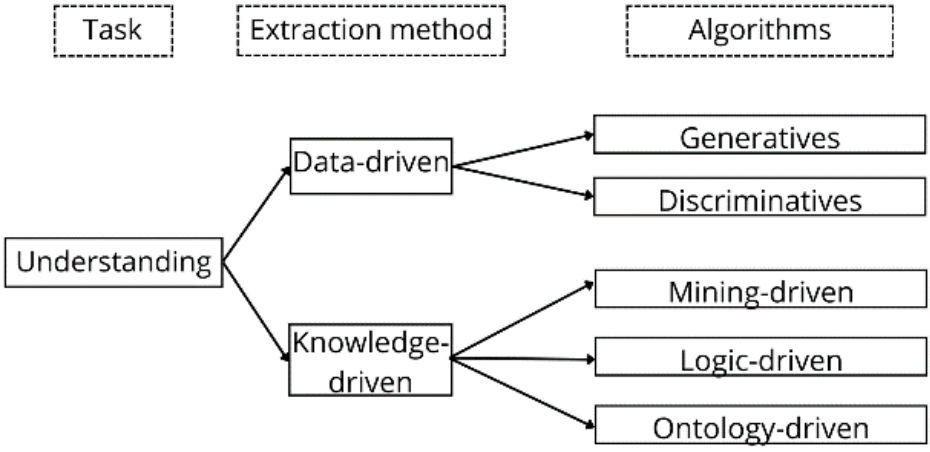


Fig. 2. Schematic representation of methods and algorithms for event understanding with information from [245], created by the authors

Data-driven algorithms in event understanding (fig. 3) can be classified into generative and discriminative methods [35]. Generative methods like Bayes classifiers, Hidden Markov Processes, and Bayesian Networks provide a complete description but require large data volumes for learning parameters. On the other hand, discriminative methods like Deep Learning Neural Networks, SVM, and Nearest Neighbor have lower computational costs but do not fully explain human events. Hybrid methods that combine both approaches have also been proposed [36].

Knowledge-driven understanding in event recognition utilizes formal knowledge [245]. Logical formalisms like Plans Recognition Theory [37-38], and Event Theory [39] are used for HER. Knowledge-driven methods (fig. 4) can be categorized into mining-driven, logic-driven, and ontology-driven approaches [40]. Mining-driven methods learn from pre-defined data to classify behaviors, while logic-driven methods use semantic representations and reasoning mechanisms. Ontology-driven methods, gaining interest in behavior recognition, offer an explicit representation of behavior definitions for broader applicability.

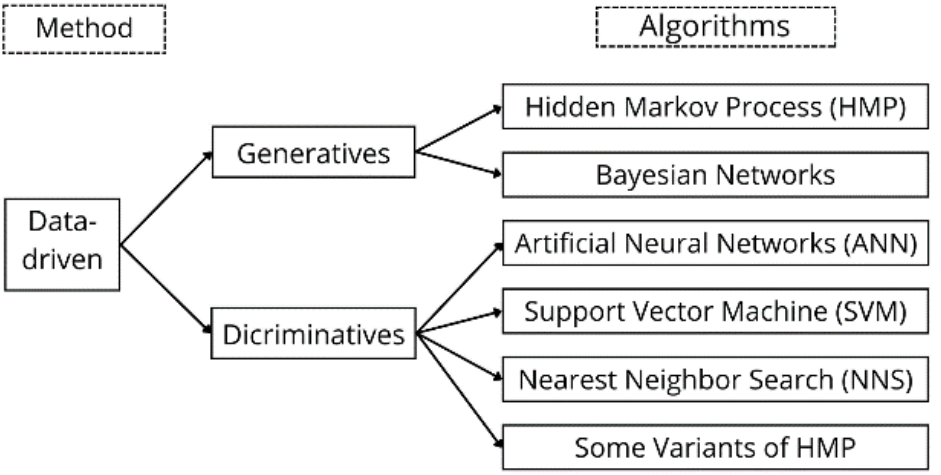


Fig. 3. Schematic representation of algorithms for data driven methods in HER, created by the authors

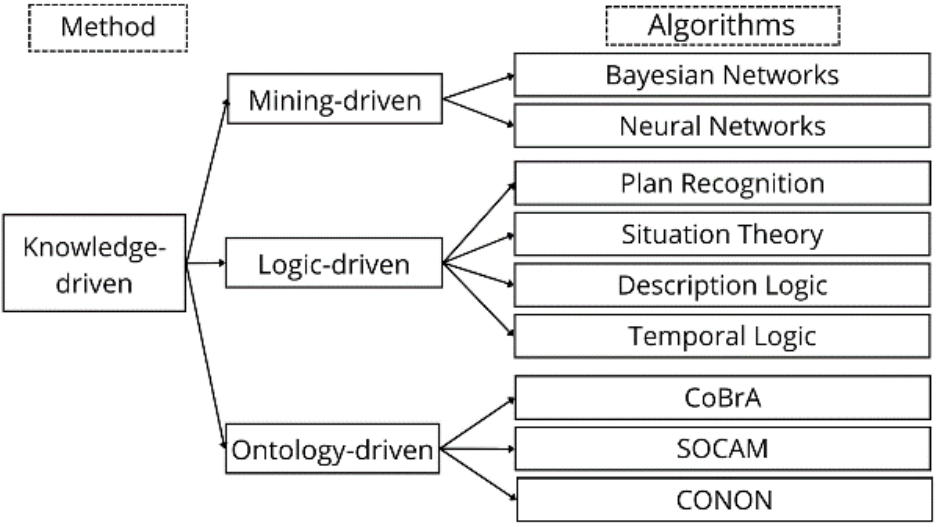


Fig. 4. Schematic representation of algorithms for knowledge-driven methods in HER, created by the authors

Ontologies offer advantages like independence from specific algorithms, promoting portability, interoperability, and reuse of technologies and systems. They have been used to model social

interaction in various domains, such as nursing homes, meeting videos, and bank monitoring. Researchers have created a video event ontology for surveillance, leading to its use in scenarios like bank and car park monitoring. While ontologies provide common terms for event definitions, scene interpretation may involve individually preferred algorithms, like rule-based systems and finite-state machines, which may share limitations with logical-based methods.

3. Method of the systematic review

The SLR method used in this study is based on the approach proposed in [52], which involves several stages depicted in Fig. 5.

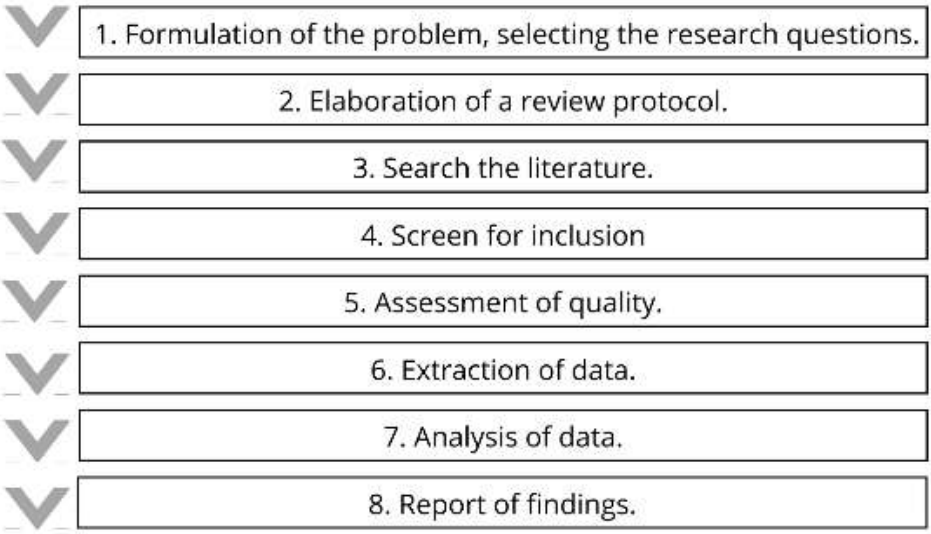


Fig. 5. Schematic representation of the methodology followed on the review, created by the authors

The research process in this study involved several stages:

1. Establishment of Research Questions:
   1.1. RQ1. What research has been conducted on acknowledging events from single-person or non-interacting behavior?
   1.2. RQ2. What research has been conducted on the recognition of events involving multiple-person interactions?
   1.3. RQ3. What research has been conducted on the recognition of events involving people-object interactions?
2. Definition of the Study Plan:

   The plan included determining information sources, inclusion criteria, search strategies, quality assessment criteria, screening procedures, and strategies for data extraction, synthesis, and reporting. The selected digital libraries were ACM, IEEE, Elsevier, and Springer, as they are prominent in the computing field and accessible to the authors. Inclusion criteria covered publications from the last ten years, containing the specified keywords in the title, abstract, or complete document, while reviews or surveys were excluded. The search string used was:

   ("event" OR "behavior" OR "action" OR "activity" OR "interaction") AND
   ("recognition" OR "detection" OR "tracking") AND

("smart classroom" OR "classroom") AND ("video" OR "vision").

3. Searching for Relevant Papers:
   The search string was adapted for each source, and relevant papers were sought in the selected digital libraries.
4. Screening and Selection of Papers:
   A two-step process was applied, involving the review of titles and abstracts for inclusion and a full-text review of selected papers.
5. Quality Assessment:
   While quality assessment is important for reviews aiming for generalization, it was not used as a criterion to exclude papers in this study, which sought to discover studies at different quality levels for a more comprehensive overview.
6. Data Extraction:
   Relevant data for answering the research questions was extracted from the selected papers.
7. Analysis
   The gathered data was thoroughly analyzed to draw meaningful conclusions.
8. Reporting:
   The collected data was analyzed, and the resulting findings were comprehensively reported in this paper.

4. Results of the review

4.1 Quantitative results

This section presents the results of the SLR conducted in this research, as shown in Fig. 6.

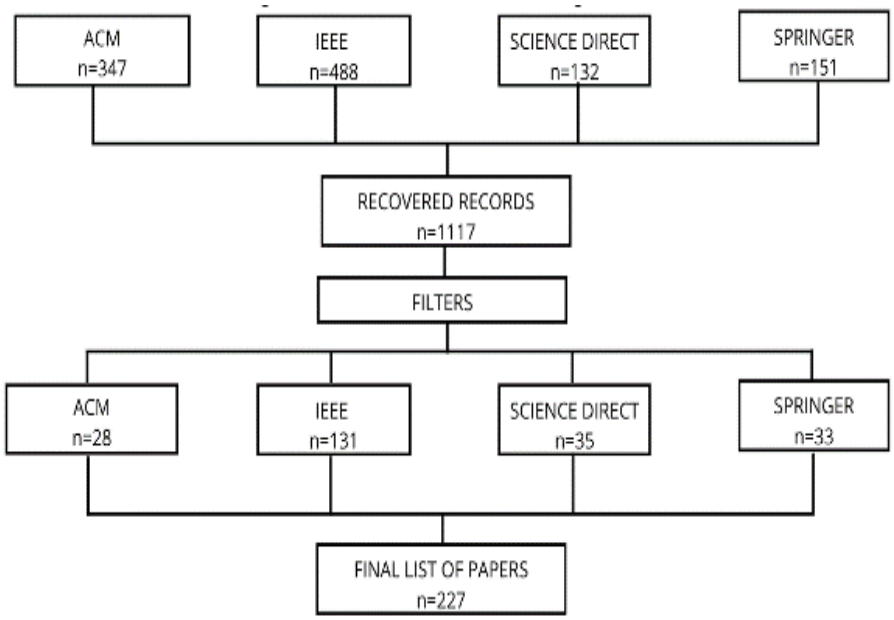


Fig. 6. Schematic representation of the process of selection of documents, created by the authors

Out of 1,117 documents initially selected, 227 papers met the inclusion-exclusion criteria after thorough reviews of titles, abstracts, and content (fig. 7). The list of papers remained unchanged throughout stages 5 and 6.

The papers were grouped by publication year and database. Over the years, the number of papers increased, showing growing interest in the field. Among the 227 documents (fig. 8), IEEE had the most papers (131, 57.7%), followed by Elsevier (35, 15.4%), SPRINGER (33, 14.6%), and ACM (28, 12.3%).

Qualitative analysis results presented by research question.

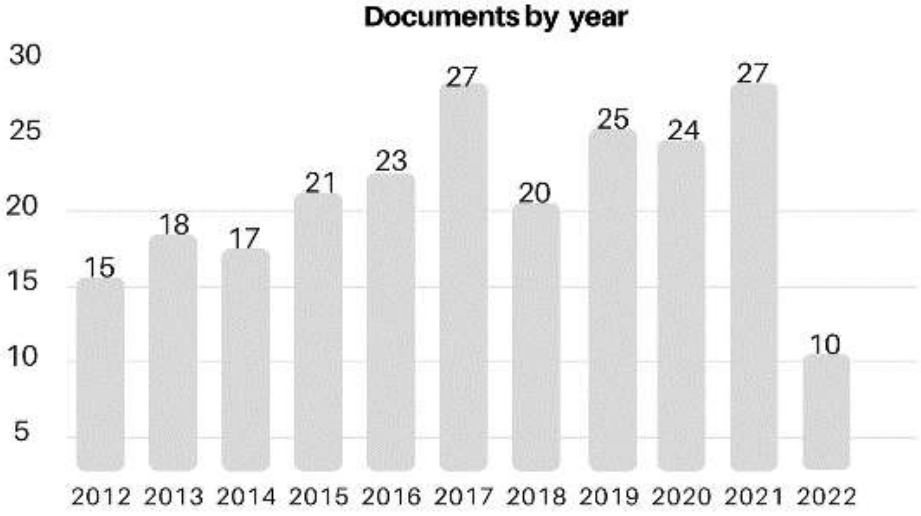


Fig. 7. A total of 227 articles with the search of the keywords, created by the authors

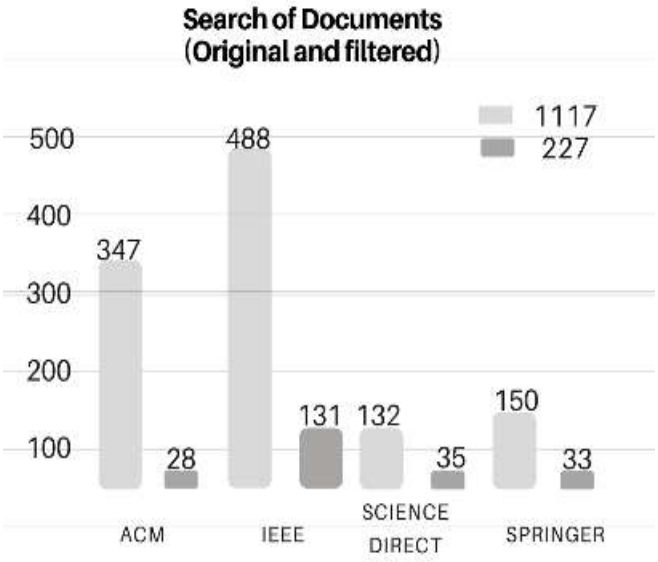


Fig. 8. Total of reviewed articles, by source, created by the authors

4.2 RQ1: Single person or non-interaction events

4.2.1 Event 1. Students being distracted from learning

In the context of educational settings, detecting events related to single person or non-interaction scenarios, such as students being distracted from learning, has been explored using various methods.

Gesture analysis has been utilized to identify boredom and lack of attention in students [92]. Additionally, facial expressions have been studied as indicators of students' feelings, and methods like image recognition and facial muscle tension measurement have been employed to capture facial expressions [93-94, 96].

Eye-gaze and face-gaze analysis have also proven to be important indicators of cognitive engagement among students [97-98]. Researchers have recorded and analyzed human gaze behavior in different scenarios, including conversational gaze and tutoring interactions [98-99, 101-103].

Pose estimation methods have been applied to detect self-absorbed or sleeping students [106-108]. These methods often involve probabilistic and compositional graphical models, but they may encounter challenges in handling errors arising from small body parts in still images [107]. Video pose estimation methods, which incorporate motion information, have been used as well [109]. However, they may have limitations in handling action datasets with larger human motion and appearance variations due to viewpoint changes.

4.2.2 Event 2. Detection of behavior related to developmental disorders

Developmental disorders such as autism and attention disorders, like ADHD, can be detected in the classroom using various computer-based methods.

For autism detection, eye-tracking from computer searching tasks has been employed as an easier, cheaper, and less-obtrusive alternative to fMRI data recording [110-113].

Regarding ADHD diagnosis, facial expression analysis has been a focus of some research works [114-117]. For instance, [114] proposed a methodology using RGBD sensors for diagnostic predictions of ADHD and ASD. Depth capturing cameras, like Microsoft Kinect, have been used to monitor the movement of children in a classroom setting [115]. These cameras allow tracking and analysis of head motion and velocity profiles to measure hyperactivity. Additionally, computerized continuous performance tests are conducted to measure inattention and impulsivity. The test results are then compared to norm data, generating reports for assessment by clinicians.

Overall, these computer-based approaches offer promising avenues for early detection and intervention for developmental disorders in educational environments.

4.2.3 Event 3. Hand-raising gesture detection

Hand-raising is a behavior studied in gaming, Human-Computer Interaction, and classroom settings [119-121]. Detecting hand-raising in a real classroom can be challenging, but vision-based models using video cameras [122-123] and Kinect [124-126] have been developed to address this. Hand gesture recognition involves tracking, representation, and conversion into meaningful commands for human-computer interaction. Techniques include contact-based and vision-based devices [127-129]. Hand gesture recognition relies on detection, tracking, and recognition using visual features like skin color, shape, and motion [131-132]. Model-based methods use tracking to enhance robustness [133-135].

Vision-based hand gesture recognition includes static and dynamic gestures, using classifiers like Hidden Markov Models [137-139]. Learning algorithms vary based on gesture representation, including supervised, unsupervised, and reinforcement learning [122, 140]. For example, static hand gestures are recognized using the Fourier descriptor of a segmentation image [142].

4.3 RQ2. Multiple-person interactions

4.3.1 Event 1. Speaking and talking

Detecting human speaking is important for Human-Computer Interaction and fatigue detection [143]. Lip movement is used to detect speaking, and video-based approaches have been proposed [145]. Methods like lip motion analysis [146-148], Viola-Jones with skin color pixel detection [149], skin-color segmentation with edge projection [150], and fuzzy c-means clustering [151] have been used for lip detection and speech recognition. Feature extraction methods like Log-polar Signature [153] and Haar-like wavelets [154] have been proposed for lip tracking and speech recognition [157].

4.3.2 Event 2. Social interactions

Video-based studies of human sociality focus on workplace settings and classrooms, observing action and sense-making practices in social interactions [158-160]. Social abilities have been linked to academic success, and Proxemics Theory is used to detect human relationships, including non-verbal relations in classrooms [163-165]. Immediacy, which enhances physical and psychological closeness between individuals [168-170], can impact effective communication in educational settings. Teachers' variable physical proximities with students foster effective communication in classrooms [172-173]. Interaction, where learners share perspectives and collaborate, is another important aspect of non-verbal behaviors [174-175]. Learner-centered approaches and collaborative learning are emphasized in education [176], and providing pre-service teachers with video scenes where students interact with each other can support their understanding of these approaches [178]. However, empirical research is needed to validate assumptions regarding video-based cases and student-student interactions in educational settings [177, 180].

4.4 RQ3. People-objects interaction

4.4.1 Unique event. Student engagement detected by interaction with objects

Various works have classified engagement in different ways [181], including student involvement in terms of effort, persistence, and concentration [179], emotional engagement related to feelings of interest or attitude, and cognitive engagement focusing on cognitive effort and strategies [182]. Agentic engagement emphasizes proactive actions taken by students during learning tasks, involving interaction with surroundings or learning objects.

To assess the level of engagement, traditional methods and measures have been introduced [183], such as using student responses as indicators in intelligent tutoring systems [184-185]. Facial movements and features extracted from them have been used [186-187], along with automated measures like response time to problems and quizzes [188-189]. Physiological and neurological measures like electroencephalogram, heart rate, and skin response have also been employed [190-193]. Some studies utilize facial features and SVM classifiers to analyze affective states of students while solving problems [194-195], while others focus on facial expressions and body movements to detect various affective states of engagement [196].

Engagement detection and localization can be performed using face and facial landmark positions in video frames [197], extracting features from small segments of video, and employing regression models or LSTM-based networks for engagement prediction [196]. Open-source utility software like OpenFace has been used to automatically track changes in body posture and facial movements to infer engagement levels through eye gaze and head movement features [198-201].

Several works have classified engagement in different ways [181]. For example, [179] explains student’s involvement in terms of effort, persistence and concentration. Emotional Engagement is related to feelings of interest or attitude towards a particular theme. Cognitive Engagement focuses on allocation of effort, a strategy used, in terms of cognitive effort, for the accomplishment of the

task. Other models have introduced another dimension known as Agentic Engagement and emphasize on proactive actions taken by the student for learning a particular task [182]. These tasks sometimes involve interaction of students with surroundings elements or learning objects.

5. Conclusion

The reviewed works show that there are relatively few studies dedicated to Smart Classroom (SC) event recognition [202-203]. While other smart environments like smart homes or smart offices have more extensive research, SC lacks conventions defining relevant events or behaviors [204]. In SC, event recognition is often a step within an application system, where it serves as input for decision-making processes aimed at assisting users [163].

Overall, video-based Human Event Recognition (HER) in SC has shown positive results, but some projects' costs may hinder widespread implementation [163]. Comparatively, other educational developments like E-learning, M-learning, and MOOCs have gained more traction, especially during the COVID-19 pandemic, but they may lack the non-verbal communication found in traditional classrooms [165]. HER has been suggested as a potential solution to address this limitation [6].

References

[1]. K. Ducatel, U. europeenne. Technologies de la societe de l'information, U. europeenne. Institut d'etudes de prospectives technologiques and U. europeenne. Societe de l'information conviviale, Scenarios for ambient intelligence in 2010 (2001).

[2]. D. J. Cook, J. C. Augusto, V. R. Jakkula. Ambient intelligence: Technologies, applications, and opportunities. Pervasive and Mobile Computing 5(4) (2009), 277–298.

[3]. M.K. Saini, N. Goel. How smart are smart classrooms? A review of smart classroom technologies. ACM Computing Surveys (CSUR) 52(6) (2019), 1–28. https://doi.org/10.1145/3365757.

[4]. D. Guinard, M. Fischer, V. Trifa. Sharing using social networks in a composable web of things. In 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops). IEEE, 2010, pp. 702–707. doi:10.1109/percomw.2010.5470524

[5]. V. Radosavljevic, S. Radosavljevic, G. Jelic. Ambient intelligence-based smart classroom model, Interactive Learning Environments (2019). 1–15. https://doi.org/10.1080/10494820.2019.1652836.

[6]. M. Kwet and P. Prinsloo, The 'smart'classroom: a new frontier in the age of the smart university, Teaching in Higher Education 25(4) (2020). 510–526. https://doi.org/10.1080/13562517.2020.1734922.

[7]. J. Candamo, M. Shreve, D.B. Goldgof, D.B. Sapper, R. Kasturi. Understanding transit scenes: A survey on human behavior-recognition algorithms. IEEE transactions on intelligent transportation systems 11(1) (2009), 206–224. 10.1109/TITS.2009.2030963.

[8]. M.A.R. Ahad. Vision and Sensor-Based Human Activity Recognition: Challenges Ahead. In Advancements in Instrumentation and Control in Applied System Applications. IGI Global, 2020, pp. 17–35. DOI:10.4018/978-1-7998-2584-5.

[9]. D.R. Beddiar, B. Nini, M. Sabokrou, A. Hadid. Vision based human activity recognition: a survey. Multimedia Tools and Applications 79(41) (2020), 30509–30555.

[10]. H.-B. Zhang, Y.-X., B. Zhong, Q. Lei, L. Yang, J.-X. Du, D.-S. Chen. A comprehensive survey of vision based human action recognition methods. Sensors 19(5) (2019), 1005.

[11]. Y. Kong, Y. Fu. Human action recognition and prediction: A survey. arXiv preprint arXiv:1806.11230 (2018).

[12]. R.K. Tripathi, A.S. Jalal, S.C. Agrawal. Suspicious human activity recognition: a review. Artificial Intelligence Review 50(2) (2018), 283–339. https://doi.org/10.1007/s10462-017-9545-7.

[13]. I. Jegham, A.B. Khalifa, I. Alouani, M.A. Mahjoub. Vision-based human action recognition: An overview and real world challenges. Forensic Science International: Digital Investigation 32 (2020), 200901. https://doi.org/10.1016/j.fsidi.2019.200901.

[14]. F. Paredes-Valles, K.Y. Scheper, G.C. De Croon. Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception. IEEE transactions on pattern analysis and machine intelligence 42(8) (2019), 2051–2064.

[15]. B. Garcia-Garcia, T. Bouwmans, A.J.R. Silva. Background subtraction in real applications: Challenges, current models and future directions. Computer Science Review 35 (2020), 100204.

[16]. M. Ahmad, I. Ahmed, K. Ullah, I. Khan, A. Adnan. Robust background subtraction based person's counting from overhead view. In 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE, 2018, pp. 746–752.

[17]. Y. Fan, G. Wen, D. Li, S. Qiu, M.D. Levine, F. Xiao. Video anomaly detection and localization via Gaussian mixture fully convolutional variational autoencoder. Computer Vision and Image Understanding (2020), 102920. https://doi.org/10.1016/j.cviu.2020.102920.

[18]. J. Cho, Y. Jung, D.-S. Kim, S. Lee, Y. Jung. Moving object detection based on optical flow estimation and a Gaussian mixture model for advanced driver assistance systems. Sensors 19(14) (2019), 3217.

[19]. H. Yang, S. Qu. Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition. IET Intelligent Transport Systems 12(1) (2018), 75-85.

[20]. I. Martins, P. Carvalho, L. Corte-Real, J.L. Alba-Castro. Bmog: boosted gaussian mixture model with controlled complexity for background subtraction. Pattern Analysis and Applications 21(3) (2018), 641–654.

[21]. N. Zerrouki, F. Harrou, Y. Sun, A. Houacine. Visionbased human action classification using adaptive boosting algorithm. IEEE Sensors Journal 18(12) (2018), 5115–5121.

[22]. L. Boregowda, A. Rajagopal. Object classification in video data. Google Patents, 2007, US Patent App.11/227,505.

[23]. N.D. Bird, O. Masoud, N.P. Papanikolopoulos, A. Isaacs. Detection of loitering individuals in public transportation areas. IEEE Transactions on intelligent transportation systems 6(2) (2005), 167–177. 10.1109/TITS.2005.848370.

[24]. R. Zhang, C. Vogler, D. Metaxas. Human gait recognition at sagittal plane. Image and vision computing 25(3) (2007), 321–330. https://doi.org/10.1016/j.imavis.2005.10.007.

[25]. V. Kukreja, D. Kumar, A. Kaur. Deep learning in Human Gait Recognition: An Overview. In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), IEEE, 2021, pp. 9–13.

[26]. U.Sevik, E. Karakullukcu, T. Berber, Y. Akbas, S. Turkyılmaz. Automatic classification of skin burn colour images using texture-based feature extraction. IET Image Processing 13(11) (2019), 2018–2028.

[27]. M.J. Taylor and T. Morris, Adaptive skin segmentation via feature-based face detection. In Real-Time Image and Video Processing 2014, Vol. 9139, International Society for Optics and Photonics, 2014, p. 91390P. https://doi.org/10.1117/12.2052003.

[28]. S.A. Rodriguez, V. Fremont, P. Bonnifait, V. Cherfaoui. An embedded multi-modal system for object localization and tracking. IEEE Intelligent Transportation Systems Magazine 4(4) (2012), 42–53. 10.1109/MITS.2012.2217855.

[29]. Y. Lu, C. Lu, C.-K. Tang. Online video object detection using association LSTM. In Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2344–2352. 10.1109/ICCV.2017.257.

[30]. R. Messing, C. Pal, H. Kautz. Activity recognition using the velocity histories of tracked keypoints. In 2009 IEEE 12th international conference on computer vision, IEEE, 2009, pp. 104–111. 10.1109/ICCV.2009.5459154.

[31]. I. Leichter, M. Lindenbaum, E. Rivlin. Mean shift tracking with multiple reference color histograms. Computer Vision and Image Understanding 114(3) (2010), 400–408. https://doi.org/10.1016/j.cviu.2009.12.006.

[32]. Y. Wu, J. Lim, M.-H. Yang. Online object tracking: A benchmark. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2013, pp. 2411–2418. 10.1109/TPAMI.2014.2388226.

[33]. E. Alpaydin. Introduction to machine learning. MIT press, 2020. https://doi.org/10.1016/j.neuroimage.2010.11.004.

[34]. Y. Li. On incremental and robust subspace learning. Pattern recognition 37(7) (2004), 1509–1518. https://doi.org/10.1016/j.patcog.2003.11.010.

[35]. F. Mairesse, M. Gasic, F. Jurcicek, S. Keizer, B. Thomson, K. Yu, S. Young. Spoken language understanding from unaligned data using discriminative classification models. In 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2009, pp. 4749–4752. 10.1109/ICASSP.2009.4960692.

[36]. K.Z. Haigh, L.M. Kiff, G. Ho. The independent lifestyle assistant: Lessons learned. Assistive Technology 18(1) (2006), 87–106. https://doi.org/10.1080/10400435.2006.10131909.

[37]. L. Chen, I. Khalil. Activity recognition: Approaches, practices and trends. In Activity Recognition in Pervasive Intelligent Environments, Springer, 2011, pp. 1–31. https://doi.org/10.2991/978-94-91216-05-3_1.
[38]. L. Chen, C.D. Nugent. Human Activity Recognition and Behaviour Analysis. Springer, 2019.
[39]. B. Bouchard, S. Giroux, A. Bouzouane. A smart home agent for plan recognition of cognitively-impaired patients. JCP 1(5) (2006), 53–62.
[40]. L. Chen, C.D. Nugent, H. Wang. A knowledge-driven approach to activity recognition in smart homes. IEEE Transactions on Knowledge and Data Engineering 24(6) (2011), 961–974. 10.1109/TKDE.2011.51.
[41]. R.L. Granada, R.F. Pereira, J. Monteiro, R.C. Barros, D. Ruiz, F. Meneguzzi. Hybrid activity and plan recognition for video streams. In Workshops at the Thirty-First AAAI Conference on Artificial Intelligence, 2017.
[42]. H. Kautz, O. Etzioni, D. Fox, D. Weld, L. Shastri. Foundations of assisted cognition systems. University of Washington, Computer Science Department, Technical Report, Tech. Rep (2003).
[43]. E.S. Chen, G.B. Melton, M.E. Engelstad, I.N. Sarkar. Standardizing clinical document names using the HL7/LOINC document ontology and LOINC codes. In AMIA Annual Symposium Proceedings, Vol. 2010, American Medical Informatics Association, 2010, p. 101.
[44]. A. Hakeem, M. Shah. Multiple agent event detection and representation in videos. In AAAI, 2005, pp. 89–94.
[45]. B. Georis, M. Maziere, F. Bremond, M. Thonnat. A video interpretation platform applied to bank agency monitoring (2004). 10.1049/ic:20040097.
[46]. J.C. SanMiguel, J.M. Martinez, A. Garcia. An ontology for event detection and its application in surveillance video. In 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, IEEE, 2009, pp. 220–225. 10.1109/AVSS.2009.28.
[47]. A.R. Francois, R. Nevatia, J. Hobbs, R.C. Bolles, J.R. Smith. VERL: an ontology framework for representing and annotating video events. IEEE multimedia 12(4) (2005), 76–86. 10.1109/AVSS.2009.28.
[48]. U. Akdemir, P. Turaga, R. Chellappa. An ontology based approach for activity recognition from video. In Proceedings of the 16th ACM international conference on Multimedia, 2008, pp. 709–712. https://doi.org/10.1145/1459359.1459466.
[49]. B. Yao, H. Hagras, M.J. Alhaddad, D. Alghazzawi. A fuzzy logic-based system for the automation of human behavior recognition using machine vision in intelligent environments. Soft Computing 19(2) (2015), 499–506. DOI10.1007/s00500-014-1270-4.
[50]. N. Ikizler, D. Forsyth. Searching video for complex activities with finite state models. In 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–8.
[51]. L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari, G. Serra. Event detection and recognition for semantic annotation of video. Multimedia tools and applications 51(1) (2011), 279–302. http://hdl.handle.net/11380/979935.
[52]. Y. Xiao, M. Watson. Guidance on conducting a systematic literature review. Journal of Planning Education and Research 39(1) (2019), 93–112.
[53]. W. Hu, T. Tan, L. Wang, S. Maybank. A survey on visual surveillance of object motion and behaviors. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 34(3) (2004), 334–352. 10.1109/TSMCC.2004.829274.
[54]. M. Valera, S.A. Velastin. Intelligent distributed surveillance systems: a review. IEE Proceedings-Vision, Image and Signal Processing 152(2) (2005), 192–204. 10.1049/ip-vis:20041147.
[55]. P. Turaga, R. Chellappa, V.S. Subrahmanian, O. Udrea. Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video technology 18(11) (2008), 1473–1488. 10.1109/TCSVT.2008.2005594.
[56]. R. Poppe. A survey on vision-based human action recognition. Image and vision computing 28(6) (2010), 976–990. https://doi.org/10.1016/j.imavis.2009.11.014.
[57]. J.K. Aggarwal, L. Xia. Human activity recognition from 3d data: A review. Pattern Recognition Letters 48 (2014), 70– 80. https://doi.org/10.1016/j.patrec.2014.04.011.
[58]. J.K. Aggarwal, M.S. Ryoo. Human activity analysis: A review. ACM Computing Surveys (CSUR) 43(3) (2011), 1–43. https://doi.org/10.1145/1922649.1922653.
[59]. A.A. Chaaraoui, P. Climent-Perez, F. Florez-Revuelta. A review on vision techniques applied to human behaviour analysis for ambient-assisted living. Expert Systems with Applications 39(12) (2012), 10873–10888. https://doi.org/10.1016/j.eswa.2012.03.005.

[60]. O.P. Popoola, K. Wang. Video-based abnormal human behavior recognition—A review. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42(6) (2012), 865–878. 10.1109/TSMCC.2011.2178594.

[61]. A.A. Sodemann, M.P. Ross, B.J. Borghetti. A review of anomaly detection in automated surveillance. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42(6) (2012), 1257–1272. 10.1109/TSMCC.2012.2215319.

[62]. P.V.K. Borges, N. Conci, A. Cavallaro. Video-based human behavior understanding: A survey. IEEE transactions on circuits and systems for video technology 23(11) (2013),1993–2008. 10.1109/TCSVT.2013.2270402.

[63]. J.M. Chaquet, E.J. Carmona, A. Fernandez-Caballero. A survey of video datasets for human action and activity recognition. Computer Vision and Image Understanding 117(6) (2013), 633–659. https://doi.org/10.1016/j.cviu.2013.01.013.

[64]. S. Vishwakarma, A. Agrawal. A survey on activity recognition and behavior understanding in video surveillance. The Visual Computer 29(10) (2013), 983–1009. https://doi.org/10.1007/s00371-012-0752-6.

[65]. G. Guo, A. Lai. A survey on still image based human action recognition. Pattern Recognition 47(10) (2014), 3343–3361. https://doi.org/10.1016/j.patcog.2014.04.018.

[66]. S.A. Lowe, G. OLaighin. Monitoring human health behaviour in one's living environment: a technological review. Medical engineering & physics 36(2) (2014), 147–168. https://doi.org/10.1016/j.medengphy.2013.11.010.

[67]. M. Amiribesheli, A. Benmansour, A. Bouchachia. A review of smart homes in healthcare. Journal of Ambient Intelligence and Humanized Computing 6(4) (2015), 495–517. https://doi.org/10.1007/s12652-015-0270-2.

[68]. P. Hurney, P. Waldron, F. Morgan, E. Jones, M. Glavin. Review of pedestrian detection techniques in automotive farinfrared video. IET intelligent transport systems 9(8) (2015), 824–832. DOI:10.1049/iet-its.2014.0236.

[69]. M. Ziaeefard, R. Bergevin. Semantic human activity recognition: A literature review. Pattern Recognition 48(8) (2015), 2329–2345. https://doi.org/10.1016/j.patcog.2015.03.006.

[70]. M. Ramezani, F. Yaghmaee. A review on human action analysis in videos for retrieval applications. Artificial Intelligence Review 46(4) (2016), 485–514. DOI10.1007/s10462-016-9473-y.

[71]. T. Subetha, S. Chitrakala. A survey on human activity recognition from videos. In 2016 International Conference on Information Communication and Embedded Systems (ICICES), IEEE, 2016, pp. 1–7. 10.1109/ICICES.2016.7518920.

[72]. J. Mahata, A. Phadikar. Recent advances in human behaviour understanding: A survey. In 2017 Devices for Integrated Circuit (DevIC), IEEE, 2017, pp. 751–755. 10.1109/DEVIC.2017.8074052.

[73]. S. Rashmi, S. Bhat, V. Sushmitha, Evaluation of human action recognition techniques intended for video analytics. In 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), IEEE, 2017, pp. 357–362.0.1109/SmartTechCon.2017.8358396.

[74]. K. Rohit, K. Mistree, J. Lavji. A review on abnormal crowd behavior detection. In 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), IEEE, 2017, pp. 1–3. 10.1109/ICIIECS.2017.8275999.

[75]. A.B. Sargano, P. Angelov, Z. Habib. A comprehensive review on handcrafted and learning-based action representation approaches for human activity recognition, applied sciences 7(1) (2017), 110. https://doi.org/10.3390/app7010110.

[76]. M. Lussier, M. Lavoie, S. Giroux, C. Consel, M. Guay, J. Macoir, C. Hudon, D. Lorrain, L. Talbot, F. Langlois et al. Early detection of mild cognitive impairment with in-home monitoring sensor technologies using functional measures: A systematic review. IEEE journal of biomedical and health informatics 23(2) (2018), 838–847. 10.1109/JBHI.2018.2834317.

[77]. C. Dhiman, D.K. Vishwakarma. A review of state-of-theart techniques for abnormal human activity recognition. Engineering Applications of Artificial Intelligence 77 (2019), 21–45. https://doi.org/10.1016/j.engappai.2018.08.014.

[78]. M. Fahim, A. Sillitti, Anomaly Detection. Analysis and Prediction Techniques in IoT Environment: A Systematic Literature Review. IEEE Access 7 (2019), 81664–81681. 10.1109/ACCESS.2019.2921912.

[79]. R. Iguernaissi, D. Merad, K. Aziz, P. Drap. People tracking in multi-camera systems: a review. Multimedia Tools and Applications 78(8) (2019), 10773–10793. https://doi.org/10.1007/s11042-018-6638-5.

[80]. A. Lentzas, D. Vrakas. Non-intrusive human activity recognition and abnormal behavior detection on elderly people: a review. Artificial Intelligence Review (2019), 1–47. https://doi.org/10.1007/s10462-019-09724-5.

[81]. S. Nigam, R. Singh, A. Misra, A Review of Computational Approaches for Human Behavior Detection. Archives of Computational Methods in Engineering 26(4) (2019), 831–863. https://doi.org/10.1007/s11831-018-9270-7.

[82]. T. Sikandar, K.H. Ghazali, M.F. Rabbi, ATM crime detection using image processing integrated video surveillance: a systematic review. Multimedia Systems 25(3) (2019), 229–251. https://doi.org/10.1007/s00530-018-0599-4.

[83]. T. Singh, D.K. Vishwakarma. Video benchmarks of human action datasets: a review. Artificial Intelligence Review 52(2) (2019), 1107–1154. https://doi.org/10.1007/s10462-018-9651-1.

[84]. R.K. Tripathi, A.S. Jalal, S.C. Agrawal. Abandoned or removed object detection from visual surveillance: a review. Multimedia Tools and Applications 78(6) (2019), 7585–7620. https://doi.org/10.1007/s11042-018-6472-9.

[85]. J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu. Deep learning for sensor-based activity recognition: A survey. Pattern Recognition Letters 119 (2019), 3–11. https://doi.org/10.1016/j.patrec.2018.02.010.

[86]. U. Bakar, H. Ghayvat, S. Hasanm, S. Mukhopadhyay. Activity and anomaly detection in smart home: A survey. In Next Generation Sensors and Systems, Springer, 2016, pp. 191–220. https://doi.org/10.1007/978-3-319-21671-3_9.

[87]. C. Zhang, Q.-S. Jia. A review of occupant behavior models in residential building: Sensing, modeling, and prediction. In 2016 Chinese Control and Decision Conference (CCDC), IEEE, 2016, pp. 2032–2037. 10.1109/CCDC.2016.7531318.

[88]. A.S. Al-Shamayleh, R. Ahmad, M.A. Abushariah, K.A. Alam, N. Jomhari, A systematic literature review on vision based gesture recognition techniques. Multimedia Tools and Applications 77(21) (2018), 28121–28184. https://doi.org/10.1007/s11042-018-5971-z.

[89]. A.B. Mabrouk, E. Zagrouba. Abnormal behavior recognition for intelligent video surveillance systems: A review. Expert Systems with Applications 91 (2018), 480–491. https://doi.org/10.1016/j.eswa.2017.09.029.

[90]. G. Sreenu, M.S. Durai. Intelligent video surveillance: a review through deep learning techniques for crowd analysis. Journal of Big Data 6(1) (2019), 48. https://doi.org/10.1186/s40537-019-0212-5.

[91]. S.M. Al-Zoubi, M.A.B. Younes. Low academic achievement: causes and results, Theory and Practice. In Language Studies 5(11) (2015), 2262–2268. http://dx.doi.org/10.17507/tpls.0511.09.

[92]. H. Kim, S. Lee, Y. Kim, S. Lee, D. Lee, J. Ju, H. Myung. Weighted joint-based human behavior recognition algorithm using only depth information for low-cost intelligent video-surveillance system. Expert systems with Applications 45 (2016), 131–141. https://www.sciencedirect.com/science/article/abs/pii/S0957417415006648.

[93]. R.L. Mandryk, M.S. Atkins, A fuzzy physiological approach for continuously modeling emotion during interaction with play technologies. International journal of humancomputer studies 65(4) (2007), 329–347. https://doi.org/10.1016/j.ijhcs.2006.11.011.

[94]. K.D. Sidney, S.D. Craig, B. Gholson, S. Franklin, R. Picard, A.C. Graesser. Integrating affect sensors in an intelligent tutoring system. In Affective Interactions: The Computer in the Affective Loop Workshop at, 2005, pp. 7–13.

[95]. M. C. Su, C. T. Cheng, M. C. Chang, Y. Z. Hsieh (2021). A video analytic in-class student concentration monitoring system. IEEE Transactions on Consumer Electronics, 67(4), 294-304.

[96]. H. Admoni, B. Scassellati. Social eye gaze in humanrobot interaction: a review. Journal of Human-Robot Interaction 6(1) (2017), 25–63. https://doi.org/10.5898/JHRI.6.1.Admoni.

[97]. K. Jokinen, H. Furukawa, M. Nishida, S. Yamamoto. Gaze and turn-taking behavior in casual conversational interactions. ACM Transactions on Interactive Intelligent Systems (TiiS) 3(2) (2013), 1–30. https://doi.org/10.1145/2499474.2499481.

[98]. S. Andrist, B. Mutlu, A. Tapus. Look like me: matching robot personality via gaze to increase motivation. In Proceedings of the 33rd annual ACM conference on human factors in computing systems, 2015, pp. 3603–3612. https://doi.org/10.1145/2702123.2702592.

[99]. C.T. Ishi, C. Liu, H. Ishiguro, N. Hagita. Head motion during dialogue speech and nod timing control in humanoid robots. In 2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI), IEEE, 2010, pp. 293–300. 10.1109/HRI.2010.5453183.

[100]. K. Otsuka, Y. Takemae, J. Yamato. A probabilistic inference of multiparty-conversation structure based on Markovswitching models of gaze patterns, head directions, and utterances. In Proceedings of

the 7th international conference on Multimodal interfaces, 2005, pp. 191–198. https://doi.org/10.1145/1088463.1088497.

[101]. C.-M. Huang, B. Mutlu. Learning-based modeling of multimodal behaviors for humanlike robots. In 2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI), IEEE, 2014, pp. 57–64. https://doi.org/10.1145/2559636.2559668.

[102]. H. Admoni, A. Dragan, S.S. Srinivasa, B. Scassellati. Deliberate delays during robot-to-human handovers improve compliance with gaze communication. In Proceedings of the 2014 ACM/IEEE international conference on Humanrobot interaction, 2014, pp. 49–56. https://doi.org/10.1145/2559636.2559682

[103]. C. Rich, B. Ponsler, A. Holroyd, C.L. Sidner. Recognizing engagement in human-robot interaction. In 2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI), IEEE, 2010, pp. 375–382. 10.1109/HRI.2010.5453163.

[104]. K. Sakita, K. Ogawara, S. Murakami, K. Kawamura, K. Ikeuchi. Flexible cooperation between human and robot by interpreting human intention from gaze information. In 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), Vol. 1, IEEE, 2004, pp. 846–851. 10.1109/IROS.2004.1389458.

[105]. S. Andrist, B. Mutlu, M. Gleicher. Conversational gaze aversion for virtual agents. In International Workshop on Intelligent Virtual Agents, Springer, 2013, pp. 249–262. https://doi.org/10.1007/978-3-642-40415-3_22.

[106]. B. Xiaohan Nie, C. Xiong, S.-C. Zhu. Joint action recognition and pose estimation from video. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,2015, pp. 1293–1301.

[107]. D.C. Luvizon, D. Picard, H. Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5137–5146. arXiv:1802.09232v2.

[108]. C. Gao, S. Ye, H. Tian, Y. Yan (2021). Multi-scale single-stage pose detection with adaptive sample training in the classroom scene. Knowledge-Based Systems, 222, 107008.

[109]. Y. Yang, D. Ramanan. Articulated human detection with flexible mixtures of parts. IEEE transactions on pattern analysis and machine intelligence 35(12) (2012), 2878–2890.10.1109/TPAMI.2012.261.

[110]. A. Cherian, J. Mairal, K. Alahari, C. Schmid. Mixing body-part sequences for human pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2353–2360. 10.1109/CVPR.2014.302.

[111]. J. Kang, X. Han, J. Song, Z. Niu, X. Li (2020). The identification of children with autism spectrum disorder by SVM approach on EEG and eye-tracking data. Computers in biology and medicine, 120, 103722.

[112]. N. V. Valtakari, I. T. Hooge, C. Viktorsson, P. Nyström, T. Falck-Ytter, R. S. Hessels (2021). Eye tracking in human interaction: Possibilities and limitations. Behavior Research Methods, 1-17.

[113]. V. Yaneva, S. Eraslan, Y. Yesilada, R. Mitkov (2020). Detecting high-functioning autism in adults using eye tracking and machine learning. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 28(6), 1254-1261.

[114]. V. Yaneva, L.A. Ha, S. Eraslan, Y. Yesilada, R. Mitkov. Detecting autism based on eye-tracking data from web searching tasks. In Proceedings of the Internet of Accessible Things, 2018, pp. 1–10. https://doi.org/10.1145/3192714.3192819.

[115]. S. Jaiswal, M. F. Valstar, A. Gillott, & D. Daley (2017, May). Automatic detection of ADHD and ASD from expressive behaviour in RGBD data. In 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017) (pp. 762-769). IEEE.

[116]. R. Gao, K. Deng, & M. Xie (2022, October). Deep learning-assisted ADHD diagnosis. In Proceedings of the 3rd International Symposium on Artificial Intelligence for Medicine Sciences (pp. 142-147).

[117]. J. Singh, G. Goyal (2021). Decoding depressive disorder using computer vision. Multimedia Tools and Applications, 80, 8189-8212.

[118]. A. Hernandez-Vela, M. Reyes, L. Igual, J. Moya, V. Violant, S. Escalera. ADHD indicators modelling based on Dynamic Time Warping from RGBD data: A feasibility study. In VI CVC Workshop on the progress of Research & Development, Barcelona, Computer Vision Center, Citeseer, 2011, pp. 59–62.

[119]. S.S. Rautaray, A. Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. Artificial intelligence review 43(1) (2015), 1–54. 10.1007/s10462-012-9356-9.

[120]. R. C. Hsu, P. C. Su, J. L. Hsu, C. Y. Wang. (2020, October). Real-time interaction system of human-robot with hand gestures. In 2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE) (pp. 396-398). IEEE.

[121]. H.T. Binh, N.Q. Trung, H.-A.T. Nguyen, B.T. Duy. Detecting Student Engagement in Classrooms for Intelligent Tutoring Systems. In 2019 23rd International Computer Science and Engineering Conference (ICSEC), IEEE, pp. 145–149. 10.1109/ICSEC47112.2019.8974739.

[122]. C.-Y. Fang, M.-H. Kuo, G.-C. Lee, S.-W. Chen. Student gesture recognition system in classroom 2.0. In 14th IASTED International Conference on Computers and Advanced Technology in Education, CATE 2011, 2011, pp. 290–297. 10.2316/P.2011.734-010.

[123]. T.S. Nazare, M. Ponti. Hand-raising gesture detection with Lienhart-Maydt method in videoconference and distance learning. In Iberoamerican Congress on Pattern Recognition, Springer, 2013, pp. 512–519. https://doi.org/10.1007/978-3-642-41827-3_64.

[124]. B. Hariharan, S. Padmini, U. Gopalakrishnan. Gesture recognition using Kinect in a virtual classroom environment. In 2014 Fourth International Conference on Digital Information and Communication Technology and its Applications (DICTAP), IEEE, 2014, pp. 118–124. 10.1109/DICTAP.2014.6821668.

[125]. S. Salous, J. Newton, L. Leroy, S. Chendeb. Gestural Recognition by a Four Kinect Module in a CAVE "Le SAS". In RoCHI, 2015, pp. 111–114.

[126]. S. Kapgate, P. Sahu, M. Das, D. Gupta. (2022, May). Human Following Robot using Kinect in Embedded Platform. In 2022 1st International Conference on the Paradigm Shifts in Communication, Embedded Systems, Machine Learning and Signal Processing (PCEMS) (pp. 119-123). IEEE.

[127]. S. Corera, N. Krishnarajah. Capturing hand gesture movement: a survey on tools, techniques and logical considerations. Proceedings of chi sparks (2011).

[128]. V. A. Shanthakumar, C. Peng, J. Hansberger, L. Cao, S. Meacham, V. Blakely (2020). Design and evaluation of a hand gesture recognition approach for real-time interactions. Multimedia Tools and Applications, 79, 17707-17730.

[129]. T. H. Tsai, C. C. Huang, K. L. Zhang (2020). Design of hand gesture recognition system for human-computer interaction. Multimedia Tools and Applications, 79, 5989-6007.

[130]. K.H. Nguyen. Method and apparatus for real-time gesture recognition. Google Patents, 2000, US Patent 6,072,494.

[131]. M. Cote, P. Payeur, G. Comeau. Comparative study of adaptive segmentation techniques for gesture analysis in unconstrained environments. In Proceedings of the 2006 IEEE International Workshop on Imaging Systems and Techniques (IST 2006), IEEE, 2006, pp. 28–33. 10.1109/IST.2006.1650770.

[132]. O. Köpüklü, A. Gunduz, N. Kose, G. Rigoll (2019, May). Real-time hand gesture detection and classification using convolutional neural networks. In 2019 14th IEEE international conference on automatic face & gesture recognition (FG 2019) (pp. 1-8). IEEE.

[133]. D. Liu, L. Zhang, T. Luo, L. Tao, Y. Wu (2020). Towards interpretable and robust hand detection via pixel-wise prediction. Pattern Recognition, 105, 107202.

[134]. Z. Sun, J. Chen, M. Mukherjee, C. Liang, W. Ruan, Z. Pan (2022). Online multiple object tracking based on fusing global and partial features. Neurocomputing, 470, 190-203.

[135]. L. Huang, B. Zhang, Z. Guo, Y. Xiao, Z. Cao, J. Yuan (2021). Survey on depth and RGB image-based 3D hand shape and pose estimation. Virtual Reality & Intelligent Hardware, 3(3), 207-234.

[136]. H. Tang, H. Liu, W. Xiao, N. Sebe (2019). Fast and robust dynamic hand gesture recognition via key frames extraction and feature fusion. Neurocomputing, 331, 424-433.

[137]. T. Song, H. Zhao, Z. Liu, H. Liu, Y. Hu, D. Sun (2021). Intelligent human hand gesture recognition by local-global fusing quality-aware features. Future Generation Computer Systems, 115, 298-303.

[138]. T. L. Dang, S. D. Tran, T. H. Nguyen, S. Kim, N. Monet (2022). An improved hand gesture recognition system using keypoints and hand bounding boxes. Array, 16, 100251.

[139]. N. Aloysius, M. Geetha (2020). Understanding vision-based continuous sign language recognition. Multimedia Tools and Applications, 79(31-32), 22177-22209.

[140]. S. U. Amin, M. Alsulaiman, G. Muhammad, M. A. Mekhtiche, M. S. Hossain (2019). Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion. Future Generation computer systems, 101, 542-554.

[141]. Z. Li, D. Lin, X. Tang. Nonparametric discriminant analysis for face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(4) (2009), 755–761. 10.1109/TPAMI.2008.174.

[142]. Y. Ren, F. Zhang. Hand gesture recognition based on MEB-SVM. In 2009 International Conference on Embedded Software and Systems, IEEE, 2009, pp. 344–349. 10.1109/ICESS.2009.21.

[143]. S. Afroze, M. M. Hoque (2020, December). Towards lip motion based speaking mode detection using residual neural networks. In International Conference on Soft Computing and Pattern Recognition (pp. 166-175). Cham: Springer International Publishing.
[144]. S. Afroze, M. M. Hoque. Talking vs Non-Talking: A Vision Based Approach to Detect Human Speaking Mode. In 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), IEEE, 2019, pp. 1–6.
[145]. A. Wuerkaixi, Y. Zhang, Z. Duan, C. Zhang (2022, August). Rethinking audio-visual synchronization for active speaker detection. In 2022 IEEE 32nd International Workshop on Machine Learning for Signal Processing (MLSP) (pp. 01-06). IEEE.
[146]. F. Haider, S. Al Moubayed. Towards speaker detection using lips movements for human machine multiparty dialogue. FONETIK 2012 (2012), 117.
[147]. W-N. Lie, H.-C. Hsieh. Lips detection by morphological image processing. In ICSP'98. 1998 Fourth International Conference on Signal Processing (Cat. No. 98TH8344), Vol. 2, IEEE, 1998, pp. 1084–1087.
[148]. M. Bendris, D. Charlet, G. Chollet, Lip activity detection for talking faces classification in TV-Content. In International Conference on Machine Vision, 2010, pp. 187–190.
[149]. I. Khan, H. Abdullah, M.S.B. Zainal. Efficient eyes and mouth detection algorithm using combination of Viola Jones and skin color pixel detection. International Journal of Engineering 3(4) (2013), 8269.
[150]. H.-Y. Huang, Y.-C. Lin. An efficient mouth detection based on face localization and edge projection. International Journal of Computer Theory and Engineering 5(3) (2013),514.
[151]. T. Azim, M.A. Jaffar, A.M. Mirza, Fully automated real time fatigue detection of drivers through fuzzy expert systems. Applied Soft Computing 18 (2014), 25–38.
[152]. R. Navarathna, P. Lucey, D. Dean, C. Fookes, S. Sridharan. Lip detection for audio-visual speech recognition in-car environment. In 10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010), IEEE, 2010, pp. 598–601.
[153]. N. Eveno, A. Caplier, P.-Y. Coulon. Accurate and quasiautomatic lip tracking, IEEE Transactions on circuits and systems for video technology 14(5) (2004), 706–715.
[154]. C. Bouvier, A. Benoit, A. Caplier, P.-Y. Coulon. Open or closed mouth state detection: static supervised classification based on log-polar signature. In International Conference on Advanced Concepts for Intelligent Vision Systems, Springer, 2008, pp. 1093–1102.
[155]. K. Saenko, K. Livescu, M. Siracusa, K. Wilson, J. Glass, T. Darrell. Visual speech recognition with loosely synchronized feature streams. In Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Vol. 2, IEEE, 2005, pp. 1424–1431.
[156]. M.I. Faraj, J. Bigun, Person verification by lip-motion. In 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06), IEEE, 2006, pp. 37–37.
[157]. P.D. Polur, G.E. Miller. Experiments with fast Fourier transform, linear predictive and cepstral coefficients in dysarthric speech recognition algorithms using hidden Markov model. IEEE Transactions on Neural Systems and Rehabilitation Engineering 13(4) (2005), 558–561
[158]. J. Katila, S. Raudaskoski, (2020). Interaction analysis as an embodied and interactive process: multimodal, co-operative, and intercorporeal ways of seeing video data as complementary professional visions. Human Studies, 43(3), 445-470.
[159]. W. Chen (2019). Knowledge-aware learning analytics for smart learning. Procedia Computer Science, 159, 1957-1965.
[160]. V. Herdel, A. Kuzminykh, A. Hildebrandt, J. R. Cauchard (2021, May). Drone in love: Emotional perception of facial expressions on flying robots. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-20).
[161]. Z. Kupper, F. Ramseyer, H. Hoffmann, S. Kalbermatten, W. Tschacher. Video-based quantification of body movement during social interaction indicates the severity of negative symptoms in patients with schizophrenia. Schizophrenia research 121(1–3) (2010), 90–100.
[162]. U. Kale. Levels of interaction and proximity: Content analysis of video-based classroom cases. The Internet and Higher Education 11(2) (2008), 119–128.
[163]. V.P. Richmond, J.C. Mccroskey, T. Mottet. Handbook of instructional communication: Rhetorical and relational perspectives. Routledge, 2015.
[164]. P. Pérez, P. Roose, Y. Cardinale, M. Dalmau, D. Masson, N. Couture (2020, November). Mobile proxemic application development for smart environments. In Proceedings of the 18th International Conference on Advances in Mobile Computing & Multimedia (pp. 94-103).

[165]. H. Kivrak, F. Cakmak, H. Kose, S. Yavuz (2021). Social navigation framework for assistive robots in human inhabited unknown environments. Engineering Science and Technology, an International Journal, 24(2), 284-298.
[166]. U. Maniscalco, P. Storniolo, A. Messina (2022). Bidirectional multi-modal signs of checking human-robot engagement and interaction. International Journal of Social Robotics, 14(5), 1295-1309.
[167]. J.S. Philpott. The relative contribution to meaning of verbal and nonverbal channels of communication: A meta-analysis. Unpublished master's thesis, University of Nebraska, Lincoln (1983).
[168]. A. Mehrabain. Some referants and measures of non-verbal behaviour, Behavioural Research Methods and Instrumentation 1 (1969), 213–217.
[169]. M. Girolami, F. Mavilia, F. Delmastro (2020). Sensing social interactions through BLE beacons and commercial mobile devices. Pervasive and Mobile Computing, 67, 101198.
[170]. R. Martínez-Maldonado, L. Yan, J. Deppeler, M. Phillips, D. Gašević (2022). Classroom analytics: Telling stories about learning spaces using sensor data. In Hybrid learning spaces (pp. 185-203). Cham: Springer International Publishing.
[171]. P.W. Miller. Nonverbal Communication. What Research Says to the Teacher., ERIC, 1988.
[172]. P.W. Miller. Body Language in the Classroom. Techniques: Connecting education and careers 80(8) (2005), 28–30.
[173]. Y. Wang, L. H. Lee, T. Braud, P. Hui (2022, July). Re-shaping Post-COVID-19 teaching and learning: A blueprint of virtual-physical blended classrooms in the metaverse era. In 2022 IEEE 42nd International Conference on Distributed Computing Systems Workshops (ICDCSW) (pp. 241-247). IEEE.
[174]. M.P. Driscoll. Psychology of learning for instruction. Needham, MA: Allyn & Bacon (2000).
[175]. S. Dhelim, H. Ning, F. Farha, L. Chen, L. Atzori, M. Daneshmand (2021). IoT-enabled social relationships meet artificial social intelligence. IEEE Internet of Things Journal, 8(24), 17817-17828.
[176]. C. Chin. Classroom interaction in science: Teacher questioning and feedback to students' responses, International Journal of Science Education 28(11) (2006), 1315–1346.
[177]. C.M. Reigeluth, J. Moore. Cognitive education and the cognitive domain, Instructional-design theories and models: A new paradigm of instructional theory 2 (1999), 51–68.
[178]. M. Teräs, J. Suoranta, H. Teräs, M. Curcher (2020). Post-Covid-19 education and education technology 'solutionism': A seller's market. Postdigital Science and Education, 2(3), 863-878.
[179]. J.A. Fredricks, P.C. Blumenfeld, A.H. Paris. School engagement: Potential of the concept, state of the evidence. Review of educational research 74(1) (2004), 59–109.
[180]. A. Nigam, R. Pasricha, T. Singh, P. Churi (2021). A systematic review on AI-based proctoring systems: Past, present and future. Education and Information Technologies, 26(5), 6421-6445.
[181]. A. Silvola, P. Näykki, A. Kaveri, H. Muukkonen. (2021). Expectations for supporting student engagement with learning analytics: An academic path perspective. Computers & Education, 168, 104192.
[182]. J. Reeve, C.-M. Tseng. Agency as a fourth aspect of students'engagement during learning activities, Contemporary Educational Psychology 36(4) (2011), 257–267.
[183]. K.J. Mach, M.C. Lemos, A.M. Meadow, C. Wyborn, N. Klenk, J.C. Arnott, ... G. Wong-Parodi (2020). Actionable knowledge and the art of engagement. Current Opinion in Environmental Sustainability, 42, 30-37.
[184]. K.R. Koedinger, J.R. Anderson, W.H. Hadley, M.A. Mark. Intelligent tutoring goes to school in the big city (1997).
[185]. L. Guo, D. Wang, F. Gu, Y. Li, Y. Wang, R. Zhou (2021). Evolution and trends in intelligent tutoring systems research: a multidisciplinary and scientometric view. Asia Pacific Education Review, 22(3), 441-461.
[186]. S.K. D'mello, A. Graesser. Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. User Modeling and User-Adapted Interaction 20(2) (2010), 147–187.
[187]. A. Behera, P. Matthew, A. Keidel, P. Vangorp, H. Fang, S. Canning (2020). Associating facial expressions and upper-body gestures with learning tasks for enhancing intelligent tutoring systems. International Journal of Artificial Intelligence in Education, 30, 236-270.
[188]. E. Joseph. Engagement tracing: using response times to model student disengagement. Artificial intelligence in education: Supporting learning through intelligent and socially informed technology 125 (2005), 88.

[189]. S. Li, S. P. Lajoie, J. Zheng, H. Wu, H. Cheng (2021). Automated detection of cognitive engagement to inform the art of staying engaged in problem-solving. Computers & Education, 163, 104114.
[190]. M. Chaouachi, C. Pierre, I. Jraidi, C. Frasson. Affect and mental engagement: Towards adaptability for intelligent. In Twenty-Third International FLAIRS Conference, 2010.
[191]. B.S. Goldberg, R.A. Sottilare, K.W. Brawner, H.K. Holden. Predicting learner engagement during well-defined and ill-defined computer-based intercultural interactions. In International Conference on Affective Computing and Intelligent Interaction, Springer, 2011, pp. 538–547.
[192]. X. Xiao, J. Wang. Understanding and detecting divided attention in mobile MOOC learning. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, 2017, pp. 2411–2415.
[193]. F. Alqahtani, S. Katsigiannis, N. Ramzan (2020). Using wearable physiological sensors for affect-aware intelligent tutoring systems. IEEE Sensors Journal, 21(3), 3366-3378.
[194]. J. Whitehill, Z. Serpell, Y.-C. Lin, A. Foster, J.R. Movellan. The faces of engagement: Automatic recognition of student engagementfrom facial expressions. IEEE Transactions on Affective Computing 5(1) (2014), 86–98.
[195]. C.R. Beal, R. Walles, I. Arroyo, B.P. Woolf. On-line tutoring for math achievement testing: A controlled evaluation. Journal of Interactive Online Learning 6(1) (2007), 43–55.
[196]. A. Kaur, A. Mustafa, L. Mehta, A. Dhall. Prediction and localization of student engagement in the wild. In 2018 Digital Image Computing: Techniques and Applications (DICTA), IEEE, 2018, pp. 1–8.
[197]. T. Baltrušaitis, P. Robinson, L.-P. Morency. Openface: an open-source facial behavior analysis toolkit. In 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2016, pp. 1–10.
[198]. B. Zhu, X. Lan, X. Guo, K. E. Barner, C. Boncelet. (2020, October). Multi-rate attention based GRU model for engagement prediction. In Proceedings of the 2020 International Conference on Multimodal Interaction (pp. 841-848).
[199]. Y. Y. Li, Y. P. Hung (2019, September). Feature fusion of face and body for engagement intensity detection. In 2019 IEEE international conference on image processing (ICIP) (pp. 3312-3316). IEEE.
[200]. V. Thong Huynh, S. H. Kim, G. S. Lee, H. J. Yang (2019, October). Engagement intensity prediction with Facial behavior features. In 2019 International Conference on Multimodal Interaction (pp. 567-571).
[201]. P. Demochkina, A. Savchenko (2022, September). Efficient Algorithms for Video-Based Engagement Prediction for a MOOC Course. In 2022 International Russian Automation Conference (RusAutoCon) (pp. 672-676). IEEE.
[202]. S. Wu (2021). Simulation of classroom student behavior recognition based on PSO-kNN algorithm and emotional image processing. Journal of Intelligent & Fuzzy Systems, 40(4), 7273-7283.
[203]. S. Chakraborty, R. Mondal, P. K. Singh, R. Sarkar, D. Bhattacharjee (2021). Transfer learning with fine tuning for human action recognition from still images. Multimedia Tools and Applications, 80, 20547-20578.
[204]. A. Nadeem, A. Jalal, K. Kim (2021). Automatic human posture estimation for sport activity recognition with robust body parts detection and entropy Markov model. Multimedia Tools and Applications, 80, 21465-21498.
[205]. I. Akhter, A. Jalal, K. Kim (2021, January). Pose estimation and detection for event recognition using Sense-Aware features and Adaboost classifier. In 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST) (pp. 500-505). IEEE.
[206]. N. Irvine, C. Nugent, S. Zhang, H. Wang, W. W. Ng (2019). Neural network ensembles for sensor-based human activity recognition within smart environments. Sensors, 20(1), 216.
[207]. Y. Y. Ghadi, I. Akhter, S. A. Alsuhibany, T. al Shloul, A. Jalal, K. Kim (2022). Multiple events detection using context-intelligence features. INTELLIGENT AUTOMATION AND SOFT COMPUTING, 34(3).
[208]. G. Mohmed, A. Lotfi, A. Pourabdollah (2020, June). Employing a deep convolutional neural network for human activity recognition based on binary ambient sensor data. In Proceedings of the 13th ACM international conference on pervasive technologies related to assistive environments (pp. 1-7).
[209]. Z. Ahmad, N. M. Khan (2019, September). Multidomain multimodal fusion for human action recognition using inertial sensors. In 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM) (pp. 429-434). IEEE.

[210]. M. Wang, Z. Yan, T. Wang, P. Cai, S. Gao, Y. Zeng, ..., X. Chen (2020). Gesture recognition using a bioinspired learning architecture that integrates visual data with somatosensory data from stretchable sensors. Nature Electronics, 3(9), 563-570.

[211]. M. Dewan, M. Murshed, F. Lin (2019). Engagement detection in online learning: a review. Smart Learning Environments, 6(1), 1-20.

[212]. Y. Du, R. G. Crespo, O. S. Martínez (2023). Human emotion recognition for enhanced performance evaluation in e-learning. Progress in Artificial Intelligence, 12(2), 199-211.

[213]. L. Wang, K. J. Yoon (2021). Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. IEEE transactions on pattern analysis and machine intelligence, 44(6), 3048-3068.

[214]. E. Baran, D. Alzoubi (2020). Human-centered design as a frame for transition to remote teaching during the COVID-19 pandemic. Journal of Technology and Teacher Education, 28(2), 365-372.

[215]. E. Fonseca, X. Favory, J. Pons, F. Font, X. Serra (2021). Fsd50k: an open dataset of human-labeled sound events. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 30, 829-852.

[216]. R. Zheng, F. Jiang, R. Shen (2020, May). Intelligent student behavior analysis system for real classrooms. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 9244-9248). IEEE.

[217]. C. Pabba, P. Kumar (2022). An intelligent system for monitoring students' engagement in large classroom teaching through facial expression recognition. Expert Systems, 39(1), e12839.

[218]. S. E. Ovur, H. Su, W. Qi, E. De Momi, G. Ferrigno (2021). Novel adaptive sensor fusion methodology for hand pose estimation with multileap motion. IEEE Transactions on Instrumentation and Measurement, 70, 1-8.

[219]. J. Xu, Z. Yu, B. Ni, J. Yang, X. Yang, W. Zhang (2020). Deep kinematics analysis for monocular 3d human pose estimation. In Proceedings of the IEEE/CVF Conference on computer vision and Pattern recognition (pp. 899-908).

[220]. P. Goldberg, Ö. Sümer, K. Stürmer, W. Wagner, R. Göllner, P. Gerjets, ..., U. Trautwein (2021). Attentive or not? Toward a machine learning approach to assessing students' visible engagement in classroom instruction. Educational Psychology Review, 33, 27-49.

[221]. T. Shou, C. Borchers, S. Karumbaiah, V. Aleven (2023). Optimizing parameters for accurate position data mining in diverse classrooms layouts.

[222]. A. Bigalke, L. Hansen, J. Diesel, C. Hennigs, P. Rostalski, M. P. Heinrich (2023). Anatomy-guided domain adaptation for 3D in-bed human pose estimation. Medical Image Analysis, 102887.

[223]. C. Luo, J. Zhang, J. Yu, C. W. Chen, S. Wang (2019). Real-time head pose estimation and face modeling from a depth image. IEEE Transactions on Multimedia, 21(10), 2473-2481.

[224]. G. D. Abowd, C. G. Atkeson, A. Feinstein, C. Hmelo, R. Kooper, S. Long, N. Sawhney, M. Tani, Teaching and learning as multimedia authoring: the classroom 2000 project. In Proceedings of the fourth ACM international conference on Multimedia, 1997, pp. 187–198.

[225]. L. Zhang, S. Lin. Research on the Design and Application of Intelligence Classroom Teaching Model with Rain Classroom Digital Support. In International Conference on Modern Educational Technology and Innovation and Entrepreneurship (ICMETIE 2020), Atlantis Press, 2020, pp. 368–373.

[226]. M. A. Mady, S. Baadel. Technology-Enabled Learning (TEL): YouTube as a Ubiquitous Learning Aid. Journal of Information & Knowledge Management 19(01) (2020), 2040007.

[227]. J. C. Augusto. Ambient intelligence: Opportunities and consequences of its use in smart classrooms. Innovation in Teaching and Learning in Information and Computer Sciences 8(2) (2009), 53–63.

[228]. I. Abdellatif. Towards A Novel Approach for Designing Smart Classrooms. In 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT), IEEE, 2019, pp. 280–284.

[229]. S. Jaiswal, A. Parmar, H. Singh, G. Rathee. Smart Classroom Automation (2018).

[230]. G. Basilaia, D. Kvavadze. Transition to online education in schools during a SARS-CoV-2 coronavirus (COVID-19) pandemic in Georgia. Pedagogical Research 5(4) (2020), 1–9.

[231]. A. Magnani. Human Action Recognition and Monitoring in Ambient Assisted Living Environments. PhD thesis, alma, 2020.

[232]. G. Cebrian, R. Palau, J. Mogas. The Smart Classroom as a means to the development of ESD methodologies. Sustainability 12(7) (2020), 3010.

[233]. L. Chen, J. Hoey, C. D. Nugent, D.J. Cook, Z. Yu. Sensor-based activity recognition. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42(6) (2012), 790–808.

[234]. B. Logan, J. Healey, M. Philipose, E.M. Tapia, S. Intille. A long-term evaluation of sensing modalities for activity recognition. In International conference on Ubiquitous computing, Springer, 2007, pp. 483–500.
[235]. E.M. Tapia, S.S. Intille, W. Haskell, K. Larson, J. Wright, A. King, R. Friedman. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In 2007 11th IEEE international symposium on wearable computers, IEEE, 2007, pp. 37–40.
[236]. M. Stikic, T. Huynh, K. Van Laerhoven, B. Schiele. ADL recognition based on the combination of RFID and accelerometer sensing. In 2008 second international conference on pervasive computing technologies for healthcare, IEEE, 2008, pp. 258–263.
[237]. N. Roy, A. Misra, D. Cook. Ambient and smartphone sensor assisted ADL recognition in multi-inhabitant smart environments. Journal of ambient intelligence and humanized computing 7(1) (2016), 1–19.
[238]. T. Diethe, N. Twomey, M. Kull, P. Flach, I. Craddock, Probabilistic sensor fusion for ambient assisted living. arXiv preprint arXiv:1702.01209 (2017).
[239]. A. Hermanis, R. Cacurs, K. Nesenbergs, M. Greitans, E. Syundyukov, L. Selavo. Wearable sensor system for human biomechanics monitoring. In Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks, 2016, pp. 247–248.
[240]. S. Jung, S. Hong, J. Kim, S. Lee, T. Hyeon, M. Lee, D.-H. Kim. Wearable fall detector using integrated sensors and energy devices. Scientific reports 5 (2015), 17081.
[241]. T.T. Um, V. Babakeshizadeh, D. Kuli´c. Exercise motion classification from large-scale wearable sensor data using convolutional neural networks. In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2017, pp. 2385–2390.
[242]. Z. Wang, S.Y. He, Y. Leung. Applying mobile phone data to travel behaviour research: A literature review. Travel Behaviour and Society 11 (2018), 141–155.
[243]. E. Dimitriadou, A. Lanitis. (2023). A critical evaluation, challenges, and future perspectives of using artificial intelligence and emerging technologies in smart classrooms. Smart Learning Environments, 10(1), 1-26.
[244]. M. Kwet, P. Prinsloo. (2020). The ‘smart’classroom: a new frontier in the age of the smart university. Teaching in Higher Education, 25(4), 510-526.
[245]. L. Chen, C. D. Nugent. (2019). Human activity recognition and behavior analysis. Springer International Publishing.
[246]. J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, R. Kasturi. (2009). Understanding transit scenes: A survey on human behavior-recognition algorithms. IEEE transactions on intelligent transportation systems, 11(1), 206-224.

Информация об авторах / Information about authors

Мария Луиза КОРДОБА-ТЛАКСКАЛЬТЕКО – лектор факультета статистики и информатики Университета Веракруса. Сфера научных интересов: формальные языки, совместные вычисления, искусственный интеллект.

María Luisa CÓRDOBA-TLAXCALTECO – Lecturer of the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Research interests: formal languages, collaborative computing, artificial intelligence.

Эдгард БЕНИТЕС-ГЕРРЕРО – профессор факультета статистики и информатики Университета Веракруса. Сфера научных интересов: человеко-машинное взаимодействие, искусственный интеллект, системы управления данными, совместные вычисления.

Edgard BENÍTEZ-GUERRERO – Professor of the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Research interests: human-Computer Interaction, artificial intelligence, data management systems, collaborative computing.

DOI: 10.15514/ISPRAS-2024-36(1)-12

Software Engineering Students, Soft and Hard Skills Got through a University Software Company

[1] *Jorge R. Aguilar Cisneros,* ORCID: 0000-0003-3040-157X <*jorge.aguilar@upaep.mx*>
[2] *Carlos A. Fernández-y-Fernández,* ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>
[1] *UPAEP Engineering Department, Puebla, México.*
[2] *UTM Computer Science Institute, Oaxaca, México.*

Abstract. This paper shows quantitative research regarding knowledge, soft & hard skills, and experience acquired by students hired by a University Software Development Company (USDC). Additionally, suggestions regarding how to set up a USDC in an academic environment, facing real customers, are shown. There have been good and bad experiences, both will be presented in this paper. Furthermore, students' perceptions will be discussed. To identify students' perceptions a questionnaire (survey) was applied. Its reliability was calculated through Cronbach's alpha coefficient (α =.89). Additionally, the Pearson correlation coefficient was calculated (r) in order to identify questions that should be deleted to increase the questionnaire's reliability. Outcomes could be useful when a software engineering faculty wishes to set up a USDC.

Keywords: software engineering students; soft skills; hard skills; student perceptions.

For citation: Aguilar-Cisneros J. R., Fernández-y-Fernández C. A. Software Engineering Students, Soft and Hard Skills Got through a University Software Company. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 199-208. DOI: 10.15514/ISPRAS-2024-36(1)-12.

Full text: Aguilar Cisneros J. R., Fernández-y-Fernández C. A. Soft and Hard Skills Gained by Students through Real Projects Developed at a University Software Company. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 599–609. DOI: 10.1134/S0361768823080029.

Личностные и технические навыки студентов-программистов, полученные в университетской компании по разработке программного обеспечения

[1] *Х. Р. Агилар Сиснерос,* ORCID: 0000-0003-3040-157X <*jorge.aguilar@upaep.mx*>
[2] *К. А. Фернандес-и-Фернандес,* ORCID: 0000-0002-1586-8772 <*caff@mixteco.utm.mx*>
[1] *Факультета инженерии, Народный автономный университет Пуэбла, Пуэбла, Мексика.*
[2] *Институт вычислений, Технологический университет Миштека, Уахуапан де Леон, Оахака, Мексика..*

Аннотация. В представленной статье описываются количественные исследования, касающиеся знаний, личностных и технических навыков и опыта, приобретенного студентами, нанятыми университетской компанией по разработке программного обеспечения (USDC). Показаны предложения по созданию компаний USDC в академической среде, работающих с реальными клиентами. В статье представлен ранее приобретенный опыт, как положительный, так и отрицательный. Описывается также восприятие сотрудничества с подобными компаниями учащимися. Для выявления восприятия учащихся был применен опрос и анкетирование участников. Надежность эксперимента вычислялась через альфа-коэффициент Кронбаха (α = .89). Кроме того, чтобы повысить надежность опроса и определить вопросы, которые лучше из него удалить, был рассчитан коэффициент корреляции Пирсона (r). Результаты могут быть полезны, если образовательная организация, ведущая обучения методам разработки программного обеспечения хочет создать USDC.

Ключевые слова: студенты программистских специальностей; личностные навыки; технические навыки; восприятие учащихся.

Для цитирования: Агилар-Сиснерос Х.Р., Фернандес-и-Фернандес К.А. Личностные и технические навыки студентов-программистов, полученные в университетской компании по разработке программного обеспечения. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 199–208 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–12.

Полный текст: Агилар Сиснерос Х.Р., Фернандес-и-Фернандес К.А. Личностные и технические навыки студентов, полученные при работе над реальными проектами в университетской компании по разработке программного обеспечения. Programming and Computer Software, 2023, т. 49, № 8, стр. 599–609 (на английском языке). DOI: 10.1134/S0361768823080029.

1. Introduction

Software system development requires a set of knowledge, hard & soft skills, experience, and values [1-2]. Traditionally, software Engineering (SE) theory and values are acquired at Universities. Experience and skills (reinforcement) are acquired through real developments. Nowadays, universities and industries follow different strategies to help students gain SE skills and professional experience. Among others,

i) Students' mobility from academia to industry [3],
ii) Internships [4],
iii) Cooperative Education (Co-Op) programs [5],
iv) Capstone projects [6],
v) Open Source contributions [7].

Additionally, there are some efforts like,

a) Companies spend resources to train new personnel, who are not ready for different SE roles [8],
b) Industry and Academic Collaboration programs,
c) Workshops and panels organized in research conferences [9],

d) Industry proposing a set of real challenges to be solved by students (hackathons).

SE teaching should be more practical than theoretical. SE programs would need a USDC (University Software Development Company) to carry out R&D activities.

It is recommendable to look for strategies to increase students' professional hard & soft skills before they finish their professional careers. In this sense, a USDC (Luminisoft) was created, more than ten years ago, at the Information and Communication Technology Faculty (ICTF) at the UPAEP, a university in México. This paper shows outcomes related to students' perception regarding LuminiSoft, and gives some advice to universities interested in replicating this model.

The paper is structured as follows. Section 2 provides a research background. Section 3 describes the research methodology. Section 4 presents the results of the study. In section 5, the results are discussed in more detail. Section 6 summarizes the key findings obtained. Finally, section 7 outlines areas for future work.

2. Research background

SE Education plays an important role in keeping students updated with software technologies, processes, and practices, unfortunately, some authors [10] point out gaps between Industry and Education [11]. A good way to understand some topics is by working in a practical way, or "learning by doing" [12].

There are a few interesting cases of this approach. For example, [13] describes using agile practices on a large-scale project. In addition, the University of Sheffield has a module, Genesys Solutions, where the students run their own software company [14].

In [15], try exposing SE students to the latest industry practice and research with an industry-academia team teaching a course. In [16], students create a consulting company where they work with a corporate sponsor on a project.

For undergraduate students, to teach proper SE they developed a "mock software company" to create a microcosm and teach real lessons about SE [17].

3. Research methodology

An online questionnaire (survey) was applied. The questionnaire was based on CLEI (Computer Laboratory Environment Inventory), and ACCC (Attitude towards Computing and Computing Courses Questionnaire) tools.

Table 1. Characteristics of SE education approaches

Paper	Business Environment	Active Learning	Practical Experience	Focus on Real-world Competencies	Emphasis on Teamwork
[8]	✓ (working with corporate sponsor)	✓	✓	✓	✓
[9]	✓ (simulated)	✓	✓	✓	✓
[7]	✓	✓	✓	✓	-
[5]	✓ (simulated)	✓	✓	-	-
[6]	✓ (students run own company)	✓	✓	✓	-

3.1 Motivation

Some research questions arise: RQ1. What kind of skills can be acquired by students when they participate in a university software development company? RQ2. What are the main aspects to consider when a university software development company will be set in a university program?

RQ3. What are the main aspects to consider when students are hired at a university software development company? and RQ4. What kind of events were meaningful to increase students' experience?

3.2 Methodology

The methodology was based on a voluntary online survey applied to students who have been hired by USDC from 2010 to 2022. The survey instrument was divided into two sections to provide quantitative and qualitative data. The first section gathered demographic questions. The second section was based on CLEI [19] Scales: Cohesiveness (C), Open-mindedness (OM), Integration (I), Technology adequacy (TA), and Facility Availability (FA).

C. Extent to which students know, help, and are supportive of each other.

OM. The extent to which the USDC activities encourage an open-minded approach to the use of computers and troubleshooting.

I. The extent to which the USDC activities encourage learning new technologies in addition to the theory learned in classes.

TA. The extent to which the hardware and software are adequate for the tasks required.

FA. The extent to which the facilities are available for use.

Additionally, ACCC [20] scales were taken into account: Anxiety (A), Usefulness of computers (UC), and Usefulness of the USDC.

Anxiety.The extent to which the student feels comfortable or has experience using computers, software, development framework, and so on.

The usefulness of computers. The extent to which the students believe computers are useful.

The usefulness of the USDC. The extent to which the students found the USDC useful.

This section measured students' perception of participating in USDC. The variables were measured using a five-point Likert scale ranging from 1 to 5, (strongly disagree to strongly agree, respectively).

Attitudes, skills, and students' experiences were identified. Independent and dependent variables were defined.

Independent variable

1. Gender,
2. Career,
3. How long were you working at LuminiSoft?

The dependent variable, some of them are shown:

1. I have enough skills to use computers
2. I have enough skills to learn new software development frameworks

...

7. I have enough skills to use CASE tools*
8. My experience as a team member improved

…

19. When technology is new, Luminisoft pays for the time required to learn the new technology*

…..

21. LuminiSoft is a complement to academic preparation in order to increase professional experience.

(* these questions were deleted in order to increase questionnaire reliability)

3.3 Instrument

For this research study, a questionnaire was designed for analyzing students' attitudes, skills, and experience, among other constructors.

3.4 Data Collection and Analysis.

Data for developing this research were collected from students who signed a contract.

This research collected data from 36.66% of hired students. A sample of 11 students was collected. The results were significant because they show a high correlation, $\alpha = 0.917$. The participants completed the questionnaire through google forms.

4. Results

First of all, the internal consistency of the questionnaire was assessed through Cronbach's alpha coefficient. The questionnaire was composed of 21 questions.

The Cronbach's alpha was computed. It was $\alpha = 0.87$. After that, Questions: Q7, and Q19 were removed because of the Pearson correlation coefficient results, see Table 2.

Table 2. Lower Mean

Questions	**Mean**	**Std. Dev.**	**K**
Q1. I have enough skills to use computers	4.7	0.516	6
Q7. I have enough skills to use CASE tool	3.8	0.752	6
Q9. I like to share my knowledge with others at LuminiSoft	4.5	0.547	6
Q10. I should have to learn new technologies at LuminiSoft	4.5	0.836	6
Q19. When it was necessary, LiminiSoft paid the required time to learn new technologies	3.7	1.211	6

Table 3 shows the correlation matrix. When question 7 was analyzed, low correlations were identified, indeed some of them were negative. Hence, question Q7 and Q19 were removed.

After this action, the Cronbach's alpha coefficient increases from 0.87 to α =0.89, see Fig. 1.

Table 3. Pearson correlation Matrix

Qs	1		3	4	5	6	7	...	21
1	1	0.67082	0.751779	0.833333	0.67082	0.833333	0.180021	...	0.388889
2	0.67082	1	0.263117	0.559017	1	0.804984	-0.06901	...	0.67082
3	0.751779	0.263117	1	0.86618	0.263117	0.470679	0.015131	...	0.392232
4	0.833333	0.559017	0.86618	1	0.559017	0.816667	-0.03858	...	0.527778
5	0.67082	1	0.263117	0.559017	1	0.804984	-0.06901	...	0.67082
6	0.833333	0.804984	0.470679	0.816667	0.804984	1	0.046291	...	0.466667
7	0.180021	-0.06901	0.015131	-0.03858	-0.06901	0.046291	1	...	-0.10287
...	...	...	...	...	...	...	...	...	...
19	-0.24721	0	-0.43633	-0.43633	0	0	-0.3433	...	0
20	0.240563	0.516398	0.198148	0.288675	0.516398	0.288675	0.111359		0.7698
21	0.388889	0.67082	0.392232	0.527778	0.67082	0.466667	-0.10287		1

5. Discussion

Regarding demographic results, there have been more male participants than female participants, see Fig. 2.

There are more male students than female students enrolled in ICTF. That's the reason for this situation.

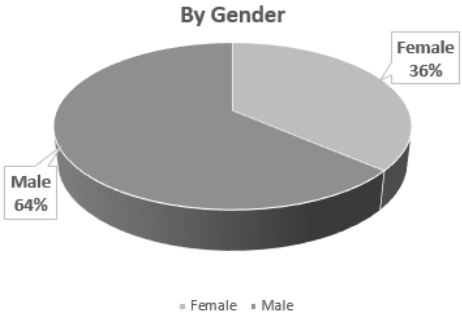
Regarding how long hired students remain working at LuminiSoft. Usually, they spend more than one year, see Fig. 3. Hired students receive payment and gain professional experience, hence it is very common that they spend more than one year working as developers at LuminiSoft.

Subject	Q1	Q2	Q3	Q4	Q5	...	Q21
1	4	4	4	3	4	...	4
2	5	5	5	5	5	...	5
3	4	5	3	3	5	...	5
4	5	5	5	5	5	...	5
5	5	5	5	4	5	...	5
6	5	5	5	5	5	...	5
7	5	5	5	5	5	...	5
8	5	5	5	5	5	...	5
9	5	5	4	4	5	...	4
10	5	5	4	4	5	...	5
11	5	5	5	5	5	...	5

Mode	5	5	5	5	5	...	5
Mean	4.8	4.9	4.5	4.4	4.9	...	4.8
Median	5	5	5	5	5	...	5

Cronbach's alpha-->	0.89

Fig. 1. Cronbach's alpha coefficient after some questions were deleted



Fig, 2. Developers by gender

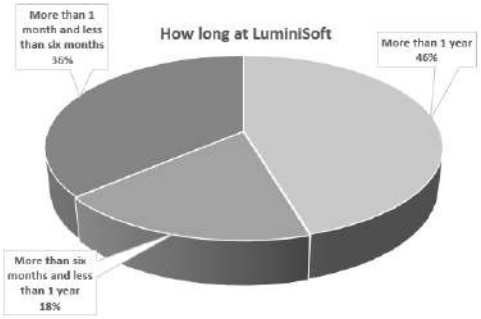


Fig. 3. How long working at LuminiSoft

Regarding attitudes towards computing, developing frameworks, sharing knowledge, hard & soft skills, and experience gained while participants worked at LuminiSoft, research questions were answered. These questions were answered taking into account CLEI and ACCC scales. The next paragraphs describe the outcomes.

The CLEI scales assessed were: C, OM, I, and TA. C, the extent to which hired students by LuminiSoft, help, and are supportive of each other. To answer this question, one question was asked: "I like to share my knowledge with others at LuminSoft". OM is the extent to which the USDC activities encourage an open-minded approach to the use of computers and troubleshooting by themselves. To answer this question, one question was asked: "If there should be a problem to solve in a software system, where I am programming, I will try to solve it". I, the extent to which the USDC activities encourage learning new technologies in addition to the theory learned in classes. To answer this question some questions were asked, one of them: "I should have to learn new development frameworks at LuminiSoft". TA is the extent to which the hardware and software are adequate for the tasks required. To answer this question, one question was asked: "When it is necessary, Luminisoft provides me with technology to develop a software project". FA is the extent to which the facilities and resources are available for use. To answer this question, one question was asked: "Luminisoft provides facilities to develop a software project".

The ACCC scales assessed were: Anxiety, the extent to which the student feels comfortable or has experience using computers, software, development framework, and so on. To answer this question, some questions were asked, one of them being: "I feel comfortable when developing a software System". The usefulness of computers, the extent to which the students believe computers are useful. To answer this question, one question was asked: "I believe software systems and computers are essential in our lives". The usefulness of the USDC, and the extent to which the students found the USDC useful. To answer this question, some questions were asked, one of them: "Participating in LuminiSoft was useful to my professional development".

RQ1, this question was answered by taking into account. ACCC Scales: C, OM, and I. Usually, hired students share knowledge among themselves, mainly tacit knowledge through face-to-face interaction or virtual interaction; additionally, pair programming is applied when junior developers are required to learn from senior developers. Students working at LuminiSoft enjoy sharing their knowledge. This is true because the question related to this information had a mean equal to 4, and the standard deviation = 0.54. It means they agreed to share their knowledge. Additionally, students working at LuminiSoft get skills by themselves, when they try to solve problems, when software projects generate a new problem, the problem has to be solved in order to reach a successful project. There are different kinds of problems. There are technical and social problems. When technical problems arise, they are solved by the whole team. Social problems are related to communication, interaction, support, and so on. To solve it, the team has to establish communication ways and has to design strategies in order to increase commitment among team members. The questions related to this aspect had a mean equal to 4.2, and the standard deviation = 0.40. It means they agreed to look for a solution when troubles arise. Students working at LuminiSoft felt encouraged to learn new technologies during their stay. This action provides them with new hard skills. This scenery was associated with: Learning new technologies, increasing software development experience, learning new development frameworks, and participating in training activities, among others. It was evident because questions related to this had an average mean equal to 4.6 and a standard deviation = 0.66.

RQ2, this question was answered by taking into account, CLEI Scales: TA, and FA. LuminiSoft is economically self-sustaining. It means the USDC does not receive economic support either from the ICTF or from the University, hence, Luminisoft must generate its own economic resources. Customers pay for developments. These economic resources are used to pay salaries and to buy software and hardware needed to develop software. This is the main reason students face real situations. When money is not involved, the situation is not real, when budgets to develop a software system are negotiated with customers, they are expecting to receive the system they have paid for.

Hence, in order to provide technology to develop a software system, the USDC must generate its economic resources by itself. Additionally, facilities must be provided by the USDC. Hence, if a University or Faculty wishes to establish a USDC, facilities must be provided.

RQ3, this question was answered taking into account, Scales: Anxiety, OM, I, and TA. Some hard skills (technical skills) include coding, database knowledge, modeling techniques, algorithms building, testing methodologies, operating systems, development IDEs, documentation techniques, and so on. Soft skills (social skills) needed, among others, are verbal communication, written communication, logical thinking, OM, teamwork, collaboration, organization, problem-solving, critical thinking, accountability, good attitude, and emotional intelligence. Some of them are required before students are hired. Hard skills are required but are not imperative because these skills are gained through practice. Soft skills are more important than hard skills when students are hired. Soft skills are difficult to acquire because they are intrinsic to each person. A strategy to reinforce these kinds of skills should be developed in academic programs. There were some indirect questions related to Soft Skills (SS) and Hard Skills (HS). For instance, I have enough skills to use computers (HS & SS), I feel comfortable using computers (HS & SS), I feel comfortable using development frameworks (HS & SS), I feel comfortable when developing a software System (HS & SS), I have enough skills to learn new software development frameworks (HS & SS), If there should be a problem to solve in a software system where I am programming, I try to solve it (SS), I like to share my knowledge with others at LuminSoft (SS), I should have to learn new technologies at LuminiSoft (HS), I increase my software development experience at Luminisoft (SS), I should have to learn new development frameworks at LuminiSoft (HS), My experience as a team member improve (SS). Questions directly related to RQ3 had an $\dot{X} = 4.6$ and a $\sigma = 0.66$. Hence, they were appropriately assessed by the participants. It means participants realized they are acquiring HS and SS.

RQ4, this question was answered by taking into account ACCC Scales, and the usefulness of the USDC. It means the extent to which the students found the USDC useful. Questions related to this scale were: "My experience as a team member improved, I like to participate in software development projects, participating in LuminiSoft was useful to my professional development, and LuminiSoft is a complement to academic preparation in order to increase professional experience". These questions had an $\dot{X} = 4.67$ and a $\sigma = 0.5$. The significant events to increase students' experience are, among others, direct contact with customers when they describe problems, they want to solve through a software system (elicitation requirements meetings), when students participate in delivery meetings (sprint review meetings) when students face requirements change request (RCR) because they have to evaluate time and money impact before they accept the RCR. Indirectly, these events were assessed with 5 questions.

As can be seen, almost all information is regarding good experiences, but there have been some bad experiences, mainly events related to staff turnover. Staff turnover must be taken into account because it is a common phenomenon. When students receive an offer from an external company they will resign from LuminiSoft. It is good and bad at the same time. It is good because students have gained enough experience to be hired by an external company. It is a little bad for LuminiSoft because the knowledge worker will be lost. In order to minimize the negative impact on Luminisoft, a knowledge database has to be maintained constantly. It means all time processes, methodologies, and solution troubles have to be documented. Knowledge wikis, and Frequently Technical Questions have to be kept up to date. Additionally, senior developers have to share their tacit knowledge with junior staff and code their tacit knowledge into explicit knowledge. Setting up a Knowledge management framework is suggested.

6. Conclusion

When the SE discipline is taught in Universities, it would be a good idea to set up a University Software Development Company for it. Mainly, because theory plus real practices are required when software engineers are formed. SE cannot be taught exclusively with theory, and academic practices. Students should be exposed to real-world situations where they can engage with customers,

understand their problems, and work to provide solutions that help maintain the customers' competitive advantage. This approach allows students to gain both hard and soft skills before graduating from university.

This research has shown, the students increased their experience, got technical knowledge, and reinforced their hard and soft skills. Additionally, Students gained the confidence to lead or participate in team works as team members or leaders.

Our USDC allowed students to face real situations before they finish their careers. It is useful because a considerable quantity of companies requires young people with professional experience, but how could Students meet this requirement? The answer can be simple, setting up an internal company in universities.

Regarding the questionnaire applied, it was reliable because Cronbach's alpha coefficient was equal to 0.89.

RQ1 to RQ4, analyzed in section 5 have shown that having a USDC is useful to increase students' professional experience, hence it is recommended to set up a USDC at universities or faculties when it is possible.

7. Future work

The same questionnaire will be applied to more hired students, at least 50% has to be reached in order to have additional information to be analyzed and compared actual outcomes with future outcomes. The USDC will try to increase student participation. Transversal capstones will be configured; several courses will be involved in each of them a single SDLC will be tackled. Additionally, a marketing department will be set up in order to increase real software development projects. So far, no marketing department exists and the Knowledge management framework must be tuned.

References

[1]. Gurcan, F., and Sevik, S., "Expertise Roles and Skills Required by the Software Development Industry". 1st International Informatics and Software Engineering Conference (UBMYK), Ankara, Turkey, 2019, pp. 1-4, doi: 10.1109/UBMYK48245.2019.8965571.

[2]. Juárez-Ramírez, R., Navarro, C. X., Licea, G., Jiménez, S., Tapia-Ibarra, V., Guerra-García, C., & Perez-Gonzalez, H. G. (2022). How COVID-19 Pandemic affects Software Developers' Wellbeing, and the Necessity to strengthen Soft Skills. Programming and Computer Software, 48(8), 614-631.

[3]. L. Kunttu, E. Huttu, and Y. Neuvo, "How doctoral students and graduates can facilitate boundary spanning between academia and industry," 2018.

[4]. Chillas, S., Marks, A., & Galloway, L. (2015). Learning to labour: an evaluation of internships and employability in the ICT sector. New technology, work and employment, 30(1), 1-15.

[5]. Liu, Q., Kovalchuk, S., Rottmann, C., & Reeve, D. (2018). Engineering co-op and internship experiences and outcomes: The roles of workplaces, academic institutions and students.

[6]. Khakurel, J., & Porras, J. (2020, November). The effect of real-world capstone project in an acquisition of soft skills among software engineering students. In 2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T) (pp. 1-9). IEEE.

[7]. Pinto, G., Ferreira, C., Souza, C., Steinmacher, I., & Meirelles, P. (2019, May). Training software engineers using open-source software: the students' perspective. In 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET) (pp. 147-157). IEEE.

[8]. Akdur, D. A Survey on Bridging the Gap between Software Industry and Academia: Preliminary Results. At Proceedings of the 13th Turkish National Software Engineering Symposium 2019.

[9]. Garousi, V., Petersen, K., and Ozkan, B., "Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review", Information and Software Technology, Volume 79, 2016, Pages 106-127, ISSN 0950-5849, https://doi.org/10.1016/j.infsof.2016.07.006.

[10]. O. Cico, L. Jaccheri, A. Nguyen-Duc, and H. Zhang, "Exploring the intersection between software industry and Software Engineering education - A systematic mapping of Software Engineering Trends," J. Syst. Softw., vol. 172, p. 110736, Feb. 2021, doi: 10.1016/j.jss.2020.110736.

[11]. Vives, L., Melendez, K., & Dávila, A. (2022). ISO/IEC 29110 and Software Engineering Education: A Systematic Mapping Study. Programming and Computer Software, 48(8), 745-755.

[12]. P. Saliou and V. Ribaud, "Learning by doing software engineering," Inform. Educ. Eur. Montp. Fr., 2006.

[13]. F. Meawad, "The virtual agile enterprise: Making the most of a software engineering course," in 2011 24th IEEE-CS Conference on Software Engineering Education and Training (CSEE&T), 2011, pp. 324–332.

[14]. O. Olayinka and M. Stannett, "Experiencing the Sheffield Team Software Project: A project-based learning approach to teaching Agile," in 2020 IEEE Global Engineering Education Conference (EDUCON), 2020, pp. 1299–1305.

[15]. A. Rusu and M. Swenson, "An industry-academia team-teaching case study for software engineering capstone courses," in 2008 38th Annual Frontiers in Education Conference, 2008, pp. F4C-18.

[16]. R. E. Bruhn and J. Camp, "Capstone course creates useful business products and corporate-ready students," ACM SIGCSE Bull., vol. 36, no. 2, pp. 87–92, 2004.

[17]. M. Bernstein, K. M. FitzGerald, J. P. Macdonell, and A. I. Concepcion, "Algorithma project: the ten-week mock software company," in Proceedings of the 36th SIGCSE technical symposium on Computer science education, 2005, pp. 142–146.

[18]. Bell, S. Project-Based Learning for the 21st Century: Skills for the Future, The Clearing House: A Journal of Educational Strategies, Issues and Ideas, Vol. 83, Issue 2, pp. 39-43. 2010. https://doi.org/10.1080/00098650903505415.

[19]. Faisal, A. M., (2012), Students' perception and attitude towards computer laboratory learning environment, International Research Journals, Vol. 3. Num, 4, pp. 402-411, ISSN: 2141-5163.

[20]. Jhurree, V., Bessoondyal, H., and Mohamudally, N., (2007). Primary Oriental Language Teachers' Attitudes towards the Computer and its Perceived Usefulness in their Teaching Profession-A Case Study. Proceedings of the 2007 Computer Science and IT Education Conference.

Информация об авторах / Information about authors

Хорхе Рафаэль АГИЛАР СИСНЕРОС имеет степень PhD по программированию, является сотрудником факультета инженерии Народного автономного университета мексиканского штата Пуэбло (UPAEP). Сфера его научных интересов включает программирование для вычислительных машин и управление знаниями.

Jorge Rafael AGUILAR CISNEROS – Ph. D. Jorge Rafael Aguilar Cisneros is a specialist at the UPAEP University's Department of Engineering. His research interests include software engineering, and knowledge Management.

Карлос Альберто ФЕРНАНДЕС-И-ФЕРНАНДЕС – имеет степень PhD университета Шеффилда по программированию, эксперт в области программирования. В настоящее время возглавляет Институт вычислений в Технологическом университете в мексиканском регионе Миштека, координирует магистерские программы по прикладным аспектам вычислительных технологий. Сфера научных интересов: визуальное моделирование, гибкие технологии разработки и формальные спецификации программного обеспечения.

Carlos Alberto FERNÁNDEZ-Y-FERNÁNDEZ – Software Engineering expert with a Ph.D from the University of Sheffield. He currently leads the Institute of Computing at Universidad Tecnológica de la Mixteca and coordinates the Master's program in Applied Computing Technologies. His research interests include visual modeling, agile methods, and formal software specification.

DOI: 10.15514/ISPRAS-2024-36(1)-13

Exploring the Role of Bots in Software Development

R. Moguel-Sánchez, ORCID: 0000-0001-7735-7110 <*mmoguelrick@gmail.com*>
C.S. Martínez-Palacios, ORCID: 0000-0001-9795-1408 <*delmanclen@hotmail.com*>
J.O. Ocharán-Hernández, ORCID: 0000-0002-2598-1445 <*jocharan@uv.mx*>
X. Limón, ORCID: 0000-0003-4654-636X<*hlimon@uv.mx*>
A.J. Sánchez-García, ORCID: 0000-0002-2917-2960 <*angesanchez@uv.mx*>
School of Statistics and Informatics, Universidad Veracruzana,
Xalapa, Veracruz, México.

Abstract. This research examines the state of applied and proposed software bots in software development through a systematic literature review. Spanning from 2003 to 2022 and encompassing 83 primary studies, the study identifies four bot archetypes: chatbots, analysis bots, repair bots, and development bots. The key benefits of utilizing bots include improved software quality, provision of information to developers, and time savings through automation. However, drawbacks such as limited effectiveness and reliance on third-party technologies are also noted. The study highlights the potential of including bots in software development but emphasizes the need for further exploration and research in this area.

Keywords: bots; development bots; software development; thematic synthesis.

For citation: Moguel-Sánchez R., Martínez-Palacios C. S., Ocharán-Hernández J. O., Limón X., Sánchez-García A. J. Exploring the Role of Bots in Software Development. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 209-224. DOI: 10.15514/ISPRAS-2024-36(1)-13.

Full text: Moguel-Sánchez R., Martínez-Palacios C. S., Ocharán-Hernández J. O., Limón X., Sánchez-García A. J. Bots in Software Development: A Systematic Literature Review and Thematic Analysis. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 712–734. DOI: 10.1134/S0361768823080145.

Исследование роли ботов в разработке программного обеспечения

Р. Могель-Санчес, ORCID: 0000-0001-7735-7110 <*mmoguelrick@gmail.com*>
С.С. Мартинес-Паласиос, ORCID: 0000-0001-9795-1408 <*delmanclen@hotmail.com*>
Х.О. Очаран-Эрнандес, ORCID: 0000-0002-2598-1445 <*jocharan@uv.mx*>
Лимон К., ORCID: 0000-0003-4654-636X <*hlimon@uv.mx*>
Санчес-Гарсия А.Х., ORCID: 0000-0002-2917-2960 <*angesanchez@uv.mx*>

Университет Веракруса, школа статистики и информатики,
Халапа, Веракрус, Мексика.

Аннотация. В работе рассматривается состояние прикладных и перспективных программных ботов в разработке программного обеспечения посредством систематического обзора литературы. Охватывая период с 2003 по 2022 год и 83 первичных исследования, исследование идентифицирует четыре архетипа ботов: боты-собеседники, боты-аналитики, боты-ремонтники и боты-разработчики. Ключевые преимущества использования ботов заключаются в повышении качества программного обеспечения, в предоставлении информации разработчикам и в экономии их времени за счет автоматизации. Отмечаются такие недостатки, как ограниченная эффективность и использование сторонних технологий. Исследование показывает значительный потенциал ботов при их подключении к разработке программного обеспечения и необходимость дальнейших исследований в этой области.

Ключевые слова: программные боты, боты-разработчики; разработка программного обеспечения; тематический синтез.

Для цитирования: Могель-Санчес Р., Мартинес-Паласиос С.С., Очаран-Эрнандес Х.О., Лимон К., Санчес-Гарсия А. Х. Исследование роли ботов в разработке программного обеспечения. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 209–224 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–13.

Полный текст: Могель-Санчес Р., Мартинес-Паласиос С.С., Очаран-Эрнандес Х.О., Лимон К., Санчес-Гарсия А. Х. Боты в разработке программного обеспечения: систематический обзор литературы и тематический анализ. Programming and Computer Software, 2023, т. 49, № 8, с. 712–734 (на английском языке). DOI: 10.1134/S0361768823080145.

1. Introduction

Aligned with the current booming in Artificial Intelligence assisted software development, led by popular projects such as GitHub Copilot and ChatGPT, in Software Engineering (SE), different tools and techniques have been applied to each phase of the software development cycle to increase the quality of the final product, while meeting their requirements. Such tools and techniques respond to the increasing complexity found in modern systems and development environments [1]. Technologies to aid software developers have been arising under Artificial Intelligence (AI) techniques, such as Machine Learning [2] and Natural Language Processing [3]. Development Bots are one such emerging technology born outside the scope of software production.

Due to the broad field of applications for bots, there are multiple types of bots. A Bot can vary in complexity based on its intended objective. "The term bot ranges from describing simple scripts that automate a task in the background to complex applications that interact with one or more humans and autonomously adapt to activities performed by humans and other systems, and even to software applications that use artificial intelligence to mimic human behavior and intelligence" [4].

Bots have been proliferating, motivating research to define and classify bots [4-6], including proposals for new bots that push the use of bots as human assistants [7-8]. Despite the increasing emergence of new bots that aid in software development tasks, little research addresses the practical use of bots in SE. To close this gap, we conducted a Literature Review to analyze published research on bots applied in software development activities, highlighting practical benefits and challenges. The results of this research may help software developers make well-informed decisions regarding

adopting bots in their software development process. Our study can also serve as a steppingstone for future research delving deeper into the emerging field of software bots.

This study is an extension, covering more research papers from 2022 and doing an iteration of Forward and Backward Snowballing to get as many papers as possible that help achieve this research's objective.

This paper is organized as follows. Section 2 presents related studies that contrast the need for this study. Section 3 describes the method used in the planning and execution of the literature review. Section 4 deals with the conduction of the research method. The results are presented in Section 5. Section 6 addresses the discussion of the obtained results. Section 7 includes the threats to validity. Finally, Section 8 draws the conclusions derived from the study.

2. Research method

To meet the stated objective, we conducted a Systematic Literature Review (RSL) following the guidelines of Evidence-Based Software Engineering and Systematic Reviews [9].

2.1 Search Process

We used automatic search as a research strategy to identify primary studies to answer RQs. Afterward, we performed a snowballing search to expand the pool of candidate studies. We applied a process based on Creswell's thematic analysis for data synthesis. The research questions are the following:

RQ-1: In which software development activities are bots involved?

RQ-1.1: What is the goal of a bot in the activity it is used in?

RQ-1.2: Which software development activities supported by bots are the most reported in the literature?

RQ-2: What are the benefits of applying a bot in a development activity?

RQ-3: Which problems arise in development activities that use bots?

RQ-4: What are the levels of intelligence of bots used for software development?

2.1.1 Search Strategy

A manual search was performed according to the steps outlined in the Automatic Search Process [9] to identify primary studies in SE. Following the process, a Quasi-Gold Standard was created to evaluate the performance of search strings to select a high-quality search string [10].

Identifying Scientific Databases: The first step was to identify venues from which to select articles through an automatic search. As primary search engines IEEE Xplore and ACM Digital Library were selected since they contain the entire catalog of articles published under the International Conference of Software Engineering (ICSE), particularly its subsidiary conference Bots in Software Engineering (BotSE), which is related to bots in software development. To bolster the rigorousness of the search process, additional databases were included. These databases were Science Direct, Springer Link, and EBSCO Host.

Afterward, a manual search was conducted for studies under ICSE to construct a Quasi-gold Standard [10]. From the manual search, 20 relevant articles were identified. From the candidate studies identified in the manual search, a set of concepts and keywords was extracted to craft search strings to test for automatic pilot searches in IEEE Xplore. After applying the method for search string selection, the selected search string is the following:

("Analysis bot" OR "Chatbot" OR "Conversational bot" OR "Conversational developer assistant" OR "Developer assistant" OR "Repair bot" OR "Automated repair" OR "Review bot" OR "Software bot") AND ("Software Development" OR "Software Project" OR "Open source project" OR "Software Engineering" OR "Repair")

2.2 Selection Process

Afterward, inclusion and exclusion criteria were defined to determine adequate studies for data extraction. We only considered articles published inclusively between 2011 and 2022 for this extraction method. This range of years of publication was chosen to drive the search scope toward recent and current articles regarding bots for software development activities. Additionally, the topic of bots has seen an increase in popularity with the creation of the BotSE conference in 2018, which centers on applying bots in SE. The inclusion and exclusion criteria used to select articles from the automatic search are shown in Table 1.

Table 1. Inclusion and Exclusion criteria for selecting studies on Automatic Search

Inclusion Criteria	Exclusion Criteria
IC-1: The study is published between 2011 and 2022	EC-1: The study is secondary or tertiary.
IC-2: The study is written in English	EC-2: The study is an opinion, presentation, or book chapter.
IC-3: The title and abstract suggest that at least RQ can be answered.	EC-3: The study is a duplicate found on a different database.
IC-4: The full-text answers at least one RQ.	EC-4: The study scores lower than six on the quality evaluation process.

2.3 Data Extraction Process

Publication data related to the demographics of each study were extracted. With this data, a particular study can be quickly identified through its characteristics and group studies, considering its publication year and publisher. For example, by cataloging the data by year, a count of how many studies about development bots were published by year can be elaborated. With this information, it can be determined whether there is a growing interest in the topic of bots in software development. Likewise, by sorting studies by the publisher, conferences with the most papers in academia on the topic of bots for SE can be identified. The Extraction Data fields are related to the RQ to be answered in this research. The type of bot identified in each article is recorded to identify the demographics of the types of bots used in SE. By extracting the development phase and activity in which a bot is used, RQ-1, regarding where a bot is applied in the software development cycle, can be answered. Similarly, the other data fields are related to RQs with Bot Adaptability, Reasoning, and Autonomy associated with different aspects of the level of intelligence observable in bots. The template used to extract the data is shown as part of a Zenodo data set [11] containing artifacts elaborated as part of this research.

2.4 Snowballing Search Process

To expand upon the studies selected from the automatic search, a Snowballing search for SE was executed based on the Guidelines for snowballing in systematic literature studies and replication in software engineering [12]. This search process selects additional papers from the references and citations of studies in an initial batch. This process was applied based on the primary studies collected from the automatic search published between 2011 and 2022. First, an iteration of backward snowballing was performed. Afterward, a single iteration of forward snowballing was executed with the aid of the Google Scholar search engine. The same selection criteria filters for the automatic search were applied to papers obtained through backward and forward snowballing, excluding the IC-1 Inclusion Criteria because we reached some articles back from 2011. Likewise, the same data extraction process was followed in the resulting studies.

2.4.1 Thematic Synthesis

Initially, an analysis was conducted through a narrative synthesis to answer the established RQs. The synthesis is based on the bots identified in 83 primary studies, resulting from the quality

evaluation of data extracted from automatic and snowballing searches. After that, a thematic synthesis was elaborated following the Recommended Steps for Thematic Synthesis in Software Engineering by Cruzes & Dybå [13] to relay the literature review results. As Result, a thematic synthesis of the analyzed data in the form of answers to established RQs can be seen, along with graphics to illustrate our responses.

3. Method conduction

The selection process was executed in five main stages, applying inclusion and exclusion criteria to identify relevant studies. First, an automatic search was conducted. Then, based on the resulting full-text studies, a backward snowballing search was performed. Afterward, a forward snowballing search was applied to the studies obtained from the automatic search. Both referenced studies from backward snowballing and citations from forward snowballing underwent the selection criteria filtering. Lastly, the quality evaluation process was performed. From the automatic search and supplementary snowballing search process, we obtained 83 primary studies. The artifacts produced as part of this research for the method conduction can be seen as a Zenodo data set [11].

4. Results

This section addresses the demographics of the identified Primary Studies (PS) and answers the RQs in order of appearance. For this section, we use the identifier PS-XX to refer to a specific study; the list with all the Primary Studies can be found in the following link *https://drive.google.com/file/d/1kx9DKS6qJ7sGDIaMtZ7DK1LunD83gjjy*.

4.1 Overview of Primary Studies

Fig. 1 shows a trend in the increase of publications of interest, especially between 2016 and 2022, with a notable surge in 2019. This trend can be partially attributed to the BotSE conference, created in 2019, contributing articles detailing the applications of bots in software development activities. It is interesting to note that there were fewer articles in 2020 compared to 2019. This may be attributed to the worldwide disruptions caused by the COVID-19 pandemic, which, to a lesser extent, also affected publications from 2021 and beyond. Nonetheless, Bots in SE achieved high international recognition partly thanks to BotSE.

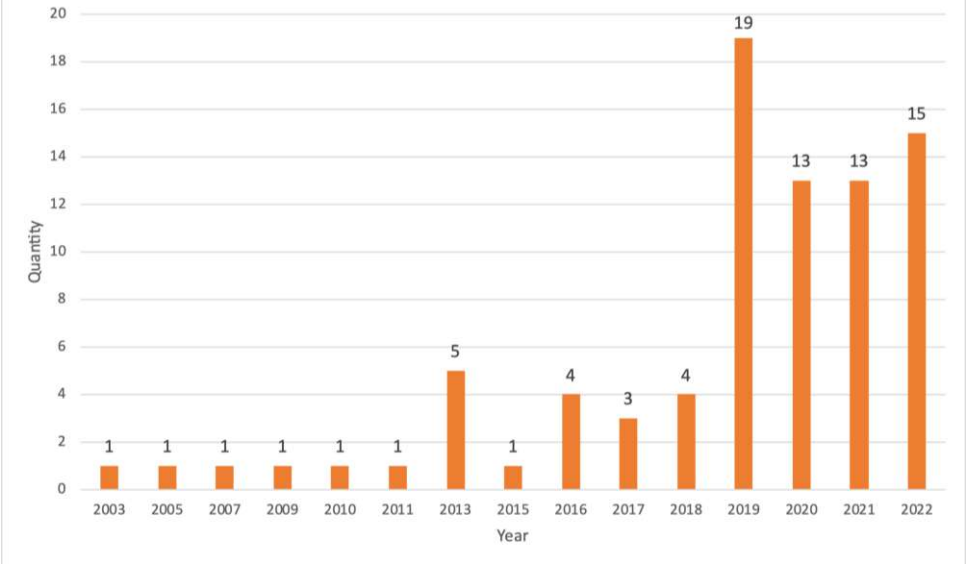


Fig. 1. Distribution of primary studies found by year

Among the different types of articles, 25 come from backward snowballing, and 29 primary studies come from IEEE Xplore Digital Library using the automatic search. The prevalence of the latter is due to the hosting of conferences covered by ICSE, including the BotSE Workshop, one of the leading conferences on this topic. With BotSE, it is evident that the topic of Bots has garnered significant interest at the international level. Prior papers show that the topic existed and only needed an international push to jump-start its popularity. Even so, existing works before 2019, although they only represent 12%, should not be disregarded.

4.2 Answers to Research Questions

As our Research Method section mentioned, we proposed 4 RQs to meet the literature review objectives. In this section, we aim to answer such questions with the results obtained from the 83 primary studies selected.

4.2.1 RQ-1

The software development activities where bots are involved. Fig. 2 presents the specific software development activities found in the primary studies. We can highlight project management as the most common activity involving bots.

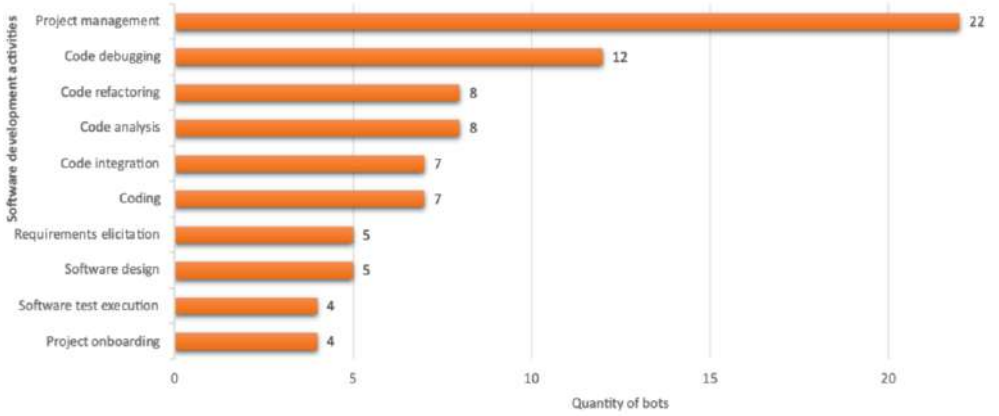


Fig. 2. Number of bots found by development activity

Software development teams spend a significant amount of time discussing solutions through messaging platforms during the development of a project [15]. As project management was found to be the most popular activity for the use of bots, it is clear there exist attempts to find different alternatives to manage software projects. Primary studies covering bots for software management follow. PS-13 showcases a chatbot to support project data analysis and team progress measurement. Similarly, TA Bot in PS-21 is a bot for solving work delays during development, which also provides graphs related to the impact that delays cause in development, recommending actions to mitigate the adverse effects. Some of this type of bots are specific to the GitHub platform, such as Stale Bot in PS-17 for identifying unattended pull requests; JITbot in PS-29 for prioritizing pull requests needing a code review; and DependaBot detailed in PS-37 and PS-40 for updating dependencies in repositories. We found some studies for source code management in GitHub, such as Code Climate Bot in PS-23, which reminds developers about pending fixes, and StyleCI in PS-34, which applies code style preferences. Magalhaes de Lima proposes another bot for code management in PS-33 to update different project branches. Devy Bot in PS-18 is the most ambitious of all management bots, being a conversational development assistant responsible for managing tedious, repetitive tasks with low levels of abstraction, thereby liberating developers to focus on more demanding and complex tasks.

We identified 12 bots related to code debugging. As a representative example, PS-1 by Belskii A. and Itsykson V. M. is a bot that starts by performing analysis, identifying bugs, generating possible solutions, offering patch fixes, and finally notifying the developer of the results.

Among the main activities identified, another one is code analysis, characterized by code reviews and static analyses executed in the software implementation and maintenance stages. The main challenge in code review is the large amount of human effort involved, even for tool-assisted code reviews [16]. This could be due to the excessive complexity of applying coding standards or reviewers prioritizing logical checking [26]. The high number of bots identified shows that this activity has great potential for bot inclusion. Repairnaitor in PS-6 identifies errors in failed builds and proposes solutions autonomously. Other bots that perform code analysis and recommend follow-up actions include Sankie in PS-11, CCBot in PS-25, and Danger Bot in PS-23. For a simple and straightforward bot companion, C-3PR Bot in PS-27 provides details for detected bugs, and Review Bot in PS-24 notifies coding standard violations. For GitHub, Saw-bot in PS-41 invokes SonarQube – a tool for static code inspection, generates a patch and then integrates its solution into a pull request. SapFix in PS-11 runs a code analysis on a pull request and then recommends different DevOps activities to perform afterward.

As a sample from code integration, Sayme Bot, in PS-15, warns of potential direct and indirect conflicts when uploading changes to a GitHub repository.

For Requirements Elicitation, there were five bot proposals. For instance, Dwitam F. and Rusli A. in PS-4 propose a bot capable of conducting interviews with stakeholders in software projects to elaborate user stories. Surana C. et al. in PS-22 propose a chatbot capable of simulating natural conversations with Stakeholders that can extract and classify requirements based on conversations.

For software design, we identified five bots. RAPID bot in PS-31 offers design recommendations based on non-functional requirements, and iContractBot in PS-39 creates models through a step-by-step conversation. Gilson F. and Weyns D. envisioned a more social bot in EP-9. Their bot generates design artifacts based on the chat logs of a development team instead of a single member.

Another notable activity is Onboarding, where new developers are introduced to the project, with four bots identified. MSR Bot in PS-7 is envisioned as an assistant focused on answering questions about a GitHub repository's documentation for project newcomers. The remaining identified bots for Onboarding address questions from development team members, such as APIBot in PS-26 and Smart Advisor in PS-16.

4.2.2 RQ-2

The benefits of applying a bot in a development activity are shown in Fig. 3.

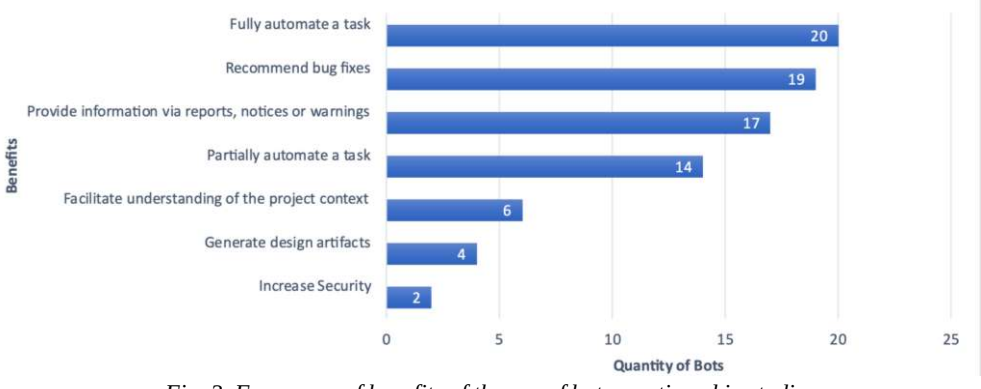


Fig. 3. Frequency of benefits of the use of bots mentioned in studies

Fig. 3 describes the reported benefits of using bots, with task automation being the most frequently observed benefit. The valuable benefit of time-saving bots is that bots help developers save time by

partially or fully automating tasks. We found 20 bots that can completely automate tasks. For example, Devy in PS-8 automates Git and GitHub tasks by creating commits to a repository, assigning code reviews, and managing a repository via voice commands. Refactoring Bot in PS-18 saves time by fully automating code refactoring to resolve code smells identified by the SonarQube static analysis tool; it then submits the changes to the developer for review via pull requests on GitHub. Lastly, Review Bot in PS-24 automates static analyses by generating checks for coding standard violations and common defect patterns and publishing the results of its analysis. In contrast, we identified 14 bots that automate parts of a task. This includes a chatbot by Surana C. et al. in PS-22, Travis CI in PS-23, and Stale Bot in PS-17. These bots partially automate tasks such as user interviews, continuous integration, and pull requests. Another bot, by Magalhaes de Lima in PS-33, updates different branches based on changes pushed to a specific repository branch. These bots provide a time-saving benefit, though to a lesser extent.

Another prevalent benefit of bots was the recommendation of fixing candidates for bugs. By getting possible solutions to a faced problem using a bot, developers can make appropriate decisions regarding the patches generated by the bot instead of spending time analyzing the problem on their own. These bots are predominantly repairing bots that help the developer identify bugs and give possible solutions that save time and improve the quality of the final product. For example, the Repairnator bot in PS-6 generates patches for undetected bugs. When it finds a bug, Repairnator attempts to replicate and repair the flaw, reporting the results to developers. Another example is R-Hero in PS-36, which repairs code with compile errors. Similarly, C-3PR Bot in PS-27 suggests bug fixes identified through static analysis on pull requests from a project repository. ConE in PS-32 also proposes recommendations for conflict resolution in pull requests, evaluating the danger of introducing defects in open pull requests.

It is also reported that 17 bots proactively provide helpful information without requiring a direct request from a developer. For instance, Sayme in PS-15 is a chatbot that provides notifications of potential direct and indirect conflicts when pushing changes to a GitHub repository. Alternatively, a chatbot can provide valuable information on demand, supporting developer decision-making, as is the case for Smart Advisor in PS-16, with the ability to answer developer questions and give appropriate alerts and recommendations during coding and maintenance.

Improvement in understanding the context of a software project is also a benefit found in six cases. In three cases, chatbots are used for training new development team members. The first was API Bot in PS-26, which introduces a developer to a documented API, answering questions about the API. Similarly, Dominic J., Ritter C., and Rodeghero P.'s chatbot in PS-5 answers questions from onboarding members. MSR Bot in PS-7 makes it easy to understand the context of a software repository and the changes made by the team. Lastly, we found two bots that increase security. These were PAutoBotCatcher in PS-35, designed to detect botnets attempting to breach a system, and SEADER in PS-76, designed to detect and generate fixes to misuses of APIs for Java.

4.2.3 RQ-3

Problems that arise in development activities that use bots are shown in Fig. 4.

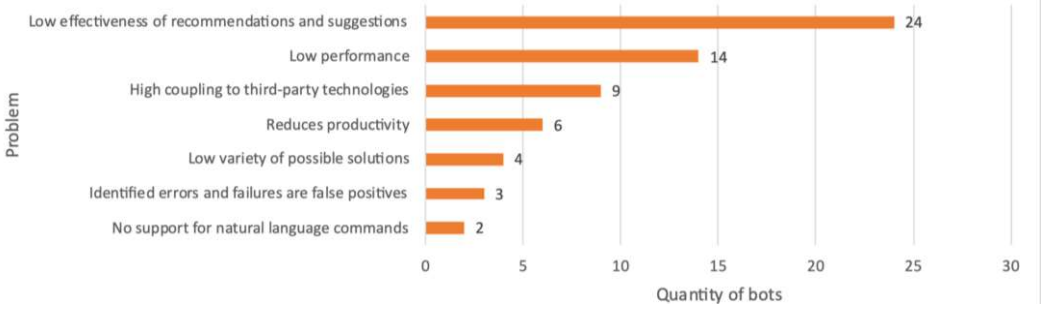


Fig. 4. Frequency of challenges with the use of bots mentioned in studies

Fig. 4 details difficulties that emerge from using bots in software development activities. We can identify two main complications: the low effectiveness of generated solutions and the poor performance of a bot in a designated activity.

From the 24 identified cases of bot ineffectiveness, C-3PR Bot in PS-27 exemplifies a typical scenario where developers reject bot fix suggestions. In this case, C-3PR Bot removed commented-out code that developers wanted to keep. Another instance of low effectiveness is PTracer in PS-19. It had a rejection rate of 32.45% for its generated solutions, as developers noted that the produced patch failed to solve identified bugs. Similarly, the bot by Brown C. and Parnin C. in PS-12 reportedly had ineffective recommendations regarding the installation of a static analysis tool; developers only accepted 3% of the bot recommendations, as the bot does not recognize the developer's working context; when trying to install the tool, source code no longer compiled correctly because it violated the file formats followed by developers. On the other hand, PS-23 reports that Travis CI in PS-23 was not beneficial to developers, as they had difficulties executing tests, preferring to debug manually.

Regarding difficulties due to bots' performance, creating fixes and patches for bugs is the limited variety of fixes. In two sample cases, bots are reported to have a limited number of possible solutions. This is the case of RefBot in PS-20, with a low variety of solutions, as the bot only focuses on recent Pull Requests and not on the current developer context; furthermore, this bot focused only on quality attributes based on the QMOOD Model, which can be counterproductive for developers who do not use this model. Another case lacking recommendation variety is the TA Bot in PS-21, as this chatbot did not have enough types of delays to give practical recommendations. The recommendations of TA Bot did not address the cause of the delays, only the symptoms.

High dependence on third-party technologies is a reported drawback in nine other cases. For example, various studies report that bots require a constant Internet connection, as in the case of the MSR Bot chatbot in PS-7. The Sankie bot from PS-11 also had a high dependency on external technologies. In this case, the bot analyzes coupling in the continuous integration and deployment pipeline, depending on the DevOps services provided by the Microsoft Azure platform. Sankie bot is also prone to respond with recommendations determined to be false negatives.

Regarding problems with false positives, the bots in PS-3 by Padgham L. et al., StaleBot in PS-38, and Task Navigator in PS-46 all reportedly flagged as error actions by developers outside of the original scope of the bot. In the case of the bot by Padgham L. et al., for categorizing warnings, exceptions, and failures during test execution, it was deemed unreliable to trust the bot’s categorization in case few unit tests were executed. StaleBot would flag as obsolete old pull requests that developers would still find relevant, and TaskNavigator would not query questions from developers in the case there were misspellings. However, the problems with TaskNavigator were alleviated by implementing an autocorrect function.

Another rare but identified problem is bots lacking support for specific commands in natural language. This is the case of the Refactoring Bot reported in PS-18, as commands in natural language were not supported. Therefore, the learning curve to use the bot was high since a developer must learn the specific commands of the chatbot. In the case of Devy in PS-8, the bot could not interpret commands from specific phrases; for example, it did not interpret the question "Who has made changes?" as a command to identify who has edited a file. The participants did not consider that the bot can interpret commands more ambiguously, needing more specific commands.

4.2.4 RQ-4

The intelligence characteristics we identify in this work are Adaptability, Reasoning, and Autonomy [7]. The subsections that follow will specify each characteristic.

Capacity for Adaptability. Fig. 5 shows a pie chart detailing whether the bots identified are cable of Adaptability, whether they can recognize changes in their environment, such as different project configurations, or if they change their behavior to adapt functionality, supporting the developer's

needs. To be able to adapt, a bot needs to be aware of the context of its development activities. In other words, the bot refines its actions for the developer's benefit as the bot learns from their mutual interactions. Of the 83 bots extracted from the primary studies, in 65% of the cases, bots can adapt and recognize a context. In contrast, 27% of the studies report that the bot does not have this capacity. The bot's intelligence level is not mentioned in the rest of the cases.

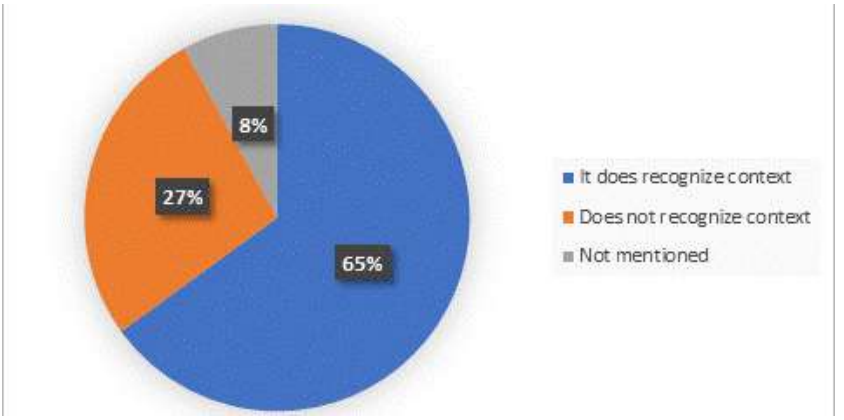


Fig. 5. Percentage of bots found with the capacity for Adaptability

Bot reasoning ability. In this case, reasoning refers to the existence of an internal bot mechanism to parse input data, perform a task to assist the developer, and provide output data to notify developers about the results of actions. Natural Language Processing is the most mentioned technique associated with bot reasoning, particularly for chatbots. Fig. 6 shows a pie chart detailing whether the bots identified are cable of Reasoning. Of the 83 bots identified, in 74% of the cases, it was possible to extract a reasoning mechanism for a bot. Of the remaining cases, in 11% of the cases, the bots do not have this ability, and in 15% of the cases, a reasoning mechanism is not mentioned.

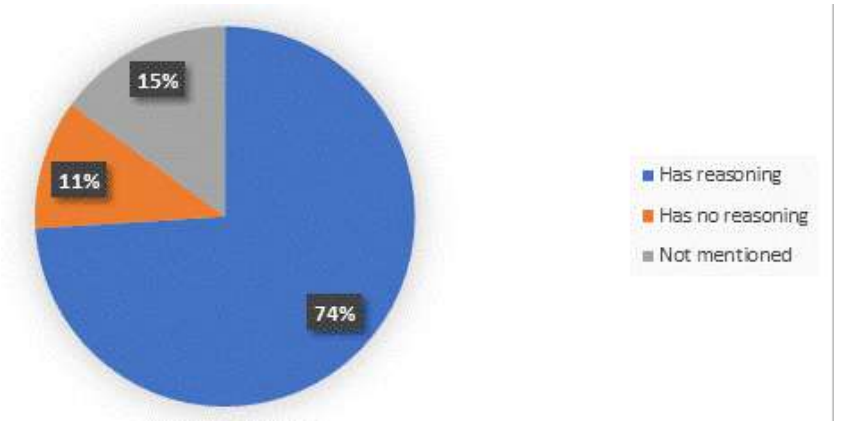


Fig. 6. Proportion of bots found with the capacity for Reasoning

Autonomy capacity. Fig. 7 presents a pie chart detailing whether the bots identified are capable of Autonomy, which is the ability to provide a beneficial service without the need for explicit human instructions. Bots that demonstrate Autonomy have a degree of independence and perform their tasks on their own, contacting the developer via messages and notifications to provide information, warnings, recommendations, or status reports, either periodically on their own or when requested by a human. A case that illustrates high independence is JIT Bot from PS-29. After performing its initial analysis, it keeps track of the repository and its changes without human intervention. Analysis Bots can also be autonomous, like the C-3PR bot in PS-27. On its own, C-3PR suggests fixes for static analysis violations via pull requests, which constructs after running an independent static analysis.

A plurality of bots at 47% lack autonomy, whereas only 42% of bots identified were determined to display it. Notably, in 11% of the group of bots identified autonomy was not mentioned and thus could not be determined.

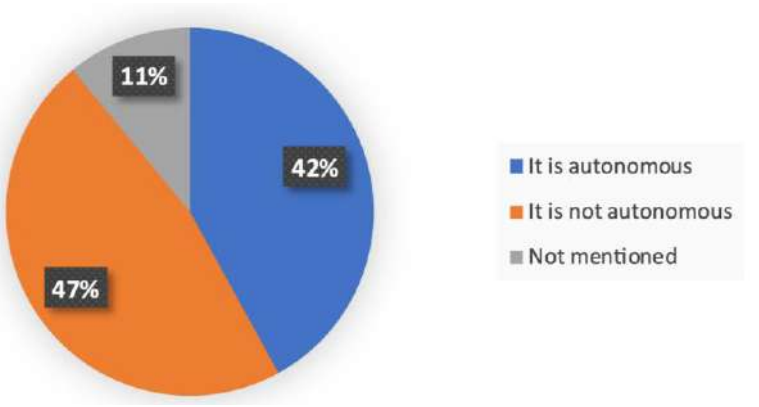


Fig. 7. Proportion of bots found with the capacity for Autonomy

4.3 Thematic Synthesis Results

To produce a thematic map, we used MaxQDA, a tool for data analysis, integrating qualitative methods. With this software, we identified and organized coded segments of data. Fig. 8 shows the higher-order thematic map produced following the thematic synthesis process for software engineering [13].

The diagram is divided into themes derived from codes based on excerpts from the 83 primary studies found in the literature review. Eight higher-order themes, representing different concepts, branch from the main theme about bots for software development. These themes, categorized in clockwise order, are Reasons why developers use bots; Bot users; Benefit from the use of bots; Activities that apply bots; Types of bots, classified by the platform for their use, and Lebeuf et al.'s Taxonomy [9]; Mechanisms for the implementation of bots; Problems encountered with the use of bots; and Areas of improvement proposed by authors. Each lower-order theme is linked to other themes in different categories.

5. Discussion

This section presents a discussion regarding the results associated with each RQ. A significant general observation is that primary studies on bots applied to software development before 2019 are scarce. The surge of research articles afterward can be attributed in part to the creation of the first international workshop on using Bots in software engineering (BotSE), a subsidiary of ICSE. It is apparent from the results that this workshop significantly boosted the research on this topic. It is also interesting to note that following the surge in 2019, in the following years, primary studies decreased slightly. This may be due to the disruptions caused by COVID-19 at a global level from 2020 onwards.

5.1 Software development activities where bots are involved

The most popular activity found is project management, in which traceability of actions and communication to facilitate collaboration between members is crucial. Other aspects include monitoring the development activities performed by a team and maintaining control in online repositories such as GitHub, where source code and documentation can be uploaded. Another frequently cited activity is onboarding, where the team members receive training, either when a new member joins and needs to understand the context of the project or even to learn the workflow of the rest of the team. Notably, bots can also support activities related to source code, such as coding,

static analysis, debugging, and code refactoring; but in this kind of activity, bots also present some issues.

The results are likely to be expanded as more types of bots arise. We believe new bots will appear focusing on other software development activities not seen in the results, such as bots for software validation, generation of test suites, monitoring a Personal Software Process (PSP), or other software development activities. This will significantly increase the usage and opportunities of bots in the context of software development.

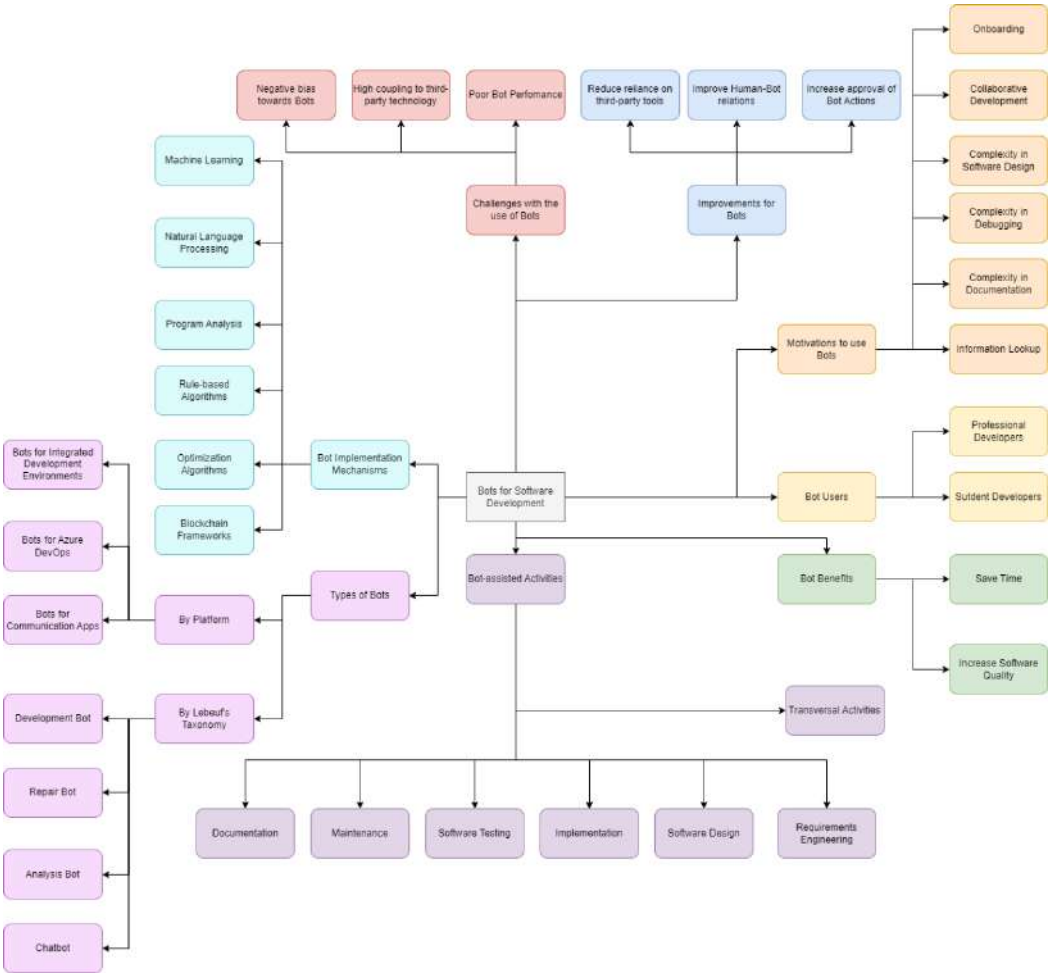


Fig. 8. Thematic map of higher-order themes about bots

5.2 Benefits of applying a bot in a development activity

The most widely reported benefit of using bots is time-saving, as bots can perform tasks or assist developers, decreasing the time to complete different tasks. As mentioned earlier, bots for project management are the most popular, helping to save time, although they come with a learning curve and have inherent drawbacks related to adopting new tools for development. As an example of time-saving, bots for repository management activities, such as organizing pull requests, code reviews, and project builds, have the main benefit of saving time by reducing developer workload and overseeing menial and relatively straightforward tasks.

A limited benefit of bots is that they can provide helpful information through recommendations. This benefit applies to bots that identify bugs and recommend a suitable fix. The caveat is that sometimes the bot's solutions are incorrect. Therefore, humans should always review bot solutions before committing them to a repository.

Another benefit we want to highlight comes from bots for onboarding. These bots help new developers in a team to learn about the project domain and workflow of existing team members so that they can integrate into the team with minimal disruptions and better productivity. We believe that these bots could be adopted to teach SE students about the context of professional software development since they will be the new software developers in a team.

The benefits provided by bots depend on the functions performed by them related to development activities. Generally, the more involved a bot is in an activity, the greater the benefits. However, a developer should weigh the risks that come with the use of bots.

5.3 Problems that arise in bot-aided development activities

We also analyzed challenges encountered by adopting bots for software development to contrast benefits. The most frequent is the poor performance of the bot contributions. The recommendations or warnings bots provide can sometimes be of little to no use to a developer.

There can also be developers misunderstanding regarding the scope and limitations of bots. This, in turn, leads to developer frustration, causing reluctance to use bots.

A different technical limitation is that bots cannot communicate on the same verbal level as humans, although significant advances in chat and voice bots aim to close this gap, as is the case in emerging technologies such as ChatGPT.

We believe that the difficulties that arise from using bots are inherent to the learning curve associated with them. This is especially true for some chatbots, as they need structured commands to perform tasks, requiring developers' memorization of them to be productive. In this sense, we can argue that bots have not yet fully achieved the goal of becoming seamless developer assistants.

5.4 Bots' intelligence level for software development

Regarding the intelligence of bots, we observed that, from the pool of 83 studies, most of the bots display all 3 of the intelligence metrics. These metrics, following [9], are the ability to exhibit Adaptability, Reasoning, and Autonomy. However, in a minority of cases, bots either did not present a particular intelligence or the study did not mention it, resulting in ambiguity, not allowing us to determine if a bot had or did not have a metric for intelligence.

We also note that these intelligence criteria were subject to our interpretation since, in most primary studies, intelligence metrics are not explicitly mentioned. Thus, we derived the appropriate intelligence metric from the bot function, the interactions of the bot with humans, and the bot implementation mechanisms.

6. Conclusions

In this paper, we conducted a literature review on bots in Software Engineering. The study selection consisted of an automatic search, followed by a supplementary Snowballing search and a quality evaluation filter. From this process, we identified 83 primary studies which answered our RQs. The primary objectives of our RQs were to outline SE activities supported by bots and the benefits and challenges that this support entails.

We found in our research that bots are mainly used for project management, to automate tasks with a low level of abstraction, such as tagging pull requests and commits, assigning team members to code reviews, performing static code analyses, and tracking changes in project repositories.

The primary benefit of including bots in software development is time-saving, automating tasks, and expediting activities such as debugging through bug identification and solution recommendation.

Time-saving in these activities means developers have more time for complex development activities related to Requirements Elicitation, Design, and Implementation.

Despite the benefits, challenges such as low bot effectiveness and performance arise. These problems come along with a perceived high learning curve for using bots, making it challenging to use bots to their full extent and understand their limitations. All the mentioned problems diminish time-saving and may detract software developers, but we believe that developers and researchers will overcome such challenges in the future as better and more sophisticated Artificial Intelligence techniques appear.

Our research points out that most bots exhibit, to some extent, adaptability, and autonomy, which may indicate that most bots have a high level of intelligence. Even so, limitations in language processing still prevent bots from being true developer assistants with the capacity to communicate and participate in the development as team member colloquially. However, the mentioned limitations may soon change thanks to breakthrough new technologies like ChatGPT.

As for future work, we envision delving deeper into the mechanisms and Artificial Intelligence principles in which bots for software development rest. This complements the current study, allowing us to associate trends in Artificial Intelligence with Software Engineering advancements to create a better picture for software developers and researchers. We believe software development assisted by Artificial Intelligence will be an ongoing trend in Software Engineering, like sign and overflow detection.

References

[1]. Nagaria B., Hall T. How software developers mitigate their errors when developing code. IEEE Transactions on Software Engineering, 2020.

[2]. Suta P., Lan X., Wu B., Mongkolnam P., Chan J. An overview of machine learning in chatbots. International Journal of Mechanical Engineering and Robotics Research, vol. 9, no. 4, pp. 502–510, 2020.

[3]. Rainey S. K., Brown B., Kirk D. B. Bots, natural language processing, and machine learning. Tax Executive, vol. 69, p. 39, 2017.

[4]. Lebeuf C. R. A taxonomy of software bots: towards a deeper understanding of software bot characteristics. Ph.D. thesis, 2018.

[5]. Lebeuf C., Zagalsky A., Foucault M., Storey M.-A. Defining and classifying software bots: A faceted taxonomy. in 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), pp. 1–6, IEEE, 2019.

[6]. Erlenhov L., de Oliveira Neto F. G., Scandariato R., Leitner P. Current and future bots in software development. in 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), pp. 7–11, IEEE, 2019.

[7]. Orgeolet L., Foulquier N., Misery L., Redou P., Pers J.-O., Devauchelle-Pensec V., Saraux A. Can artificial intelligence replace manual search for systematic literature? Review on cutaneous manifestations in primary Sjögren's syndrome. British Journal of Rheumatology, vol. 59, no. 4, pp. 811–819, 2020.

[8]. A. Ciupe, S. Meza, and B. Orza, "Systematic assessment of interactive instructional technologies in higher engineering education," in International Conference on Interactive Collaborative Learning, pp. 797–804, Springer, 2020.

[9]. Kitchenham P., Budgen D., Brereton P. (2015). Evidence-based software engineering and systematic reviews. CRC Press, DOI: 10.1201/b19467.

[10]. Zhang H., Babar M. A., Tell P. (2011). Identifying relevant studies in software engineering. Information and Software Technology, 53(6), 625–637. DOI: 10.1016/j.infsof.2010.12.010.

[11]. Moguel-Sánchez R., Martínez-Palacios C. S., Ocharán-Hernández J. O., Limón X., Sanchez Garcia A. J. (2022). Zenodo: Bots and their Uses in Software Development: A Systematic Mapping Study. [Data Set] https://doi.org/10.5281/zenodo.7872403.

[12]. Wohlin C. (2014). Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering. Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering.

[13]. Cruzes D., Dybå T. Recommended Steps for Thematic Synthesis in Software Engineering, Dept. of Computer and Information Science, Trondheim, Norway, 2011.

[14]. Matthies C., Dobrigkeit F., Hesse G., An additional set of (automated) eyes: chatbots for agile retrospectives. In 2019 IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE), pp. 34–37, IEEE, 2019.

[15]. Balachandran V. Reducing human effort and improving quality in peer code reviews using automatic static analysis and reviewer recommendation. In 2013 35th International Conference on Software Engineering (ICSE), pp. 931–940, IEEE, 2013.

[16]. McConnell S. Professional software development: shorter schedules, higher quality products, more successful projects, enhanced careers. Addison-Wesley, 2004.

Информация об авторах / Information about authors

Рикардо МОГЕЛЬ-САНЧЕС имеет степень бакалавра по программной инженерии. С 2024 года является профессором Младшего колледжа Маффлс. Сфера научных интересов: кибербезопасность, искусственный интеллект, технологическая доступность.

Ricardo MOGUEL-SÁNCHEZ – Bachelor of Science in Software Engineering. Professor, Muffles Junior College since 2024. His research interests include Cybersecurity, Artificial Intelligence, and Technological Accessibility.

Сесар Серхио МАРТИНЕС-ПАЛАСИОС окончил Университет Веракруса, получив степень бакалавра по программной инженерии. Работает над комплексным созданием веб-продуктов по технологии .NET. Сфера научных интересов: искусственный интеллект, технологическая доступность, трехмерная графика.

César Sergio MARTÍNEZ-PALACIOS – graduated from the University of Veracruz with a Bachelor’s degree in Software Engineering. Dedicated to full stack .NET development in different environments. With Research interest in Artificial intelligence, accessibility and 3D Graphics.

Хорхе Октавио ОЧАРАН-ЭРНАНДЕС – профессор факультета статистики и информатики Университета Веракруса. Получил степень бакалавра по стратегическим информационным технологиям в кампусе Веракруса сети университетов Анауака, степень магистра по программной инженерии и степень PhD по программированию в Университете Веракруса. В сферу его научных интересов входят программная инженерия, инженерия требований, проектирование и разработка архитектуры программного обеспечения, разработка прикладных программных интерфейсов, гуманитарные аспекты программной инженерии. Является членом ассоциаций ACM и IEEE.

Jorge Octavio OCHARÁN-HERNÁNDEZ – is a Full Professor at the Faculty of Statistics and Informatics of Universidad Veracruzana. He holds a Ph.D. in Computer Science and a Master’s in Software Engineering from Universidad Veracruzana and a BC in Strategic Information Technologies from Universidad Anáhuac Campus Veracruz. His research interests include Software Engineering, Requirement Engineering, Software Design and Architecture, API Design, and Human Aspects of Software Engineering. He is a member of the ACM and IEEE Computer Society.

Ксавьер ЛИМОН – профессор факультета статистики и информатики Университета Веракруса. Является членом мексиканской Национальной системы поддержки исследователей. Имеет степень бакалавра по информатике, а также степени магистра и PhD по искусственному интеллекту. Сфера научных интересов: мультиагентные системы, добыча данных, кибербезопасность и распределенные системы. Является автором многочисленных публикаций в этих предметных областях.

Xavier LIMÓN – works as a full-time professor at the Faculty of Statistics and Informatics at the University of Veracruz. He is currently a member of the National System of Researchers. He holds a Bachelor's degree in Informatics, as well as a Master's and Doctorate degree in Artificial Intelligence. His areas of research interest include Multi-Agent Systems, Data Mining, Cybersecurity, and Distributed Systems. He has numerous publications in these fields.

Анхель Хуан САНЧЕС-ГАРСИЯ имеет степень PhD по искусственному интеллекту. Профессор и исследователь Школы статистики и информатики Университета Веракруса с 2012 года. Научные интересы: машинное обучение в приложении к программной инженерии,

эволюционные вычисления, измерения программного обеспечения, машинное зрение и робототехника.

Angel Juan SÁNCHEZ-GARCÍA – Doctor of Artificial Intelligence, Professor and researcher of the School of Statistics and Informatics of the Universidad Veracruzana since 2012. Research interests: machine learning applied to software engineering, evolutionary computation, software measurement, computer vision and robotics.

DOI: 10.15514/ISPRAS-2024-36(1)-14

Determining Relevant Risk Factors for Breast Cancer

Z.J. Ibarra-Cuevas, ORCID 0000-0002-0084-2393 <*zaziil.97@gmail.com*>
J.I. Nunez-Varela, ORCID 0000-0002-9633-3453 <*jose.nunez@uaslp.mx*>
A. Nunez-Varela, ORCID 0000-0003-4813-8992 <*alberto_snv@hotmail.com*>
F.E. Martinez-Perez, ORCID 0000-0002-3133-9045 <*eduardo.perez@uaslp.mx*>
S.E. Nava-Muñoz, ORCID 0000-0001-9345-4391 <*senavam@uaslp.mx*>
C.A. Ramirez-Gamez, ORCID 0000-0002-1509-0980 <*crgamez@uaslp.mx*>
H.G. Perez-Gonzalez, ORCID 0000-0003-3331-2230 <*hectorgerardo@uaslp.mx*>
School of Engineering, Universidad Autónoma de San Luis Potosí, San Luis Potosí, México.

Abstract. Breast cancer is a serious threat to women's health worldwide. Although the exact causes of this disease are still unknown, it is known that the incidence of breast cancer is associated with risk factors. Risk factors in cancer are any genetic, reproductive, hormonal, physical, biological, or lifestyle-related conditions that increase the likelihood of developing breast cancer. This research aims to identify the most relevant risk factors in patients with breast cancer in a dataset by following the *Knowledge Discovery in Databases* process. To determine the relevance of risk factors, this research implements two feature selection methods: the *Chi-Squared test* and *Mutual Information*; and seven classifiers are used to validate the results obtained. Our results show that the risk factors identified as the most relevant are related to the age of the patient, her menopausal status, whether she had undergone hormonal therapy, and her type of menopause.

Keywords: data mining; breast cancer; risk factors.

For citation: Ibarra-Cuevas Z.J., Nunez-Varela J.I., Nunez-Varela A., Martinez-Perez F.E., Nava-Muñoz S.E., Ramirez-Gamez C.A., Perez-Gonzalez H.G. Determining Relevant Risk Factors for Breast Cancer. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 225-238. DOI: 10.15514/ISPRAS-2024-36(1)-14.

Full text: Ibarra-Cuevas Z.J., Nunez-Varela J.I., Nunez-Varela A., Martinez-Perez F.E., Nava-Muñoz S.E., Ramirez-Gamez C.A., Perez-Gonzalez H.G. Determination of Relevant Risk Factors for Breast Cancer Using Feature Selection. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 671–681. DOI: 10.1134/S0361768823080091.

Определение релевантных факторов риска для рака молочной железы

С.Х. Ибарра-Куэвас, ORCID 0000-0002-0084-2393 <*zaziil.97@gmail.com*>
Х.И. Нунес-Варела, ORCID 0000-0002-9633-3453 <*jose.nunez@uaslp.mx*>
А. Нунес-Варела, ORCID 0000-0003-4813-8992 <*alberto_snv@hotmail.com*>
Ф.Э. Мартинес-Перес, ORCID 0000-0002-3133-9045 <*eduardo.perez@uaslp.mx*>
С.Э. Нава-Муньос, ORCID 0000-0001-9345-4391 <*senavam@uaslp.mx*>
С.А. Рамирес-Гамес, ORCID 0000-0002-1509-0980 <*crgamez@uaslp.mx*>
Э.Х. Перес-Гонсалес, ORCID 0000-0003-3331-2230 <*hectorgerardo@uaslp.mx*>

Инженерная школа Автономного университета Сан-Луис-Потоси, Сан-Луис-Потоси, Мексика.

Аннотация. Рак молочной железы представляет собой серьезную угрозу для здоровья женщин во всем мире. Хотя точные причины этого заболевания до сих пор неизвестны, известно, что заболеваемость раком молочной железы связана с некоторыми факторами. Факторы риска при раке – это любые генетические, репродуктивные, гормональные, физические, биологические или связанные с образом жизни состояния, которые увеличивают вероятность развития рака молочной железы. Настоящее исследование направлено на выявление наиболее значимых факторов риска у пациентов с раком молочной железы по набору данных, следуя процессу «Обнаружение знаний в базах данных». Чтобы определить актуальность факторов риска, реализованы два метода отбора признаков: критерий Хи-квадрат и взаимная информация; для проверки полученных результатов используются семь классификаторов. Результаты показывают, что наиболее важные факторы риска связаны с возрастом пациентки, ее менопаузальным статусом, прохождением гормональной терапии и типом менопаузы.

Ключевые слова: добыча данных; рак молочной железы; факторы риска.

Для цитирования: Ибарра-Куэвас С.Х., Нунес-Варела Х.И, Нунес-Варела А., Мартинес-Перес Ф. Э., Нава-Муньос С.Э., Рамирес-Гамес С.А., Перес-Гонсалес Э.Х. Определение релевантных факторов риска для рака молочной железы. Труды ИСП РАН, том. 36, вып. 1, 2024. стр. 225-238 (на английском языке). DOI: 10.15514/ISPRAS-2024-36(1)-14.

Полный текст: Ибарра-Куэвас С.Х., Нунес-Варела Х.И, Нунес-Варела А., Мартине-Перес Ф.Э., Нава-Муньос С.Э., Рамирес-Гамес С.А., Перес-Гонсалес Э.Х. Определение релевантных факторов риска для рака молочной железы на основе отбора признаков. Programming and Computer Software, 2023, т. 49, № 8, стр. 671–681 (на английском языке). DOI: 10.1134/S0361768823080091.

1. Introduction

Globally, breast cancer is the most common and widespread type of cancer among women with more than 2.2 million new cases and about 680,000 deaths in 2020, according to the Global Cancer Observatory [1]. The early detection of breast cancer is key to increase the chance of treatment and recovery; this is normally done by screening tests, such as a mammography. Studies have also identified what are known as risk factors, that are associated with the likelihood of developing breast cancer. There are a wide variety of risk factors that include genetic, reproductive, hormonal, physical, biological, lifestyle-related, among others [2]. It is important to analyze and understand the possible impact each factor could have in the development of breast cancer so that physicians could suggest preventive strategies to women who are known to have some of these risk factors.

A common trend in recent years is the analysis of data obtained from clinical records [3, 13, 14]. This has been achieved by using methodologies that extract potentially valuable information. Knowledge Discovery in Databases (KDD) [4] is a process that follows different phases or stages (Figure 1), such as selection, preprocessing and transformation of data, so that machine learning

methods could be applied with the aim of classifying information (prediction) or identifying new knowledge (discovery).

In this research we follow the KDD process, and our main contribution is the integration of feature selection methods and ensemble learning algorithms to determine and validate relevant risk factors from a breast cancer dataset. The most relevant factors identified are related to the patient's age, whether she had undergone hormone therapy, her type of menopause, and her menopausal status.

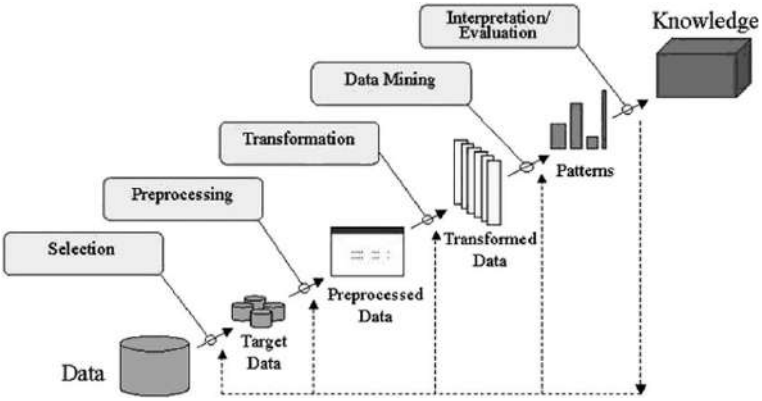


Fig. 1. Knowledge Discovery in Databases Process (taken from [4])

Being able to determine whether there is a risk of breast cancer or not solely from information readily known to most people is an important tool that would be widely available without the need to have specialized equipment. Of course, this is not meant to substitute screening tests and the knowledge of medical personnel. However, these tools could provide useful information and be part of the strategies for breast cancer risk control.

The rest of this paper is organized as follows. Section 2 reviews the related works for determining breast cancer risk factors. Section 3 explains the dataset used in this research. Section 4 describes the data pre-processing stage. Section 5 explains how the relevant risk factors are selected. Section 6 shows the results of classification methods on the dataset. Section 7 presents the validation of those selected risk factors. Section 8 provides our final conclusions.

2. Related work

Li et al. [5] present a prevention and control system for breast cancer by means of item rule association algorithms applied on a private dataset with 2,966 records and 83 attributes. An important characteristic of their work is the creation of their own dataset by interviewing patients from 22 hospitals over a one-year period and storing clinical, personal, and socio-economical information. Three types of rules defining the more relevant risk factors were identified; 35 rules were obtained using a single factor, 19 rules were obtained combining two factors, and 9 rules were obtained combining three factors. The main difference with our work is the creation of their own dataset, that provides more information and control. Kabir et al. [6] also generated risk factor rules by means of association rule mining, using the Breast Cancer Surveillance Consortium's (BCSC)[1] Risk Factors dataset. This public dataset contains 6,318,638 cases and 13 attributes, although all records containing at least one missing value were discarded. The logit model was used to select those factors that may affect the likelihood of breast cancer. A set of 5 rules was obtained for breast cancer cases and 4 rules for non-cancer cases. However, because of the class imbalance problem, they had to adjust the algorithm for the breast cancer cases.

[1] Breast Cancer Surveillance Consortium page: https://www.bcsc-research.org

The class imbalance is a problem that is commonly found in cancer-related datasets, since there are fewer positive cases compared to the number of negative cases. Kabir and Ludwig [7] focused on this issue by implementing six data-level resampling approaches. These techniques were applied on the BCSC's Risk Factors dataset, after discarding all records containing at least one missing value. The authors used three different classification algorithms: Decision Tree, Random Forest, and XGBoost. Their results showed that performance improves when resampling techniques are used compared to when no techniques are applied. The difference with our work is that we use a resampling approach at the algorithm level.

In summary, the main distinction between the described works and ours is that we make use of feature selection methods, a resampling technique, and use classification to validate the relevance of the selected risk factors.

3. Data selection

The breast cancer dataset used in this research was obtained from the Breast Cancer Surveillance Consortium (BCSC)[2]. The BCSC provides four datasets related to risk factors of breast cancer. For our analysis, the Risk Estimation (v.2) dataset [8] was selected (with information ranging from 1996-2002) for three reasons: i) it provides an attribute indicating the presence of breast cancer, that is used to classify each case, ii) it contains information about 11 risk factors, and iii) patients had no previous diagnosis of breast cancer up until the screening test recorded in the dataset. This last point is important because we are interested in determining relevant risk factors when no cancer has been diagnosed before. For instance, the Risk Factors (v.2) dataset also includes information of patients that have had cancer at some point in their life. This dataset could be useful to analyze the relationship between risk factors in women that have had cancer and those that have not.

Table 1 contains the description of the 16 attributes within the Risk Estimation (v.2) dataset and the values that can be assigned to each attribute, as well as their meaning. Table 2 shows the number of breast cancer cases, and their corresponding percentage, within the Risk Estimate (v.2) dataset. In total the dataset contains 1,007,660 cases. However, notice the difference between positive cancer (0.73%) and non-cancer (99.27%) cases. This imbalance in the data is an issue commonly present in this type of problems and will be further discussed in Section 6.

4. Data preprocessing

The preprocessing phase for our research consisted in taking the original dataset and apply four different operations.

4.1 Simple conversion operations

First, we converted all data types from numerical to categorical, except the count attribute which remained as a numeric attribute. Second, we converted all 9 values to the categorical value of unknown in all attributes that contain this value (i.e., attributes 1 and 3 to 12).

4.2 Attribute transformation

After analyzing the values of three attributes, specifically, value 1 of the menopause attribute, value *9* of the *surgmeno* attribute, and value *9* of the *hrt* attribute (attributes 1, 11 and 12 in Table 1 respectively); we decided to transform these three attributes to clarify the information given by those values. For the *menopause* attribute, value *1* refers to postmenopausal women *or* women of more than 55 years old. It is possible to identify true postmenopausal cases by means of the *surgmeno*

[2] Data collection and sharing was supported by the National Cancer Institute-funded Breast Cancer Surveillance Consortium (HHSN261201100031C). http://www.bcsc-research.org/

attribute. If the *surgemeno* attribute contains a *0* or *1*, it means that the record refers to a postmenopausal woman, and these records are assigned a value of *1* in the *menopause* attribute. A new value *2* was created and assigned to those cases where it is not possible to define whether a woman is postmenopausal or is older than 55 years. The attribute was renamed as *menopause_new* to differentiate from the original (see Table 3). Originally, value *1* was assigned to 140,843 records; after the transformation 107,810 records were detected as true postmenopausal cases (that were left with a value of *1*), and the rest were assigned the new value of *2*.

Table 1. Description of attributes of the Risk Estimation (v.2) dataset

No.	Attribute	Description	Values
1	*menopause*	Menopausal status	0 = premenopausal 1 = postmenopausal or age>=55 9 = unknown
2	*agegrp*	Age (years) in 5-year groups	1 = 35-39 2 = 40-44 3 = 45-49 4 = 50-54 5 = 55-59 6 = 60-64 7 = 65-69 8 = 70-74 9 = 75-79 10 = 80-84
3	*density*	BI-RADS breast density codes	1 = Almost entirely fat 2 = Scattered fibro glandular densities 3 = Heterogeneously dense 4 = Extremely dense 9 = Unknown or different measurement system
4	*race*	Race	1 = white 2 = Asian/Pacific Islander 3 = black 4 = Native American 5 = other/mixed 9 = unknown
5	*hispanic*	Patient is Hispanic	0 = no 1 = yes 9 = unknown
6	*bmi*	Body mass index	1 = 10-24.99 2 = 25-29.99 3 = 30-34.99 4 = 35 or more 9 = unknown
7	*agefirst*	Age at first birth	0 = Age < 30 1 = Age 30 or greater 2 = Nulliparous 9 = unknown
8	*nrelbc*	Number of first-degree relatives with breast cancer	0 = zero 1= one 2 = 2 or more 9 = unknown
9	*brstproc*	Previous breast procedure	0 = no 1 = yes 9 = unknown
10	*lastmamm*	Result of last mammogram before the index mammogram	0 = negative 1 = false positive 9 = unknown
11	*surgmeno*	Type of menopause	0 = natural 1 = surgical 9 = unknown or not menopausal (menopause=0 or menopause=9)
12	*hrt*	Current hormone therapy	0 = no 1 = yes 9 = unknown or not menopausal (menopause=0 or menopause=9)
13	*invasive*	Diagnosis of invasive breast cancer within one year of the index screening mammogram	0 = no 1 = yes

14	*cancer*	Diagnosis of invasive or ductal carcinoma in situ breast cancer within one year of the index screening mammogram	0 = no 1 = yes
15	*training*	Training data	0 = no (validation) 1 = yes (training)
16	*count*	Frequency count of this combination of covariates and outcomes (all variables 1 to 15)	

Table 2. Distribution of positive and non-cancer cases

Breast Cancer Diagnosis	Cases	%
Yes	7,319	0.73
No	1,000,341	99.27
Total	1,007,660	100

For the *surgmeno* attribute, value *9* is given to women that have not undergone menopause yet *or* the status of menopause is unknown. A new value *2* was created to refer to cases that are still not menopausal by checking if the *menopause* attribute is *0*. The attribute was renamed as *surgmeno_new* to differentiate from the original (see Table 3). Originally, value *9* was assigned to 83,545 records; after this operation 29,542 records were given the value of *2*, and 54,003 remained as *unknown*.

Similarly, for the *hrt* attribute, the same value *9* is assigned to cases that have not presented menopause *or* to cases where the use of hormone restitution therapy is unknown. A new value *2* was created to refer to cases that are still not menopausal by checking if the *menopause* attribute is *0*. The attribute was renamed as *hrt_new* to differentiate from the original (see Table 3). Originally, value *9* was assigned to 64,489 records; after this operation 29,542 records were given the value of *2*, and 34,947 remained as *unknown*.

Table 3. New attributes after being transformed

Attribute	Values
menopause_new	0 = premenopausal 1 = postmenopausal 2 = postmenopausal or age>=55 9 = unknown
surgmeno_new	0 = natural 1 = surgical 2 = not menopausal 9 = unknown or unknown menopausal (menopause=9)
hrt_new	0 = no 1 = yes 2 = not menopausal 9 = unknown or unknown menopausal (menopause=9)

4.3 Attribute removal

Three attributes were removed from the dataset. The *invasive* attribute, that refers to the diagnosis of invasive or ductal carcinoma, was not considered due to the causality of correlation with the *cancer* attribute of interest. The *training* attribute suggests whether that record in the dataset is to be considered for training or validation. However, because of the next transformations to be described we cannot use this division of records, thus the attribute is removed. Finally, the *last_mammogram* attribute indicates the result of the last mammogram taken before the index mammogram that relates to the *cancer* attribute. Since it only contains information about negative and false positive results, then, it can be removed without affecting our analysis.

4.4 Elimination of records with unknown values

Most of the attributes, as shown in Table 1, contain the *unknown* value. After careful analysis we decided to remove all records containing one or more *unknown* values and work only with records containing true values. After this operation, out of the 1,007,660 cases in the dataset (see Table 2), we are left with 160,390 cases.

5. Risk factors selection

To determine the ranking of attributes, this research makes use of two feature selection methods: *Chi-squared test* and *Mutual Information*.

5.1 Chi-squared test

The *Chi-squared test* is a nonparametric statistical technique used to determine if a distribution of observed frequencies differs from the theoretical expected frequencies [9]. Table 4 presents the *Chi-squared* values obtained for each of the 11 risk factors within the dataset. The values are sorted in descending order. The higher the value of an attribute the more relevant it is considered. We also verified the resulting values with a confidence of 95% (p-value of 0.05). Attributes from 1 to 9 are statistically significant at the 0.05 level. Only attributes 10 and 11 are not statistically significant.

According to the obtained values the first four attributes could be considered as more relevant, i.e., the patient's age (*agegrp*), whether she had undergone hormone therapy (*hrt_new*), her type of menopause (*surgmeno_new*), and her menopausal status (*menopause_new*). The next two attributes are also interesting, whether the patient have had a breast procedure (*brstproc*) and the patient's breast density (*density*). The rest of the attributes could be considered less relevant for this specific dataset.

Table 4. Chi-squared results for all risk factors

No.	Attribute	Chi-squared
1	*agegrp*	170.285
2	*hrt_new*	84.667
3	*surgmeno_new*	82.352
4	*menopaus_new*	82.306
5	*brstproc*	49.163
6	*density*	40.555
7	*nrelbc*	21.018
8	*Hispanic*	16.404
9	*agefirst*	6.721
10	*race*	4.456
11	*bmi*	1.374

5.2 Mutual Information

Mutual Information [10] is calculated between two variables and measures the reduction in uncertainty for one variable given a known value of the other variable. Table 5 presents the values obtained from the *Mutual Information* with normalization. Again, the values are sorted in descending order. The higher the value of an attribute the more relevant it is considered. Here, a threshold (cutoff) value was calculated in order to determine which attributes should be selected. Our threshold value was calculated by means of the standard deviation (*S*). For an attribute to be selected, its *Mutual Information* value must be greater than the threshold value *S*. In this case, only the first four attributes are greater than our calculated $S = 0.00022$. Notice that these four selected attributes are the same most relevant calculated by the *Chi-squared test*. The rest of the attributes have a similar ranking as given by the *Chi-squared test*.

Table 5. Mutual Information results for all risk factors

No.	Attribute	Mutual Information
1	*agegrp*	0.000740
2	*hrt_new*	0.000398
3	*surgmeno_new*	0.000390
4	*menopaus_new*	0.000390
5	*brstproc*	0.000202
6	*density*	0.000196
7	*Hispanic*	0.000092
8	*nrelbc*	0.000085
9	*agefirst*	0.000032
10	*race*	0.000021
11	*bmi*	0.000006

5.3 Definition of subsets of relevant attributes

To synthesize and validate the results obtained by the *Chi-Squared test* and *Mutual Information*, three subsets are defined based on the values given in the rankings of both methods as seen in Table 6.

Table 6. Attributes of the defined subsets

Subset ID	Attributes
Subset(4)	*{agegrp, hrt_new, surgmeno_new, menopause_new}*
Subset(7)	*{Subset(4), brstproc, density, nrelbc}*
Subset(11)	*{Subset(7), Hispanic, agefirst, race, bmi}*

6. Imbalance classification problem

This type of problem occurs when the number of records of some class label is much larger than the other class (as shown in Table 2). This problem remains after the preprocessing phase described in Section 4, where all records with an unknown value were eliminated. The resulting dataset ended up with 95.83% of non-cancer records versus 4.17% of positive cancer records. The problem of class imbalance has been actively addressed and several techniques to deal with this problem have been proposed, both at the data-level and algorithm-level [11]. Because it is important to maintain the integrity of our dataset, we follow an algorithm-level approach by implementing an ensemble learning method, particularly the *Bagging* method [12].

The *Bagging* method creates independent and parallel sub-classifiers with a single machine learning algorithm. First, from the initial data, several subsets of the same size are generated, thus ensuring diversity and independence. Then, for each sample, a sub-classifier is constructed and, finally, using a majority vote the final classification is obtained (Fig. 2).

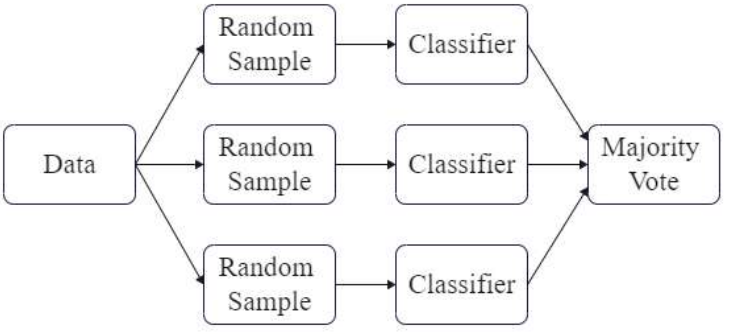


Fig. 2. Bagging diagram

Following this method, it was necessary to create a resampling of the data according to the *cancer* attribute. Fig. 3 shows the process used to perform such resampling. From the dataset, after being preprocessed, twenty-three sample groups were randomly generated, combining all the positive cancer records with a subset of the same number of randomly selected non-cancer records. Since the dataset ended up with 1,053 positive cancer records after the preprocessing phase, each sample group contains that number of records plus a random selection of 1,053 non-cancer records (2,106 records per sample group).

7. Risk factors validation

Section 5 defined two similar rankings for the risk factors within the dataset. The aim of identifying which risk factors are more relevant than others, is to use those relevant attributes to determine breast cancer cases, or at least, to pay more attention to those specific factors. In this section, experiments will be performed to determine the predictive performance of the attribute subsets as defined in Table 6, i.e., *Subset(4)*, *Subset(7)*, and *Subset(11)*, where the latter will be used as baseline for the previous two subsets. For our experiments, the RapidMiner software platform[3] was used, as it provides preprocessing procedures and the implementation of machine learning algorithms, among other features. Seven different algorithms were selected to cover multiple machine learning techniques: Decision Tree, Decision Stump, Random Tree, Deep Learning, Generalized Linear Model, Naïve Bayes, and k-NN (k-Nearest Neighbors). All algorithms were executed considering the default settings given by the software platform. To validate each subset of attributes, the seven classification algorithms were trained only with the attributes that belong to the subset being evaluated. Also, a 10-fold cross validation was used to obtain the performance metrics of accuracy, precision, and recall.

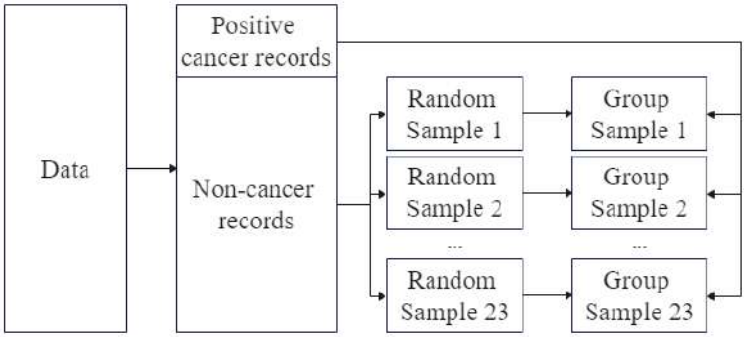


Fig. 3. Resampling process for the class imbalance problem

Table 7 presents the results for the three subsets of attributes as defined in Table 6. The first thing to note is the column that refers to *Subset(11)*; this is our baseline, as it considers all attributes. The classifiers with the highest accuracy (Acc.) are Decision Tree and Deep Learning with 97.45% and 97.21% respectively, while the least accurate is Random Tree with 78.38%.

Table 7. Performance metrics of the subsets of relevant attributes.

Algorithm	Metric	Subset(4)	Subset(7)	Subset(11)
Decision Stump	Acc.	86.32%	86.32%	86.32%
	Prec.	99.79%	99.79%	99.79%
	Rec.	72.83%	72.83%	72.83%
Decision Tree	Acc.	86.32%	96.18%	97.45%
	Prec.	99.79%	99.82%	99.77%

[3] Rapid Miner page: https://rapidminer.com

	Rec.	72.83%	92.53%	95.12%
Random Tree	Acc.	85.24%	80.67%	78.38%
	Prec.	98.29%	94.39%	87.79%
	Rec.	72.15%	67.16%	70.08%
Deep Learning	Acc.	93.32%	96.16%	97.21%
	Prec.	99.41%	99.65%	99.52%
	Rec.	87.19%	92.65%	94.88%
Generalized Linear Model	Acc.	92.87%	95.56%	96.62%
	Prec.	99.62%	99.68%	99.71%
	Rec.	86.09%	91.44%	93.51%
Naïve Bayes	Acc.	92.51%	93.87%	93.93%
	Prec.	98.77%	98.64%	98.70%
	Rec.	86.31%	89.11%	89.16%
k-NN	Acc.	93.10%	87.27%	81.30%
	Prec.	100.00%	99.94%	99.91%
	Rec.	86.20%	74.59%	62.67%

It is important to also consider the metrics of precision (Prec.) and recall (Rec.), that provide more information with regard of the classification of positive cancer cases. The higher the precision value the fewer false positives being classified. On the other hand, the higher the recall value the more positive records are classified correctly. In our experiments for *Subset(11)*, the precision values for all algorithms are high. However, the recall value for k-NN is low, which means that only 62.67% of the positive cancer cases were correctly classified. In terms of the three metrics, Decision Tree, Deep Learning, and Generalized Linear Model obtained the best results for all attributes.

In order to validate whether the selected attributes could be truly relevant in our study, we need to compare the results against those obtained by the baseline (*Subset(11)*). First, notice that Decision Stump reported the same results for the three subsets. This is because the algorithm generates a decision tree with only one division obtained from the evaluation of one of the most significant attributes. In our case, the algorithm chose the attributes of *agegrp* and *menopause_new* as a single node, and since both attributes are part of the three subsets then the results are the same. Although these results do not provide new information, as they are the same, the algorithm does support the relevance of these two attributes as stated in Section 5.

After analyzing these results, it is possible to conclude that the four selected risk factors: the patient's age (*agegrp*), whether she had undergone hormone therapy (*hrt_new*), her type of menopause (*surgmeno_new*), and her menopausal status (*menopause_new*); are relevant for the classification of positive cancer cases.

8. Conclusions

Predicting the risk of breast cancer occurrence is an important challenge for clinical oncologists as this has a direct influence on their daily practice and clinical service. The study of risk factors for breast cancer is an option that has been investigated to create control and risk assessment strategies in women. The main objective of this research is to identify relevant risk factors that could accurately predict whether a woman can develop breast cancer or not. Our research explores two feature selection techniques, *Chi-squared test* and *Mutual Information*, combined with an ensemble method (*Bagging*) to detect breast cancer cases with information on risk factors. We found that the most relevant risk factors in breast cancer cases, according to the dataset analyzed, are the patient's age (*agegrp*), whether she had undergone hormone therapy (*hrt_new*), her type of menopause (*surgmeno_new*), and her menopausal status (*menopause_new*). These four risk factors were validated by means of seven classification algorithms. It is possible to obtain a predictive performance similar to that obtained using all 11 attributes of the dataset. These are significant results that should also be validated by physicians. It is difficult to directly compare our results with

other similar works because of the different datasets and methods being used. Datasets may contain clinical, personal, demographical, therapeutical, or pathological information, and the availability of this information and the number of attributes of each type will affect the results obtained. As future work, one of the most important issues is to have as much data as possible. We are looking at the possibility of creating our dataset in collaboration with local hospitals. Also, we are interested in exploring other feature selection methods and resampling techniques, along with other classification algorithms. We expect that this work could further advance our understanding in topics as relevant such as this.

References

[1]. Global Cancer Observatory, “Cancer Today”, https://gco.iarc.fr/today/online-analysis-pie (accessed Apr. 25, 2023).
[2]. Cancer.Net, “Breast Cancer: Risk Factors and Prevention”, https://www.cancer.net/cancer-types/breast-cancer/risk-factors-and-prevention (accessed Apr. 25, 2023).
[3]. P. H. Abreu, M. S. Santos, M. H. Abreu, B. Andrade, and D. C. Silva, “Predicting Breast Cancer Recurrence Using Machine Learning Techniques”, ACM Comput. Surv., vol. 49, no. 3, pp. 1–40, Dec. 2016, doi: 10.1145/2988544.
[4]. H. Kawano, “Knowledge Discovery and Data Mining”, J. Japan Soc. Fuzzy Theory Syst., vol. 9, no. 6, pp. 851–860, 1997, doi: 10.3156/jfuzzy.9.6_851.
[5]. A. Li et al., “Association Rule-Based Breast Cancer Prevention and Control System”, IEEE Trans. Comput. Soc. Syst., vol. 6, no. 5, pp. 1106–1114, Oct. 2019, doi: 10.1109/TCSS.2019.2912629.
[6]. M. F. Kabir, S. A. Ludwig, and A. S. Abdullah, “Rule Discovery from Breast Cancer Risk Factors using Association Rule Mining”, in 2018 IEEE International Conference on Big Data (Big Data), Dec. 2018, pp. 2433–2441, doi: 10.1109/BigData.2018.8622028.
[7]. M. F. Kabir and S. Ludwig, “Classification of Breast Cancer Risk Factors Using Several Resampling Approaches”, in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Dec. 2018, pp. 1243–1248, doi: 10.1109/ICMLA.2018.00202.
[8]. W. E. Barlow et al., “Prospective Breast Cancer Risk Prediction Model for Women Undergoing Screening Mammography”, JNCI J. Natl. Cancer Inst., vol. 98, no. 17, pp. 1204–1214, Sep. 2006, doi: 10.1093/jnci/djj331.
[9]. K. Pearson, “On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling”, London, Edinburgh, Dublin Philos. Mag. J. Sci., vol. 50, no. 302, pp. 157–175, Jul. 1900, doi: 10.1080/14786440009463897.
[10]. D. J. C. MacKay, "Information Theory, Inference & Learning Algorithms". USA: Cambridge University Press, 2002.
[11]. H. Kaur, H. S. Pannu and A. K. Malhi, "A Systematic Review on Imbalanced Data Challenges in Machine Learning: Applications and Solutions", ACM Computing Surveys, vol. 52, no. 4, pp. 1-36, 2019, doi: 10.1145/3343440.
[12]. L. Breiman, “Bagging Predictors”, Mach. Learn., vol. 24, no. 2, pp. 123–140, 1996, doi: 10.1023/A:1018054314350.
[13]. I. Volkov, G. Radchenko, and A. Tchernykh, "Digital Twins, Internet of Things and Mobile Medicine: A Review of Current Platforms to Support Smart Healthcare". Programming and Computer Software, vol. 47, pp. 578–590, 2021, doi: 10.1134/S0361768821080284.
[14]. I. Vasilev, M. Petrovskiy, I. Mashechkin, et al. "Predicting COVID-19-Induced Lung Damage Based on Machine Learning Methods". Programming and Computer Software, vol. 48, pp. 243–255, 2022, doi: 10.1134/S0361768822040065.

Информация об авторах / Information about authors

Сасиль Хосефина ИБАРРА-КУЭВАС – магистр компьютерных наук и разработчик программного обеспечения. С 2022 года работает в коммерческой компании, где ведет разработку программного обеспечения. Научные интересы: интеллектуальный анализ данных, базы данных и разработка программного обеспечения.

Zazil Josefina IBARRA-CUEVAS – Master of Science in Computer Science and software developer working at a private company since 2022. Research interests: Data mining, data bases and software engineering.

Хосе Игнасио НУНЕС-ВАРЕЛА – доктор компьютерных наук, профессор, координатор бакалаврской программы по инженерии интеллектуальных систем в Автономном университете Сан-Луис-Потоси с 2017 года. Научные интересы: машинное обучение, наука о данных, интеллектуальная робототехника.

Jose Ignacio NUNEZ-VARELA – Doctor of Computer Science, professor, coordinator of the Intelligent Systems Engineering undergraduate program at the Autonomous University of San Luis Potosi since 2017. Research interests: Machine learning, data science, intelligent robotics.

Альберто НУНЕС-ВАРЕЛА – доктор компьютерных наук и доцент Автономного университета Сан-Луис-Потоси с 2014 года. Область научных интересов: разработка программного обеспечения, вывод на основе формальных грамматик, обработка естественного языка и машинное обучение.

Alberto NUNEZ-VARELA – Doctor of Computer Science and associate professor at the Autonomous University of San Luis Potosi since 2014. Research interests: Software engineering, grammatical inference, natural language processing, and machine learning.

Франсиско Эдуардо МАРТИНЕС-ПЕРЕС – доктор компьютерных наук, профессор, координатор бакалаврской программы по программированию в Автономном университете Сан-Луис-Потоси с 2023 года. Научные интересы: обработка изображений, окружающий интеллект, повсеместные вычисления, человеко-машинное взаимодействие и медицинская информатика.

Francisco Eduardo MARTINEZ-PEREZ – Doctor of Computer Science, professor, coordinator of the Computer Engineering undergraduate program at the Autonomous University of San Luis Potosi since 2023. Research interests: Image processing, ambient intelligence (AmI), ubiquitous computing, human–computer interaction, and medical informatics.

Сандра Э. НАВА-МУНЬОС – доктор компьютерных наук, профессор, координатор аспирантской программы по информатике в Автономном университете Сан-Луис-Потоси с 2023 года. Научные интересы: разработка программного обеспечения, человеко-машинное взаимодействие, контекстно-зависимые вычисления и медицинская информатика.

Sandra E. NAVA-MUÑOZ – Doctor of Computer Science, professor, coordinator of the Computer Science postgraduate program at the Autonomous University of San Luis Potosi since 2023. Research interests: Software engineering, human-computer interaction, context aware computing, and medical informatics.

Сесар Аугусто РАМИРЕС-ГАМЕС имеет степень магистра компьютерных наук, соискатель степени доктора философии. С 2023 года работает в коммерческой компании, где ведет разработку программного обеспечения. Научные интересы: компьютерное зрение, обработка изображений и машинное обучение.

César Augusto RAMÍREZ-GÁMEZ – Master of Science in Computer Science, his Ph.D. degree, and software developer working at a private company since 2023. Research interests: Computer vision, image processing, and machine learning.

Эктор Херардо ПЕРЕС-ГОНСАЛЕС – штатный профессор-исследователь Автономного университета Сан-Луис-Потоси (Мексика), имеет ученую степень доктора компьютерных

наук. Автор научных статей и глав в книгах по автоматизации проектирования программного обеспечения и человеко-машинного взаимодействия, выступал с научными докладами на международных конференциях в США, Канаде, Великобритании, Португалии и в Сингапуре. Область научных интересов: проектирование программного обеспечения, преподавание методов разработки программного обеспечения, обработка цифровых изображений, разработка программного обеспечения для квантовых компьютеров.

Hector Gerardo PEREZ-GONZALEZ – Full-time research professor at Universidad Autónoma de San Luis Potosi, Mexico. PhD in Computer Science from the University of Colorado in 2003. Author of research articles and book chapters on Automatic Software Design and Human-Computer Interaction. He has been a speaker at international conferences in the USA, Canada, UK, Portugal, and Singapore. His research areas are software design, computer science education, and quantum software engineering. He is a member of the National Researchers System in Mexico.

DOI: 10.15514/ISPRAS-2024-36(1)-15

Software Architecture for the Development of a Collaborative Medical Activities System in the Rehabilitation of Strokes

S.I. Fernández Gregorio, ORCID: 0009-0007-0509-6530 <*zS21000291@estudiantes.uv.mx*>
L.G. Montané-Jiménez, ORCID: 0000-0003-2732-5430 <*lmontane@uv.mx*>
C. Mezura Godoy, ORCID: 0000-0002-5386-107X <*cmezura@uv.mx*>
V.Y. Rosales-Morales, ORCID: 0000-0003-2890-3343 <*vivrosales@uv.mx*>

Universidad Veracruzana,
Veracruz, México.

Abstract. A person who has had a stroke needs rehabilitation to recover from the effects of the incident. A multidisciplinary team of experts performs rehabilitation, offering treatment from many fields, including neurology, nutrition, psychology, and physiotherapy. In the rehabilitation process, physicians interact with medical computing software and devices. The interactions represent medical activities that follow rehabilitation. Nevertheless, how specialists collaborate to do medical tasks is poorly understood using technologies since no particular means of communication enable interdisciplinary cooperation for integral rehabilitation of strokes. Therefore, we present a collaborative software architecture to assist and enable the monitoring of medical activities through multimodal human-computer interactions. The architecture has three layers: the first is to perceive interactions and monitor activities, the second is to manage information sharing and interdisciplinary access, and the third is to assess how well multidisciplinary activities were carried out. The physicians are assisted in their decision-making on the execution of the treatment plan by evaluating how the activities are carried out, which are recollected through the architecture proposed. As a result, we provide a prototype with a user-centered design that understands how the architecture supports human-computer interactions.

Keywords: Medica activities; architecture; groupware; collaborative activities; rehabilitation of stroke.

For citation: Fernández-Gregorio S.I., Montané-Jiménez L.G., Mezura-Godoy C., Rosales-Morales V.Y. Software architecture for the development of a collaborative medical activities system in the rehabilitation of strokes. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 239-250. DOI: 10.15514/ISPRAS-2024-36(1)-15.

Full text: Fernández Gregorio S.I., Montané-Jiménez L.G., Mezura Godoy C., Rosales-Morales V.Y. Architecture for Groupware Oriented to Collaborative Medical Activities in the Rehabilitation of Strokes. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 643–656. DOI: 10.1134/S0361768823080078.

Acknowledgments. This research was funded by “Universidad Veracruzana” and “Consejo Nacional de Ciencia y Tecnología” (CONACYT) through the fellowship number 478122.

Архитектура программного обеспечения для разработки системы совместной медицинской деятельности при реабилитации инсультов

С.И. Фернандес-Грегорио, ORCID: 0009-0007-0509-6530 <*zS21000291@estudiantes.uv.mx*>
Л.Х. Монтане-Хименес, ORCID: 0000-0003-2732-5430 <*lmontane@uv.mx*>
К. Месура-Годой, ORCID: 0000-0002-5386-107X <*cmezura@uv.mx*>
В.Я. Росалес-Моралес, ORCID: 0000-0003-2890-3343 <*vivrosales@uv.mx*>

Университет Веракруса, Веракрус, Мексика.

Аннотация. Человек, перенесший инсульт, чтобы оправиться от его последствий, нуждается в реабилитации, Многопрофильная команда экспертов проводит реабилитацию, предлагая многоплановое лечение, включая диету, неврологию, психологию и физиотерапию. В процессе реабилитации врачи взаимодействуют с медицинскими приборами и программным обеспечением. Такая работа представляет собой медицинскую врачебную деятельность, сопровождающую процесс реабилитации. Тем не менее, в силу отсутствия подходящих средств взаимодействия, позволяющих осуществлять междисциплинарное сотрудничество при комплексной реабилитации инсультов, способы взаимодействия специалистов, совместно решающих медицинские задачи, технологически понимаются плохо. Поэтому мы представляем совместную программную архитектуру, способную обеспечить мониторинг медицинской деятельности посредством мультимодального взаимодействия человека и компьютера. Архитектура имеет трехуровневое строение: первый уровень служит для восприятия взаимодействия и мониторинга деятельности, второй – для управления обменом информацией и междисциплинарным доступом, а третий – для оценки того, насколько хорошо были выполнены междисциплинарные мероприятия. Врачам помогают в принятии решений по выполнению плана лечения путем оценки того, как выполняются действия, которые рекомендуются предложенной архитектурой. В результате мы предоставляем прототип, спроектированный с ориентацией на пользователя, который понимает, как архитектура поддерживает взаимодействие человека с компьютером.

Ключевые слова: врачебная деятельность; архитектура; рабочая группа; совместная деятельность; реабилитация инсульта.

Для цитирования: Фернандес-Грегорио С.И., Монтане-Хименес Л.Х., Месура-Годой К., Росалес-Моралес В.Я. Архитектура программного обеспечения для разработки системы совместной медицинской деятельности при реабилитации инсультов. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 239–250 (на английском языке). DOI: 10.15514/ISPRAS-2024-36(1)-15.

Полный текст: Фернандес-Грегорио С.И., Монтане-Хименес Л.Х., Месура Годой К., Росалес-Моралес В.Я. Архитектура программного обеспечения рабочих групп, ориентированного на совместную медицинскую деятельность в реабилитации инсультов. Programming and Computer Software, 2023, т. 49, № 8, стр. 643–656 (на английском языке). DOI: 10.1134/S0361768823080078.

Благодарности. Настоящее исследование финансировалось Университетом Веракрусы и мексиканским Национальным советом по науке и технике (КОНАСИТ), стипендия № 478122.

1. Introduction

Medical activities are carried out with interactions of various devices. Also, several physicians are involved in rehabilitating patients who have suffered cerebrovascular accidents. A human-computer interaction that allows different modes of data entry is called multimodal [1]. Assume that any interaction of this kind qualifies as a recorded event. In this situation, monitoring medical actions to assess their behavior and efficiency during the event-driven rehabilitation process is feasible. This type of scenario can be represented with the design of user interfaces [2]. For this, it is important to consider that a multidisciplinary team carries out all these activities [3]. Therefore, when receiving the users' multimodal interactions; it is essential to coordinate the activities to maintain a correct

record of the executed rehabilitation process. In addition, other aspects that intervene to coordinate events must be considered, such us: How are the activities related? Who or what triggers an activity? How is user collaboration perceived? How to represent the activities to evaluate the progress of the rehabilitation?

Therefore, there is no clear conception of how specialists coordinate to carry out medical activities to evaluate the rehabilitation process of people who have suffered an ictus. Particularly, there are no specific communication channels that allow multidisciplinary collaboration during the whole rehabilitation process. Hence, not having precise coordination of medical activities in a team of specialists collaborating on rehabilitation treatment could prolong care time. Also, maintaining ineffective communication between members regarding the integral treatment activities affects the patient treatment since some indications might be contradictory, so coordination to generate an integral treatment is mandatory and must be guaranteed.

This paper proposes a new collaborative software architecture focused on monitoring medical activities in collaborative teams using a groupware approach that reinforces team coordination, ordering events, and evaluating activities involved in rehabilitating patients who have suffered ictus. Also, we developed a new prototype with a user-centered design that perceives human-computer interactions supported by the new architecture.

This paper is structured as follows. Section 2 describes the state of the art. Section 3 proposes a conceptual architecture for a collaborative system focused on stroke rehabilitation activities. Section 4 shows the user-oriented prototype with the proposed architecture as a base. Finally, section 5 presents the conclusions and future work.

2. State of the art

We explored works that consider a multidisciplinary collaboration for the complete rehabilitation of patients that have suffered a stroke, see Table 1.

The papers in the state-of-the-art review explain case studies of medical situations where collaboration is required. Mainly, they explain an analysis related to how the collaboration is carried out [6-7, 10, 14] and which patient data should be used to generate statistics [6, 9-12, 14-15]. In addition, in some works [10, 14], they establish collaboration measures to evaluate the coordination.

The works [6-7, 10, 14] explain the presence of teamwork, although only [10] and [14] contemplate collaborative activities. In contrast, [6, 9-12, 14-15] perform data treatment with statistical analysis, meta-analysis, and results graphs, but only [8-9, 13-15] have defined a workflow. Therefore, it is reflected that there is little attention to collaborative activities. On the other hand, exploring data processing is necessary since few works have considered workflows and teamwork.

Some papers [16-20] propose an entity relationship diagram to define a database. Other authors established, Workflow for data management [17, 19-20]. Three works [16-17] and [20] provide care in the subacute phase or hospital care. [19] provides care in the acute phase or primary care. [18] focuses on chronic care, which refers to follow-up in rehabilitation. Three authors [17, 19] and [20] defined a workflow, although none of these contemplated teamwork or collaborative activities. In contrast, [16] and [18], even when both included performance teamwork, only [16] approached collaborative activities.

The works [21, 23-26, 28-29] are oriented to the acute phase. [21] and [23] approached their works considering the subacute phase. Instead, [23-24, 26, 28] and [30] focused on the area of critical care. We showed that [21-25] and [27] presented data processing, and only [22, 24-25, 27] and [30] carried out collaborative activities. [21-22] and [24] use teamwork, and [24, 26-27] have a defined workflow. In the works [21-30], they proposed using desktop software, mobile applications, network services, and implementations of robotic arms, video games, and bio-robotic aids to support primary care medical decisions or assistance to develop physical or cognitive rehabilitation. These works generally have focused on primary care treatment or physical or cognitive rehabilitation; however,

they do not consider the interaction between the medical team and the patient as part of the rehabilitation encompassing medical care from various specialties, an integral rehabilitation.

Table 1. Comparison of the state of the art works
1 - Acute phase care. 2 - Medical treatment. 3 - Physical and/or cognitive rehabilitation. 4 - Teamwork. 5 - Collaborative activities. 6 - Data treatment. 7 - Defined workflows. 8 - Systems and applications.

Publication	1	2	3	4	5	6	7	8
Obana et al. [6]	✓	✓		✓		✓		
Pristipino et al. [7]		✓		✓				
Watson et al. [8]	✓	✓					✓	
Tiu et al. [9]		✓				✓	✓	
De Lecinana et al. [10]	✓	✓		✓	✓	✓		
Macisaac et al. [11]		✓				✓		
Chowdhury et al. [12]	✓	✓				✓		
Baskar et al. [13]	✓	✓					✓	
Hunter et al. [14]	✓	✓		✓	✓	✓	✓	
Daemen et al. [15]		✓				✓	✓	
Grigoriev et al. [16]		✓		✓	✓	✓		
Esensoy et al. [17]		✓				✓	✓	
Ferrante et al. [18]			✓	✓		✓		
Yang et al. [19]	✓					✓	✓	
Wantaka et al. [20]		✓				✓	✓	
Chang et al. [21]	✓	✓		✓		✓		✓
Gibson et al. [22]				✓	✓	✓		✓
Tang et al. [23]	✓	✓	✓			✓		✓
Sun et al. [24]	✓		✓	✓	✓	✓	✓	✓
Li et [25]	✓				✓	✓		✓
Ilieva et al. [26]	✓		✓				✓	✓
Park et al. [27]					✓	✓	✓	✓
Wang et al. [28]	✓		✓					✓
Ramesh et al. [29]	✓							✓
Tsoupikova et al. [30]			✓		✓			✓

We remark that monitoring and communications are central concerns in the rehabilitation process of stroke patients in the chronic phase. Only a few works have considered the chronic phase and mainly focused on the rehabilitation oriented in developing systems and applications where the patients interact or monitor a particular problem derived from the stroke, leaving behind the communication and coordination of the medical team. There is a significant appearance of works that consider data processing. However, the presence of this characteristic is essential to work with data from collaborative environments to develop applications and implement technologies. Works focused on collaborative environments have considered the development of workflow, teamwork, and collaborative activities. Nevertheless, only one work has contemplated all these characteristics. Instead, only have generated efforts to include one of two of them.

The state-of-the-art review shows a lack of consideration for the communication, collaboration, and monitoring of the activities needed to rehabilitate cerebrovascular accidents in order to evaluate how the treatments and the medical collaboration are helping the patients. As far as we know, no equivalent software or project has been proposed to follow the medical rehabilitation process considering a groupware architecture for collaborative medical activities. Therefore, in this work, we propose monitoring and evaluating the rehabilitation of cerebrovascular accidents, including technological developments and the application of emerging technologies with collaborative activities.

3. Software architecture proposal

This paper presents an architecture for software development to monitor collaborative medical activities for patients suffering from stroke consequences (see Fig. 1). We developed the architecture considering three main layers 1) User presentation layer, 2) Control layer of the multidisciplinary team, and 3) Control layer of medical activities. Besides, a component focused on data administration services is contemplated.

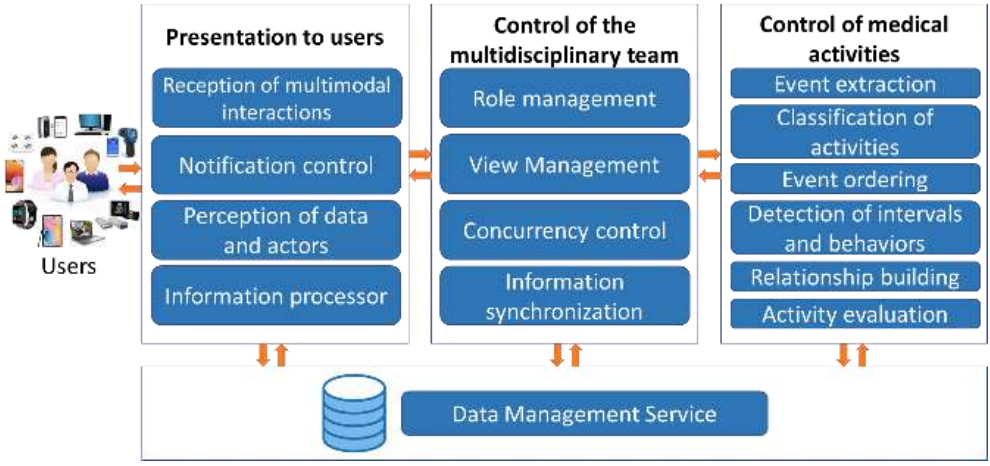


Fig. 1. Software architecture proposal

3.1 User presentation layer

The user presentation layer consists of four components. The first component, "Reception of multimodal interactions," is designed to receive data interactions from various devices. Therefore, it is considered a multimodal system, which allows data entry from different types of devices, for instance, laptops, tablets, smartphones, and smartwatches. Each human-computer interaction will be recognized as an event logged on to the devices. The data is collected through Wi-Fi or Bluetooth connections between the devices and the application to record the events in an orderly manner according to their appearance over time. Each device will enable the reading of data, such as the manual recording of information, in addition to automatic readings, such as blood pressure, heart rate, and physical activity, among other medical data that smart devices can detect.

The second component is "Notification control". Controls the sending, receiving, and status checking of notifications, that is, checking if the notification has been sent, received, and read. Notifications can be sorted by importance and use different notification channels such as email, SMS, and smartphone notifications. The notifications will be delivered to users to warn or remember situations, for example: reminding the patient of the date of a medical appointment, informing a specialist of the treatment ordered by another doctor for his patient, request for consultation from one treating doctor to another specialist, to name a few. In addition, since it is a system for medical collaboration, all the specialists who care for a patient must be kept informed of the progress of their treatment and rehabilitation. Therefore, when an activity is carried out that changes or affects this treatment, the participants must be kept informed; so it is important to control the sending and receiving of notifications to users.

The component "Perception of data and actors" involved in treating the patient refers to keeping the participants aware of all the activities carried out as part of the treatment. This component focuses on implementing awareness to perceive the changes made by the actors and controlling the identification of activities. Therefore, the system must identify the significant changes to generate notifications about the changes or actions performed.

The "Information processor" sends and receives user data through communication channels to ensure delivery. This component considers using the AES algorithm for data encryption to maintain the data's security and the information exchange between the system users.

3.2 Control layer of the multidisciplinary team

The control layer of the multidisciplinary team contains four components that aim to address the problems of coordination, collaboration, and communication. The first component is "Role management", assigning permissions and access according to the user's role. For example, a patient will have different permissions than a medic, or a specialist physician will have different permissions than a treating physician. A treating physician is in charge of carrying out the comprehensive follow-up of the patient, while a specialist only focuses on caring for the patient according to his specialty. The treating physician will decide on the appropriate information to share with each specialist.

The "View management" is a component closely linked to the previous one since it on the roles and permissions of the users, the views, or interfaces that each one can have.

We include a "Concurrency control" component focused on controlling the multiple inputs of users simultaneously since we must ensure that each user has updated data according to the activities carried out by each participant. The "Concurrency control" component controls the multiple inputs that can be given simultaneously from different devices to access the system. It provides the consistency of the information according to the events obtained from multimodal human-computer interactions.

The "Information synchronization" component is essential to maintain control of the information. All information entered or registered by users must be synchronized so that, when the data are consulted by any other member from any other device, the latest data processed should be displayed, thus maintaining the integrity of the information.

3.3 Control layer of medical activities

The control layer of medical activities includes six components necessary to process collaborative medical behavior. The first component is "Event extraction", responsible for identifying each event carried out from any device. An event occurs when an action is spontaneously generated by a user/process at any instant within the system to monitor the patient's rehabilitation.

Component, "Classification of activities", groups events into activities since an activity is a set of ordered events. This component is in charge of controlling the activities according to the established treatment. Besides, the "Event ordering" component is focused on registering and maintaining the events' order using a causal algorithm to visualize the events as they have occurred in the system, we can use works such as [31, 32]. The activities are ordered according to the events generated by the users. For example, an activity can be the request of a treating physician for a consultation with a specialist, and this activity is composed of the following events:

- Doctor: Sends the request for consultation by a specialist
- Specialist: Receives the consultation request information
- Patient: Receives the data of the request for a specialist consultation
- Patient: Sends the data of the scheduled time for a consultation with a specialist
- Specialist: Receives information on the consultation schedule
- Doctor: Receives information on the consultation schedule

The fourth component is "Detection of intervals and behaviors" to determine the beginning and end of the activities. In this component, we must identify from the set of events those that represent an interval. An interval is an activity with internal events composed of a subset of total events. On the other hand, detect behaviors refers to the set of events that define a specific activity. In this way, specific activities are detected to evaluate the collaboration in the last component.

The component “Relationships building” establishes the relationships between the sending and receiving data generated in the system. In this sense, it defines the relations between the activities. A weighted graph is used to establish the relationships between the activities.

Finally, the component “Activity evaluation” the evaluation of activities will be the component that processes the activities, events, intervals, behaviors, and relationships. Considering all these factors, it will be in charge of determining the fulfillment of the treatment activities for rehabilitation at a certain point in time. To generate the evaluation, a fuzzy cognitive map will be used to obtain the performance of the group's behavior, according to the activities carried out at a specific time. Thus, according to the evaluation, doctors will be able to analyze whether the activities carried out are effective and adequate according to the treatment recommended in the rehabilitation process.

3.4 Data administration services

This last component is responsible for storing the data and sending and receiving data. Besides, it is responsible for executing the queries and stored procedures for information management. This service must process the information in real-time to be available and accessible from any device that the multidisciplinary team uses as an access point.

4. Prototype of the medical activities

This section shows a prototype representing the events' interactions described in Table 2. The proposed architecture presented in Section 3 has been used as a developed base. Hence, the prototype reflects the action component. However, explaining the medical activities is essential to reflect the architecture in a prototype.

Table 2. Medical activities of multimodal human-computer interactions

NP	Activities
1.	Record of medical indications
2.	Access to study reports
3.	Checking attendance
4.	Measurement of vital signs: Body temperature, Pulse, Respiratory rate, respiration,
5.	Blood pressure
6.	Administrate medication
7.	Physical activity
8.	sleep monitoring
9.	SpO2 sensing
10.	Detect stress levels
11.	Attention to falls
12.	Therapies with augmented reality walks
13.	Exercises with an electrical stimulation machine of interactive therapy
14.	Record of therapies with rehabilitation equipment (balls, dumbbells, mirror boxes, putty)

The list of activities in Table 2 is designed to obtain multimodal data. For example, it is possible to detect a person's physical activity with a smartwatch. More data can be obtained from other devices used during rehabilitation (e.g., Smartphones, PCs, scales, thermometers, ultrasound machines). The medical activities in Table 2 have been considered to develop the prototype. Following, we present the prototype screens, where data on medical activities are obtained from multimodal human-computer interactions.

We remark that activity one is carried out to confirm and attend a medical appointment. Therefore, there are multimodal interactions to execute the complete activity. This activity is composed of four

events. In event one, the patient confirms the appointment from his smartphone. Besides, in event 2, the medical doctor receives the confirmation notification, as shown in Fig. 2. In event 3, the medical doctor sends the information from the medical consultation record, an interaction from a laptop. In event 4, the patient receives the information from his medical consultation, providing an exchange from a Tablet, as shown in Fig. 3.

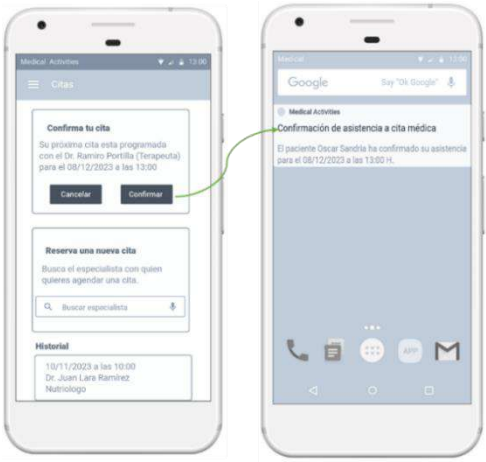


Fig. 2. Screens of the events 1 and 2

Fig. 3. Screens of events 3 and 4

Fig. 3 shows screen A, where a PC displays the medical system. In the system, the medical specialist keeps track of his patients; the doctors also record patient data and treatments. Screen B shows the treatments and medical indications the patient can consult; in this case, the device is a Tablet.

Fig. 4 shows the data collected from devices. In this case, the information from a scale is collected to obtain weight, fat percentage, and muscle, among other data. On the other hand, the data of SYS, DIA, and beats per minute from a blood pressure monitor are obtained. The information can be consulted in the medical system from another device, in this case, from a laptop. The data are collected from the devices via Wi-Fi or Bluetooth, according to the requirements of each device. These prototype screens provide multimodal human-computer interactions of medical activities considered. On the other hand, the internal behavior of the prototype considers the layers mentioned in section 3.

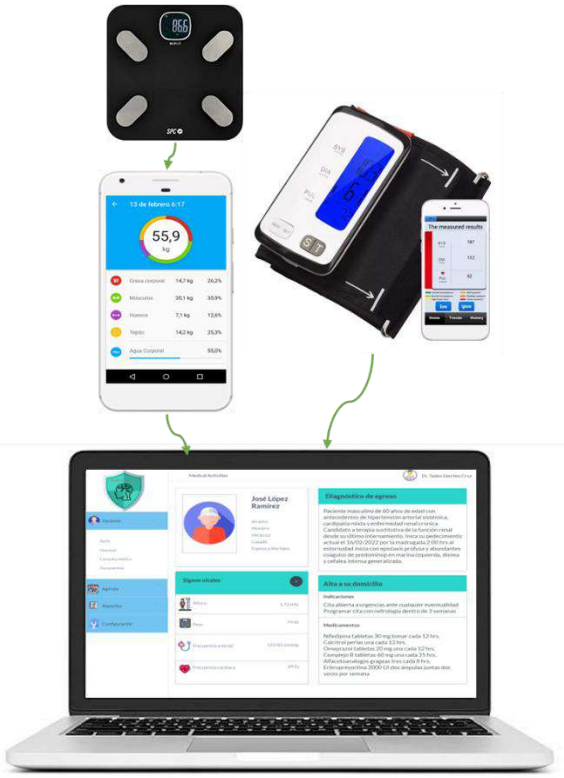


Fig. 4. Representation the data collected from devices

5. Conclusions and future work

In this paper, we proposed an architecture centered on tracking the interactions and teamwork of a medical team that treats patients with the chronic phase of stroke.

The architecture for a collaborative environment enables the collection, comprehension, and control of data produced by multimodal interactions and the coordination of a multidisciplinary team while collaborating with rehabilitating individuals who have had an ictus. We can determine how the medical activities were created to assess the state of the rehabilitation at a certain period from the events collected from the multimodal interactions. This helps the multidisciplinary team see how well the activities assessing rehabilitation progress from implementing the treatment plan created for each patient are working. With this knowledge, medical professionals might suggest modifying the rehabilitation plan or continuing to carry out the previously determined activities.

Future work will involve setting the assessment mechanism into practice with the architecture described to assist doctors who attend stroke rehabilitation in determining if patients comply with treatment regimens.

References

[1]. Interaccion. Revista digital de AIPO. Asociación Intercción Persona-Ordenador. [En línea]. Disponible en https://revista.aipo.es/index.php/INTERACCION/CFP-multimodal-HCI [Accedido: 25-abr-202

[2]. Lukin V. N., Dzyubenko A. L., Chechikov Y. B. (2020). Approaches to user interface development. Programming and Computer Software, 46, 316-323.

[3]. Alessandro L., Olmos L. E., Bonamico L., Muzio D. M., Ahumada M. H., Russo M. J., Allegri R. F., Gianella M. G., Campora H., Delorme R., Vescovo M. E., Lado V., Mastroberti L. R., Butus A., Galluzzi H. D., Décima G., Ameriso S. F. (n.d.). Rehabilitación multidisciplinaria para pacientes adultos con accidente cerebrovascular. MEDICINA (Buenos Aires) 2020, 80: 54-68.

[4]. Gregorio S. I. F., Montané-Jiménez L. G. (2022, October). Towards the improvement of computer-assisted medical activities for stroke rehabilitation. In 2022 10th International Conference in Software Engineering Research and Innovation (CONISOFT) (pp. 102-111). IEEE.

[5]. Hayyolalam V., Kazem A. A. P. (2018). A systematic literature review on QoS-aware service composition and selection in cloud environment. Journal of Network and Computer Applications, 110, 52-74.

[6]. Obana M., Furuya J., Matsubara C., Tohara H., Inaji M., Miki K., Numasawa Y., Minakuchi S., Maehara T. (2019). Effect of a collaborative transdisciplinary team approach on oral health status in acute stroke patients. "Journal of Oral Rehabilitation", 46 (12), 1170–1176. https://doi.org/10.1111/joor.12855.

[7]. Pristipino C., Anzola G. P., Ballerini L., Bartorelli A., Cecconi M., Chessa M., Donti A., Gaspardone A., Neri G., Onorato E., Palareti G., Rakar S., Rigatelli G., Santoro G., Toni D., Ussia G. P., Violini R. (2013). Management of patients with patent foramen ovale and cryptogenic stroke: A collaborative, multidisciplinary, position paper. "Catheterization and Cardiovascular Interventions", 82 (1). https://doi.org/10.1002/ccd.24637.

[8]. Watson T., Tiu J., Clissold B. (2020). Addressing inequity in acute stroke care requires attention to each component of regional workflow. "Medical Journal of Australia", 212 (1), 8-10. https://doi.org/10.5694/mja2.50440.

[9]. Tiu J., Watson T., Clissold B. (2021). Mechanical thrombectomy for emergent large vessel occlusion: an Australian primary stroke centre workflow analysis. "Internal Medicine Journal", 51 (6), 905–909. https://doi.org/10.1111/imj.14843. https://doi.org/10.1016/j.future.2017.09.061 (in press)

[10]. Alonso de Leciñana M., Fuentes B., Ximénez-Carrillo A., Vivancos J., Masjuan J., Gil-Nuñez A., Martínez-Sánchez P., Zapata-Wainberg G., Cruz-Culebras A., García-Pastor A., Díaz-Otero F., Fandiño E., Frutos R., Caniego J. L., Méndez J. C., Fernández-Prieto A., Bárcena-Ruiz E., Díez-Tejedor E. (2016). A collaborative system for endovascular treatment of acute ischaemic stroke: The Madrid Stroke Network experience. "European Journal of Neurology", 23 (2), 297–303. https://doi.org/10.1111/ene.12749.

[11]. Macisaac R. L., Khatri P., Bendszus M., Bracard S., Broderick J., Campbell B., Ciccone A., Dávalos A., Davis S. M., Demchuk A., Diener H. C., Dippel D., Donnan G. A., Fiehler J., Fiorella D., Goyal M., Hacke W., Hill M. D., Jahan R., … Lees K. R. (2015). A collaborative sequential meta-analysis of individual patient data from randomized trials of endovascular therapy and tPA vs. tPA alone for acute ischemic stroke: ThRombEctomy And tPA (TREAT) analysis: Statistical analysis plan for a sequential meta-anal. "International Journal of Stroke", 10 (A100), 136–144. https://doi.org/10.1111/ijs.12622.

[12]. Chowdhury S. Z., Baskar P. S., Bhaskar S. (2021). Effect of prehospital workflow optimization on treatment delays and clinical outcomes in acute ischemic stroke: A systematic review and meta-analysis. "Academic Emergency Medicine", 28 (7), 781–801. https://doi.org/10.1111/acem.14204.

[13]. Santana Baskar P., Cordato D., Wardman D., Bhaskar S. (2021). In-hospital acute stroke workflow in acute stroke – Systems-based approaches. "Acta Neurologica Scandinavica", 143 (2), 111–120. https://doi.org/10.1111/ane.13343

[14]. Rogers H., Madathil K. C., Joseph A., Holmstedt C., Qanungo S., McNeese N., Morris T., Holden R. J., McElligott J. T. (2021). An exploratory study investigating the barriers, facilitators, and demands affecting caregivers in a telemedicine integrated ambulance-based setting for stroke care. "Applied Ergonomics", 97. https://doi.org/10.1016/J.APERGO.2021.103537

[15]. Daemen E. M. L., Flinsenberg I. C. M., van Loenen E. J., Cuppen R. P. G., Rajae-Joordens R. J. E. (2013). Adaptive Daily Rhythm Atmospheres for stroke patients: A staff evaluation. Proceedings of the 2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops, PervasiveHealth 2013, 121–128. https://doi.org/10.4108/icst.pervasivehealth.2013.252090

[16]. Grigoriev O. G., Kobrinskii B. A., Osipov G. S., Molodchenkov A. I., Smirnov I. V. (2018). Health Management System Knowledge Base for Formation and Support of a Preventive Measures Plan. "Procedia Computer Science", 145, 238–241. https://doi.org/10.1016/J.PROCS.2018.11.050/

[17]. Esensoy, A. V., Carter, M. W. (2018). High-fidelity whole-system patient flow modeling to assess health care transformation policies. “European Journal of Operational Research”, 266(1), 221–237. https://doi.org/10.1016/J.EJOR.2017.09.019.

[18]. Ferrante S., Bonacina S., Pinciroli F. (2013). Modeling stroke rehabilitation processes using the Unified Modeling Language (UML). “Computers in Biology and Medicine”, 43 (10), 1390–1401. https://doi.org/10.1016/J.COMPBIOMED.2013.07.012.

[19]. Yang Y., Xu H., Qi B., Niu X., Li M., Zhao D. (2020). Stroke screening data modeling based on openEHR and NINDS Stroke CDE. “Proceedings - 2020 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2020”, 2147–2152. https://doi.org/10.1109/BIBM49941.2020.9313127.

[20]. Wantaka C., Kitidumrongsuk P., Soontornpipit P., Sillabutra J. (2018). Design and Development of Data Model for Stroke FAST Track System. “IEECON 2018 - 6th International Electrical Engineering Congress”. https://doi.org/10.1109/IEECON.2018.8712241.

[21]. Chang T. M., Kao H. Y., Wu J. H., Su Y. F. (2016). Improving physicians’ performance with a stroke CDSS: A cognitive fit design approach. “Computers in Human Behavior”, 54, 577–586. https://doi.org/10.1016/J.CHB.2015.07.054.

[22]. Gibson O. J., Balami J. S., Pope G. A., Tarassenko L., Reckless I. P. (2012). “Stroke Nav”: A wireless data collection and review system to support stroke care delivery. “Computer Methods and Programs in Biomedicine”, 108(1), 338–345. https://doi.org/10.1016/J.CMPB.2012.02.001.

[23]. Tang Z., Lawson S., Messing D., Guo J., Smith T., Feng J. (2016). Collaborative Rehabilitation Support System: A Comprehensive Solution for Everyday Rehab. “Proceedings - 2015 IEEE International Symposium on Multimedia, ISM 2015”, 61–64. https://doi.org/10.1109/ISM.2015.62.

[24]. Sun M., Dai D., Wu X., Wang S., Zang T., Xu X. (2016). SS4CSHC: A Services System for the Collaboration in Stroke Healthcare Cycle. “Proceedings of International Conference on Service Science, ICSS</i>, 2016-Febru”, 174–180. https://doi.org/10.1109/ICSS.2015.29.

[25]. Li S., Manogaran G. (2019). Design and Implementation of Networked Collaborative Service System for Brain Stroke Prevention and First Aid. “IEEE Access”, 7, 14825–14836. https://doi.org/10.1109/ACCESS.2019.2892947.

[26]. Ilieva R. Y., Damyanov V. (2019). NextGen HighTech Solutions to Improve the QoL in Cerebral Stroke Consequences. “10th National Conference with International Participation, ELECTRONICA 2019 – Proceedings”, Ml, 2019–2022. https://doi.org/10.1109/ELECTRONICA.2019.8825640.

[27]. Park E., Kim J. H., Nam H. S., Chang H. J. (2018). Requirement Analysis and Implementation of Smart Emergency Medical Services. “IEEE Access”, 6, 42022–42029. https://doi.org/10.1109/ACCESS.2018.2861711.

[28]. Wang Y. (2012). When music, information technology, and medicine meet. MIRUM 2012 - Proceedings of the 2nd International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies, Co-Located with ACM Multimedia 2012, 43–44. https://doi.org/10.1145/2390848.2390859.

[29]. Ramesh V., Kim S., Nguyen H. A., Agrawal K., Meyer B. C., Weibel N. (2020). Developing aids to assist acute stroke diagnosis. “Conference on Human Factors in Computing Systems – Proceedings” https://doi.org/10.1145/3334480.3383039.

[30]. Tsoupikova D., Triandafilou K., Solanki S., Barry A., Preuss F., Kamper D. (2016). Real-time diagnostic data in multi-user virtual reality post-stroke therapy. “SA 2016 - SIGGRAPH ASIA 2016 VR Showcase”, 1–2. https://doi.org/10.1145/2996376.2996387.

[31]. Virbitskaite I. B., Dubtsov R. S. (2008). Semantic domains of timed event structures. Programming and Computer Software, 34, 125-137.

[32]. Zhuklinets I. A., Khotimsky D. A. (2002). Logical time in distributed software systems. Programming and Computer Software, 28, 174-18.

Информация об авторах / Information about authors

София Исабель ФЕРНАНДЕС ГРЕГОРИО имеет степень магистра прикладных вычислений Национальной лаборатории передовых вычислений (LANIA) в Веракрусе, Мексика. Она является докторантом по информатике в Университете Веракрузана. Сфера научных интересов: разработка программного обеспечения, приложения для мобильного здравоохранения, совместная работа с компьютерной поддержкой (CSCW) и человеко-машинное взаимодействие.

Sofía Isabel FERNÁNDEZ GREGORIO has her Master Degree in Applied Computing from the National Laboratory of Advanced Computing (LANIA) in Veracruz, Mexico. She is a doctoral student in computer science from the Universidad Veracruzana in Mexico. Her areas of interest are Software Engineering, mHealth Applications, Computer-Supported Cooperative Work (CSCW), and Human-Computer Interaction.

Луис Херардо МОНТАНЕ-ХИМЕНЕС имеет степень PhD по программированию, окончил в мексиканский Университет Веракруза со степенью магистра прикладных вычислений в Национальной лаборатории передовых вычислений (LANIA). В настоящее время является штатным профессором и научным сотрудником факультета статистики и информатики Университета Веракруза. Сфера научных интересов – совместная работа с компьютерной поддержкой (CSCW), визуализация данных, человеко-машинное взаимодействие, контекстно-зависимые вычисления и разработка видеоигр.

Luis Gerardo MONTANÉ-JIMÉNEZ – PhD in Computer Science graduated from the Universidad Veracruzana in Mexico, with a master's degree in Applied Computing from the National Laboratory of Advanced Computing (LANIA). He is currently Full-time Professor and Researcher at the Faculty of Statistics and Computer Science of the University Veracruzana (México). His areas of interest are Computer-Supported Cooperative Work (CSCW), Data Visualization, Human-Computer Interaction, Context-Aware Computing and Videogame Development.

Кармен МЕСУРА ГОДОЙ имеет степень PhD по программированию от Университета Савойи во Франции. Профессор факультета статистики и информатики Университета Веракруса в Мексике. Основные научные интересы: человеко-машинное взаимодействие, пользовательский опыт (UX), совместная работа с компьютерной поддержкой (CSCW), визуализация и многоагентные системы.

Carmen MEZURA GODOY – Ph.D. in Computer Science from the University of Savoie in France. Professor at the Faculty of Statistics and Informatics of the University of Veracruz in Mexico. Main research interests: Human Computer Interaction, User eXperience (UX), Computer Supported Collaborative Work (CSCW), Visualization and Multiagent Systems.

Вивиана Ярель РОСАЛЕС-МОРАЛЕС имеет степень магистра по программированию и степень PhD по техническим наукам от Технологического института Орисабы (Веракрус, Мексика). Участвовала в нескольких мексиканских исследовательских проектах и присоединилась к факультету статистики и информатики Университета Веракруса в Мексике в рамках программы "Кафедры" комитета CONACYT в 2019 году. Сфера ее многочисленных научных интересов включает в себя: человеко-машинное взаимодействие, пользовательский опыт, серьезные игры и приложения электронного здравоохранения.

Viviana Yarel ROSALES-MORALES – MSc in Computer Systems and PhD in Engineering Sciences from the Technological Institute of Orizaba, Veracruz, Mexico. She has involved in some Mexican research projects and joined the Faculty of Statistics and Informatics of the Universidad Veracruzana in Mexico, through the Cátedras CONACYT program in 2019. Her research interests include: Human-Computer Interaction, User Experience, Serious Games and eHealth Applications, to name a few.

DOI: 10.15514/ISPRAS-2024-36(1)-16

Usability Test for Teachers in Their Training to Care for Children with Hearing Disabilities Mediated by ICT

E. Archundia-Sierra, ORCID: 0000-0001-9686-5305 <*etelvina.archundia@correo.buap.mx*>

Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, Pue., Mexico.

Abstract. The training of teachers in the inclusive classroom in attention to children with hearing disabilities is important for an educational system in equal conditions. The User-Centered Design (UCD) methodology and the System test Usability Scale (SUS) provided perception data to support teacher training in the inclusive classroom, especially for children with hearing impairment. The test (SUS) was applied to 12 teachers, the result of the study indicates that the usability of all the tools is above the standards (72.5), equivalent to a very good rating. The tool fostered acceptance by teachers for inclusive classroom training, in addition to needing a teacher training program where children with disabilities and learning disorders are cared for.

Keywords: Inclusive classroom; educational technology; teacher training; hearing disability.

For citation: Archundia-Sierra E. Usability test for teachers in their training to care for children with hearing disabilities mediated by ICT. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 251-258. DOI: 10.15514/ISPRAS-2024-36(1)-16.

Тест на удобство использования для учителей при обучении уходу за детьми с нарушениями слуха на основе информационно-коммуникационных технологий

Э. Арчундия-Сьерра, ORCID: 0000-0001-9686-5305 <*etelvina.archundia@correo.buap.mx*>

Бенемеритский автономный университет Пуэбла, факультет информатики, Пуэбла, Пуе., Мексика.

Аннотация. Обучение учителей в инклюзивном классе с уделением внимания детям с нарушениями слуха имеет важное значение для достижения равных условий в системе образования. Применение методологии проектирования, ориентированной на пользователя (UCD), и проведение системного тестирования по шкале удобства использования (SUS) предоставили данные о восприятии для поддержки обучения учителей в инклюзивном классе, особенно для детей с нарушениями слуха. Тестированию (SUS) подверглись 12 учителей, результат исследования показывает, что удобство использования всех инструментов выше стандартов (72,5), что эквивалентно очень хорошей оценке. В дополнение к Программе подготовки учителей, в которой заботятся о детях с ограниченными возможностями и нарушениями обучения, выработанный подход способствовал принятию учителями инклюзивного обучения в классе.

Ключевые слова: инклюзивный класс; образовательные технологии; обучение учителей; нарушения слуха.

Для цитирования: Арчундия-Сьерра Э. Тест на удобство использования для учителей при обучении уходу за детьми с нарушениями слуха на основе информационно-коммуникационных технологий. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 251–258 (на английском языке). DOI: 10.15514/ISPRAS–2024–36(1)–16.

1. Introduction

ICTs (Information and Communication Technologies) improve access to information, reduce labor costs and increase the interconnection between people with all the advantages of ICTs. However, digitization is not within the reach of all people, affecting the digital divide. Previously, the digital divide was attributed to underdevelopment and it was assumed that it was a temporary trend that would fade as technology became more widely used, but inequality persists today, despite the widespread commercialization of devices with Internet access [1], furthermore, the concept of equity is based on equality and in it the principle of social justice is applied, taking into account the individual needs of each person and attending to diversity, diminishing differences of any kind, therefore, equity is considered a fundamental indicator of educational quality, as some authors affirm, both terms are inseparable and there is no quality without equity, some research with a gender perspective in education considers that promoting equality would be possible to achieve a more just society [2, 3]. Technology in the classroom could create a major impact if it is designed to be accessible for children with disabilities, which could support teaching/learning processes that are usually adapted to the particular children's needs. An example of this process is literacy teaching for deaf children, where strategies must differ from those used with hearing children. One strategy that has been proven to be effective in teaching and learning processes is storytelling [4].

Dudley-Marling and Burns [5] argued that there were two dominant perspectives among educators regarding inclusive education. The first has been described as a deficit position or medical model because students are said to lack the skills and/or ability to succeed in school. The second perspective is a social constructivist perspective that is based on the notion that the responsibility for the disability does not rest with the student and that structural reform should take precedence over remediation. In other words, the social constructivist perspective suggests that there are sociocultural factors that mediate the success of students in school. Further exploration of how teachers, already in the field, are prepared to teach all students, as well as their willingness to implement inclusive education, is warranted, as studies in this area are scarce.

According to Darling Hammond [6] cited by Larios and Zetlin [7], it is to create a stronger and more equitable educational system, presenting seven elements for effective professional development: (a) it focuses on content, (b) incorporates active learning, (c) supports collaboration, (d) uses effective practice models, (e) provides coaching and expert support, (f) offers feedback and reflection, and (g) is sustained in duration.

The Program for Inclusion and Educational Equity is linked to the 2030 Agenda for sustainable development through the objective of guaranteeing inclusive, equitable and quality education and promoting lifelong learning opportunities for all, ensuring technical, professional and higher education [8]. The World Health Organization (WHO) estimates that more than 400 million people, including 34 million children, have hearing loss that affects their health and quality of life, furthermore, it indicates that by 2050, some 2.5 billion people (1 in 4) will have some form of hearing loss, and that almost 700 million (1 in 14) will have moderate or advanced levels of hearing loss in the better-hearing ear [9]. According to Instituto Nacional de Estadística y Geografía (INEGI) data dating from 2010, in Mexico there are 498,640 people with hearing limitations. On the other hand, in the country there are 401,534 people with limitations to speak and communicate in the country [10].

Hearing plays an important role from birth and occurs at three levels: basic, which allows us to be aware of sounds and our body, medium, which controls the sense of distance, and superior, which allows us to communicate and understand spoken language. Hearing is a sensory process that informs us of what happens beyond a visual field, even while we sleep, it allows us to be in contact

with the social context through linguistic codes. Partial or total hearing loss influences the use of language, learning and the relationship in the environment [11].

Ensuring equitable and quality education in the primary stages is one of the educational goals included in the 2030 Agenda, being the core of the Sustainable Development Goal (SDG 4), although the vision of inclusive education (SDG 4) encompasses all children, youth and adults, this education has historically been associated with the education of children with disabilities, and has often been conceptualized as such, therefore, the struggle of people with disabilities has shaped the understanding of inclusion. From the perspective of the teaching profession, this equitable and quality teaching is aimed at developing skills for work and for life of all students without exception, even among the most vulnerable groups of students or groups [12]. One of the challenges that educational policies have faced has been to improve the equity of our educational system.

The research work meets the recommendations of (WHO) and (SDG 4) at the international level and at the national level in Mexico with the Programs for Inclusion and Educational Equity, in the training of teachers in the inclusive classroom for children with hearing disabilities, designing the content structure, interactive learning activities and feedback through evaluation. The implementation was carried out in an open-source tool (open source) facilitating the creation of content trees, multimedia elements and interactive evaluation activities. The usability test System Usability Scale (SUS) was applied to primary school teachers to verify the acceptance and importance of teacher training in the inclusive classroom in attention to children with hearing disabilities mediated by (ICT).

The structure of the research in section 2 indicates the methodology (DCU) focused on teachers to promote inclusive classroom training for children with hearing disabilities, section 3 presents the design and implementation of the tool and Section 4 shows the results of the usability test (SUS).

2. Methodology

Inclusion must imply more than the placement of students with disabilities in regular classes, it must promote teacher training, therefore, the research question of this paper asks: Children with hearing disabilities, what is the perception of teachers in their training in the inclusive classroom? The objective is determined to know the perception of teachers in their training in the inclusive classroom for attention to children with hearing disabilities. The specific objectives are established in the requirements to train teachers in inclusive classroom care for children with hearing disabilities, the (DCU) is contemplated in the development of content and interactive activities implemented in an open source tool and the usability test (SUS) [13] applied to 12 primary school teachers in a study with a quantitative approach and descriptive scope with non-probabilistic sampling.

User-Centered Design (UCD) is a design methodology focused on the needs of users to develop products and services by improving usability, accessibility and user experience, which translates into greater user satisfaction in the use of the product or service. According to Norman and Draper [14], the (DCU) identifies the needs, requirements and limitations of users to design products that meet those needs effectively and efficiently, to achieve this the design process must be iterative and collaborative, with the active participation of users and designers in all stages of the process, for example , conversational agents for informal caregivers , applying a descriptive phenomenological qualitative study to carry out a questionnaire of open questions for the care of people with dementia and piloted with a commercial device [15] and the automatic translate service of the Nahuatl language , on topics of inclusion when Nahuatl spoken in regions where English the dominant language . This leads native people to in some way forget their mother language in favor of Spanish. In this environment, the language slowly disappears or, even worse, the situation leaves the people of these remote communities excluded from the technological advances and vulnerable to laws or services that are not written in Nahuatl [16].

3. Development

The design of the tool to promote teacher training in the inclusive classroom in care for children with hearing disabilities mediated by ICT integrates the characterization of the dimensions of user-centered design, technology and pedagogy. The user-centered dimension corresponds to the development of the (DCU), the Concur method tasks Tree (CTT) and Application Usability Testing (SUS). The technological dimension is addressed by the educational content creation structure which allows the use of multimedia elements, interactive self-assessment activities and games, it also facilitates the export of content in multiple formats, for example: HyperText markup Language (HTML), Shareable Content Object Reference Model (SCORM) and IMS Content Packaging and the pedagogical dimension, where content, interaction and evaluation activities are analyzed.

The pedagogical dimension of the teacher training tool is organized by the beginning, introduction and four units. Unit I. Educational inclusion, Unit II. Hearing impairment, Unit III. Educational attention and Unit IV. Strategies for educational inclusion. Each Unit is integrated by theme and interactive activities such as: fill in the gaps, interactive videos, true-false questions and unordered list.

4. Results

The usability method, to measure the software of the inclusive classroom in care for children with hearing disabilities, is carried out through the questionnaire (SUS) with the following ten questions [17] (Table 1).

Table 1. Questionnaire of questions (SUS)

Q	Question statement
Q1	I think I would use this system frequently.
Q2	I find this system unnecessarily complex.
Q3	I think the system was easy to use.
Q4	I think I would need help from a technically savvy person to use this system
Q5	The functions of this system are well integrated.
Q6	I think the system is very inconsistent.
Q7	I imagine that most people would learn to use this system very quickly.
Q8	I find the system very difficult to use.
Q9	I feel confident using this system
Q10	I needed to learn a lot of things before being able to use this system.

The answers to each statement are requested following the Likert Scale:

1. Strongly disagree
2. Disagree
3. Neutral
4. Agree
5. Totally agree

The demographic characteristics of the participation of a total of 12 teaching participants from the municipalities of the State of Puebla, Mexico, where 33.33% are men and 66.66% are women, with an average age of 45 years, 83.33% have a Bachelor's Degree in Primary Education and 16.66% have a master's degree, who have an average experience of 18 years at the primary level.

Teachers respond to the 10 questions of (SUS) obtaining the following data (Table 2).

The equations to calculate the values of (SUS) result from applying Equation (1) is used to calculate questions 1, 3, 5, 7 and 9. Equation (2) is used to calculate questions 2, 4, 6, 8 and 10.

$$SUS_{impar} = \left(\sum_{i=1}^{10} score - 1\right) * 2.5 \qquad (1)$$

Equation 1: calculation (SUS) for odd question

$$SUS_{par} = \left(\sum_{i=1}^{10} 5 - score\right) * 2.5. \qquad (2)$$

Equation 2: calculation (SUS) for even question

Table 2. Results of the questionnaire (SUS)

Teacher	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
D1	4	2	4	2	4	2	4	2	5	1
D2	5	4	4	2	4	2	4	4	4	2
D3	4	3	4	2	3	2	4	2	4	2
D4	4	2	5	2	4	2	5	2	4	2
D5	4	4	4	2	4	2	4	2	4	2
D6	5	5	5	5	5	1	4	5	5	1
D7	5	3	4	2	4	2	4	2	4	2
D8	4	4	5	4	4	2	4	4	4	4
D9	4	3	4	2	4	2	4	2	4	1
D10	4	3	4	5	5	1	4	2	5	1
D11	1	3	4	2	4	2	4	4	4	2
D12	5	2	5	1	5	1	5	1	5	1
Total	49	38	52	31	50	21	50	32	52	21

The applied results of the scoring rules of (SUS) (Table 3), the average scores of (SUS) [18] and the adjective rating scales for the digital application for teacher training in the inclusive classroom for children with hearing disabilities. The general average of the tool for teachers is 72.5, equivalent to a very good rating. The results of the study show that the perceived usability of the tool in general is very good for all participants, given the deficiencies in inclusion issues.

The open question asked to the teachers: What experience have you had in the inclusive classroom with disabled students in your workplace? 83.33% of the teachers surveyed have had or have students with the following disabilities: auditory, visual and intellectual, in addition to identifying students with autism and speech and language disorders. Most of the teachers surveyed mentioned that they have had children with different types of disabilities, but they do not have the knowledge, tools and resources to be able to teach and integrate the group in the inclusive classroom.

5. Conclusions

The application of the (SUS) questionnaire is an important process to involve teachers in the use of technological tools on inclusion issues for children with hearing disabilities. The data has highlighted some of the benefits and challenges faced in inclusive classroom teacher training to address diversity in schools, emphasizes the need for teachers in training to generate strategies and learning activities aimed at inclusive education immersed in a process of attention to the diversity

of needs of all students to reduce exclusion, in addition to addressing a continuous need for a teacher training program in inclusive classrooms.

As future work, it is contemplated to cover more thoroughly other inclusion topics, such as learning disorders, autism and dyslexia.

Table 3. Scale results (SUS)

Teacher number	Addition	Average	Adjective
D1	32	80	Excellent
D2	27	67.5	Well
D3	28	70	Well
D4	32	80	Excellent
D5	28	70	Well
D6	27	67.5	Well
D7	30	75	Very Well
D8	23	57.5	Well
D9	30	75	Very Well
D10	30	75	Very Well
D11	24	60	Well
D12	39	97.5	The best imaginable

References

[1]. CORPORATE I. Digital divides throughout the world and why it causes inequality. Iberdrola n.d. https://www.iberdrola.com/social-commitment/what-is-digital-divide (accessed April 9, 2022).

[2]. L. Vega Caro, A. Vico Bosch. Equality and educational quality: opportunities and challenges of teaching. 1st edition, no. 34. Dykinson, 2021. [Online]. Available at: https://search.ebscohost.com/login.aspx?direct=true&AuthType=cookie,ip,url,custuid&custid=s4231244&db=nlebk&AN=3142830&site=ehost-live&ebv=EB&ppid=pp_19.

[3]. N. Gulya, A. Fehérvári. The impact of literary works containing characters with disabilities on students' perception and attitudes towards people with disabilities. International Journal of Educational Research, vol. 117, p. 102132, Jan. 2023, doi: 10.1016/j.ijer.2022.102132.

[4]. L. Flórez-Aristizábal, S. Cano, CA Collazos, F. Benavides, F. Moreira, H. M. Fardoun. Digital transformation to support literacy teaching to deaf Children: From storytelling to digital interactive storytelling. Telematics and Informatics, vol. 38, p. 87–99, May 2019, doi: 10.1016/j.tele.2018.09.002.

[5]. C. Dudley-Marling, M.B. Burns. Two Perspectives on Inclusion in the United States. Global Education Review, vol. 1, no. 1, p. 14–31, 2014.

[6]. L. Darling-Hammond, M. Hyler, M. Gardner. Effective Teacher Professional Development. Learning Policy Institute, June. 2017. doi: 10.54300/122.311.

[7]. R.J. Larios, A. Zetlin, Challenges to preparing teachers to instruct all students in inclusive classrooms. Teaching and Teacher Education, vol. 121, p. 103945, Jan. 2023, doi: 10.1016/j.tate.2022.103945.

[8]. Operating Rules of the Program for Educational Inclusion and Equity, agreement number 02/04/19 by which the Operating Rules of the Program for Educational Inclusion and Equity are issued. Official Journal of the Federation. Available at https://dof.gob.mx/nota_detalle.php?codigo=5551602&fecha=28/02/2019 (accessed April 13, 2023).

[9]. World report on hearing. Washington, DC: Pan American Health Organization; 2021. License: CC BY-NC-SA 3.0 IGO. https://doi.org/10.37774/9789275324677.

[10]. National Institute of Statistics and Geography (INEGI), Bank of indicators. https://www.inegi.org.mx/app/indicadores/?t=151 (accessed August 10, 2022).

[11]. General Directorate of Special Basic Education (2013), Hearing and auditory disability, pages 9-24. Guidelines for the educational attention of students with hearing disabilities. Ministry of Education. Available at: https://ceadi.com.mx/audicion-y-discapacidad-auditiva.

[12]. C. Bodenhofer, L. Baez, Mar Botero, R. Carrasco. World Education Monitoring Report, 2020, Latin America and the Caribbean: Inclusion and Education: Everyone Without Exception - UNESCO Digital

Library. Paris, UNESCO, 2020. Accessed: February 6, 2023. [Online]. Available at: https://unesdoc.unesco.org/ark:/48223/pf0000374615.
[13]. D. A Norman, S.W. Draper. User Centered System Design: New Perspectives on Human-Computer Interaction. CRC Press, 1986.
[14]. S. Jiménez, R. Juárez-Ramírez, A. Ramírez-Noriega et al. A Usability and Persuasion Evaluation of Mobile Apps for Diabetes Type 2. Program computer Soft 48, 552–565 (2022). https://doi.org/10.1134/S0361768822080138.
[15]. S. Jiménez, J. Favela, A. Quezada et al. Towards Conversational Agents to support Informal Caregivers of People with Dementia: Challenges and Opportunities. program computer Soft 48, 606–613 (2022). https://doi.org/10.1134/S036176882208014X.
[16]. S. K. Garcia, E. S. Lucero, E. B. Huerta et al. Implementation of Neural Machine Translation for Nahuatl as a Web Platform: A Focus on Text Translation. program computer Soft 47, 778–792 (2021). https://doi.org/10.1134/S0361768821080168.
[17]. J. Brooke, SUS: Quick and Dirty Usability Scale. Usability Evaluation. In Industry, London, 1995. https://doi.org/10.1201/9781498710411-35.
[18]. J. Sauro. 5 Ways to Interpret a SUS Score – MeasuringU. September 19, 2018. https://measuringu.com/interpret-sus-score/ (accessed December 20, 2022).

Информация об авторах / Information about authors

Этельвина АРЧУНДИЯ-СЬЕРРА имеет ученую степень PhD в области информационных технологий и анализа решений, с 1997 года – профессор факультета информатики Бенэмерийского Автономного университета в городе Пуэбла. Сфера научных интересов: модели рабочих потоков, совместная работа с использованием компьютеров, человеко-машинное взаимодействие.

Etelvina ARCHUNDIA-SIERRA – Doctor in the Information Technology and Decision Analysis, professor of the Facultad de Ciencias de la Computación of the Benemérita Universidad Autónoma de Puebla, since 1997. Research interests: workflow models, computer supported cooperative work and computer human interaction.

DOI: 10.15514/ISPRAS-2024-36(1)-17

Strategies for Automatic Detection of Fallacious Arguments in Political Speeches during Electoral Campaigns in Mexico

[1] *K. Nieto-Benitez*, ORCID: 0000-0002-0748-4859 <*kenianieto@cenidet.edu.mx*>
[1] *N. A. Castro-Sanchez*, ORCID: 0000-0002-8083-3891 <*noe.cs@cenidet.tecnm.mx*>
[2] *H. Jimenez-Salazar*, ORCID: 0000-0002-0121-361X <*hjimenez@cua.uam.mx*>
[3] *G. Bel-Enguix*, ORCID: 0000-0002-1411-5736 <*gbele@iingen.unam.mx*>
[1] *D. Mújica-Vargas*, ORCID: 0000-0001-8665-4096 <*dante.mv@cenidet.tecnm.mx*>
[1] *J. G. González-Serna*, ORCID: 0000-0002-1874-9402 <*gabriel.gs@cenidet.tecnm.mx*>
[1] *N. González-Franco*, ORCID: 0000- 0002-1051-1379 <*nimrod.gf@cenidet.tecnm.mx*>

[1] *Tecnológico Nacional de México, Centro Nacional de Investigación y Desarrollo Tecnológico (TecNM/CENIDET), Morelos, México.*
[2] *Universidad Autonoma Metropolitana, Ciudad de México, México.*
[3] *Universidad Nacional Autónoma de México, Ciudad de México, México.*

Abstract. This study proposes a machine learning approach to automatically detect "appeal to emotion" fallacies. The objective is to establish a set of elements that enable the application of fallacy mining. Our method uses a lexicon of emotions to distinguish valid arguments from fallacies, employing Support Vector Machine and Multilayer Perceptron models. The Multilayer Perceptron obtained an F1 score of 0.60 in identifying fallacies. Based on our analysis, we suggest using lexical dictionaries to effectively identify "appeal to emotion" fallacies.

Keywords: fallacies; corpus; arguments; appeal to emotions.

For citation: Nieto-Benitez K., Castro-Sanchez N. A., Jimenez-Salazar H., Bel-Enguix G., Mújica-Vargas D., González-Serna J. G., González-Franco N. Strategies for Automatic Detection of Fallacious Arguments in Political Speeches During Electoral Campaigns in Mexico. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 259-276. DOI: 10.15514/ISPRAS-2024-36(1)-17.

Full text: Nieto-Benitez K., Castro-Sanchez N. A., Jimenez Salazar H., Bel-Enguix G., Mújica Vargas D., González Serna J. G., González Franco N. Elements for Automatic Identification of Fallacies in Mexican Election Campaign Political Speeches. *Programming and Computer Software*, 2023, Vol. 49, No. 8, pp. 762–774. DOI: 10.1134/S0361768823080170.

Acknowledgments. This work was partially supported by the Government of Mexico (CONACYT grant with project number 653661, SNI).

Nieto-Benitez K., Castro-Sanchez N. A., Jimenez Salazar H., Bel-Enguix G., Mújica Vargas D., González Serna J. G., González Franco N. Strategies for Automatic Detection of Fallacious Arguments in Political Speeches During Electoral Campaigns in Mexico. *Trudy ISP RAN/Proc. ISP RAS*, vol. 36, issue 1, 2024. pp. 259-276.

Стратегии автоматического выявления ошибочных аргументов в политических речах во время избирательных кампаний в Мексике

[1] *К. Ньето-Бенитес,* ORCID: 0000-0002-0748-4859 <*kenianieto@cenidet.edu.mx*>
[1] *Н. А. Кастро-Санчес,* ORCID: 0000-0002-8083-3891 <*noe.cs@cenidet.tecnm.mx*>
[2] *Э. Хименес Саласар,* ORCID: 0000-0002-0121-361X <*hjimenez@cua.uam.mx*>
[3] *Х. Бель-Энгикс,* ORCID: 0000-0002-1411-5736 <*gbele@iingen.unam.mx*>
[1] *Д. Мухина-Варгас,* ORCID: 0000-0001-8665-4096 <*dante.mv@cenidet.tecnm.mx*>
[1] *Х. Г. Гонсалес-Серна,* ORCID: 0000-0002-1874-9402 <*gabriel.gs@cenidet.tecnm.mx*>
[1] *Н. Гонсалес-Франко,* ORCID: 0000- 0002-1051-1379 <*nimrod.gf@cenidet.tecnm.mx*>

[1] *Национальный технологический центр Мексики, Национальный центр исследований и технологических разработок, Морелос, Мексика.*
[2] *Автономный университет Метрополитен, г. Мехико, Мексика.*
[3] *Национальный автономный университет Мексики, г. Мехико, Мексика.*

Аннотация. Для автоматического обнаружения ошибок «обращения к эмоциям» авторами предлагается подход на основе машинного обучения. Цель состоит в том, чтобы сформировать набор элементов, которые позволят построить приложение для выявления ошибок. Чтобы отличить реальные аргументы от ошибочных, наш метод, основанный на моделях опорных векторов и многослойного перцептрона, использует словарь эмоций. При выявлении ошибок многослойный перцептрон получил оценку по метрике F1, равную 0,60. Основываясь на проведенном анализе, мы предлагаем использовать лексические словари для эффективного выявления ошибок «обращения к эмоциям».

Ключевые слова: ложные высказывания; корпус; аргументы; обращение к эмоциям.

Для цитирования: Ньето-Бенитес К., Кастро-Санчес Н.А., Саласар Э.Х., Бель-Энгикс Г., Мухика-Варгас Д., Гонсалес-Серна Х.Г., Гонсалес-Франко Н. Стратегии автоматического выявления ошибочных аргументов в политических речах во время избирательных кампаний в Мексике. Труды ИСП РАН, том 36, вып. 1, 2024 г., стр. 259–276 (на английском языке). DOI: 10.15514/ISPRAS-2024-36(1)-17.

Полный текст: Ньето-Бенитес К., Кастро-Санчес Н.А., Саласар Э.Х., Бель-Энгикс Г., Мухика-Варгас Д., Гонсалес Серна Х.Г., Гонсалес Франко Н. Элементы автоматического выявления ошибок в политических речах мексиканской избирательной кампании. Programming and Computer Software, 2023, т. 49, № 8, с. 762–774 (на английском языке). DOI: 10.1134/S0361768823080170.

Благодарности. Настоящая работа была частично поддержана Правительством Мексики, грант Национального совета по науке и технике (CONACYT) номер 653661, SNI.

1. Introduction

Existing research on fallacy identification in several types of texts has provided the types of fallacies committed by political candidates and confirmed their use in political debates and speeches. These studies involve the analysis of texts in the English language. However, these investigations lack a method for identifying fallacies by implementing natural language processing techniques. Although [1] identified some lexical and characteristic syntactic elements of the *Straw man* fallacy and proposed an approximate model of its structure for mining, no method was implemented to automatically identify whether a proposition (argument) is a fallacy. The system developed by [2] and [3] identifies formal fallacies in natural dialogues between two people, but the process used does not allow for the identification of informal fallacies in monological political speeches.

This paper structures the mechanisms for identifying fallacies and presents the main elements to be considered for the development of systems that allow for their identification from unstructured texts. Therefore, our goal is to propose a set of elements that allow for fallacy mining and to discuss the

challenges involved in this task. Moreover, this paper assumes that it is possible to implement machine learning-based techniques that allow for the automatic fallacy detection.

This paper presents the identification of emotional appeal fallacies in political speeches in the Spanish language by implementing two machine learning methods: Support Vector Machine, Multilayer Perceptron, and the use of two features: affective terms and lexical diversity. In addition, the conceptualizations of the "fallacy" term are structured, the mechanisms for their identification are presented, a set of elements to consider for the development of systems that allow for fallacy mining is proposed, and the challenges involved in this task are discussed.

2. Motivation

The fallacies have received little attention from the linguistic community. There is insufficient characterization of their form and determination, involving semantic, pragmatic, and communicative analysis. It is important to emphasize that the same reasoning error in arguments can be classified into different types of fallacies. Although there are diverse taxonomies, there is no certain and unique taxonomy. The complexity of classifying them arises from the absence of precise rules that determine absolutes regarding errors in reasoning, and even from the intrinsic problem of the definition, purpose, meaning, or effects of fallacies on the audience or readers [29].

To implement machine learning techniques, a collection of labeled data is required to validate the performance of any implemented technique [51]. Within the literature, there are few corpora available in Spanish language to experiment with methods for identifying fallacies [30]. Corpus have been created with specific objectives and are hardly adaptable to identify arguments that have no valid basis.

Moreover, it is important that criteria identification could be implemented with machine learning techniques. The criteria established to identify fallacies through manual analysis may not be processable through a computational method. Additionally, these criteria may vary according to the types or categories of informal fallacies to be processed.

If we take the example of fallacies by appealing to emotions, emotional appeals can arise in any context as people advocate for what they feel is important, but there are contexts in which they are inappropriate [30]. Similarly, two propositions considered irrelevant to each other in one context, may be considered relevant in another, and there may be references to emotions that are not a fallacy in an argument.

3. Fallacies

There have been numerous attempts to establish concepts that enable an understanding of the term "fallacy" in any argumentation theory [4]. This Inconsistency and disagreement have led to the emergence of several approaches and definitions of the term fallacy.

According to [5], the term "fallacy" is not precise due to its ambiguity and can refer to: "(a) a type of error in an argument, (b) a type of error in reasoning (including arguments, definitions, explanations, among others), (c) a false belief, or (d) the cause of any of the above errors".

In our case, similar to [5], but based on the monological, dialogical, and rhetorical models for argumentation analysis presented in [6], as well as the conceptualization of the term fallacy in the field of formal and informal logic presented in [7], a fallacy can refer to:

1. A type of error in argument form: Fallacies can be defined as arguments that have errors in their form by infringing on any of the deductively valid structures [7, 8] or identifiable instances of invalid logical forms [9].
2. A type of error in the argument reasoning: Fallacies are arguments that contain errors in their content due to mishandling of their propositions [9], or they are an invalid, failed, or fraudulent argumentation [9, 10].

3. A violation of rules and/or criteria: Reference is made to rules or criteria that must be followed in speech or argument construction. In this context, fallacies are characterized by infringing on the critical discussion rules and interrupting the resolving a dispute process [11, 12]; They are arguments that lead to error by infringing one of the rules or criteria for constructing good arguments [13, 14] or are considered arguments lacking in solidity [8].
4. Something implausible: Unlike the previous ones, non-linguistic aspects are considered, and reference is made to the argument persuasive intention and the effects it produces on the audience [10, 15].

Regardless of their definition, fallacies are grouped into formal and informal. Informal fallacies are speeches that pretend to be good argumentation [10] and are found in everyday language. This type of fallacy is analyzed in definitions 2, 3, and 4. Formal fallacies arise from errors in their structure and are independent of the content they deal with [7] or the context in which they arise [16], as specified in 1, and are typically presented in syllogisms.

4. Identification of fallacies

4.1 Related works

During electoral campaigns, argumentative strategies are used to persuade and manipulate citizens with the aim of obtaining their vote. One of these strategies is the use of fallacies, which are commonly presented in structured political speeches such as debates, press conferences, position papers, among others, to offer apparently coherent and solid positions [17].

Most of the research on identifying fallacies has focused on analyzing texts written in the English language. In 1986, [18] demonstrated that fallacies are common in political speeches by identifying more than 40 types of fallacies in two presidential debates, including *Ad Populum* and *Ad Hominem*. [16] found 25 fallacies in a presidential debate, with the most frequent being *Straw Person* and *Ad Hominem*. And in [20], 550 texts (press releases and journalistic articles) were analyzed, and almost one-third (32.5%) of the texts included at least one fallacious argument, with fallacies appearing more frequently in press releases than in journalistic articles.

In [4], a set of stages was described for resolving a critical discussion, where violating one of these stages results in a fallacy. According to these stages, [21] analyzed four political debates and found a concentration of fallacies in the argumentation and confrontation stages of the debates, with *Ad Hominem* being the preferred fallacy by politicians in the confrontation stage and *Ad Misericordiam* in the argumentation stage.

In [22], the criteria for a good argument were used to identify fallacies in four presidential debates. The relevance and acceptability criteria were violated most frequently, appearing in 12 of the 32 identified fallacies. The most frequent fallacy was *False Alternatives*, which occurred 10 times.

Another way to identify erroneous arguments in debates was by using the 10 rules of reasoning described in [12]. Considering these rules, [23] analyzed a presidential debate and concluded that politicians most frequently violate rule four (relevance of arguments), which was present in 25% of the data.

Unlike previous works, in addition to identifying fallacies, [24] also obtained the structure and pragmatic strategies of a fallacy. The pragmatic structure was established in three stages: Starting Point, Argument, and Endpoint. In the argument stage, they found that 60% of the arguments in the speeches appealed to self-interest, 20% to fear, 10% to commitment, 10% to flattery, and 0% to reciprocity and authority.

In [25], a taxonomy of fallacies was obtained through an analysis of arguments about security. The authors assumed that security arguments do not contain causal fallacies or emotional appeals, and based on these assumptions, these types of fallacies were excluded. In [26], students' ability to

identify fallacies was examined, taking into account their argumentative context, where an argument can be considered fallacious in certain types of contexts only. In [27], the critical thinking skills of 25 students were measured to detect six types of fallacies. The 25 students were able to correctly identify and name three of the six fallacies: *Irrelevant Authority*, *False Dilemma*, and *Ad Hominem*. In the analysis conducted in [28] of a social debate on religion, the *Ad Hominem* fallacy was found to occur most frequently.

In [29], fallacies were identified using a set of nine presidential speeches in Spanish. Seventeen types of fallacies were identified in the opening and closing campaign speeches of presidential candidates. Among the most relevant fallacies in terms of frequency were *False Dichotomy*, *Ad Populum*, *Argumentum in Terrorem*, *Ad Hominem*, and *False Attribution*.

Regarding the approach to automatic identification of fallacies through machine learning algorithms, [1] identified some lexical and syntactic characteristic elements of the *Straw Man* fallacy. Based on the analysis performed, an approximate model of the structure of the *Straw Man* fallacy was proposed for its detection without implementation using Natural Language Processing (NLP) techniques. In [30], a baseline was proposed for the fallacies identification by emotional appeal using three machine learning models: Support Vector Machine, Logistic Regression, and Decision Trees. A set of 601 arguments obtained from 80 political speeches in Spanish was used. As a result, an F-score of 0.55 was obtained using textual similarity between the components of the argument and 0.62 by combining similarity with the affective terms used in the arguments.

In addition, research has focused on identifying informal fallacies to verify their use, understanding student's abilities to identify fallacies, understanding the fallacies relationship with populist communication, and the strategic and/or manipulative use made of them in debates, political speeches, and other media. Among the most common fallacies that appear most frequently in political speeches are *Ad Hominem*, *Ad Misericordiam*, and *Ad Populum* [17, 19, 18, 21, 22, 24, 29].

In comparison to the referenced paper [30], the present article provides a study on related works regarding fallacy identification, as well as the elements and features to be considered for the implementation of machine learning models. In fallacies identification by appealing to emotions, emotional traits and lexical diversity are employed as argumentative patterns to distinguish valid arguments from fallacies. The Multilayer Perceptron neural network is implemented.

4.2 Features

There are several features that can be used to analyze arguments or documents in order to identify fallacies in valid arguments. Here, we will focus on describing the most common features that frequently appear in a political argumentation context and are centered on the identification of fallacies by appealing to emotions.

In fallacies by appealing to emotions (*Ad Populum* argument), the support given to the argument's conclusion is an inappropriate appeal, because instead of evidence and a rational argument, it relies on expressive language and other mechanisms designed to provoke an emotion in the audience. This type of fallacy incorporates the *Ad Misericordiam* fallacy: a fallacy where the argument relies on *generosity*, *altruism*, or *pity* [7]. Other authors refer to *Ad Populum* as "the speaker appeals to the support that a large number of people give to the presented theses" [29].

Each research presents a proposal of features. For example, argumentative patterns were used in [18, 20, 29]; critical discussion resolution rules were used in [23]; and construction of good arguments criteria were used in [22].

Argumentative patterns are related to the expressive language used in premises to justify the argument's conclusion [7]. For example, the *Ad Misericordiam* fallacy can be identified by the use of words that allow taking advantage of the audience's sympathy or pity [20]. Some patterns are established according to the axes for emotion reconstruction in speech: involved people, intensity/quantity, and time [31].

Other features based on criteria have been established that allow for the development of good arguments, such as the criteria of acceptability, truth, relevance, and sufficiency. When one or more of these criteria are violated, the argument is considered fallacious [32]. Of these criteria, only the acceptability, relevance, and sufficiency criteria were considered in [14], and the refutation structure and effectiveness criterion were included, with relevance being the criterion used for the identification of fallacies by appealing to emotions (FAE). In [33], acceptability, relevance, and sufficiency were proposed as the three main aspects that should be examined to determine if an adequate basis is provided for accepting the conclusion given in the argument; otherwise, the argument is an FAE. Finally, in [34], two of these criteria, relevance and sufficiency, were established to evaluate the argument's components (premise and conclusion) and determine which arguments are fallacious.

Among other features, it includes analyzing arguments through critical questions or considering rules for constructing good arguments. Questions help to distinguish legitimate strategies for supporting the assertion in the argument. For the identification of the *Ad Populum* argument, the questions evaluate whether the arguer has relied on any kind of evidence and whether the appeal is relevant to the conclusion in the context of the argument. In the case of *Ad Misericordiam*, the questions evaluate the appeals in the argument context, as well as the relationship between the premise and conclusion through relevance [35]. Both fallacies can also be identified through the rules established for constructing good arguments [13].

Other criteria have been established that judge arguments within a dialogue structure or systematically evaluate the movements or sequences of the argument in the dialogue context. Under this context, three criteria of a good argument were established in [32]: anticipating an objection to a premise, anticipating other criticisms, dealing with alternative positions; two criteria were established in [34]: Dialectical Relevance and Dialectical Shift; and a set of rules for resolving a critical discussion was established in [11,12].

Based on these rules or criteria, an argument is considered fallacious if it violates one or more of them. The three features can be used in a dialogic speech, unlike the criteria used to evaluate the internal argument structure, which focus more on monologic speeches [24].

According to [26], informal fallacies can be detected by examining the argument's context. In argumentation, the context can be defined using the dimensions suggested by [36], as cited by [26]: initial situation that motivates the dialogue, method of dialogue, and the objective of the dialogue. These dimensions differentiate types of dialogue, which in turn from the argumentative context.

Finally, a critical evaluation of the argument can be performed in two steps to determine whether the approach is fallacious. First, the argument is reconstructed from the speech. Once the argument has been obtained, the three sources of objective evidence are evaluated: the speech text, the dialogue context, and the abstract model of dialogue. These steps involve an evaluation of both the argument structure and the dialogue in which the argument is presented [37].

The set of features is grouped into two categories based on [32] and [34], as cited by [24]. One category groups the features that analyze the dialogue structure, and the other evaluates only the propositions of the argument, that make up the argument structure (Table 1).

Table 1. Features used in the identification of informal fallacies

Dialogue structure	**Argument structure**
Rules for resolving a critical discussion	Argumentative patterns
Dimensions of the context	Rules for constructing good arguments
Dialectical relevance and Dialectical shift	Criteria of a good argument
Sources of objective evidence	

While both categories allow the identification of fallacies, the process for selecting the features depends on their definition or approach. The first category can be used to evaluate arguments from

a dialectical approach, and the second from a logical approach. In the case of rhetoric, the acceptability criterion [32], space [31], or any non-linguistic features can be used.

Some of the research in the literature has used features focused on the analysis of the argumentative structure [18, 29, 30]. We consider that this type of feature can be implemented to some extent using natural language processing techniques, especially argumentative patterns. For example, the patterns found in the identification of fallacies by appealing to emotions are based on emotive language that allows for the justification of the assertions made in the speeches.

4.3 Taxonomy of fallacies

Several taxonomies have been established in the literature to group some of the informal fallacies. However, existing taxonomies differ in their length, categories, sets, and names (Table 2). For example, fallacies used to win an argument were grouped together [38]. Other taxonomies were established according to the criteria that are violated in the construction of a good argument [14], or based on the most common types of reasoning errors [7]. When any of these criteria or errors are violated, the argument is considered fallacious. Lists of fallacies that involve types of errors committed in the content of the reasoning or in the structure of the argument are also presented [5], or fallacies found in security documents were grouped [25].

Some investigations have grouped together a set of fallacies without defining a taxonomy as such. Here, fallacies were grouped that break some of the rules presented in the stages through which the resolution of a dispute must pass in a critical discussion [12]. Also, the types of fallacies that occur when the premises of an argument are irrelevant or when its conclusions are based on faulty analogies were grouped together [39].

Table 2. Taxonomy of fallacies. TF represents the number of fallacies proposed by the taxonomy

Ref.	TF	Categories
[5]	224	A list of uncategorized fallacies is established
[7]	15	Relevance, faulty induction, presupposition, and ambiguity
[12]	34	Opening, confrontation, argumentation, and closing
[14]	60	Structure, relevance, acceptability, sufficiency, and effectiveness of refutation
[25]	33	Circular reasoning, divergent arguments, fallacious appeals, mathematical fallacies, unfounded claims, anecdotal arguments, omission of key evidence, and linguistic fallacies
[38]	64	Linguistic factors, relevance of omission, relevance of intrusion, and relevance of presumption
[39]	32	Irrelevance and analogy

The list of different types of fallacies is extensive, and the features that distinguish them from one another are quite varied. Attempting to address the problem of identifying fallacies using a general method and taxonomy would be inadequate, due to the variety of fallacies, concepts, rules, and criteria established by different authors. For instance, the categories proposed in [7, 38, 14] are oriented towards evaluating arguments from a logical approach, while the categories proposed in [12] are based on a pragmadialectical approach.

5. Elements of identification

Discourse analysis consists of a set of strictly related tasks designed to distinguish good arguments from fallacies. Considering that a fallacy is an argument with an error in its content due to mishandling of its propositions [7], or a claim that has a reasoning error [29], and according to research in literature, a set of elements was obtained to consider in fallacy identification (Fig. 1). These elements are grouped into two sections: Argument Mining and fallacy identification.

5.1 Argument Mining

The main goal of Argument Mining (AM) is to "automatically extract arguments from generic textual corpus, in order to provide structured data for computational models" [6]. The AM systems implement a pipeline architecture, process unstructured documents, and produce a set of annotated arguments as a result.

Research on AM in the literature is characterized by the use of English language texts [6, 40, 41, 45, 46, 50], and few studies have processed Spanish language texts [30, 42, 43, 44].

The tasks involved in systems developed for argument extraction from plain text begin with text segmentation, and the boundary of the text that is considered argumentative (argumentative sentence) is defined. Subsequently, these segments are classified according to their function (premise or conclusion) within the argument (classified sentences), and links between segments (support or attack) are predicted to build the argument structure. Finally, the relationships between the existing arguments in the text are inferred (Fig. 1).

5.2 Identification of fallacies

In literature, the argumentative sentences identification and the components classification stages in AM are used as an initial stage in identifying fallacies [24, 29, 30]. Subsequently, the fallacy concept to be used and the selection of fallacies to be identified are determined. From this, features are selected or searched that allow for the determination of whether a sentence or argument is a fallacy. Therefore, once the arguments or sentences are extracted from the texts, an analysis is carried out for each of them, considering the type of fallacy to be identified. The analysis can be carried out by considering the relationship between argument components: evaluating the justifications present in the premises that support the statement given in the conclusion (inference); by considering the relationship between arguments by evaluating them within the dialogue structure; or by evaluating the arguments in relation to their acceptance in the audience (Fig. 1).

This paper focuses on identifying fallacies automatically by appealing to emotions through the analysis of argumentative components using affective terms (patterns) and measuring the lexical diversity of each component.

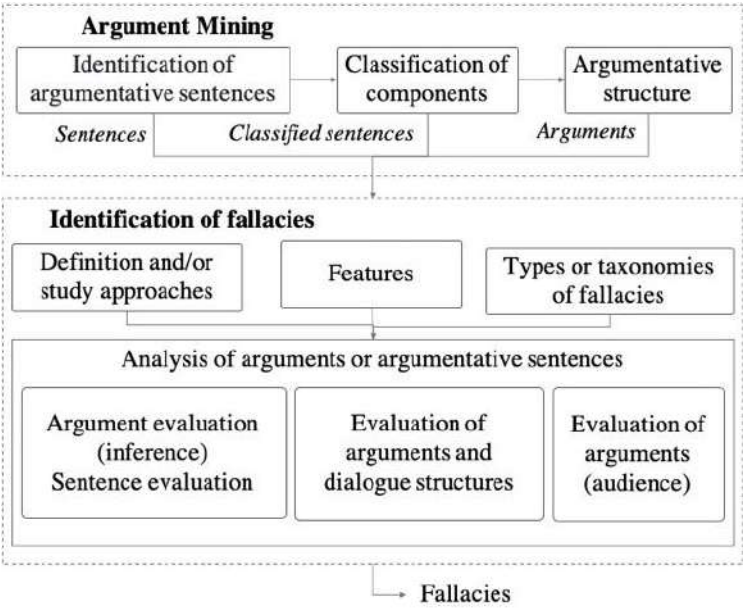


Fig. 1. Elements for fallacy identification

6. Identification of fallacies by appeal to emotions

6.1 Data

The appeal to emotion fallacy corpus consists of arguments obtained from a set of political speeches [30]. The corpus contains 601 arguments labeled according to their argumentative structure (premise and conclusion) and classified into fallacies by appealing to emotions and valid arguments:

1. (*Premise*) Aunque existen otros asuntos, el principal tema de la agenda con el gobierno estadounidense tiene que ser la migración, por todo lo que aquí se ha dicho. (*Premise*) Los flujos migratorios masivos y el creciente rechazo en la frontera constituyen una de las principales fuentes de fricción entre las dos naciones. (*Conclusion*) Por ello, hoy más que nunca es necesaria la cooperación entre ambos gobiernos para buscar soluciones de fondo que atemperen y ordenen el fenómeno migratorio. (*Valid*)
2. (*Premise*) No se puede gobernar un país en un mar, en un océano de desigualdad. (*Premise*) Esto se debe de entender: No vamos a tener seguridad pública, si sigue habiendo tanta desigualdad social. (*Premise*) Esto conviene a todos. (*Conclusion*) Por eso cuando planteamos que "Por el bien de todos, primero los pobres", no estamos proponiendo imponer las cosas, sino convencer y persuadir. (*Fallacy*)

For argumentative component classification, an agreement with Cohen's Kappa index (kc) of 0.692 and an agreement with Fleiss' Kappa index (kf) of 0.648 were obtained; both results with a substantial agreement degree. In identifying fallacies, a *kc* of 0.442 and a *kf* of 0.282 were obtained [30].

6.2 Feature Selection

The fallacies by appealing to emotions are characterized by the use of emotive language to support an opinion or position in an argument or as a resource to achieve a goal. This type of language is presented in arguments in a positive or negative manner and includes words that serve only to manipulate emotions [7, 13]. Emotive language can be detected through certain argumentative patterns. For instance, words that appeal to emotions [7]:

1. (*Premise*) Somos un país de gente alegre, ingeniosa y trabajadora; de mujeres y hombres que luchan, que están de pie y que saben salir adelante. (*Conclusion*) Por eso, por todos ustedes, aquí hoy les digo: México va a estar mejor y México va a cambiar.

The affective terms can be classified as having either a positive or negative polarity (iSOL lexical dictionary) [47] or according to the type of emotion they convey (SEL dictionary) [48] (Table 3). Some terms in SEL dictionary are classified with more than one type of emotion (TE), and the difference lies in the frequency of use in each TE.

Table 3. Sample of affective terms found in the iSOL and SEL dictionaries

Term	Polarity	Type of emotion	Frequency of use in TE
Abandon	Negative	Sadness	0,898
Admirable	Positive	Happiness	0,764
Admirable	Positive	Surprise	0,73
Abysmal	Negative	-	-
Torment	-	Anger	0,365
Torment	-	Sadness	0,53

There is another way to express emotional language, which is through the use of words that convey emotional features [31]:

2. Nosotros, en la Alianza por México, tenemos un *gran compromiso,* porque en el año 2000 nuestro país tomó el camino de una *aventura política,* que hoy está viviendo nuestro país

las consecuencias, un país sin rumbo, un país sin dirección, un país en donde las cosas están al revés, donde la *inseguridad crece* y los *delitos aumentan* y las víctimas nadie las defiende.

These affective traits are classified into three axes [31]:

1. Involved individuals. The discourse either focuses on the speaker or involves the audience.
2. Intensity/Quantity. It affects categories such as distance, time, or the quality of people through quantitative modulation.
3. Time. It focuses on the description of the period in which the events being narrated occur.

Other traits that do not fall into these axes are also considered, and for these types of terms, a general class was determined (Table 4).

Table 4. Sample of emotional traits in arguments

Involved individuals	Intensity/ Quantity	Time	General
Democrats.	Many, more, most	Two months	Real change
Mexicanos	Minimum, huge	Future	Single mothers, widows
Veracruzanos	Millions, hundreds, thousands	Present	Disability
People	Marginalized Mexico	Half a century	Lack
	Very serious situation	Past generations	Criminal acts

Expressive language is related to the lexical diversity of the argument. Lexical diversity measures whether a text uses a wide range of terms or is limited to recycled lexical items [49]. The simplest measure of diversity is the type/token ratio (TTR). This diversity expresses the ratio of types (word forms) to the ratio of tokens (continuous words) in the text (Eq. (1)). The interpretation is based on these two parameters, the greater the number of word forms relative to the number of all words in the text, the more lexically varied the text or corpus.

$$TTR = \frac{no.\ type}{no.\ tokens} \quad (1)$$

The affective terms are used in argument components to justify or establish the idea and topic discussed in the argument. This can result in a decrease in the number of different lexical elements used in the argument. Therefore, if the affective terms are removed or repeated (Argument 5 and 6), diversity decreases. Hence, the argument is considered a fallacy when it has lower lexical diversity and a higher number of affective terms.

1. *(Conclusion) D*ebemos estar unidos de cara a la nación, *(Premise)* porque sólo unidos podremos vencer a quienes son nuestros verdaderos *enemigos:* la *pobreza*, la *delincuencia,* el *desempleo,* la *desigualdad. (Premise)* Divididos perderíamos la fuerza que necesitamos para construir un *México mejor.*
2. *(Premise)* No se puede *manipular*, como se hacía antes, ya no se puede pensar poner vino nuevo en botellas viejas. *(Premise)* Puede seguir la misma estructura de poder, la misma estructura de control y de *manipulación*, pero es otra la mentalidad de nuestro pueblo. *(Premise)* El pueblo de México no es *tonto, tonto* es el que piensa que el *pueblo* es *tonto.* (*Conclusion*) Por eso no les va a funcionar su estrategia. *(Premise)* Ellos tienen el dinero, mucho dinero para comprar espacios en la televisión, en la radio, para *difamarnos*, pero no tienen lo mero principal, no tienen el apoyo de la *mayoría* de la gente, eso se los puedo asegurar.

6.3 Features analysis

Emotive traits were not found in the iSOL and SEL dictionaries. These texts are represented as syntagms in the arguments, primarily as nominal syntagms (NS) or adjective syntagms (S-ADJ). A dictionary of emotive features was made with 1,093 syntagms classified as affective and non-affective (Table 5). The labeling was performed by groups of two and three annotators. An agreement of 0.2478 was obtained with Cohen's Kappa index and 0.2302 with Fleiss' Kappa index (Table 6).

Table 5. Sample of nominal and adjective syntagms

Nº	Emotive traits	Classification
1	a failed strategy	Affective
2	a political adventure	Affective
3	foreign policy	Non-Affective
4	an exacerbated presidentialism	Affective
5	economic policies	Non-Affective

Table 6. Inter-annotator agreement for labeling emotive traits

Group	Affective	Non- Affective	Total	*k*
A1 – A2	459	147	606	0,2240
A1 – A3	670	65	735	0,2123
A2 – A3	486	133	619	0,2478
A1 – A2 – A3	436	61	497	0,2302

There is a set of 601 arguments. Each argument (ARG) has a structure with a conclusion (CO) and one or more premises (PRE). The argumentative structure was analyzed using three dictionaries: type of emotion (TE), polarity (PO), and emotive traits (ET), along with the lexical diversity of each component and the argument itself (Table 7).

Table 7. Sample of the features obtained in the arguments.

#ARG	Lexical Diversity			Affective Terms		
	CO	PRE	ARG	TE	PO	ET
1	0,695	0,654	0,658	15	10	3
2	0,799	0,772	0,648	15	10	3
3	0,652	0,631	0,675	2	7	3
4	0,421	0,401	0,446	3	9	2
5	0,781	0,752	0,812	3	3	1

The relationship between lexical diversity and affective terms indicates that arguments use a limited vocabulary and frequently employ affective language. Compared to [30], using the emotional features dictionary, most fallacies contain at least one affective term (Fig. 2 and 3). However, the number of affective terms increases in both fallacies and valid arguments (Fig. 4 and 5).

7. Results

The identification of fallacies is carried out through the analysis of argumentative components. The corpus contains classification of arguments, which is the first element of Argument Mining to initiate the identification of fallacies. Affective terms and lexical diversity of each component are used in the analysis, and the result of these features is evaluated using Support Vector Machine (SVM) and Multilayer Perceptron (MLP) methods. The MLP network has three layers, and the number of neurons in input and hidden layer depends on the number of features to be evaluated. x_n are the features used for fallacy identification, while y represents the classes (valid argument and fallacy). The logistic sigmoid function was used in the network (Fig. 6).

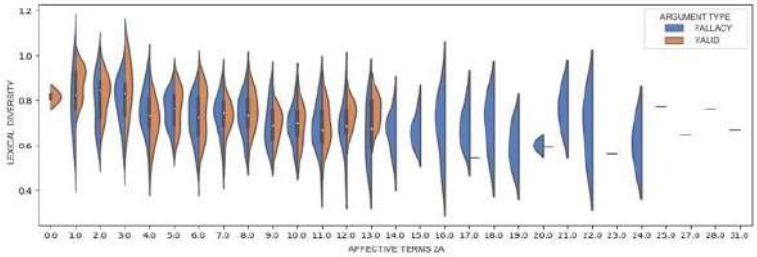


Fig. 2. Data distribution by affective terms and lexical diversity. Affective terms added from the emotional traits dictionary, and considering agreement between two annotators (2A)

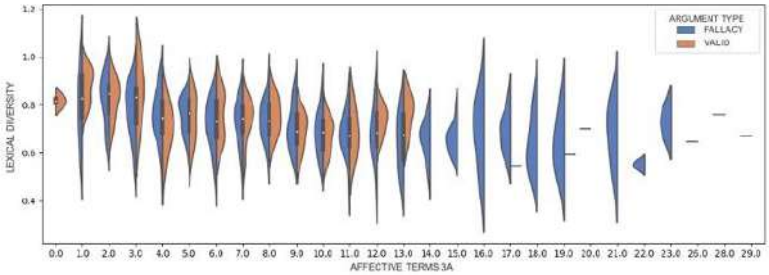


Fig. 3. Data distribution by affective terms and lexical diversity. Affective terms added from the emotional traits dictionary, and considering agreement between three annotators (3A)

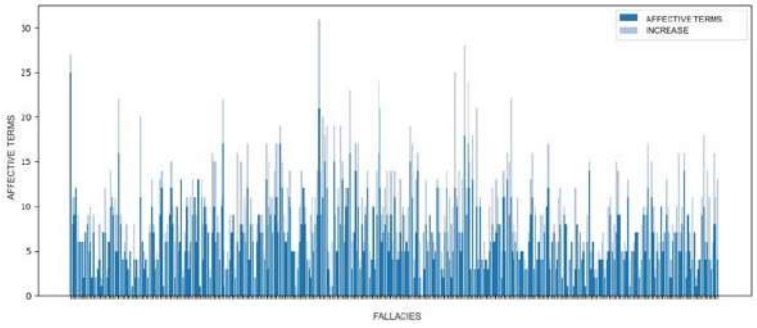


Fig. 4. Increase of affective terms in fallacies with the emotional traits dictionary

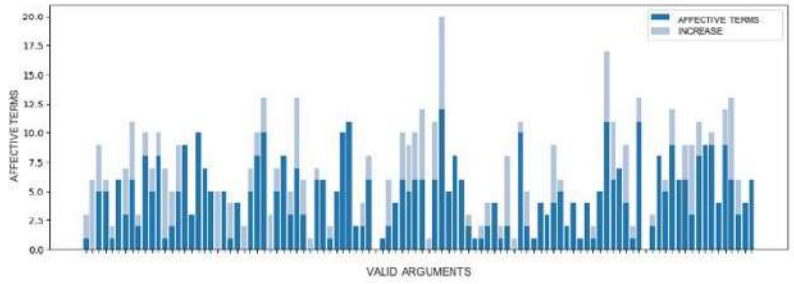


Fig. 5. Increase of affective terms in valid arguments with the emotional features dictionary

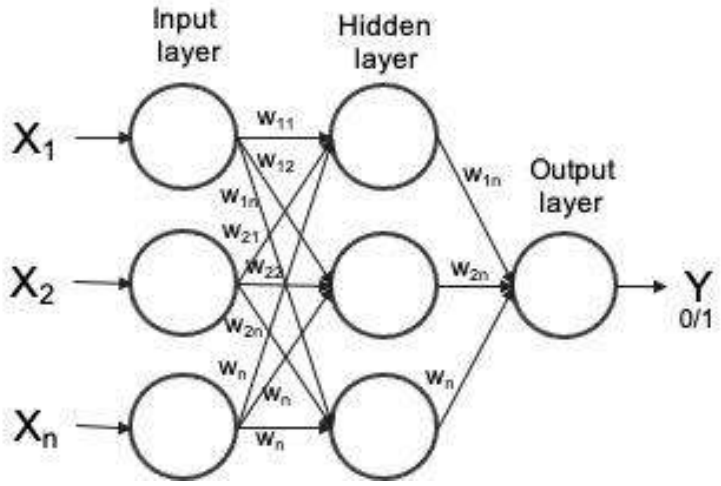


Fig. 6. MLP method architecture

The evaluation of features with SVM and MLP models was performed using a 10-fold cross-validation with 70% of the data for training, and the results are presented using the F1-score metric.

In the experiments, affective terms were evaluated by independently considering the dictionaries and then calculating the information gain (IG) of the affective term (AT) set (Table 8). In this test, a performance of 0.48 was obtained by processing the GI (TA) with the MLP model.

The result increased to an F1-score of 0.56 when processing the set of ATs (Table 9); and when grouping lexical diversity (LD) and ATs with tokens (Table 10). However, the best performance, an F1-score of 0.60, was obtained when grouping the gain of affective terms, lexical diversity, and tokens (Table 10).

Table 8. Result using affective terms and information gain from them

Features	**SVM**	**MLP**
ISOL	0,45	0,45
SEL	0,44	0,44
ET	0,43	0,45
IG(AT)	**0,48**	**0,48**

Table 9. Results considering affective terms, lexical diversity and Tokens

Features	**SVM**	**MLP**
Tokens	0,54	0,54
AT	0,46	**0,56**
LD	0,48	0,53

Table 10. Results considering groups of features

Features	**SVM**	**MLP**
Tokens + IG (AT)	0,54	0,54
AT + *Tokens*	0,52	**0,56**
AT + LD	0,55	0,55
LD + *Tokens*	0,55	**0,56**
LD + IG (AT)	0,48	0,53
Tokens + IG (AT) + LD	0,58	**0,60**
Tokens + AT + LD	0,48	0,53

The obtained result was lower than expected according to [30]. We believe that increasing the affective lexicon and using lexical diversity instead of textual similarity of arguments would improve the results. This is because there is a relationship between diversity and affective terms,

i.e., both are evaluated based on the lexical set used in arguments. However, the result obtained is higher when considering only affective terms and increasing the lexical set with noun and adjective syntagms: in [30], an F1-score of 0.42 was obtained with SVM method, while in this study, an F1-score of 0.46 was obtained with SVM and 0.56 with MLP (Table 9).

Arguments contain several affective terms, including valid arguments. By including noun and adjective syntagms, fallacies have at least one affective term, but valid arguments contain more affective terms. This compared to the affective terms set used in [30]. Additionally, lexical diversity slightly decreases when increasing the number of affective terms in arguments (Fig. 2 and 3). Therefore, valid arguments are classified as fallacies (Fig. 7). Although lexical diversity and textual similarity are widely used models in machine learning, and lexical dictionaries are currently little used linguistic resources in automatic identification, obtaining of new affective terms related to political discourse has increased the result obtained in [30].

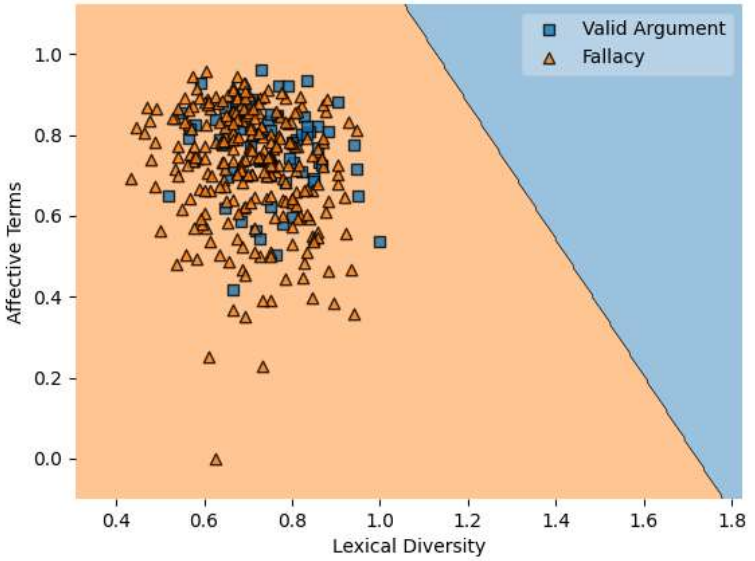


Fig. 7. Plotting the SVM data with affective terms and lexical diversity

8. Conclusions

There is a limited systematic study on fallacies to determine their complexity of treatment from a linguistic perspective and the difficulty this phenomenon poses to Language Technologies. This paper proposes a set of main elements to consider in the development of systems using machine learning methods. We believe that evaluating arguments using a logical perspective and argumentative patterns is the best option for developing systems that allow for fallacy mining.

In fallacy identification, features related to expressive language are used, such as affective terms and lexical diversity of arguments; Two machine learning models were implemented: Support Vector Machine and Multilayer Perceptron. As a result, an F1-score of 0.56 was obtained with affective terms processing, 0.53 using lexical diversity, and 0.60 when grouping the information gain of affective terms, lexical diversity, and argument tokens.

The use of affective terms is considered the main feature to determine if an argument is a fallacy. Despite obtaining low performance with this feature, by obtaining a set of terms used specifically in political speeches, the results increased in relation to previous work. Therefore, affective lexical dictionaries related to political discourse are necessary to identify fallacies by appealing to emotions.

Based on our results, it is proposed to increase the corpus data, get new affective terms, and use conventional neural models such as recurrent networks. Additionally, the topic discussed in the

argument should be considered as an additional feature for fallacy identification. This will help to elaborate semantic fields according to the argument's topic, in order to identify and create an affective lexical dictionary related to specific themes. Lastly, perform tests incorporating the textual similarity presented in [30]. Also, balance the corpus data by having an equal number of fallacies and valid arguments.

References

[1]. Wells G. Minería de falacias en el discurso político. Grade Thesis, Barcelona: Universidad de Barcelona, 2018.
[2]. Kacprzak M., Sawicka A. Identification of formal fallacies in a natural dialogue, Fundamenta Informaticae, vol. 135, no. 4, 2014, pp. 403-417.
[3]. Kacprzak M., Yaskorska O. Dialogue protocols for formal fallacies, Argumentation, vol. 28, no. 3, 2014, pp. 349-369.
[4]. Van Eemeren F.H., Garssen B., Meuffels B. Fallacies and judgments of reasonableness: Empirical research concerning the pragma-dialectical discussion rules, Springer, 2009.
[5]. Dowden B. Fallacies. The Internet Encyclopedia of Philosophy. https://iep.utm.edu/fallacy/. Accessed may, 2019.
[6]. Lippi M., Torroni P. Argumentation mining: State of the art and emerging trends, ACM Transactions on Internet Technology, vol. 16, no. 2, 2016, pp. 1-25. https://doi.org/10.1145/2850417.
[7]. Copi I.M., Cohen C. Introducción a la lógica, México: Limusa, 2013
[8]. Capaldi N. Como Ganar una Discusión: El Arte de la Argumentación, Barcelona: Gedisa, 2011.
[9]. Hansen, H. Fallacies. Stanford Encyclopedia of Philosophy Archive. https://plato.stanford.edu/archives/fall2019/info.html. Accessed December 2019].
[10]. Vega R.L. La fauna de las falacias, Madrid: Trotta, 2013.
[11]. Van Eemeren F.H. Grootendorst R. Fallacies in pragma-dialectical perspective, Argumentation, vol. 1, no. 3, 1987, pp. 283-301. https://doi.org/10.1007/BF00136779.
[12]. Van Eemeren F.H., Grootendorst R. Argumentación, comunicación y falacias: una perspectiva pragma-dialéctica, Chile: Ediciones universidad católica de chile, 2002.
[13]. Weston A. Las claves de la argumentación, Barcelona: Ariel, 2006.
[14]. Damer T.E. Attacking Faulty Reasoning: A Practical Guide to Fallacy-Free Arguments, Estados Unidos: Wadsworth Pub Co, 2009.
[15]. Zurloni V., Anolli L. Fallacies as argumentative devices in political debates, International Workshop on Political Speech, pp. 245-257, 2013. https://doi.org/10.1007/978-3-642-41545-6_18.
[16]. Harada O.E. 2009: Argumentos, formalización y lógica informal, CIENCIA ergo-sum, vol.16, no. 2, 2009, pp. 125-13.
[17]. Jason G. Fallacies are common, Informal Logic, vol. 11, no. 2, 1989, pp. 101-106. https://doi.org/10.22329/il.v11i2.2624.
[18]. Jason G. Are Fallacies Common? A Look at Two Debates, Informal Logic, vol. 8, no. 2, 1986, pp. 81-92. https://doi.org/10.22329/il.v8i2.2685.
[19]. Santoso J.M.A. Fallacy Analysis of the Arguments on the First US Presidential Debate Between Hillary Clinton and Donald Trump, Kata Kita, vol. 5, no. 2, 2017, pp. 65-71.
[20]. Blassnig S., Büchel F., Ernst N., Engesser S. Populism and Informal Fallacies: An Analysis of Right-Wing Populist Rhetoric in Election Campaigns, Argumentation, vol. 33, no. 1, 2018, pp. 107-136. https://doi.org/10.1007/s10503-018-9461-2.
[21]. Cabrejas-peñuelas A.B. Manipulation in Spanish and American pre-election political debates: The Rajoy–Rubalcaba vs. Obama–McCain debates, Intercultural Pragmatics, vol. 12, no. 4, 2015, pp. 515–546. https://doi.org/10.1515/ip-2015-0025.
[22]. Warman J.S., Hamzah H. An Analysis of Logical Fallacy on Prabowo Subianto's Argumentation during 2019 Indonesia Presidential Debate, Lingua Didaktika: Jurnal Bahasa dan Pembelajaran Bahasa, vol. 14, no. 1, 2020, pp. 70-80.
[23]. Hameed S.K., Al-Asadi R.A.N.M. Analysis of Fallacies in Hillary and Trump's Second Presidential Debate, International Journal of English Literature and Social Sciences, vol. 3, no. 4, 2018, pp. 625-635.
[24]. Al-Hindawi F.H., Alkhazaali M.A., Al-Awadi D. A Pragmatic Study of Fallacy in David Cameron's Political Speeches, Journal of Social Science Studies, vol. 2, no. 2, 2015, pp. 214-239.

[25]. Greenwell W.S., Knight J.C., Holloway C.M. Pease J. J. A Taxonomy of Fallacies in System Safety Arguments. International System Safety Conference, pp. 1-10, 2006.
[26]. Neuman Y., Weinstock M.P., Glasner A. The effect of contextual factors on the judgement of informal reasoning fallacies, Quarterly Journal of Experimental Psychology, vol. 59, no. 2, 2006, pp. 411-425.
[27]. Hiba B. When Education Becomes an Empowering Act: Learning Fallacies Analysis, American Journal of Educational Research, vol. 8, no. 9, 2020, pp. 731-738.
[28]. Tobolka S. J. Competencia de los hablantes en la identificación de falacias: una perspectiva pragma-dialéctica, Onomázein, vol. 15, 2007, pp. 129-155. https://doi.org/10.7764/onomazein.15.05.
[29]. Morales G.I.M. Falacias en los discursos de los candidatos presidenciales en México (2012), Revista latinoamericana de estudios del discurso, vol. 12, no. 2, 2016, pp. 11-32.
[30]. Nieto-Benitez K., Castro-Sánchez N. A., Salazar H. J., Bel-Enguix G. Corpus de falacias por apelación a las emociones: una aproximación a la identificación automática de falacias, Linguamática, vol. 14, no. 2, 2022, pp. 59-72.
[31]. Camargo L.S.P. La construcción de la emoción en los discursos políticos de campaña, Pragmalingüística, vol. 26, 2018, pp. 199-220.
[32]. Johnson R. H. Manifest rationality: A pragmatic theory of argument, Routledge, 2000.
[33]. Gilbert M.A. Emotion, argumentation and informal logic, Informal Logic, vol. 24, no. 3, 2004, pp. 245-264. https://doi.org/10.22329/il.v24i3.2147.
[34]. Walton D.N. A Pragmatic Theory of Fallacy, London: University of Alabama Press, 1995.
[35]. Tindale C.W. Fallacies and argument appraisal, Nueva York: Cambridge University Press, 2007.
[36]. Walton D.N. Informal logic: A handbook for critical argument, Cambridge University Press, 1989.
[37]. Walton D.N. Dialogue theory for critical thinking, Argumentation, vol. 3, no. 2, 1989, pp. 169-184.
[38]. Pirie M. How to Win Every Argument: The Use and Abuse of Logic, Lodon/New York: Continuum, 2006.
[39]. Govier T. A Practical Study of Argument, Estados Unidos: Cengage Learning, 2013.
[40]. Lawrence, J., Reed, C. Argument mining: A survey, Computational Linguistics, vol. 45, no. 4, 2020, pp. 765-818. https://doi.org/10.1162/coli_a_00364.
[41]. Cabrio E., Villata S. Five Years of Argument Mining: a Data-driven Analysis, IJCAI, vol. 18, pp. 5427-5433, 2018.
[42]. Garcia-Gorrostieta J.M., Lopez-Lopez A. Argument component classification in academic writings, Journal of Intelligent & Fuzzy Systems, vol. 34, no. 5, 2018, pp. 3037-3047.
[43]. Garcia-Gorrostieta J.M., Lopez-Lopez A. A corpus for argument analysis of academic writing: argumentative paragraph detection, Journal of Intelligent and Fuzzy Systems, vol. 36, no. 5, 2019, pp. 4565-4577. 10.3233/JIFS-179008.
[44]. García-Gorrostieta J.M., López-López A., González-López S. Automatic argument assessment of final project reports of computer engineering students, Computer Applications in Engineering Education, vol. 26, no. 5, 2018, pp. 1217-1226.
[45]. Stab C., Gurevych I. "Recognizing insufficiently supported arguments in argumentative essays," Proceedings of th 15th Conference of the European Chapter of the Association for Computational Linguistics, vol. 1, pp. 980-990, 2017.
[46]. Habernal I., and Gurevych I. "What makes a convincing argument? empirical analysis and detecting attributes of convincingness in web argumentation," Conference on Empirical Methods in Natural Language Processing, pp. 1214-1223, 2016.
[47]. Molina-González M. D., Martínez-Cámara E., Martín-Valdivia M. T., Perea-Ortega J. M. Semantic orientation for polarity classification in Spanish reviews, Expert Systems with Applications, vol. 40, no. 18, 2013, pp. 7250-7257.
[48]. Rangel I.D., Sidorov G., Suárez-Guerra S. Creación y evaluación de un diccionario marcado con emociones y ponderado para el español, Onomazein, vol. 29, 2014, pp. 31-46.
[49]. Brezina V. Statistics in Corpus Linguistics: A Practical Guide, Cambridge: Cambridge University Press, 2018.
[50]. Habernal I., Gurevych I. Argumentation mining in user-generated web discourse, Computational Linguistics, vol. 43, no. 1, 2017, pp. 125-179.
[51]. Sierra M.G. Introducción a los Corpus Lingüísticos. México: Universidad Nacional Autónoma de México-Instituto de Ingeniería, 2017.

Информация об авторах / Information about authors

Кения НЬЕТО-БЕНИТЕС – докторант Национального центра технологических исследований и разработок (TecNM/CENIDET) по специальности "Программирование". Получила степень магистра программирования от TecNM/CENIDET в 2017. В настоящее время занимается вопросами обработки текстов на естественных языках, в частности, изучает язык политических рассуждений. Сфера научных интересов: естественные языки, машинное обучение и формализация естественных языков.

Kenia NIETO-BENITEZ – Doctoral student in Computer Science at Centro Nacional de Investigación y Desarrollo Tecnológico (TecNM/CENIDET). She received the master's degree in computer science from TecNM/CENIDET in 2017. She currently works in Natural Language Processing, especially in the study of language of political discourse. Her current research interests include NLP, machine learning and the denaturation of language.

Ноэ Алехандро КАСТРО-САНЧЕС прошел обучение в магистратуре и аспирантуре Вычислительного центра Национального политехнического института, специализирующегося в области исследований естественных языков. Является членом мексиканской Национальной системы поддержки исследователей, руководящего комитета мексиканского общества искусственного интеллекта, латиноамериканской ассоциации языковых технологий. В настоящее время является преподавателем Национального центра исследований и технологического развития (CENIDET).

Noe Alejandro CASTRO-SANCHEZ completed his master's and doctoral studies at the Research Center in Computing of the National Polytechnic Institute, specializing in Natural Language Processing. He is a member of the National Researchers System, a board member of the Mexican Society of Artificial Intelligence, and a member of the Latin American Association of Language Technologies. He serves as a faculty member at the National Center for Research and Technological Development (CENIDET).

Эктор Хименес САЛАСАР – имеет степень бакалавра по математике, позднее получил степени магистра и PhD по программированию. В течение 15 лет он является активным членом мексиканской Ассоциации по исследованию естественных языков. Преподавал программирование в Автономном университете Пуэбла, а с 2007 года является профессором Автономного университета "Метрополитен". Сфера научных интересов: исследование естественных языков и процессы преподавания и обучения.

Hector Jimenez SALAZAR studied for a bachelor's degree in mathematics and later a master's and PhD in computer science. He has been an active member of the Mexican Natural Language Processing Association for more than 15 years. He has taught computing at the Autonomous University of Puebla, and since 2007 he has been a professor at the Autonomous Metropolitan University. In addition to his interest in NLP is the feedback in the teaching-learning process.

Хемма БЕЛЬ-ЭНГИКС имеет степень PhD по вычислительной лингвистике от государственного Университета Ровира и Вирхилий в Тарагоне. С 2016 года она ведет исследования в Институте инженерии Национального автономного университета Мексики. В настоящее время специализируется в области исследований естественных языков, в частности, изучает языки социальных платформ и сетей, сложность и проблемы выявления половых различий, а также языки выражения агрессии. Являлась научным редактором 8 книг и автором более 100 научных статей в индексируемых журналах, глав в книгах и трудах конференций.

Gemma BEL-ENGUIX – PhD in Computational Linguistics from the Rovira i Virgili University (Tarragona). Since 2016 she is a researcher at the Instituto de Ingeniería, at the Universidad Nacional

Autónoma de Mexico. She currently works in Natural Language Processing, especially in the study of language of social platforms, graphs, complexity and detection of sexism and aggresive language. She is the editor of 8 books and author of more than 100 research articles in indexed journals, book chapters and conference proceedings.

Данте МУХИКА ВАРГАС получил степень PhD по телекоммуникациям и электронике в подразделении дополнительного образования и исследований инженерного факультета Национального политехнического института Мексики. С 2015 года является профессором Отделения вычислительных наук Национального технологического центра Мексики. В настоящее время его научные интересы включают в себя глубокое обучение, нечеткую кластеризацию, нечеткие нейросистемы, цифровую обработку сигналов и биомедицинские приложения.

Dante MÚJICA VARGAS received the Ph.D. degree in Communications and Electronics from Seccion de Estudios de Posgrado e Investigación, ESIME-Culhuacán, Instituto Politécnico Nacional in Mexico. He is a professor from 2015 at the Departamento de Ciencias Computacionales, Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico. His current research interests include deep learning, machine learning, fuzzy clustering, neuro-fuzzy systems, digital signal processing and biomedical applications.

Хуан Габриель ГОНСАЛЕС СЕРНА получил степень PhD по программированию в Исследовательском вычислительном центре Национального политехнического института в 2006 году. До этого, в 1995 году он получили степень магистра по программированию в Национальном центре исследований и технологического развития. С тех пор до настоящего времени он работает профессором и ведет исследования на факультете Программирования TecNM/CENIDET. Область научных интересов: человеко-машинное взаимодействие, аффективные вычисления и анализ мнений, а также изучение пользовательского опыта.

Juan Gabriel GONZÁLEZ SERNA earned his Ph.D. in Computer Science from the Research Center in Computing of the National Polytechnic Institute (CIC-IPN) in 2006. He obtained a Master's degree in Computer Science from the National Research and Technological Development Center (TecNM/CENIDET) in 1995. He has been a Professor-Researcher in the Department of Computer Science at TecNM/CENIDET since 1995 to the present. His research areas include Human-Computer Interaction, Affective Computing and Sentiment Analysis, and User Experience (UX) Evaluation.

Нимрод ГОНСАЛЕС ФРАНКО работает профессором-исследователем в Национальном центре исследований и технологического развития TecNM/CENIDET в Куэрнаваке, Мексика, где с 2019 года изучает гибридные интеллектуальные системы. Ведет рецензирование научных статей для различных журналов и научных конференций, включая такие известные события, как Всемирная мультиконференция по системности, кибернетике и информатике, а также Мексиканской международной конференции по искусственному интеллекту. Сфера его научных интересов охватывает различные области, с акцентом на системы интерфейсов мозг-компьютер и машинное обучение.

Nimrod GONZÁLEZ FRANCO – joined TecNM/CENIDET, located in Cuernavaca, Mexico, as a research professor in the field of Intelligent Hybrid Systems in 2019. He has served as a reviewer for scientific articles across multiple journals and conferences, including prominent events like the World Multi-Conference on Systemics, Cybernetics and Informatics, as well as the Mexican International Conference on Artificial Intelligence. His research spans diverse areas, with a focus on brain-computer interface systems and machine learning.