DOI: 10.15514/ISPRAS-2025-37(6)-23

Применение контрастного обучения для семантической интерпретации русскоязычных таблиц

К.В. Тобола, ORCID: 0009-0006-1014-451X <kirilltobola@gmail.com>
Н.О. Дородных, ORCID: 0000-0001-7794-4462 <nikidorny@icc.ru>
Институт динамики систем и теории управления имени В.М. Матросова СО РАН,
Россия, 664033, г. Иркутск, ул. Лермонтова, д. 134.

Аннотация. Таблицы широко используются для представления и хранения данных, но, как правило, они не сопровождаются явной семантикой необходимой для машинной интерпретации своего содержания. Семантическая интерпретация таблиц является ключевой задачей для интеграции структурированных данных с графами знаний, однако существующие методы сталкиваются с проблемами при обработке русскоязычных таблиц из-за недостатка размеченных данных и языковой специфики. В данной работе предложен подход на основе контрастного обучения, направленный на устранение зависимости от ручной разметки и улучшение качества аннотирования столбцов редкими семантическими типами. Подход включает адаптацию алгоритма контрастного обучения для табличных данных с использованием аугментаций (удаление и перестановка ячеек), а также дистиллированной мультиязычной модели DistilBERT для эффективного обучения на неразмеченных данных корпуса RWT, содержащего 7.4 млн. столбцов. Обученные табличные представления интегрируются в конвейер аннотирования фреймворка RuTaBERT, что позволяет снизить вычислительные затраты. Эксперименты показали, что предложенный подход достигает микро-F1 97% и макро-F1 92%, превосходя некоторые базовые решения, что подтверждает его эффективность в условиях разреженности данных и языковых особенностей русского языка. Результаты демонстрируют, что контрастное обучение позволяет моделировать семантическое сходство между столбцами без явной разметки, что особенно важно для данных редких типов.

Ключевые слова: русскоязычные таблицы, табличные данные; семантическая интерпретация таблиц; семантическое аннотирование столбцов; графы знаний; самообучение; контрастное обучение; табличные представления.

Для цитирования: Тобола К.В., Дородных Н.О. Применение контрастного обучения для семантической интерпретации русскоязычных таблиц. Труды ИСП РАН, том 37, вып. 6, часть 2, 2025 г., стр. 107–122. DOI: 10.15514/ISPRAS–2025–37(6)–23.

Благодарности: Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации (тема № 1023110300006-9).

Using Contrastive Learning for Semantic Interpretation of Russian-Language Tables

K.V. Tobola, ORCID: 0009-0006-1014-451X <kirilltobola@gmail.com>
N.O. Dorodnykh, ORCID: 0000-0001-7794-4462 <nikidorny@icc.ru>

Matrosov Institute for System Dynamics and Control Theory of the Siberian Branch of Russian Academy of Sciences (ISDCT SB RAS),
134, Lermontov st., Irkutsk, 664033, Russia.

Abstract. Tables are widely used to represent and store data, but they are typically not accompanied by explicit semantics necessary for machine interpretation of their contents. Semantic table interpretation is critical for integrating structured data with knowledge graphs, but existing methods struggle with Russian-language tables due to limited labeled data and linguistic specificity. This paper proposes a contrastive learning-based approach to reduce dependency on manual labeling and improve column annotation quality for rare semantic types. The proposed approach adapts contrastive learning for tabular data using augmentations (removing/shuffling cells) and a distilled multilingual DistilBERT model trained on unlabeled RWT corpus (7.4M columns). The learned table representations are integrated into the RuTaBERT pipeline, which reduces computational costs. Experiments show micro-F1 0.974 and macro-F1 0.924, outperforming some baselines. This highlights the approach's efficiency in handling data sparsity and Russian language features. Results confirm that contrastive learning captures semantic column similarities without explicit supervision, crucial for rare data types.

Keywords: Russian-language tables, tabular data; semantic table interpretation; semantic column annotation; knowledge graphs; self-supervised learning; contrastive learning; table representations.

For citation: Tobola K.V., Dorodnykh N.O. Using contrastive learning for semantic interpretation of Russian-language tables. Trudy ISP RAN/Proc. ISP RAS, vol. 37, issue 6, part 2, 2025, pp. 107-122 (in Russian). DOI: 10.15514/ISPRAS-2025-37(6)-23.

Acknowledgements. This work was supported by the state assignment of Ministry of Science and Higher Education of the Russian Federation (theme No. 1023110300006-9).

1. Введение

Табличные данные являются одним из ключевых форматов представления структурированной информации в различных областях: от научных исследований до бизнес-аналитики. Они используются в реляционных базах данных, электронных таблицах, веб-ресурсах и документах, что делает их обработку критически важной для автоматизации анализа данных. Однако таблицы, как правило, не сопровождаются явной семантикой необходимой для машинной интерпретации своего содержания. Поэтому семантическая интерпретация таблиц, особенно на языках отличных от английского, остается сложной задачей [1-2]. Основные вызовы связаны с необходимостью соотнесения отдельных элементов таблиц (столбцов, строк, ячеек) с понятиями из графов знаний, такими как DBpedia или Wikidata, а также с обработкой структурного и языкового разнообразия данных.

Русскоязычные таблицы представляют особую проблему из-за ограниченного количества специализированных инструментов и размеченных данных. Большинство современных методов, в частности, основанные на предварительно обученных языковых моделях по типу BERT [3-9], требуют огромных объемов размеченных данных, которые для русского языка часто недоступны или не сбалансированы. Кроме того, существующие решения, разработанные для английского языка, плохо адаптируются к другим языкам из-за различий в токенизации и контекстуальной семантике.

В данной работе предлагается новый подход для семантического аннотирования столбцов русскоязычных таблиц на основе контрастного обучения. Проверяется утверждение, что контрастное обучение на неразмеченных табличных данных улучшает способность модели различать семантические типы столбцов без использования вручную размеченного корпуса

табличных данных. Таким образом, подход позволяет эффективно использовать неразмеченные табличные данные для обучения устойчивых векторных представлений, снижая зависимость от ручной разметки. Наш вклад включает:

1) Адаптацию контрастного обучения для русскоязычных табличных данных с применением аугментаций. Обычно под аугментацией данных понимается техника искусственного увеличения размера обучающей выборки путем применения некоторых преобразований к исходным данным. Для табличных данных в этот процесс также включают удаление и перестановку ячеек.
2) Использование дистиллированной мультиязычной модели DistilBERT, что обеспечивает баланс между производительностью и вычислительными затратами.
3) Интеграцию предобученных табличных представлений в существующий конвейер аннотирования на базе фреймворка RuTaBERT [9], что демонстрирует гибкость подхода.
4) Эксперименты на крупномасштабном русскоязычном наборе данных RWT-RuTaBERT [10] показали, что предложенный подход превосходит некоторые базовые решения, что подтверждает его эффективность в условиях разреженности данных и языковой специфики.

Статья организована следующим образом: раздел 2 представляет современное состояние исследований в области семантической интерпретации таблиц. В разделе 3 описывается предложенный подход для семантического аннотирования столбцов русскоязычных таблиц, включая подготовку данных, архитектуру модели и алгоритм обучения. Раздел 4 содержит экспериментальные оценки тестирования производительности предлагаемого подхода. В заключении (раздел 5) дается обсуждение полученных результатов и планы будущей работы.

2. Современное состояние исследований

Под семантической интерпретацией (аннотированием) таблиц (*semantic table interpretation*) понимается процесс распознавания и связывания табличных данных с понятиями из некоторого целевого графа знаний, онтологии или другого внешнего словаря (например, DBpedia, Wikidata, Yago, Freebase, WordNet) [2, 11]. Одной из основных задач семантической интерпретации таблиц является аннотирование столбцов (*column type annotation*), при котором осуществляется сопоставление столбцов таблицы с семантическими типами (классами и свойствами) из целевого графа знаний.

За последние несколько лет существующие методы и модели использовали передовые достижения в области глубокого машинного обучения, формулируя задачу аннотирования столбцов как задачу классификации нескольких классов (*multi-class classification*). Так в работе [12] применяли нейронные сети и множество извлеченных групп признаков, таких как векторные представления слов и символов, а также глобальные статистики столбцов. В работе [13] добавлен анализ локального (внутри-табличного) контекста таблицы (соседних столбцов относительно целевого столбца), а в работе [14] добавился еще и межтабличный контекст для улучшения предсказаний. Однако особый интерес в данном контексте представляют работы, использующие предварительно обученные языковые модели на основе архитектуры трансформер (*Transformer*). Блоки трансформера используют механизм внимания, что позволяет модели генерировать полезные контекстуализированные векторные представления для структурных компонентов табличных данных, таких как ячейки, столбцы или строки. Также языковые модели, предварительно обученные на крупномасштабных текстовых корпусах, могут хранить семантику из обучающего текста в форме параметров модели, что делает процесс дообучения таких моделей на конкретных последующих задачах (*downstream tasks*) достаточно эффективным. Примерами таких работ являются модели TURL [3], TaPas [4], TaBERT [5], TABBIE [6], TUTA [7], Doduo [8].

Существующие решения в этой области имеют высокую производительность, которая достигается за счет большого количества размеченных обучающих данных. В частности, англоязычные наборы данных могут включать сотни тысяч размеченных столбцов (например, VizNet-Sato [13] ~ 100 000, WikiTables-TURL [3] ~ 600 000), а русскоязычный набор табличных данных RWT-RuTaBERT [10] насчитывает более 1.4 миллиона столбцов. Создание таких наборов является достаточно трудоемким процессом, требующим большого количества времени и ресурсов. Более того, для существующих наборов таблиц свойственна проблема разреженности данных, которая выражается в, довольно, несбалансированном распределении семантических типов (так называемое «*распределение с длинным хвостом*»). Так некоторым семантическим типам соответствуют сотни тысяч столбцов, а некоторым лишь несколько десятков. В результате модели сложно уловить достаточное количество сигналов для семантических типов, относящихся к меньшинству (редким типам, таким как «*атлет*», «*горный хребет*» или «*страховая компания*»), даже при контролируемых (*supervised*) настройках. Например, график распределения 30 наиболее встречающихся семантических типов для набора данных RWT-RuTaBERT, демонстрирующий эту проблему, представлен на рис. 1.

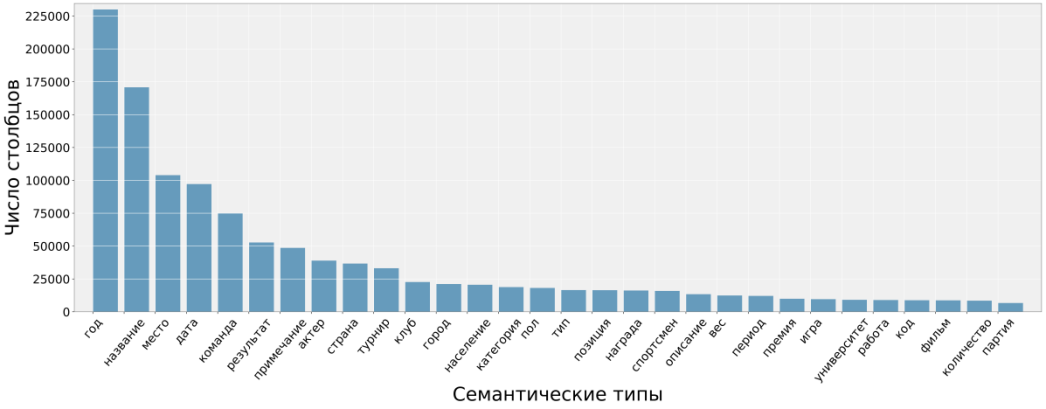

Рис. 1. Пример проблемы разреженности данных для набора RWT-RuTaBERT.
Fig. 1. Example of a data sparsity problem for the RWT-RuTaBERT dataset.

Существующие методы на основе предварительно обученных языковых моделей не являются универсальными. Наблюдается разрыв между эффективностью существующих решений на тестовых примерах и их применимостью на практике. Особенно это касается таблиц с различной языковой принадлежностью (представленных не на английском языке) и имеющих разную структурную компоновку.

Для повышения способности общего понимания таблиц и решения различных табличных задач появились работы, использующие большие языковые модели, которые часто имеют лучшую производительность по сравнению с предварительно обученными языковыми моделями, такими как BERT. Они также более устойчивы к невидимым примерам из-за специфичных эффектов, возникающих в результате размера модели, обученной на огромных объемах текста. Примерами таких работ являются модели Table-GPT [15] и TableLlama [16], а также подход [17]. Однако основным недостатком таких решений является то, что они требуют огромных вычислительных ресурсов, что затрудняет их практическое использование.

Для решения вышеуказанных проблем предлагается использовать методы самообучения (*self-supervised learning*), в частности, контрастного обучения (*contrastive learning*) для изучения табличных представлений, полученных на основе обширного корпуса неразмеченных табличных данных. Данные табличные представления могут быть

использованы как для определения родства (*relatedness*) между двумя таблицами (путем вычисления косинусного сходства между векторными представлениями), так и для тонкой настройки с дополнительными размеченными данными, содержащего небольшое количество таблиц, под конкретные последующие задачи (*downstream tasks*).

3. Предлагаемый подход

3.1 Постановка задачи

Таблица – это двухмерная структура данных, состоящая из строк и столбцов. В ячейках таблиц могут содержаться текстовые данные, числовые, дата и время и так далее. Можно выделить три вида таблиц с точки зрения структурированности информации:

1) *сильно структурированные* (таблицы реляционных баз данных);
2) *полуструктурированные* (электронные таблицы, составленные в специализированном программном обеспечении, например, MS Excel и так далее);
3) *неструктурированные* (изображения таблиц в документах формата PDF).

Те же таблицы можно классифицировать, вводя три основные группы в зависимости от ориентации:

1) *вертикальные* – таблицы, в которых данные расположены в виде вертикальных колонок (то есть идут "сверху вниз");
2) *горизонтальные* – таблицы, в которых данные расположены в виде горизонтальных линий (то есть идут "слева направо");
3) *матричные* – таблицы, в которых каждая запись индексируется ключом(ями) строки и ключом(ями) столбца.

В данной работе рассматриваются только вертикальные, сильно структурированные и полуструктурированные таблицы. Формальное описание входной таблицы можно представить как:

$$T = \{col_1, \dots, col_n\}, col_i = \{cell_1, \dots, cell_m\}, i \in \overline{1,n}, \quad (1)$$

где T – вертикальная таблица; col_i – i-столбец; $cell_j$ – j-ячейка i-столбца, при этом $j \in \overline{1,m}$.

Наша цель предсказать тип столбца, то есть классифицировать каждый столбец по его семантическому типу, например, «*Книга*», «*Писатель*», «*Жанр*» и «*Дата публикации*», вместо стандартных типов данных, таких как строка (*string*), число (*integer*) или дата (*datetime*). Предлагаемый подход подразумевает использование 170 различных семантических типов, которые были сформированы на основе отобранных классов и свойств (свойств-значений и объектных свойств) из графа знаний общего назначения DBpedia [18]. При этом брались только русские обозначения этих типов через метку языка (*label*), так как подход ориентирован на аннотирование русскоязычных таблиц. Формально данную задачу можно описать следующим образом:

$$P(col_i) \in KG_{st}, KG_{st} = \{st_1, \dots, st_{170}\}, \quad (2)$$

где $P(col_i)$ – предсказанный семантический тип для i-столбца; KG_{st} – множество всех семантических типов, мощность которого в данном случае равна 170.

Пример решения задачи аннотирования столбцов для входной таблицы представлен на рис. 2.

Основная идея подхода заключается в создании кодировщика устойчивых табличных представлений на основе контрастного обучения, которые затем можно использовать на последующих задачах (*downstream tasks*), в частности, для семантического аннотирования столбцов русскоязычных таблиц. Общая схема предлагаемого подхода представлена на рис. 3.

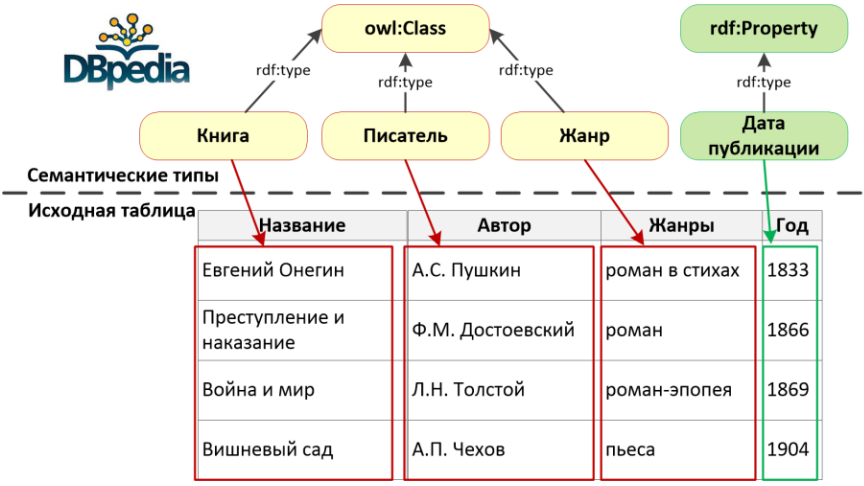

Рис. 2. Пример решения задачи аннотирования столбцов.
Fig. 2. Example of solving the column type annotation task.

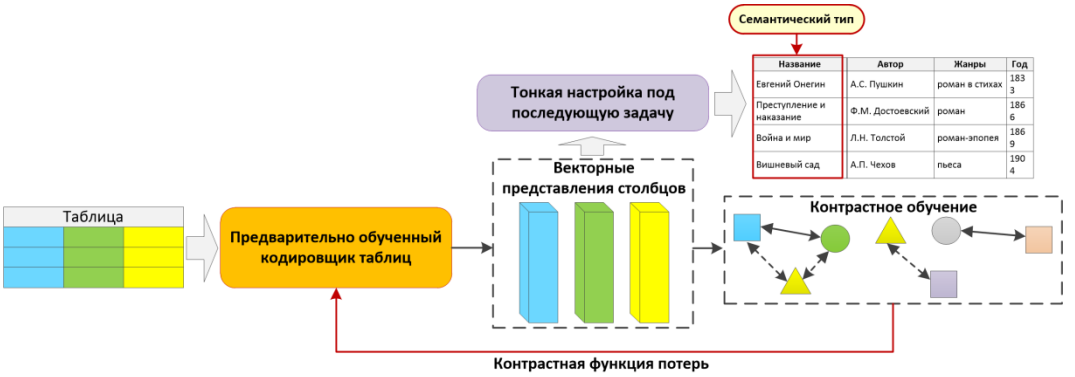

Рис. 3. Общая схема предлагаемого подхода.
Fig. 3. General outline of the proposed approach.

3.2 Описание набора данных

Предварительно обученный кодировщик таблиц (*Encoder*) обучается на огромном количестве табличных данных, которые не нуждается в ручной разметке. В качестве набора исходных таблиц используются крупномасштабный корпус RWT (Russian Web Tables) [19]. Данный набор представляет собой срез таблиц из русскоязычной Википедии за 13 сентября 2021 года. Основные статистики по набору RWT представлены в табл. 1.

На первом этапе предварительной обработки данных, из исходного корпуса RWT были отобраны вертикальные таблицы. При этом каждый столбец из такой таблицы представляется в качестве строки данных с помощью разделителя ячеек «<<». Пример хранения извлеченных столбцов приведен в табл. 2.

Далее была проведена очистка данных при помощи следующих операций:

- фильтрация пустых столбцов;
- удаление служебной информации парсера, оборачивающей текст при помощи регулярных выражений;
- удаление ссылок на статьи Википедии;

- удаление специальных символов (например, «@», «&», «?» и «!»);
- удаление пустых ячеек в столбце;
- удаление столбцов, имеющих длину меньше трех ячеек, так как данные столбцы после применения аугментаций удаления ячеек становятся не репрезентативными.

В результате всех операций очистки был получен набор неразмеченных русскоязычных табличных данных, состоящий из 4 656 668 столбцов.

Табл. 1. Статистика корпуса таблиц RWT.
Table 1. Statistics of the RWT table corpus.

Статистика	**Значение**
Число таблиц	1 266 731
Число столбцов	7 419 771
Число ячеек	99 638 194
Процент пустых столбцов	6%
Среднее количество ячеек в столбце	13.42

Табл. 2. Формат хранения извлеченных столбцов из корпуса RWT.
Table 2. Storage format for extracted columns from the RWT corpus.

id	**table_id**	**column_id**	**column_header**	**column_data**
0	7545708	0	Название	Сан-Хуан (исп. San Juan) << Вальe-Нуэво...
1	5433710	3	Зрители	100 << 200 << 50 << 300 << 500
…	…	…	…	…

В проведенном эксперименте предварительная обработка таблиц осуществлялась в автоматизированном режиме с использованием специального средства [9].

3.3 Алгоритм обучения

Контрастное обучение (*contrastive learning*) – это одна из техник самообучения, предназначенная для получения информативных векторных представлений. Она заключается в максимизации некоторой метрики согласованности, в нашем случае косинусного сходства, между положительными парами (экземплярами данных) и минимизации данной метрики между отрицательными парами. Контрастное обучение позволяет эффективно обучаться на неразмеченном корпусе данных.

В данной работе адаптируется концепция контрастного обучения, предложенная в работе [20], для табличных данных. Алгоритм контрастного обучения для табличных данных представлен на рис. 4.

Основная идея заключается в построении во время обучения для каждого столбца в пакете данных двух аугментаций. Для полученных аугментаций формируются векторные представления при помощи модели кодировщика. Векторные представления для аугментаций, полученные на одном столбце, считаются положительной парой, наша задача максимизировать метрику косинусного сходства для этой пары. В свою очередь векторные представления аугментаций, построенных для различных столбцов, считаются отрицательными парами. Для этих пар решается задача минимизации метрики косинусного сходства.

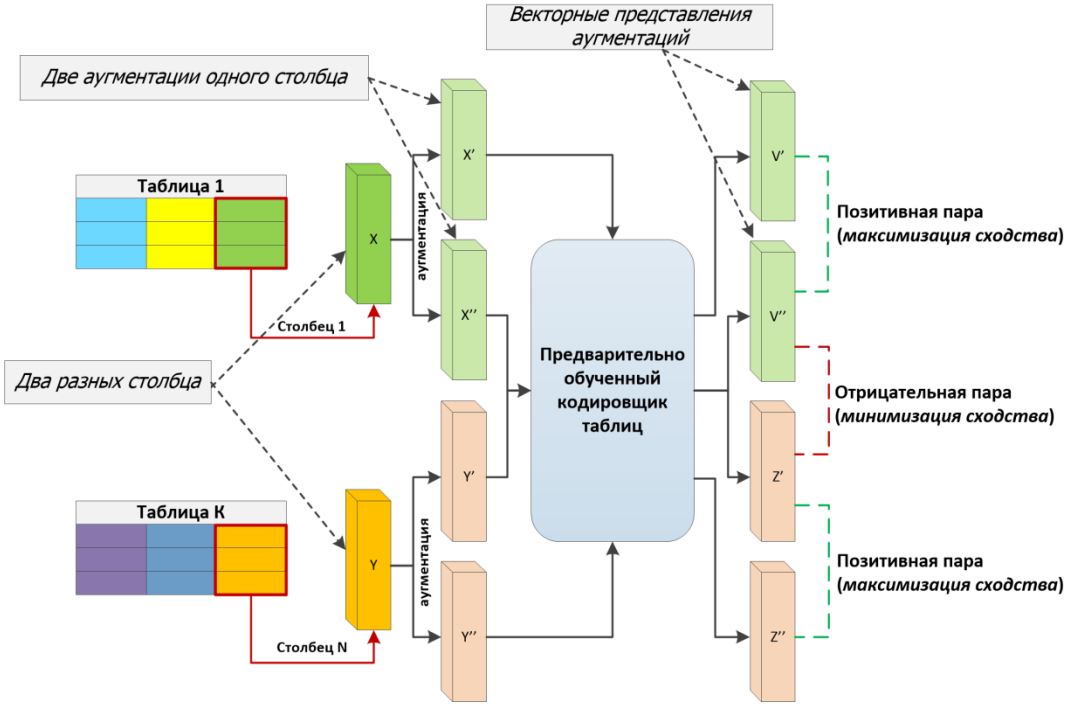

Рис. 4. Алгоритм контрастного обучения для табличных данных.
Fig. 4. Contrastive learning algorithm for tabular data.

3.3.1 Аугментация данных

Техника аугментации данных широко используется в условиях ограниченного количества размеченных данных или их отсутствии для увеличения обобщающей способности модели. В контрастном обучении аугментации играют определяющую роль в формировании семантически согласованных положительных пар.

Как правило, для табличных данных выделяют следующие аугментации:

- удаление случайной ячейки;
- удаление/перестановка/замена токенов в ячейке;
- выборка строк (например, в размере 50%);
- перестановка ячеек в строке таблицы;
- удаление столбцов;
- перестановка столбцов в таблице;

На данный момент нет исследований по определению аугментаций, работающих наилучшим образом для формирования семантически согласованных пар в контексте обработки табличных данных. Поэтому в данной работе, были выбраны две аугментации, которые по нашему предположению, являются наиболее перспективными, а именно: удаление случайных ячеек и перестановка ячеек в столбце. При удалении случайных ячеек, удаляются 10% от всех ячеек в столбце.

3.3.2 Контрастная функция потерь

В задачах обучения представления активно используют контрастные функции потерь (*contrastive loss*), так как с их помощью модель способна лучше различать внутренние

структуры данных, и как следствие лучше извлекать полезные представления. Контрастная функция потерь направлена на максимизацию согласования между положительными парами и минимизацию согласования между отрицательными парами в векторном пространстве.

Существует несколько вариаций для контрастных функций потерь. В данной работе выбрана функция NT-Xent loss (Normalized Temperature-Scaled Cross-Entropy Loss), используемая в работе [20]. Данная функция определяется следующим образом:

$$L_{NT-Xent} = \frac{1}{2N} \times \sum_{k=1}^{N} [l(2k-1,2k) + l(2k,2k-1)],$$

$$l(i,j) = -\log \frac{exp\left(\frac{s(i,j)}{\tau}\right)}{\sum_{k=1}^{2N} 1_{[k \neq i]} exp\left(\frac{s(i,k)}{\tau}\right)},$$

$$s(i,j) = \frac{z_i \times z_j}{\|z_i\| \times \|z_j\|}, \quad (4)$$

где $1_{[k \neq i]} \in \{0,1\}$ принимает значение 1, если $k \neq i$, иначе 0; τ – параметр температуры; $s(i,j)$ – косинусное сходство; $z_i = [z_1, \dots, z_{256}], z \in \mathbb{R}$ – векторное представление (выход модели).

3.4 Архитектура модели

В настоящий момент модели архитектуры трансформер являются ключевыми в решении задач обработки естественного языка. Данные модели являются универсальными инструментами для решения задач обработки текстов за счет их возможности учитывать контекстные зависимости между словами в последовательностях, а также возможности обучаться на неразмеченных или частично размеченных данных. Причем они делают это достаточно эффективно, за счет высокого параллелизма, что делает их предпочтительными для обучения на больших объемах данных.

Согласно работе [20] двумя ключевыми гиперпараметрами контрастного обучения являются размер пакета данных и количество эпох. Чем больше размер пакета данных и количество эпох, тем более репрезентативными получаются векторные представления обучаемой модели, тем лучше она показывает себя на последующих конкретных задачах при тонкой настройке.

Исходя из этого, в качестве базовой модели кодировщика была взята дистиллированная мультиязычная модель BERT [21]. Данная модель обучена на корпусе статьей из Википедии, написанных на 104 различных языках. В отличии от базовой версии [22] данная модель состоит всего из 6 слоев, что в два раза меньше, чем у базовой версии, и 12 модулей механизма внимания. Данная модель насчитывает 134 миллиона параметров (у базовой версии 177 миллионов параметров).

Дистилляция моделей машинного обучения – это техника, при которой осуществляется перенос знаний от более сложной модели, также называемой учителем, к более компактной, называемой учеником. При этом сохраняется качество предсказаний модели.

Данная техника в сочетании с уменьшением максимальной длины последовательности токенизатора, инструмента для разбиения текстового документа на более мелкие единицы (токены), до 256 токенов, позволила обучить модель с размером пакета данных, равным 800, это в 25 раз больше, чем у аналогичного передового англоязычного решения [23].

В адаптируемой работе [20] было проведено исследование об использовании проекции выходного слоя кодировщика в некоторое латентное векторное пространство, в котором рассчитывается контрастная функция потерь. Результаты показывают, что применение нелинейной проекции во время обучения положительно влияет на качество представлений. Таким образом, в данной работе используется двухслойный перцептрон (Multilayer

perceptron, MLP), идущий после выходного слоя кодировщика, для проекции в латентное векторное пространство, размерности 128, в котором рассчитывается контрастная функция потерь по указанным выше формулам.

4. Экспериментальные оценки и обсуждение

Все эксперименты проводились на кластере «Академик В.М. Матросов» на базе Института Динамики систем и Теории управления СО РАН (ИДСТУ СО РАН). В конфигурацию кластера входят два 16-ти ядерных процессора Intel Xeon Gold 6326 «Ice Lake» 2.9 GHz, четыре графических процессора NVIDIA A100 80 GB PCIe и 2 TB оперативной памяти DDR4-3200.

4.1 Настройки контрастного обучения

Подход реализован на языке Python с использованием библиотек PyTorch и Transformers. В качестве оптимизатора градиентного спуска был выбран классический метод AdamW (lr = 5–e5, eps = 1e-6). Для ускорения сходимости модели была применена техника косинусного отжига (*cosine annealing*), предназначенная для динамического уменьшения скорости обучения (*learning rate*). Параметр температуры, являющийся гиперпараметром контрастной функции потерь, равен значению 0.1, так как данное значение является оптимальным, согласно работе [20].

В таких настройках модель предварительно обученного кодировщика была обучена на протяжении 100 эпох, на четырех графических ускорителях NVIDIA A100 с использованием технологии Distributed-Data-Parallel фреймворка PyTorch. Обучение продлилось на протяжении 9 дней 9 часов 53 минут. Количество потребляемой памяти графического процессора составило 290 гигабайт.

4.2 Настройки модели для семантического аннотирования столбцов

В настоящей работе в качестве последующей конкретной задачи выступает семантическая интерпретация (аннотирование) столбцов таблиц. Для решения этой задачи, ранее в работе [9] был предложен фреймворк RuTaBERT, основанный на тонкой настройке предварительно обученной мультиязычной языковой модели BERT с использованием специально подготовленного набора таблиц RWT-RuTaBERT [10]. Данный набор содержит примерно 1.56 миллиона размеченных столбцов. Основная идея заключается в том, чтобы использовать уже готовый конвейер (*pipeline*) этого фреймворка с заменой стандартной модели BERT на специализированный предварительно обученный табличный кодировщик. При этом в качестве набора данных для обучения также выступает размеченный ранее набор RWT-RuTaBERT со всеми стандартными настройками. Размер валидационной выборки составил 5% от общего числа тренировочного подмножества. В качестве разложения значений столбцов в последовательности токенов используется техника сериализации соседних столбцов (*neighboring column serialization*), также предложенная в работе [9].

Согласно работе [20], слой проекции обучен быть инвариантным к преобразованию данных, из-за чего он может потерять информацию, которая может быть полезна для последующих задач (*downstream tasks*). Поэтому для дальнейшего дообучения табличного кодировщика использовался выход с первого линейного слоя проекции с применением функции активации LeakyReLU. При этом применялись стандартные настройки обучения, заданные во фреймворке RuTaBERT. Таким образом, модель была дообучена на протяжении 30 эпох с размером пакета данных равным 32 на наборе данных RWT-RuTaBERT. Для этого использовались 2 графических ускорителя NVIDIA A100, обучение продлилось на протяжении 2 дней 20 часов 15 минут. Потребление памяти графического процессора составило 9.9 гигабайт. Дополнительно была обучена модель с размером пакета данных равным 256 с сохранением всех остальных гиперпараметров. При таких настройках обучение

заняло 4 дня 3 часа 1 минуту. Количество потребляемой памяти графического процессора составило 52 гигабайта.

4.3 Метрики оценки

В качестве основных метрик оценки производительности предлагаемого подхода выступает усредненная F1-мера, поскольку решается задача классификации нескольких классов (*multi-class classification*). В частности, используется микро F1-мера (*micro F1*), макро F1-мера (*macro F1*) и взвешенная F1-мера (*weighted F1*) так как набор данных RWT-RuTaBERT не сбалансирован.

Микро F1-мера рассчитывается по всей матрице ошибок и определяется следующим образом:

$$MicroPrecision = \frac{TP_1+TP_2+\cdots+TP_n}{TP_1+\cdots+TP_n+FP_1+\cdots+FP_n}, \quad MicroRecall = \frac{TP_1+TP_2+\cdots+TP_n}{TP_1+\cdots+TP_n+FN_1+\cdots+FN_n},$$

$$MicroF1 = \frac{2\times MicroPrecision\times MicroRecall}{MicroPrecision+MicroRecall}, \tag{5}$$

Макро F1-мера – это средняя оценка F1-меры для каждого семантического типа (класса). При этом все классы являются эквивалентными, то есть не учитывается дисбаланс классов. Макро F1-мера рассчитывается по формуле:

$$MacroF1 = \frac{\sum_{i=1}^{N} F1_i}{N}, \tag{6}$$

где N – число семантических типов (классов); $F1_i$ – F1-мера для i-семантического типа.

Взвешенная F1-мера рассчитывается для каждого класса и затем суммируется как средневзвешенное значение с учетом количества записей для каждого класса. Данная метрика в отличие от микро F1-меры учитывает дисбаланс классов. Взвешенная F1-мера рассчитывается по формуле:

$$WeightedF1 = \sum_{i=1}^{C}[w_i \times F1_i], w_i = \frac{n_i}{N}, \tag{7}$$

где C – число семантических типов (классов); n_i – количество экземпляров в i-классе; N – общее количество экземпляров; $F1_i$ – F1-мера для i-семантического типа.

4.4 Результаты и обсуждение

Результаты экспериментальной оценки представлены в табл. 3. При этом осуществлено сравнение производительности предлагаемого подхода с некоторыми аналогами, которые выбраны в качестве базовых решений (*baseline*).

Во-первых, выбрана предварительно обученная языковая модель RuBERT [24], которая специализируется на обработке русского языка. При этом применена одна из техник трансферного обучения (*transfer learning*), при которой в процессе обучения веса слоев кодировщика остаются неизменными. Таким образом, во время дообучения RuBERT на наборе данных RWT-RuTaBERT изменялись только параметры слоя, предназначенного для классификации данных.

Во-вторых, выбран современный фреймворк Doduo [8], который является лидером решения задачи семантического аннотирования столбцов и связей между ними. Таким образом, в данном случае также применялась техника трансферного обучения с заморозкой слоев кодировщика и дообучением только последнего линейного слоя классификатора (*Doduo*). Помимо этого, было рассмотрено альтернативное базовое решение (*Translated Doduo*), основанное на переводе русскоязычного набора табличных данных на английский язык и его применения для англоязычной модели Doduo. Таким образом, семантические типы RWT-RuTaBERT были отображены в английские типы набора данных VizNet-Sato (например,

«*год*»: «*year*», «*место*»: «*location*», «*альбом*»: «*album*» и так далее). При отображении семантических типов, 16 типов из VizNet-Sato были исключены, в связи с невозможностью их отображения («*affiliate*», «*birthPlace*», «*brand*», «*collection*», «*command*», «*currency*», «*family*», «*fileSize*», «*gender*», «*grades*», «*isbn*», «*jockey*», «*plays*», «*sales*», «*species*», «*teamName*».). В свою очередь из набора RWT-RuTaBERT 5 семантических типов не получилось отобразить ни в один тип VizNet-Sato («*площадь*», «*звук*», «*отношение*», «*ввп*», «*лес*»). Сами столбцы из тестового набора RWT-RuTaBERT были переведены на английский язык с помощью машинного перевода. Кроме того, была осуществлена полноценная тонкая настройка мультиязычной модели BERT согласно подходу Doduo на наборе данных RWT-RuTaBERT (*Fine-tuned Doduo*).

В-третьих, взят оригинальный подход RuTaBERT из работы [9].

Табл. 3. Результаты экспериментальной оценки на наборе данных RWT-RuTaBERT.
Table 3. The results of the experimental evaluation on the RWT-RuTaBERT dataset.

Подход	**micro F1**	**macro F1**	**weighted F1**
Doduo	0.140	0.043	—
Translated Doduo	0.364	0.127	—
RuBERT	0.612	0.417	0.592
Fine-tuned Doduo	0.962	0.891	0.962
RuTaBERT	0.964	0.904	0.963
Предлагаемый подход (bs32)	0.969	0.910	0.969
Предлагаемый подход (bs256)	**0.974**	**0.924**	**0.974**

Полученная оценка показала, что предлагаемый подход в обеих конфигурациях обучения (размер пакета данных 32 и 256) превзошел все базовые решения. В частности, эксперимент продемонстрировал, что модель RuBERT хоть и ориентирована на обработку русского языка, но не направлена напрямую на решение табличных задач, которые оказались сложны для этой модели. Таким образом, существующие русскоязычные модели не могут быть эффективно применены к решению задачи семантического аннотирования столбцов.

Модель Doduo обученная с применением техники трансферного обучения показала достаточно низкие результаты оценки. Это связано с тем, что модель обучалась на табличных данных, представленных только на английском языке. В частности, в токенизаторе этой модели практически отсутствуют русскоязычные токены. Вследствие чего можно прийти к выводу, что невозможно взять модель, обученную на английском языке, и использовать ее на некотором другом языке, в данном случае русском. Для этого необходимо менять базовый кодировщик, способный различать этот самый язык.

При этом тонко настроенный мультиязычный кодировщик фреймворка Doduo и подход RuTaBERT показали почти сопоставимые результаты по метрикам оценки. Однако можно заметить, что использование предварительно обученного кодировщика таблиц на основе контрастного обучения положительно влияет на результат. С использованием меньшей модели при тех же настройках получилось достичь точно таких же результатов, как у классической модели RuTaBERT и тонко настроенной Doduo. При этом модель потребляет примерно в 3 раза меньше видеопамяти в процессе обучения, менее 10 гигабайт (при одинаковом для всех трех моделей размере пакета данных равным 32), что позволяет запускать обучение на стационарном домашнем компьютере. Кроме того, при использовании

большего размера пакета данных (256, в отличие от 32), получилось достичь выигрыш в 1.5% относительно классической модели RuTaBERT и почти 3% по сравнению с тонко настроенной Doduo. Полученные экспериментальные результаты указывают на потенциал нашего подхода для семантической аннотации русскоязычных таблиц.

Для дальнейшей оценки эффективности предлагаемого подхода, был проведен статистический анализ по трем аспектам:

1) Группировка по типам данных: Все столбцы из собранных таблиц были классифицированы на 5 основных групп: «*Дата*», «*Число*», «*Ссылка*», «*Короткий текст*» и «*Длинный текст*». Столбцы с типом «*Дата*» включают даты, годы или время в различных форматах. Столбцы типа «*Число*» содержат только числа, например, результаты измерений длины, веса или возраста. Столбцы с типом «*Ссылка*» включают различные виды ссылок, включая URL-адреса. Текстовые столбцы были разделены на «*Короткий текст*» (значение ячейки содержит менее четырех лексем) и «*Длинный текст*» (значение ячейки содержит четыре и более лексем). Также был выделен отдельный тип данных «*Персона*», который учитывает распространенность таких семантических типов как «*работодатель*», «*сценарист*», «*спортсмен*», «*футболист*» и так далее. В табл. 4 представлены результаты микро F1-меры для каждой группы типов данных.

Табл. 4. Результаты экспериментальной оценки по микро F1 для выделенных типов данных.
Table 4. The results of the experimental evaluation of micro F1 for selected data types.

Тип данных	RuTaBERT	Предлагаемый подход
Дата	0.941	**0.948**
Длинный текст	**0.885**	0.858
Число	0.749	**0.760**
Персона	0.692	**0.716**
Короткий текст	0.926	**0.932**
Ссылка	**0.699**	0.611

2) Сходимость модели: Для оценки сходимости модели были проведены эксперименты для отдельных контрольных точек (*checkpoints*) обученной модели (*bs32*) в сравнение с моделью RuTaBERT. При этом брались две контрольных точки моделей, обученных в течение 10 и 30 эпох. Результаты представлены в табл. 5. В скобках представлено значение прироста оценки относительно первой контрольной точки.

Можно заметить, что модель, обученная по предлагаемому подходу, сходится быстрее, чем модель RuTaBERT, при этом имеет на 1-3% выше производительность на тестовой выборке по метрике F1. Данная особенность позволяет использовать меньшее количество эпох на этапе обучения, получая при этом сравнимую или даже превосходящую производительность в сравнении с моделью RuTaBERT.

3) Редкие семантические типы: Оценка производительности также проводилась для 15 наименее встречаемых семантических типов. Для сравнения были использованы контрольные точки обученной модели (bs32) и RuTaBERT, достигшие наивысшего результата на обучающем наборе данных по метрике макро F1. Результаты представлены в табл. 6. Результаты показывают, что благодаря полученным устойчивым табличным представлениям, предлагаемый подход значительно превосходит существующее современное русскоязычное решение, в частности, RuTaBERT по производительности, в особенности для редко встречающихся семантических типов.

Табл. 5. Результаты экспериментальной оценки сходимости моделей.
Table 5. The results of an experimental assessment of the convergence of the models.

Модель	micro F1	macro F1	weighted F1
RuTaBERT (10 эпох)	0.952	0.856	0.952
Предлагаемый подход (10 эпох)	0.966	0.888	0.966
RuTaBERT (30 эпох)	0.964 (+0.012)	0.904 (+0.048)	0.963 (+0.011)
Предлагаемый подход (30 эпох)	0.969 (+0.003)	0.910 (+0.022)	0.969 (+0.003)

Табл. 6. Результаты экспериментальной оценки по микро F1 для 15 редко встречающихся семантических типов.
Table 6. The results of an experimental evaluation of micro F1 for 15 rarely occurring semantic types.

Семантический тип	Кол-во вхождений (в тестовой выборке)	RuTaBERT	Предлагаемый подход
камера	102 (4)	0.250	0.75
работодатель	101 (10)	0.899	1.000
устройство	101 (8)	0.625	0.875
животное	93 (7)	0.857	1.000
журнал	93 (9)	0.440	0.440
континент	92 (8)	0.625	0.750
роман	89 (11)	0.818	0.909
закон	89 (9)	1.000	1.000
борец	88 (5)	0.400	0.600
колледж	87 (5)	0.000	0.200
музей	86 (4)	0.500	0.750
фирма	85 (6)	0.333	0.333
префектура	83 (10)	0.600	0.699
дорога	83 (6)	0.500	0.666
цитата	76 (7)	0.857	1.000

5. Заключение

В работе предложен подход семантического аннотирования столбцов русскоязычных таблиц, основанный на контрастном обучении. Подход реализован в форме инструментального средства. Исходный код [25] и обученная модель [26] опубликованы в открытом доступе. Экспериментальные результаты демонстрируют, что подход устраняет зависимость от

больших объемов размеченных данных за счет самообучения на неразмеченных таблицах. При этом он превосходит существующие базовые решения (Doduo и RuTaBERT) по метрикам оценки, особенно для редких семантических типов. Подход также обеспечивает вычислительную эффективность за счет использования дистиллированной модели, что снижает требования к памяти на 60% по сравнению с аналогами.

Полученные результаты экспериментальной оценки показали эффективность предлагаемого решения. В будущем, в рамках исследовательского проекта с Институтом системного программирования имени В.П. Иванникова Российской академии наук (ИСП РАН), планируется интегрировать эти результаты в виде специального обработчика таблиц, входящего в состав платформы Talisman [27]. Также планируется расширение подхода на таблицы с горизонтальной и матричной компоновкой. Дополнительно будет исследоваться вопрос использования новых аугментаций для улучшения устойчивости табличных представлений.

В целом предлагаемый подход открывает возможности для создания универсальных систем семантической интерпретации таблиц, что актуально для задач интеграции различной структурированной и слабоструктурированной информации, и бизнес-аналитики.

Список литературы

[1]. Badaro G., Saeed M., Papotti P. Transformers for Tabular Data Representation: A Survey of Models and Applications. Transactions of the Association for Computational Linguistics, vol. 11, 2023, pp. 227-249. DOI: 10.1162/tacl_a_00544.

[2]. Liu J., Chabot Y., Troncy R., Huynh V.-P., Labbe T., Monnin P. From tabular data to knowledge graphs: A survey of semantic table interpretation tasks and methods. Journal of Web Semantics, vol. 76, 2023, 100761. DOI: 10.1016/j.websem.2022.100761.

[3]. Deng X., Sun H., Lees A., Wu Y., Yu C. TURL: Table Understanding through Representation Learning. Proc. the VLDB Endowment, vol. 14, no. 3, 2020, pp. 307-319. DOI: 10.14778/3430915.3430921.

[4]. Herzig J., Nowak P. K., Muller T., Piccinno F., Eisenschlos J. M. TaPas: Weakly Supervised Table Parsing via Pre-training. Proc. 58th Annual Meeting of the Association for Computational Linguistics, Online, 2020, pp. 4320-4333. DOI: 10.18653/v1/2020.acl-main.398.

[5]. Yin P., Neubig G., Yih W. TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data. Proc. the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8413-8426. DOI: 10.18653/v1/2020.acl-main.745.

[6]. Iida H., Thai D., Manjunatha V., Iyyer M. TABBIE: Pretrained Representations of Tabular Data. Proc. the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 3446-3456. DOI: 10.18653/v1/2021.naacl-main.270.

[7]. Wang Z., Dong H., Jia R., Li J., Fu Z., Han S., Zhang D. TUTA: Tree-based Transformers for Generally Structured Table Pre-training. Proc. the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD'21), 2021, pp. 1780-1790. DOI: 10.1145/3447548.3467434.

[8]. Suhara Y., Li J., Li Y. Annotating Columns with Pre-trained Language Models. Proc. the 2022 International Conference on Management of Data (SIGMOD'22), 2022, pp. 1493-1503. DOI: 10.1145/3514221.3517906.

[9]. Tobola K. V., Dorodnykh N. O. Semantic Annotation of Russian-Language Tables Based on a Pre-Trained Language Model. Proc. the 2024 Ivannikov Memorial Workshop (IVMEM), Velikiy Novgorod, Russian Federation, 2024, pp. 62-68. DOI: 10.1109/IVMEM63006.2024.10659709.

[10]. RWT-RuTaBERT, Available at: https://huggingface.co/datasets/sti-team/rwt-rutabert, accessed 06.05.2025.

[11]. Ji S., Pan S., Cambria E., Marttinen P., Yu P.S. A Survey on Knowledge Graphs: Representation, Acquisition and Applications. IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 2, 2021, pp. 494-514. DOI: 10.1109/TNNLS.2021.3070843.

[12]. Hulsebos M., Hu K., Bakker M., Zgraggen E., Satyanarayan A., Kraska T., Demiralp Ç., Hidalgo C. Sherlock: A Deep Learning Approach to Semantic Data Type Detection. Proc. the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'19), 2019, pp. 1500-1508. DOI: 10.1145/3292500.3330993.

[13]. Zhang D., Suhara Y., Li J., Hulsebos M., Demiralp C., Tan W.-C. Sato: Contextual semantic type detection in tables. Proc. the VLDB Endowment, vol. 13, no. 11, 2020, pp. 1835-1848. DOI: 10.14778/3407790.3407793.
[14]. Wang D., Shiralkar P., Lockard C., Huang B., Dong X. L., Jiang M. TCN: Table Convolutional Network for Web Table Interpretation. Proc. the Web Conference (WWW'21), Ljubljana, Slovenia, 2021, pp. 4020-4032. DOI: 10.1145/3442381.3450090.
[15]. Li P. , He Y., Yashar D., Cui W., Ge S., Zhang H., Fainman D. R., Zhang D., Chaudhuri S. Table-GPT: Table Fine-tuned GPT for Diverse Table Tasks. Proceedings of the ACM on Management of Data, vol. 2, no. 3, 2024, pp. 1-28. DOI: 10.1145/3654979.
[16]. Zhang T., Yue X., Li Y., Sun H. TableLlama: Towards Open Large Generalist Models for Tables. Proc. the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Mexico City, Mexico, 2024, pp. 6024-6044. DOI: 10.18653/v1/2024.naacl-long.335.
[17]. Korini K., Bizer C. Column Property Annotation Using Large Language Models. Proc. the Semantic Web: ESWC 2024 Satellite Events, Hersonissos, Crete, Greece, 2024, pp. 61-70. DOI: 10.1007/978-3-031-78952-6_6.
[18]. DBpedia, Available at: https://www.dbpedia.org/, accessed 06.05.2025.
[19]. Ru-Wiki-Tables-dataset, Available at: https://gitlab.com/unidata-labs/ru-wiki-tables-dataset, accessed 06.05.2025.
[20]. Chen T., Kornblith S., Norouzi M., Hinton G. A simple framework for contrastive learning of visual representations. Proc. the 37th International Conference on Machine Learning (ICML'20), Online, 2020, pp. 1597-1607. DOI: 10.5555/3524938.3525087.
[21]. Model Card for DistilBERT base multilingual (cased), Available at: https://huggingface.co/distilbert/distilbert-base-multilingual-cased, accessed 06.05.2025.
[22]. BERT multilingual base model (cased), Available at: https://huggingface.co/google-bert/bert-base-multilingual-cased, accessed 06.05.2025.
[23]. Miao Z., Wang J. Watchog: A Light-weight Contrastive Learning based Framework for Column Annotation. Proceedings of the ACM on Management of Data, vol. 1, no. 4, 2023, pp. 1-24. DOI: 10.1145/3626766.
[24]. Kuratov Y., Arkhipov M. Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language, 2019, arXiv:1905.07213.
[25]. CoLeM framework, Available at: https://github.com/YRL-AIDA/CoLeM, accessed 06.05.2025.
[26]. CoLeM base cased, Available at: https://huggingface.co/sti-team/colem-base-cased, accessed 06.05.2025.
[27]. Talisman, Available at: http://talisman.ispras.ru, accessed 06.05.2025.

Информация об авторах / Information about authors

Кирилл Владимирович ТОБОЛА – аспирант Института динамики систем и теории управления им. В.М. Матросова Сибирского отделения РАН (ИДСТУ СО РАН) с 2024 года. Сфера научных интересов: табличные представления, большие языковые модели для реляционных данных, извлечение данных из табличных источников.

Kirill Vladimirovich TOBOLA is a postgraduate student at the Matrosov Institute of System Dynamics and Control Theory named SB RAS (ISDCT SB RAS) since 2024. Research interests: table embedding, large language models for relational data, and data extraction from tabular sources.

Никита Олегович ДОРОДНЫХ – кандидат технических наук, старший научный сотрудник Института динамики систем и теории управления им. В.М. Матросова Сибирского отделения РАН (ИДСТУ СО РАН) с 2021 года. Сфера научных интересов: автоматизация создания интеллектуальных систем и баз знаний, получение знаний на основе преобразования концептуальных моделей и электронных таблиц.

Nikita Olegovych DORODNYKH is PhD, senior associate researcher at the Matrosov Institute of System Dynamics and Control Theory named SB RAS (ISDCT SB RAS) since 2021. Research interests: computer-aided development of intelligent systems and knowledge bases, knowledge acquisition based on the transformation of conceptual models and tables.