DOI: 10.15514/ISPRAS-2025-37(2)-16

Поиск именованных сущностей в инструкциях по медицинскому применению лекарственных средств с использованием глубокого обучения и методов обработки естественного языка

Ю.П. Титов, ORCID: 0000-0002-9093-6755 <kalengul@mail.ru>
Н.В. Кильмишкин, ORCID: 0009-0007-4732-3260 <kilmiskinn@gmail.com>
Д.Д. Кубраков, ORCID: 0000-0003-2986-9343 <kubrakoff.dmitry@yandex.ru>
П.М. Иванова, ORCID: 0009-0005-4579-4603 <polianna_654@mail.ru>

Российский экономический университет имени Г.В. Плеханова,
Россия, 115054, г. Москва, Стремянный пер., д. 36.

Аннотация. В рамках работы создан специализированный словарь для поиска ключевых терминов в текстах медицинских инструкций, с использованием данных из глобальной базы данных VigiAccess, классификации МКБ-10 и ресурса rlsnet.ru. Текстовый корпус был предварительно очищен и приведён к единому формату для улучшения качества обучения модели. В дальнейшем планируется использовать источник grls.rosminzdrav.ru, как более авторитетный и полный, для получения информации о зарегистрированных лекарственных средствах. Для автоматизации аннотации данных разработан алгоритм, который выполняет поиск и разметку терминов из словаря в формате BIO (Begin, Inside, Outside), обеспечивая структурированную разметку для обучения моделей. Модель на основе глубоких нейронных сетей продемонстрировала высокую эффективность в распознавании именованных сущностей благодаря учёту контекстных зависимостей. Построение семантического графа лекарственных средств осуществлялось с помощью алгоритмов нахождения связей между именованными сущностями. Однако автоматическое выявление более глубоких связей между узлами графа затруднено и требует ручной доразметки данных для учёта сложных грамматических структур, что позволит улучшить анализ взаимодействий в текстах медицинских инструкций.

Ключевые слова: машинное обучение; глубокое обучение; нейронные сети; обработка естественного языка NLP; инструкции к лекарственным средствам; семантический граф.

Для цитирования: Титов Ю.П., Кильмишкин Н.В., Кубраков Д.Д., Иванова П.М. Поиск именованных сущностей в инструкциях по медицинскому применению лекарственных средств с использованием глубокого обучения и методов обработки естественного языка. Труды ИСП РАН, том 37, вып. 2, 2025 **г.**, стр. 217–236. DOI: 10.15514/ISPRAS–2025–37(2)–16.

Благодарности: Исследование выполнено при финансовой поддержке РНФ № 23-75-30012.

Use of Deep Learning and Natural Language Processing Techniques for Searching Named Entities in the Medical Instructions for Use of Drugs

Y.P. Titov, ORCID: 0000-0002-9093-6755 <kalengul@mail.ru>
N.V. Kilmishkin, ORCID: 0009-0007-4732-3260 <kilmiskinn@gmail.com>
D.D. Kubrakov, ORCID: 0000-0003-2986-9343 <kubrakoff.dmitry@yandex.ru>
P.M. Ivanova, ORCID: 0009-0005-4579-4603 <polianna_654@mail.ru>
Plekhanov Russian University of Economics,
36, Stremyanny Lane, Moscow, 115054, Russia.

Abstract. As part of the work, a specialized dictionary has been created to search for key terms in the texts of medical instructions, using data from VigiAccess, ICD-10 and rlsnet.ru. The text corpus was previously cleaned and brought to a single format to improve the quality of model training. In the future, it is planned to use the source grls.rosminzdrav.ru, as more authoritative and complete, for information about registered medicines. To automate data annotation, an algorithm has been developed that searches and marks terms from the dictionary in BIO (Begin, Inside, Outside) format, providing structured markup for model training. The model based on deep neural networks has demonstrated high efficiency in recognizing named entities by taking into account contextual dependencies. The semantic graph of medicines was constructed using algorithms for finding connections between named entities. However, automatic identification of deeper connections between graph nodes is difficult and requires additional data markup to account for complex grammatical structures, which will improve the analysis of interactions in the texts of medical instructions.

Keywords: machine learning; deep learning; neural networks; natural language processing; medical drug instructions; semantic graph.

For citation: Titov Y.P., Kilmishkin N.V., Kubrakov D.D., Ivanova P.M. Use of deep learning and natural language processing techniques for searching named entities in the medical instructions for use of drugs. Trudy ISP RAN/Proc. ISP RAS, vol. 37, issue 2, 2025. pp. 217-236 (in Russian). DOI: 10.15514/ISPRAS-2025-37(2)- 16.

Acknowledgements. The study was carried out with the financial support of the Russian Science Foundation No. 23-75-30012.

1. Введение

Анализ медицинских инструкций к лекарственным средствам представляет собой сложную задачу, требующую обработки значительных объемов текстовой информации, содержащей данные о фармакодинамике, фармакокинетике, показаниях, противопоказаниях, дозировке, побочных эффектах и других аспектах, критически важных для безопасности и эффективности применения лекарств. Объем и сложность таких документов делают ручной анализ трудоемким и подверженным ошибкам, поэтому современные методы обработки естественного языка (NLP) и глубокого обучения становятся важным инструментом автоматизации этого процесса, позволяя выделять именованные сущности (Named Entity, NE) и связывать их внутри текста.

Методы NLP обеспечивают преобразование текстов медицинских инструкций в машиночитаемый формат, что открывает возможности для последующего анализа и построения семантических графов, отображающих взаимосвязи между словами и словосочетаниями, такими как механизмы действия и побочные эффекты. Такие графы позволяют выявить ключевые связи и узлы взаимодействия, упрощая интерпретацию информации.

Распознавание именованных сущностей (Named Entity Recognition, NER) [1] является сложной задачей NLP, решающей проблему автоматического выделения категорий объектов

в тексте. NER играет важную роль в структурировании текстовой информации, выделяя важные медицинские сущности и облегчая их дальнейший анализ [2]. Методы NER, использующие глубокое обучение, демонстрируют высокую точность по сравнению с традиционными методами, основанными на правилах и словарях [3]. Современные трансформерные модели, обученные на больших объемах данных, учитывают контекст и специфику медицинского языка, где одно слово может приобретать различные значения в зависимости от контекста.

Выделенные именованные сущности могут использоваться для построения семантических графов, отображающих свойства и взаимодействие различных лекарственных средств, их воздействие на организм и потенциальные взаимодействия между лекарственными средствами. Такой граф способен стать важным элементом в системах поддержки принятия решений для медицинских специалистов, оказывая помощь при назначении лечения и подборе медикаментозных средств.

Исследования, проведенные на основе обычных методов машинного обучения и ансамблевом подходе, используют различные наборы данных для обучения или различные алгоритмы обучения для создания базовых классификаторов. Различные алгоритмы машинного обучения, такие как SVM и CRF, использовались для Clinical-NER [4]. Керетна (Keretna) и др. [5] представили гибридный подход, использующий подходы на основе правил и словарей для идентификации названий лекарств в неструктурированных и неформальных текстах, оценена на наборе данных для испытания лекарств i2b2 2009 и сообщила о 66,97% f-score. Словари и подходы на основе правил широко использовались для извлечения клинических сущностей в клинических информационных системах, таких как MedLEE, разработанная Фридманом и др. [6], MetaMap, разработанная Арнсоном и Лангом [7], и cTAKES, разработанная Савовой и др. [8]. Гурулингаппа (Gurulingappa) и др. [9] обучили CRF на текстовых признаках, улучшенных с помощью вывода системы NER на основе правил. Они оценили свою работу с использованием набора данных медицинских задач i2b2/VA 2010 и сообщили о значении метрики f-score, равном 81,2%. Халгрим (Halgrim) и др., [10] разработали гибридный подход, который включал CRF и подход на основе правил для клинического NER. Чжан и Эльхадад (Zhang и Elhadad) [11] разработали неконтролируемый подход для извлечения клинических сущностей из свободного текста. Они использовали обратную частоту документов в качестве основы для фильтрации потенциальных клинических именованных сущностей. Экбал и Саха (Ekbal и Saha) [12] использовали подход с накопленным ансамблем для извлечения биомедицинских именованных сущностей. Шаширеха (Shashirekha) и Найел (Nayel) [13] изучали производительность биомедицинского NER с использованием различных моделей. Керетна (Keretna) и др. [14] представили метод повышения клинического NER путем расширения модели IOBES и ввели новый тег для решения проблемы неоднозначности. Они оценили предложенную методику на наборе данных медицинских задач i2b2/VA 2010. Ву (Wu) и др. [15] обучили глубокую модель нейронной сети для извлечения клинических сущностей из китайских текстов. Последние годы серьёзного успех в решении задачи NER были получены с помощью методов на основе искусственных нейронных сетей, в частности, на основе архитектуры трансформеров, о чём говориться в работах Джорджи Дж.М. и Бадер Г.Д. (Giorgi J.M. и Bader G.D) [16], Хабиби (Habibi) и др. [17], Ван (Wang) и др. [18], Юн (Yoon) и др. [19].

Предложенные алгоритмы не учитывают специфику медицинских документов, инструкций к лекарственным средствам (ЛС), поэтому для применения данных методов требуется интеллектуальная предобработка текстов. Термины в текстах инструкций к лекарственным средствам часто состоят из нескольких слов, которые представляют различный смысл как по отдельности, так и в совместном виде. Так как инструкции пишутся для лекарственных средств, которые могут быть зарегистрированы под различными торговыми названиями, торговым маркам, то для одного лекарственного средства (рассматривается международное

непатентованное название, МНН) может быть множество инструкций, написанных в разное время различными фирмами, что, в совокупности с отсутствием стандартов написания, затрудняют исследование текстов представленными методами статистического анализа слов.

2. Постановка задачи

Задача, рассматриваемая в данной работе, заключается в автоматизированной обработке текстовых данных инструкций к медицинским лекарственным средствам для построения графовой семантической модели. Предложенная модель должна определять риски возникновения побочных эффектов при одновременном применении нескольких (более 3-х) лекарственных средств – при полифармакотерапии.

На первом этапе необходимо осуществить чтение текста из файлов, содержащих инструкции, и выполнить его предварительную обработку, включая форматирование, разбиение на главы и отдельные предложения для дальнейшего анализа. Далее создается словарь именованных сущностей и алгоритм для автоматической разметки данных, которые будут использованы в качестве тренировочного набора для последующего обучения модели глубокого обучения для задачи NER. На следующем этапе применяется модель глубокого обучения, обученная на соответствующей области медицинских текстов, для автоматического выделения именованных сущностей. Этот процесс направлен на структурирование неупорядоченной текстовой информации и получение набора значимых сущностей, связанных с медицинским контекстом. После извлечения сущностей проводится анализ их взаимосвязей. Данные взаимосвязи используются для построения семантического графа, который представляет собой структурированное представление информации, извлеченной из текста.

3. Модели и методы

3.1 Создание словаря именованных сущностей и подготовка корпуса текста

В контексте NER словарь может включать в себя различные категории именованных сущностей, такие как побочные действия, названия лекарственных средств и другие слова, которые стоит учитывать при построении семантического графа. Для формирования специализированного словаря были использованы источники:

- VigiAccess – глобальная база данных Всемирной организации здравоохранения, содержащая информацию о возможных побочных эффектах лекарственных средств [20]. Информация содержится в табличном виде, где для каждого лекарственного средства можно получить количество случаев выявления побочного эффекта;
- МКБ-10 – международная классификация болезней десятого пересмотра, используемая для систематизации заболеваний и патологических состояний [21];
- Тексты инструкций к лекарственным средствам, взятые с rlsnet.ru [22];

Процесс отбора лексических единиц из вышеупомянутых источников был организован в полуавтоматическом режиме. Исходные тексты подвергались разбиению на отдельные слова, после чего происходил отбор лексем на основании морфологических признаков, таких как префиксы и суффиксы, а также с применением алгоритмов для выделения аббревиатур: алгоритм в тексте анализирует каждое слово, считывая количество заглавных букв. Если слово содержит две или более заглавных буквы, оно классифицируется как аббревиатура. Этот этап позволил автоматизировать часть рутинной работы по выделению терминов и облегчить дальнейшую обработку данных.

Данные из базы VigiAccess и классификации МКБ-10 потребовали дополнительной обработки и преобразования для их структурирования и приведения к формату, который позволяет взаимодействовать с ними в рамках построения словаря. Такие преобразования

включали, в частности, нормализацию и унификацию текстовых данных, устранение дублированных элементов, а также разработку специализированных алгоритмов для выделения и категоризации терминов.

Программа для подготовки корпуса обрабатывает исходный текст, выполняя его очистку, форматирование и структурирование. Этапы включают чтение текста, удаление избыточной информации (например, содержания в скобках), упоминание таблиц, сегментацию на предложения и запись в файл, где каждое предложение занимает отдельную строку. Использование библиотеки «razdel» обеспечивает высокую точность сегментации текста на предложения для русского языка, что улучшает качество данных для задач классификации [23].

3.2. Токенизация слов и объединение в нотацию BIO

После того как был подготовлен текстовый корпус и сформирован словарь, слова помечаются тегами. Для этого для каждой строки выполняется предобработка предложений и последующая классификация слов с использованием библиотеки «spaCy» [24]. Алгоритм разметки слов включает этапы:

1) токенизация и определение частей речи – используется библиотека «spaCy» для разделения строки на отдельные токены. Для каждого токена сохраняется его текстовое представление и часть речи;
2) нормализация токенов – приведение их к единой форме (например, к леммам). Это позволяет исключить вариативность, связанную с различными грамматическими формами одного и того же слова;
3) тегирование слов – для каждого нормализованного слова вызывается функция, которая на основе самого слова и его части речи (полученной в предыдущем шаге) назначает соответствующий тег, слова могут быть помечены как существительные, прилагательные, глаголы или другие части речи в зависимости от их морфологической характеристики и контекста;
4) учёт отрицания – для корректной обработки отрицаний в предложении вызывается функция, которая изменяет тег отрицательной частицы "не", на тот, который стоит после этой частицы;
5) нахождение зависимых слов для поиска словосочетаний – функция выполняет поиск зависимостей между словами в предложении. Главные слова (например, глаголы или существительные) могут иметь зависимые слова (например, прилагательные, наречия), и этот этап позволяет связать их в одну серию для анализа синтаксических связей.

Структурная схема алгоритма представлена на рис. 1.

Задача извлечения именованных сущностей NER заключается в автоматическом выявлении в тексте сегментов, представляющих собой имена собственные или другие значимые объекты, с последующей их категоризацией по типу сущности. Основной целью NER является улучшение качества текстовой классификации за счёт уменьшения разреженности данных. Это позволяет моделям работать более эффективно, избегая ошибок, связанных с редкостью определённых токенов. Ключевым стандартом для разметки текста в задачах NER является пословная разметка последовательности с использованием BIO-нотации. Она представляет собой схему для маркировки слов в предложении, которая помогает структурировать текст и выделить границы сущностей:

- B (beginning) — помечает начало именованной сущности, например, первый токен имени, компании или другого объекта;
- I (inside) — используется для пометки последующих токенов, которые являются частью той же сущности, что и токен с меткой B;

- O (outside) — обозначает слова, не относящиеся к именованным сущностям, таким образом маркируя всю остальную часть текста.

BIO-разметка помогает формализовать процесс выделения сущностей, определяя границы между ними и уточняя их роль в тексте (табл. 1).

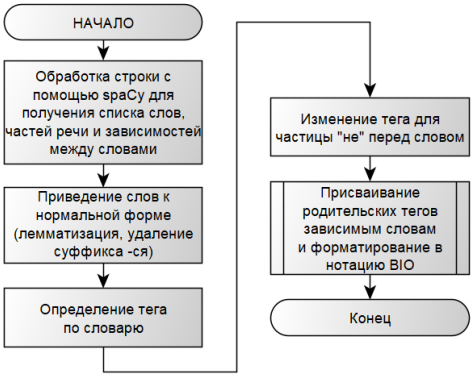


Рис. 1. Структурная схема алгоритма разметки текста словами из словаря.
Fig. 1. Block diagram of the text markup algorithm with words from the dictionary.

Табл. 1. Пример разметки в нотации BIO текстов инструкций к лекарственным средствам.
Table 1. Example of marking in BIO notation of texts of instructions for medicinal products.

Слово	Разметка BIO
Аценокумарол	B-prepar
усиливает	B-glag
эффект	B-noun
пероральных	I-noun
гипогликемических	I-noun
средств	I-noun
,	O
токсическое	B-noun
действие	I-noun
фенитоина	I-noun
,	O
ульцерогенное	B-noun
действие	I-noun
глюкокортикоидов	B-prepar
.	O

В качестве анализа предложений были рассмотрены такие библиотеки как: spaCy, MyStem и UDPipe. MyStem не поддерживает построение деревьев синтаксических зависимостей. Она предназначена для морфологического анализа и лемматизации текста. Сравнительный анализ разметки текста (рис. 2) с использованием инструментов udpipe и spaCy показал, что второй инструмент в большинстве случаев обеспечивает более точные и логичные зависимости. В частности, в udpipe наблюдаются неточности в установке зависимостей, такие как использование связи parataxis (параллельный синтаксис) для слова "ЦОГ", что не всегда соответствует смыслу, поскольку в контексте это слово является уточняющим приложением. В отличие от этого, spaCy правильно интерпретирует зависимость как conj (сочинение). Также стоит отметить, что в udpipe лемматизация, хотя и в целом корректная, содержит некоторые несоответствия, например, лемматизация "ЦОГ-1" и "ЦОГ-2" в "Цог", что не отражает оригинальное написание. В spaCy лемматизация также не всегда идеальна, как в случае с леммой для слова "Ингибирует", но в целом она более точна. Кроме того, в udpipe часто наблюдаются ошибки в разметке пунктуации, особенно в сложных конструкциях, где

она неправильно связывается с зависимыми словами. В spaCy пунктуация правильно связана с предыдущими элементами, что улучшает понимание синтаксической структуры предложения. В целом, несмотря на наличие некоторых ошибок в лемматизации и разметке, spaCy демонстрирует более высокую точность, особенно при работе с синтаксическими зависимостями и сложными структурами предложений, что делает его более подходящим инструментом для анализа текста в рамках данной задачи.

a)

id	form	lemma	upostag	head	deprel
1	Ингибирует	ингибировать	VERB	0	root
2	циклооксигеназу	циклооксигеназа	NOUN	3	obj
3	(	(	PUNCT	4	punct
4	ЦОГ-1	ЦОГ-1	PROPN	3	appos
5	и	и	CCONJ	3	cc
6	ЦОГ-2	ЦОГ-2	PROPN	3	appos
7	)	)	PUNCT	3	punct
8	и	и	CCONJ	9	cc
9	необратимо	необратимо	ADV	10	advmod
10	тормозит	тормозить	VERB	0	root
11	циклооксигеназный	циклооксигеназный	ADJ	12	amod
12	путь	путь	NOUN	10	obj
13	метаболизма	метаболизм	NOUN	12	nmod
14	арахидоновой	арахидоновый	ADJ	15	amod
15	кислоты	кислота	NOUN	13	nmod
16	.	.	PUNCT	10	punct

b)

id	form	lemma	upostag	head	deprel
1	Ингибирует	Ингибировать	VERB	0	root
2	циклооксигеназу	циклооксигеназ	NOUN	1	obj
3	(	(	PUNCT	4	punct
4	ЦОГ	Цог	PROPN	2	parataxis
5	-	-	PUNCT	4	punct
6	1	1	NUM	4	nummod
7	и	и	CCONJ	8	cc
8	ЦОГ	Цог	PROPN	2	conj
9	-	-	PUNCT	8	punct
10	2	2	NUM	8	nummod
11	)	)	PUNCT	10	punct
12	и	и	CCONJ	14	cc
13	необратимо	необратимо	ADV	14	advmod
14	тормозит	тормозить	VERB	1	conj
15	циклооксигеназный	циклооксигеназный	ADJ	16	amod
16	путь	путь	NOUN	14	nsubj
17	метаболизма	метаболизм	NOUN	16	nmod
18	арахидоновой	арахидоновый	ADJ	19	amod
19	кислоты	кислота	NOUN	17	nmod
20	.	.	PUNCT	19	punct

c)

id	form	lemma	upostag	head	deprel
1	Ингибирует	ингибирует	VERB	1	ROOT
2	циклооксигеназу	циклооксигеназу	NOUN	1	obj
3	(	(	PUNCT	4	punct
4	ЦОГ-1	цог-1	PROPN	2	parataxis
5	и	и	CCONJ	6	cc
6	ЦОГ-2	цог-2	PROPN	4	conj
7	)	)	PUNCT	4	punct
8	и	и	CCONJ	10	cc
9	необратимо	необратимый	ADV	10	advmod
10	тормозит	тормозить	VERB	1	conj
11	циклооксигеназный	циклооксигеназный	ADJ	12	amod
12	путь	путь	NOUN	10	obj
13	метаболизма	метаболизм	NOUN	12	nmod
14	арахидоновой	арахидоновой	ADJ	15	amod
15	кислоты	кислота	NOUN	13	nmod
16	.	.	PUNCT	1	punct

Рис. 2. Анализ синтаксических зависимостей:
a – разметка вручную; b — разметка с помощью UDPipe; c — разметка с помощью spaCy.
Fig. 2. Syntactic dependency analysis:
a – manual markup; b – markup using UDPipe; c – markup using spaCy.

Библиотека «spaCy» разбирает предложения с помощью токенизации и строит дерево зависимостей, где каждая связь показывает грамматическое отношение между словами. Например, подлежащее (субъект) связано с глаголом (предикатом), а дополнение – с глаголом. Зависимости обозначаются по классификации «Universal Dependencies».

Пример для предложения «*Амлодипин ингибирует трансмембранный переход ионов кальция в кардиомиоциты*» приведен на рис. 3.

Структурная схема алгоритма нахождения зависимых слов от главного показана на рис. 4.

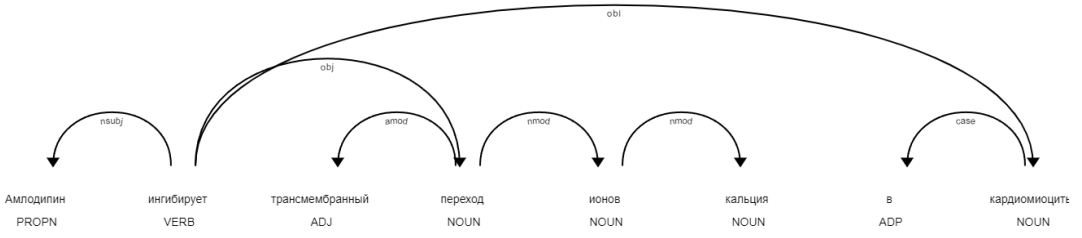
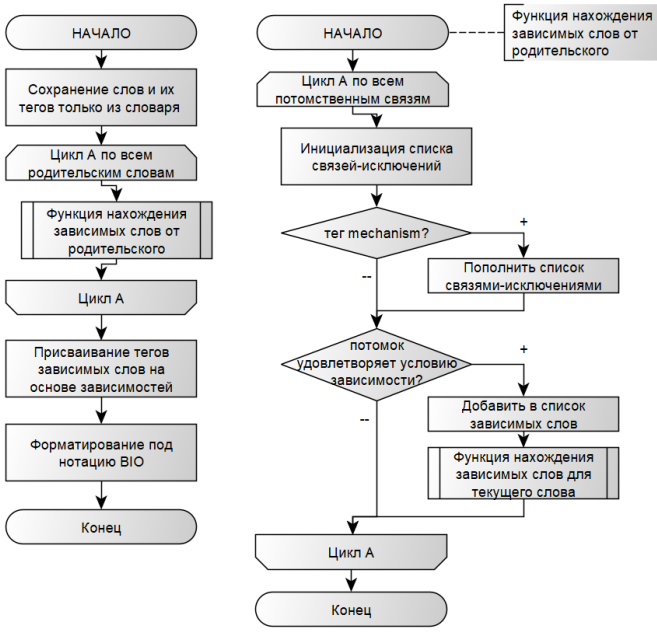


Рис. 3. Дерево зависимостей слов в предложении.
Fig. 3. The dependency tree of words in a sentence.



a) *b)*

Рис. 4. Структурная схема алгоритма нахождения зависимых слов от главного:
a – подготовка и изменение тегов зависимых слов,
b — рекурсивная функция для нахождения зависимых слов от родительского.
Fig. 4. A block diagram of the algorithm for finding dependent words from the main one:
a – preparation and modification of tags of dependent words;
b – a recursive function for finding dependent words from the parent.

В алгоритме, показанном на структурной схеме рис. 4a, сначала происходит фильтрация токенов и тегов для извлечения только тех слов, которые имеют специфичные теги. Затем осуществляется рекурсивный поиск зависимых слов у этих тегов (структурная схема на рис. 4b). Эта функция предназначена для извлечения зависимых токенов из данного слова с учетом определенных исключений в типах зависимостей.

В процессе выполнения алгоритм инициализирует пустой список, который будет содержать зависимые токены. Затем происходит итерация по всем дочерним токенам текущего слова. В зависимости от тега текущего слова, алгоритм задает список исключений, который, в случае тега 'mechanism', включает зависимости 'nmod' и 'nsubj' по классификации «Universal Dependencies». Далее осуществляется проверка условий для добавления дочернего токена в список зависимых. Если уровень равен 1 и зависимость токена имеет тип 'amod' или 'nmod', при этом не находится в списке исключений, или если уровень больше 1 и зависимость принадлежит к одному из типов 'amod', 'nmod', 'cc', 'nsubj', 'conj' или 'case', то дочерний токен добавляется в список зависимых токенов. В случае добавления токена в список, осуществляется рекурсивный вызов функции для получения зависимых токенов от текущего дочернего токена, при этом уровень увеличивается на единицу. В завершение рекурсивная функция возвращает полный список зависимых токенов, собранный на основе заданных условий и исключений.

На следующем этапе алгоритм модифицирует теги зависимых токенов в зависимости от тегов головного токена. Для каждого головного слова, представленного в словаре зависимостей, осуществляется обход его зависимых токенов, и тег каждого зависимого токена заменяется на тег головного. После обновления тегов всех токенов выполняется их преобразование в

последовательность BIO-тегов, где используется функция для маркировки каждого токена в формате "Begin", "Inside" или "Outside" в зависимости от его роли в последовательностях сущностей.

Результатом работы алгоритма является последовательность BIO-тегов, которая отражает начало и продолжительность именованных сущностей в тексте. Использование BIO-нотации даёт возможность точно выделять в тексте именованные сущности, не смешивая их с другими элементами, а также правильно обозначать связи между зависимыми словами. Это улучшает обобщающие способности модели и позволяет создавать более точные и детализированные модели для классификации текста и других задач обработки естественного языка.

3.3. Обучение модели для задачи NER

Для задачи NER применяются системы на основе правил (с использованием словарей терминов), методы машинного и глубокого обучения. Наилучшую предсказательную способность обеспечивают методы глубокого обучения с использованием искусственных нейронных сетей (ИНС), которые лучше улавливают сложные зависимости в данных, особенно при их большом объеме. Среди моделей ИНС часто используют свёрточные сети (CNN), рекуррентные сети (RNN) и трансформеры. Для решения задачи NER рассматривались модели BERT, ruBERT, RuBioBERT, ruT5 и ruT5-multitask, загруженные из huggingface. BERT, использующий только механизм энкодера трансформера, отличается двунаправленным вниманием и хорошо подходит для задач классификации и извлечения информации (NLU). Модель ruBERT обучена на общих русскоязычных текстах, RuBioBERT – на медицинских статьях, что делает ее более подходящей для медицинского текста. Модели T5, такие как ruT5 и ruT5-multitask, решают широкий спектр задач NLP в формате «входной текст → выходной текст» и подходят для генеративных задач. Все четыре модели дообучались на инструкциях к лекарственным средствам, размеченных по нотации BIO, что позволяет применять их для распознавания именованных сущностей в текстах медицинских инструкций. Инструкции к лекарственным средствам были размечены по нотации BIO, таким образом, чтобы каждому предложению как последовательности слов соответствовала последовательность тег BIO, то есть получается формат «слово» – «тег BIO».

Для подбора оптимальных гиперпараметров модели был использован метод Grid Search. Полученные результаты визуализированы с помощью тепловой карты, что позволило наглядно оценить влияние различных комбинаций параметров на качество модели. Тепловая карта для гиперпараметров скорости обучения и количества эпох (рис. 5).

Обучение всех четырёх моделей проводилось с одинаковыми гиперпараметрами. Скорость обучения модели (learning rate) со значением 2e-5. Скорость обучения – величина, с которой модель обновляет свои веса на каждом шаге обучения. Контролирует, насколько сильно обновляются веса модели после каждой итерации. Слишком большое значение может привести к нестабильному обучению, тогда как слишком малое — к очень медленному обучению или застреванию в локальном минимуме.

Размер обучающего пакета (train batch size) пакета (батча) – 16 точек данных. Размер батча (пакета) для обучения – количество примеров, которые обрабатываются моделью за один шаг на одном устройстве. Больший размер батча может ускорить обучение и стабилизировать градиенты, но требует больше памяти. Размер батча нужно выбирать в зависимости от доступных ресурсов и особенностей задачи.

Размер проверяющего (валидирующего) (eval batch size) пакета – 16 точек данных. Он необходим для оценки модели. Больший батч для оценки позволяет быстрее пройтись по всем данным, однако также требует больше памяти.

Число эпох обучения (итераций обучения) (epochs) – 6 эпох. Одна эпоха — это полный проход по всему тренировочному набору данных. Большее количество эпох может улучшить

качество модели, но, если их слишком много, модель может переобучиться. Обычно с каждой эпохой модель всё лучше подстраивается под тренировочные данные, но улучшение метрик на валидационном наборе замедляется.

Коэффициент регуляризации (затухания весов) (weight decay) – 0.01. Коэффициент регуляризации (затухания весов), который применяется для контроля величины весов модели, чтобы избежать переобучения. Добавляет штраф за слишком большие значения весов, уменьшая их с каждым шагом. Это помогает сделать модель более обобщающей и устойчивой к шуму, особенно в случае сложных моделей или небольших наборов данных.

Обучение проводилось на корпусе, содержащем 11 801 предложение, из которых 15% были выделены в качестве выборки для оценки качества модели.

Набольшая эффективность обнаружена при числе эпох равном 6. Большее количество эпох ведёт к переобучению, из-за чего качество работы модели перестаёт расти и начинает падать. Для численной оценки качества работы моделей использовались специальные метрики: правильность (accuracy), точность (precision), полнота (recall), F-score (F1).

Эти метрики вычисляются как соотношение правильных и не правильных ответов модели. Для подсчёта ответов модели используется матрица ошибок (confusion matrix) [25]. Таким образом, ошибки классификации бывают двух видов: ложноотрицательные (False Negative, FN) и ложноположительные (False Positive, FP).

Правильность (accuracy) – доля правильных ответов модели среди всех предсказаний, метрика, которая характеризует качество модели, агрегированное по всем классам [26]. Ее использование полезно, когда классы для нас имеют одинаковое значение [27]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Точность (precision) – доля истинно-положительных ответов среди всех положительных ответов модели [28]:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

Полнота (recall) – доля истинно-положительных ответов среди всех правильных ответов:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

Стремление к увеличению точности не позволяет сводить все объекты в один класс, так как в этом случае возникает рост уровня ложноположительных решений. Метрика полноты демонстрирует способность алгоритма обнаруживать данный класс вообще, а метрика точности – способность отличать этот класс от других классов [29]. Точность и полнота не зависят (в отличие от правильности) от соотношения классов и потому применимы в условиях несбалансированных выборок.

a)

b)

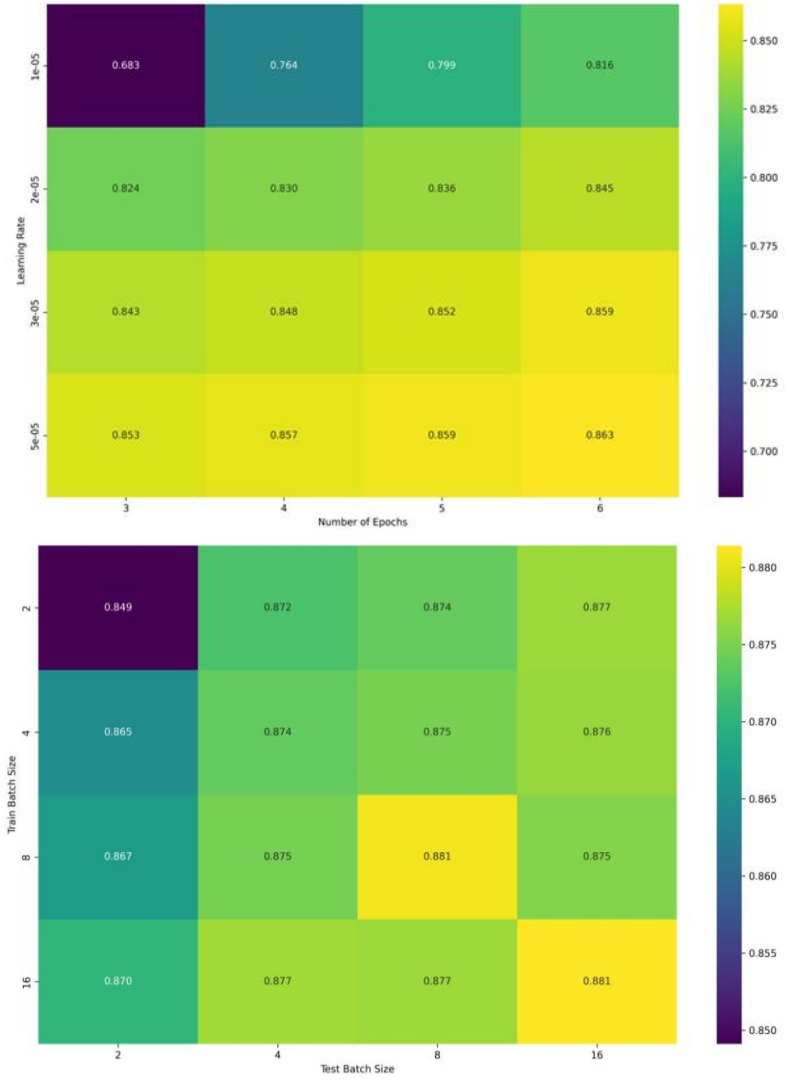


Рис. 5. Тепловая карта метрики F1 score: a – скорость обучения/количество эпох; b – размер обучающего пакета/размер валидирующего пакета.
Fig. 5. Heat map of the F1 score metric: a – learning rate/number of epochs; b – size of the training package /size of the validating package.

Существует несколько подходов к объединению точности (Precision) и полноты (Recall) в единую метрику качества. Одним из таких подходов является использование F-меры (F-score) [30], которая представляет собой взвешенное гармоническое среднее между точностью и полнотой:

$$F_\beta = (1 + \beta^2) * \frac{precision * recall}{(\beta^2 * precision) + recall} \quad (4)$$

Когда коэффициент β, регулирующий баланс между полнотой и точностью, равен 1, F-мера сводится к частному случаю – F1-мере (F1-score), которая равномерно учитывает точность и полноту (Precision = Recall = 1). При $\beta > 1$ большее значение придается полноте, при $\beta < 1$ – точности.

Сравнительный анализ ruBERT и RuBioBERT показал, что качество работы модели ruBERT выше, чем модели RuBioBERT. Результаты сравнительного анализа показаны ниже (табл. 3-5).

Интересно отметить, что, хотя RuBioBERT специально обучена для работы с медицинской терминологией, полученные результаты, вероятно, связаны с тем, что её обучение проводилось на медицинских статьях. Научные статьи сами по себе имеют определённую специфику, а тексты медицинских инструкций к лекарственным средствам – ещё более узкую. Эта особенность обуславливает различия в закономерностях, характерных для научных и медицинских текстов, по сравнению с общеупотребительными текстами, на которых была обучена модель ruBERT.

Следовательно, для приближения закономерностей, свойственных текстам инструкций к лекарствам, больше подходят закономерности, характерные для общих текстов, поскольку они отражают более универсальные свойства естественного языка. В отличие от специализированных текстов, эти общие закономерности применимы к любым разновидностям языка и дают более точные результаты.

Сравнительный анализ, проведённый авторами данного исследования, показал, что качество однозадачных моделей ruT5 ниже, чем у многозадачной модели ruT5 (см. табл. 3–5). Это является интересным результатом, учитывая, что многозадачные модели, как правило, демонстрируют худшие результаты по сравнению с однозадачными. Предполагается, что причина этого заключается в том, что однозадачная модель была обучена на меньшем объёме данных и является многоязычной.

Кроме того, в рамках проведённого анализа было установлено, что оптимальное количество эпох для обучения модели BERT в задаче распознавания именованных сущностей (NER) составляет 6. Увеличение числа эпох за пределы этого значения не приводит к существенному улучшению качества модели, при этом продолжительность обучения значительно возрастает. Таким образом, авторы пришли к выводу, что шесть эпох являются оптимальным числом для обучения данной модели.

3.4 Поиск зависимостей между именованными сущностями для построения семантического графа

Из кортежей вычленяются именованные сущности и создаются узлы. Главным узлом является наименование лекарственного средства. В каждом предложении определяются связи между словами:

- подлежащего и сказуемого;
- зависимых слов от глагола;
- зависимых глаголов от текущего глагола.

После добавления всех необходимых узлов производится поиск глаголов, не зависящих от других глаголов. Именованные сущности, ассоциированные с этими независимыми глаголами, связываются с главной вершиной графа. Образованное ребро подписывается наименованием соответствующего независимого глагола (рис. 6). Например, в предложении «Флуконазол повышает плазменную концентрацию сиролимуса предположительно из-за ингибирования метаболизма сиролимуса» слово «повышает» зависит от отглагольного существительного «ингибирования», что позволяет идентифицировать «ингибирование» как независимый элемент, не имеющий дополнительных зависимостей. Так как у глагола «ингибирование» имеется зависимая именованная сущность – «метаболизма сиролимуса», данная связь будет отображена на графе в виде ребра между главной вершиной и зависимой сущностью «метаболизма сиролимуса» с указанием на независимый глагол «ингибирование». Структурная схема алгоритма представлена на рис. 7.

Затем устанавливаются связи между именованными сущностями в случаях, когда один глагол зависит от другого. Для этого создаётся словарь, в котором каждому глаголу сопоставлен список слов, зависящих от него. Если обнаруживается зависимость одного глагола от другого, то выстраивается иерархическая структура связей: зависимые существительные подчинённого глагола будут также зависеть от существительных главного глагола (рис. 8). Например, в предложении «Флуконазол повышает плазменную концентрацию сиролимуса предположительно из-за ингибирования метаболизма сиролимуса» глагол «повышает» находится в зависимости от отглагольного существительного «ингибирования». В этом случае на графе устанавливается связь между зависимыми словами глаголов «повышает» и «ингибирования». Структурная схема алгоритма представлена на рис. 9.

На завершающем этапе в граф добавляются именованные сущности, играющие роль подлежащих, при этом сказуемое оформляется в виде ребра, выходящего из узла подлежащего (рис. 10). Например, в предложении «Флуконазол повышает плазменную концентрацию сиролимуса предположительно из-за ингибирования метаболизма сиролимуса» подлежащим является «Флуконазол», а сказуемым – «повышает». Следовательно, необходимо произвести редактирование графа, удаляя лишние ребра и располагая узел, соответствующий подлежащему «Флуконазол», перед ребром с пометкой «повышает». Структурная схема алгоритма представлена на рис. 11.

Реализованные авторами алгоритмы позволили построить семантический граф на основе текста инструкции к лекарственному средству (рис. 12).

Табл.3. Сравнение оценок качества работы ruBERT и RuBioBERT.
Table 3. Comparison of ruBERT and RuBioBERT work quality ratings.

Название модели	Accuracy	Precision	Recall	F1	AUC
rubert-base-cased_3	0,890909	0,736842	0,7	0,717949	0,899238
RuBioBERT_3	0,868182	0,672414	0,65	0,661017	0,887158
rubert-base-cased_4	0,9	0,745763	0,733333	0,739496	0,902762
RuBioBERT_4	0,927273	0,852459	0,866667	0,859504	0,923765
rubert-base-cased_5	0,922727	0,822581	0,85	0,836066	0,915715
RuBioBERT_5	0,913636	0,737705	0,75	0,743802	0,914775
rubert-base-cased_6	0,931818	0,868852	0,883333	0,876033	0,919964
RuBioBERT_6	0,909091	0,725806	0,75	0,737705	0,913922

Табл.4. Сравнение оценок качества работы ruT5 и ruT5 – multitask.
Table 4. Comparison of ruT5 and ruT5- multitask performance ratings.

Название модели	Accuracy	Precision	Recall	F1	AUC
ruT5 3	0,653958	0,482142	0,290322	0,362416	0,624223
ruT5 multitask 3	0,700879	0,533333	0,430107	0,476190	0,725815
ruT5 4	0,706744	0,5	0,440860	0,468571	0,741036
ruT5 multitask 4	0,703812	0,452380	0,408602	0,429378	0,755976
ruT5 5	0,695014	0,487804	0,430107	0,457142	0,740040
ruT5 multitask 5	0,712609	0,476190	0,430107	0,451977	0,748181
ruT5 6	0,689149	0,425287	0,397849	0,411111	0,737326
ruT5 multitask_6	0,697947	0,464285	0,419354	0,440677	0,742045

Табл.5. Сравнение результатов работы rubert-base-cased, RuBioBERT, rut5-base, rut5-base-multitask.
Table 5. Comparison of the results of rubert-base-cased, RuBioBERT, ruT5-base, ruT5-base-multitask.

rubert-base-cased		RuBioBERT		rut5-base		rut5-base-multitask	
[CLS]	O	[CLS]	O	Эффективность	B-noun	Эффективность	B-noun
Эффективность	B-noun	Эффективность	B-noun	арипипразола	I-noun	арипипразола	I-noun
арипипразола	I-noun	арипипразола	I-noun	при	O	при	O
при	O	при	O	адъюнктивном	B-noun	адъюнктивном	B-noun
адъюнктивном	B-noun	адъюнктивном	B-noun	лечении	O	лечении	O
лечении	I-noun	лечении	I-noun	БДР	I-noun	БДР	I-noun
БДР	I-noun	БДР	I-noun	была	O	была	O
была	B-glag	была	B-glag	продемонстрирована	B-glag	продемонстрирована	B-glag
продемонстрирована	B-glag	продемонстрирована	B-glag	в	O	в	O
в	O	в	O	двух	O	двух	O
двух	O	двух	O	краткосрочных	O	краткосрочных	O
краткосрочных	B-noun	краткосрочных	B-noun	плацебо-контролируемых	O	плацебо-контролируемых	O
плацебо	B-glag	плацебо	B-glag	исследованиях	B-noun	исследованиях	B-noun
-	B-glag	-	B-glag	у	O	у	O
контролируемых	B-glag	контролируемых	B-glag	взрослых	O	взрослых	O
исследованиях	B-noun	исследованиях	B-noun	пациентов,	I-noun	пациентов,	I-noun
у	O	у	O	соответствующих	B-glag	соответствующих	B-glag
взрослых	B-noun	взрослых	B-noun	критериям	B-noun	критериям	B-noun
пациентов	I-noun	пациентов	I-noun	DSM-IV	I-noun	DSM-IV	B-noun
,	O	,	O	для	O	для	O
соответствующих	B-glag	соответствующих	B-glag	БДР,	B-noun	БДР,	B-noun
критериям	O	критериям	O	у	O	у	O
DSM	B-noun	DSM	B-noun	которых	B-noun	которых	B-noun
-	B-noun	-	B-noun	был	B-glag	был	B-glag
IV	B-noun	IV	B-noun	неадекватный	B-noun	неадекватный	B-noun
для	O	для	O	ответ	I-noun	ответ	O
БДР	B-noun	БДР	B-noun	на	O	на	O
,	O	,	O	предыдущее	B-noun	предыдущее	B-noun
у	O	у	O	лечение	O	лечение	O
которых	B-noun	которых	B-noun	антидепрессантами	I-noun	антидепрессантами	I-noun
был	B-glag	был	B-glag	в	O	в	O
неадекватный	B-noun	неадекватный	B-noun	текущем	O	текущем	O
ответ	O	ответ	O	эпизоде	I-noun	эпизоде	I-noun
на	O	на	O	и	O	и	O
предыдущее	B-noun	предыдущее	B-noun	которые	O	которые	O
лечение	I-noun	лечение	I-noun	также	O	также	O

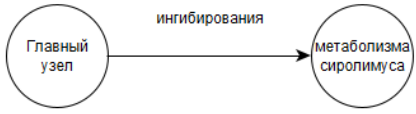


Рис. 6. Пример построения связи от главного узла к независимому глаголу.
Fig. 6. An example of building a connection from the main node to an independent verb.

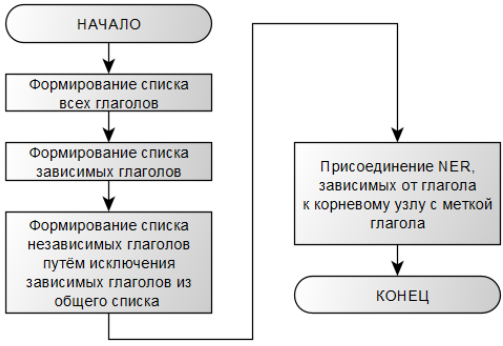


Рис. 7. Структурная схема алгоритма построения связи от главного узла к независимому глаголу.
Fig. 7. A block diagram of the algorithm for building a connection from the main node to an independent verb.

Рис. 8. Пример построения связи от независимого глагола к зависимому глаголу.
Fig. 8. An example of building a connection from an independent verb to a dependent verb.

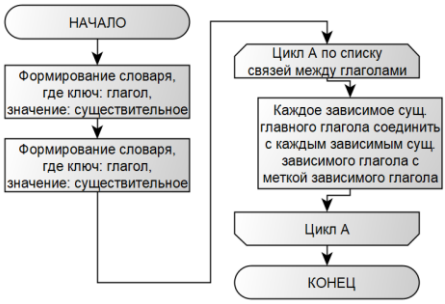


Рис. 9. Структурная схема алгоритма построения связи от независимого глагола к зависимому глаголу.
Fig. 9. A block diagram of an algorithm for building a connection from an independent verb to a dependent one.

Рис. 10. Пример построения связей подлежащего и сказуемого.
Fig. 10. An example of building links between a subject and a predicate.

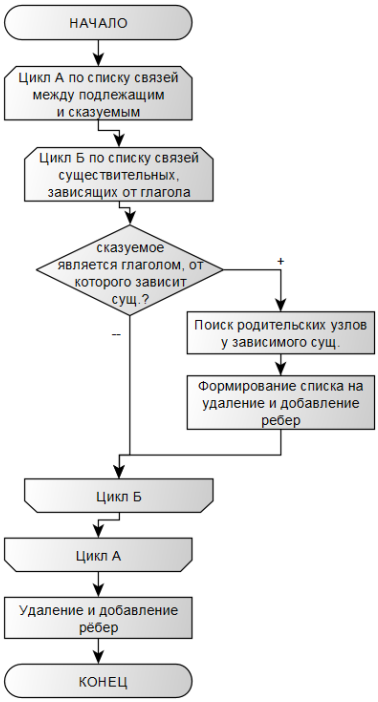


Рис. 11. Структурная схема алгоритма построения связей подлежащего и сказуемого.
Fig. 11. A block diagram of the algorithm for building connections between the subject and predicate.

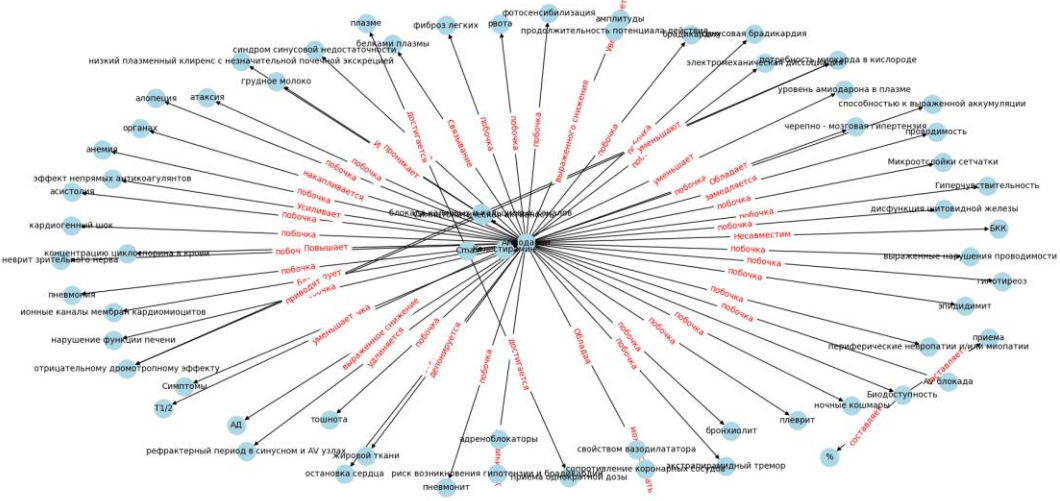


Рис. 12. Семантический граф текста инструкции к лекарственному средству «Амиадарон».
Fig. 12. Semantic graph of the text of the medical drug "Amiadarone".

3.5. Формализация графа для байесовской сети определения рисков возникновения побочных эффектов при полифармакотерапии

В результате анализа большого количества инструкций к различным лекарственным средствам возможно объединение одинаковых вершин. В результате возможно построить большую графовую модель, описания медицинских процессов, правил приема, особенностей приема и другой важной информации с точки зрения текстов инструкций к лекарственным средствам. Первичными вершинами графа для построения байесовской сети являются названия лекарственных средств, имеющие два состояния: принимается (A_1) или не принимается (A_0). $P = \{p_1, p_2 \dots p_i, \dots p_n\}$; $p_i \in \{A_{0,i}, A_{1,i}\}$, где, P – множество всех введенных в систему лекарственных средств; p_i – i-ое лекарственное средство; $A_{0,i}, A_{1,i}$ – состояния i-го лекарственного средства. Для препаратов с различными способами ввода или дозировками возможно добавление состояний с уменьшенными вероятностями действия лекарственного средства $p_i \in \{A_{0,i}, A_{1,i} \dots A_{q_i,i}\}$, где q_i – количество состояний для i-го лекарственного средства. Представленные вершины могут объединяться в группы аналогично медицинской классификации для удобства определения аналогов, альтернатив и обобщения действий. $H = \{h_1, h_2 \dots h_s, \dots h_S\}$, для $\forall p_i \; \exists h_s$: $p_i \in h_s$, где, H – множество групп лекарственных средств $S \ll n$.

При назначении лекарственных средств происходит активация соответствующих вершин, перевод состояний вершин p_i для каждого назначенного лекарственного средства с $A_{0,i}$ – лекарственное средство не принимается, до $A_{1,i}$ – прием лекарственного средства, или другое состояние, если для лекарственного средства указывается дозировка или способ приема. Требуется оценить вероятности различных состояний ($C_{0,k}$ – побочный эффект не наблюдается, $C_{1,k}$ – наблюдается побочный эффект) для всех вершин побочных эффектов. Оценки вероятностей побочных эффектов позволяют оценить риски при одновременном приеме нескольких лекарственных средств. В результате одновременной активации нескольких вершин лекарственных средств определяются риски возникновения различных побочных эффектов, риски при полифармакотерапии.

4. Заключение

В процессе выполнения работы был создан специализированный словарь для поиска ключевых терминов в текстах медицинских инструкций. Этот словарь включает в себя термины, отобранные из различных авторитетных источников, таких как база данных VigiAccess, Международная классификация болезней (МКБ-10), а также справочные материалы с сайта rlsnet.ru. Собранный текстовый корпус был тщательно подготовлен и отформатирован для обеспечения высокого качества обучения модели, включающего процедуры очистки данных и приведения их к единому формату.

В будущем, для сбора текстового корпуса, планируется перейти к использованию более авторитетного источника grls.rosminzdrav.ru, чтобы использовать тексты инструкций с официального сайта, поскольку данный источник считается более достоверным с точки зрения точности и полноты информации о зарегистрированных лекарственных средствах.

Для автоматизации процесса аннотации данных был разработан алгоритм, который осуществлял поиск терминов из созданного словаря и автоматически размечал их в формате BIO (Begin, Inside, Outside). Это позволило упорядочить процесс меток, обеспечивая структурированную разметку, необходимую для последующего обучения моделей машинного обучения.

В ходе работы была разработана модель на основе глубоких нейронных сетей, продемонстрировавшая высокую эффективность в задаче распознавания именованных сущностей. Применение методов глубокого обучения позволило учитывать контекстные зависимости между словами, что обеспечило более точное выделение медицинских терминов в текстах.

Решение задачи построения семантического графа лекарственных средств опирается на реализованные алгоритмы нахождения связей между именованными сущностями, которые, однако, не предусматривают нахождение глубоких связей между узлами графа, так как в текстах важную роль играют подлежащие и сказуемые. Для углубления связей на уровень ниже необходимо, чтобы в тексте зависимое слово, которое занимает конечное положение в графе, являлось подлежащим в другом предложении. Однако структура естественного языка не всегда соответствует этому требованию, что важно для работы алгоритма, но не обязательно для текста. В связи с этим возникает потребность в ручной доразметке инструкций, позволяющей выявлять более сложные зависимости.

Список литературы / References

[1]. Popov A. M. Adaskina Yu. V. Andreyeva D. A. Charabet Ja. K. Moskvina A. D. Protopopova E. V. Yushina T. A. Named Entity Normalization for Fact Extraction Task. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2016” Moscow, June 1–4, 2016.

[2]. Sysoev A. A. Andrianov I. A. Named Entity Recognition in Russian: the Power of Wiki-Based Approach. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2016” Moscow, June 1–4, 2016.

[3]. Stepanova M. E. Budnikov E. A. Chelombeeva A. N. Matavina P. V Skorinkin D. A. Information Extraction Based on Deep Syntactic-Semantic Analysis. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2016” Moscow, June 1–4, 2016.

[4]. Dingcheng Li, Karin Kipper-Schuler, and Guergana Savova. Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. In Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing, 2008, BioNLP ’08, pages 94–95, Stroudsburg, PA, USA. Association for Computational Linguistics.

[5]. S. Keretna, C. P. Lim, and D. Creighton. A hybrid model for named entity recognition using unstructured medical text. In 2014 9th International Conference on System of Systems Engineering (SOSE), 2014, pages 85–90, June.

[6]. Carol Friedman, Philip O Alderson, John HM Austin, James J Cimino, and Stephen B Johnson. A general natural-language text processor for clinical radiology. Journal of the American Medical Informatics Association, 1994, 1(2):161–174.
[7]. Alan R Aronson and Franc¸ois-Michel Lang. An overview of metamap: historical perspective and recent advances. Journal of the American Medical Informatics Association, 2010, 17(3):229–236.
[8]. Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C KipperSchuler, and Christopher G Chute. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. Journal of the American Medical Informatics Association, 2010, 17(5):507–513.
[9]. Gurulingappa H, Hofmann-Apitius M, and Fluck J. Concept identification and assertion classification in patient health records. In Proceedings of the 2010 i2b2/VA Workshop on Challenges in Natural Language Processing for Clinical Data. 2010.
[10]. Scott Halgrim, Fei Xia, Imre Solti, Eithon Cadag, and Ozlem Uzuner. Extracting medication information from discharge summaries. In Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents, Louhi'10, 2010, pages 61–67, Stroudsburg, PA, USA. Association for Computational Linguistics.
[11]. Shaodian Zhang and Noemie Elhadad. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts. Journal of Biomedical Informatics, 2013, 46(6):1088 – 1098.
[12]. Asif Ekbal and Sriparna Saha. Stacked ensemble coupled with feature selection for biomedical entity extraction. Knowledge-Based Systems, 2013, 46(0):22 – 32.
[13]. H. L. Shashirekha and H. A. Nayel. A comparative study of segment representation for biomedical named entity recognition. In 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, pages 1046– 1052, Sept.
[14]. Sara Keretna, Chee Peng Lim, Doug Creighton, and Khaled Bashir Shaban. Enhancing medical named entity recognition with an extended segment representation technique. Computer Methods and Programs in Biomedicine, 2015, 119(2):88 – 100.
[15]. Yonghui Wu, Min Jiang, Jianbo Lei, and Hua Xu. Named entity recognition in chinese clinical text using deep neural network. Studies in health technology and informatics, 2015, 216:624.
[16]. Giorgi,J.M. and Bader,G.D. Transfer learning for biomedical named entity recognition with neural networks. Bioinformatics, 2018, 34, 4087.
[17]. Habibi,M. et al. Deep learning with word embeddings improves bio- medical named entity recognition. Bioinformatics, 2017,33, i37–i48.
[18]. Wang,X. et al. Cross-type biomedical named entity recognition with deep multi-task learning. Bioinformatics, 2018, 35, 1745–1752.
[19]. Yoon,W. et al. Collabonet: collaboration of deep neural networks for biomedical named entity recognition. BMC Bioinformatics, 2019, 20, 249.
[20]. VigiAccess. Глобальная база данных Всемирной организации здравоохранения (ВОЗ). https://vigiaccess.org. 2024.
[21]. МКБ-10. Международная классификация болезней 10-го пересмотра. https://mkb-10.com. 2024.
[22]. Регистр лекарственных средств. Энциклопедия лекарств РЛС. https://www.rlsnet.ru. 2024.
[23]. Razdel. rule-based system for Russian sentence and word tokenization. https://github.com/natasha/razdel. 2024.
[24]. Б.И. Гельцер, Т.А. Горбач, В.В. Грибова, О.В. Карпик, Э.С. Клышинский, Н.А. Кочеткова, Д.Б. Окунь, М.В. Петряева, К.И. Шахгельдян, Синтаксический анализ текстов предметной области при помощи онтологии. Труды ИСП РАН, 2021, том 33, вып. 4.
[25]. Sebastian Raschka, Model Evaluation. Model Selection, and Algorithm Selection in Machine Learning. University of Wisconsin–Madison Department of Statistics November 2018.
[26]. Gaël Varoquaux, Olivier Colliot. Evaluating machine learning models and their diagnostic value. HAL open science Submitted on 21 Jan 2023 (v4), last revised 20 Apr 2023 (v5).
[27]. Pedregosa F, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011. 12(85):2825–2830.
[28]. Vickers AJ, Van Calster B, Steyerberg EW. Net benefit approaches to the evaluation of prediction models, molecular markers, and diagnostic tests, 2016, bmj 352.
[29]. Powers D Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. Journal of Machine Learning Technologies. 2011. 2(1):37–63.

[30]. Perez-Lebel A, Morvan ML, Varoquaux G. Beyond calibration: estimating the grouping loss of modern neural networks. ICLR. 2023.

Информация об авторах / Information about authors

Юрий Павлович ТИТОВ – кандидат технических наук, доцент, ведущий научный сотрудник научной лаборатории «Перспективных систем хранения и обработки сверхбольших массивов данных» Российского экономического университета имени Г.В. Плеханова. Сфера научных интересов: метаэвристическая оптимизация, графовые модели, машинное обучение, нечеткая логика и имитационные модели.

Yuri Pavlovich TITOV – Cand. Sci. (Tech.), Associate Professor, Leading Researcher of the Scientific Laboratory of “Advanced Systems for Storage and Processing of Ultra-Large Data Arrays” of the Plekhanov Russian University of Economics. Research interests: metaheuristic optimization, graph models, machine learning, fuzzy logic and simulation models.

Никита Владимирович КИЛЬМИШКИН является сотрудником лаборатории «Прикладное моделирование» Российского экономического университета имени Г.В. Плеханова. Его научные интересы включают машинное обучение.

Nikita Vladimirovich KILMISHKIN is an employee of the laboratory "Applied Modeling" of the Plekhanov Russian University of Economics. His research interests include machine learning.

Дмитрий Дмитриевич КУБРАКОВ – является сотрудником лаборатории «Перспективных систем хранения и обработки сверхбольших массивов данных» Российского экономического университета имени Г.В. Плеханова. Его научные интересы включают машинное обучение, машинная лингвистика, обработка больших данных.

Dmitry Dmitrievich KUBRAKOV is an employee of the laboratory of "Advanced Storage and Processing systems for Ultra-large data arrays" of the Plekhanov Russian University of Economics. His research interests include machine learning, machine linguistics, and big data processing.

Полина Михайловна ИВАНОВА – является сотрудницей лаборатории «Перспективных систем хранения и обработки сверхбольших массивов данных» Российского экономического университета имени Г.В. Плеханова. Ее научные интересы включают обработка больших данных.

Polina Mikhailovna IVANOVA – is an employee of the laboratory of "Advanced Storage and Processing systems for Ultra–large data arrays" of the Plekhanov Russian University of Economics. Her research interests include big data processing.