УДК 004.912 + 004.8

Связывание терминов из научных текстов с сущностями баз знаний

Кузовлев Д.Д. (Институт систем информатики им. А.П. Ершова СО РАН),
Батура Т.В. (Институт систем информатики им. А.П. Ершова СО РАН),
Старцев П.А. (Институт систем информатики им. А.П. Ершова СО РАН)

В данной статье предлагаются новые алгоритмы связывания научных терминов с сущностями Википедии и MeSH (Medical Subject Headings), работающие в условиях ограниченного количества данных. Алгоритм связывания с Википедией для части коллекции текстов на русском языке использует поисковую систему Википедии для генерации кандидатов и библиотеку spaCy для получения векторного представления текста. Подсчет семантической близости между описанием сущности из Википедии и термином из текста выполняется не только на основе самого научного термина, но и контекста, в котором он расположен. Для части коллекции по медицинской тематике, содержащей переводы с русского на английский язык, описан и реализован алгоритм связывания терминов, который опирается на словарь медицинских предметных рубрик MeSH. Результаты экспериментов показывают значения F1-меры 50.77 % для Википедии и 40.05 % для MeSH, что является хорошим показателем в условиях малого объема размеченных данных. Проведенное исследование подчеркивает необходимость развития специализированных русскоязычных баз знаний по аналогии с MeSH. Перспективным направлением является применение мультиязычных моделей для кросс-лингвистического связывания, что особенно актуально для редких терминов. Полученные результаты могут применяться при создании интеллектуальных систем анализа научных текстов и автоматизированных научных ассистентов, что особенно важно для узких предметных областей.

Ключевые слова: *обработка текстов, извлечение информации, связывание сущностей, поиск и ранжирование, Википедия, MeSH*

1. Введение

Научные тексты всегда были и остаются основным источником новых знаний. Однако с увеличением объема публикуемых материалов возникает проблема эффективного извлечения и структурирования информации. Одной из ключевых задач в области обработки естественного языка является связывание сущностей (Entity Linking, EL), которая заключается в сопоставлении упоминаемых понятий в текстах, с соответствующими записями в базах знаний [1]. Данная статья посвящена исследованию

этой важной задачи для научных текстов, где в качестве терминов выступают слова или фразы, используемые в определенной предметной области для точного обозначения конкретных понятий, явлений или объектов; под сущностями подразумеваются элементы (записи) базы знаний. Решение этой задачи открывает новые возможности для создания интеллектуальных систем, способных автоматически анализировать научные публикации, извлекать ключевые концепции и связывать их с уже существующими структурированными знаниями.

Связывание терминов с сущностями баз знаний представляет собой сложную задачу, основные трудности которой связаны с многозначностью терминов, различиями в терминологии между дисциплинами, а также с необходимостью учета контекста использования терминов в тексте. Так, современные методы EL активно используют контекстуальное усиление (contextual augmentation) для улучшения качества связывания. Авторы [2] предлагают подход, основанный на больших языковых моделях (Large Language Model, LLM), которые генерируют дополнительные контексты для упоминаний сущностей, помогая разрешать неоднозначности. Метод включает динамическое расширение контекста запроса и семантическое ранжирование кандидатов, что особенно полезно для коротких или малоинформативных упоминаний. Важным преимуществом является адаптивность под разные области знаний, что делает подход универсальным для специализированных баз знаний.

В статье [3] представлен масштабируемый метод EL, работающий для 100 языков, включая редкие и малоресурсные. Авторы используют двунаправленные трансформеры (mBERT) и обучают модель на данных с автоматически сгенерированной разметкой (так называемых “слабо размеченных данных”), применяя техники дистилляции знаний и кросс-языкового переноса. Особенность подхода заключается в эффективном использовании мультиязычных эмбеддингов, что позволяет достигать высокой точности даже для языков с ограниченным количеством обучающих данных.

Упомянутые выше работы требуют огромного количества вычислительных мощностей, поэтому в условиях ограниченных вычислительных ресурсов актуальны эффективные и экономичные методы EL. Авторы работы [4] предлагают систему ReLiK, сочетающую быстрый поиск кандидатов на основе индексированных эмбеддингов и точное ранжирование с использованием легковесных нейросетевых архитектур. Этот подход демонстрирует конкурентоспособную точность при значительно меньших затратах по сравнению с тяжелыми LLM, что делает его перспективным для академических исследований.

В статье [5] описан метод, основанный на плотном поиске сущностей (Dense Entity Retrieval, DER). В отличие от традиционных подходов, использующих разреженные представления (например, TF-IDF или BM25), авторы применяют нейросетевые эмбеддинги для кодирования как текстовых упоминаний, так и описаний сущностей из базы знаний. Этот подход не требует обучения на размеченных данных для конкретной области, поскольку использует общие семантические представления, что особенно полезно для редких или новых терминов. Для русского языка такие методы являются перспективным, так как позволяют обойти проблему нехватки аннотированных корпусов, полагаясь на мультиязычные эмбеддинги (например, из mBERT).

В целом, можно заметить, что большая часть исследований ориентирована на английский язык, конкретно русскому языку посвящено лишь ограниченное число работ [6, 7]. Однако актуальность исследований для русского языка обусловлена необходимостью развития инструментов, учитывающих его морфологическую развитость и ограниченность размеченных данных. В данной статье предложен новый алгоритм связывания терминов из научных текстов на русском языке с сущностями Википедии. Эксперименты проведены с текстами по четырем областям знаний: информационные технологии, медицина, психология и лингвистика.

Особое внимание в статье уделяется медицинской области. Ясно, что специализированные медицинские терминологические базы являются более полными по сравнению с Википедией. В частности, MeSH – одна из наиболее полных и авторитетных терминологических систем в биомедицине, что позволяет использовать ее в качестве эталонного ресурса для связывания сущностей. Русскоязычная версия MeSH существенно уступает английскому оригиналу по охвату терминов, частоте обновлений и отсутствию столь же строгой системы верификации. Кроме того, в отличие от англоязычного MeSH, который доступен через официальные ресурсы NLM (National Library of Medicine), русскоязычная версия не имеет централизованной открытой платформы. Хотя частичные переводы находятся в специализированных базах данных (например, в Российском индексе научного цитирования или каталогах РГМУ), но их структура и доступность варьируются, что усложняет интеграцию русскоязычных терминов в системы автоматического извлечения и связывания сущностей. Поэтому для медицинской области было решено дополнительно рассмотреть переводы текстов на английский язык. В данной статье предложен алгоритм связывания терминов из медицинских текстов, переведенных на английский язык, с сущностями MeSH. Поскольку основа словаря – это упорядоченное множество рубрик, то сущностями на данном этапе разработки алгоритма связывания являются рубрики. Разработанный алгоритм в дальнейшем может быть относительно

просто адаптирован для русского языка при наличии довольно полной регулярно обновляемой медицинской терминологической базы.

2. Существующие методы связывания сущностей

При традиционном подходе связывание сущностей проходит в два этапа: генерация кандидатов и ранжирование.

Генерация кандидатов. Генерация кандидатов в задаче связывания сущностей — это процесс отбора потенциальных сущностей из базы знаний, которые могут соответствовать данному термину в тексте. Этот этап снижает размерность поиска, позволяя более эффективное ранжирование на следующем шаге.

Существует несколько подходов для генерации кандидатов. Одним из них является подход, основанный на словаре имен [8–10]. Словарь имен подразумевает отображение из множества ключевых значений *key* в множество возможных кандидатов *value*. Например, ключевое значение термина “*поле*” сопоставляется с множеством кандидатов, в которое входят сущности “*алгебраическое поле*”, “*магнитное поле*”, “*поле (иконопись)*”, “*сельскохозяйственное поле*” и т.п. Другим подходом для генерации кандидатов является техника расширения словоформ. В работах [11, 12] термин дополняется контекстом.

Для генерации кандидатов может использоваться статистическая информация о встречаемости термина в различных текстах. На основе статистической информации строится априорная вероятность сущности. Априорная вероятность представляет эмпирическую оценку вероятности того, что термин связан с определенной сущностью. Например, в работе [13] авторы предлагают подход, основанный на использовании априорного распределения.

Для использования методов, основанных на словарях имен или вычислении априорной вероятности, необходимо иметь обучающие данные. В таких случаях используются предварительно заполненные словари имен или предварительно собранная статистическая информация о сущностях. Такие методы не подходят для областей знаний, по которым нет достаточного количества обучающих данных.

В данной статье для генерации кандидатов используется поисковая система Википедии, что позволяет опираться на уже существующую одноименную базу знаний. Подобный подход используется в работе [14]: применяются словари, основанные на Википедии, для поиска сущностей с похожими именами (с учётом аббревиатур, синонимов, вариантов написания), семантическое расширение кандидатов с помощью контекста (использование ключевых слов и других упоминаний в документе), обращение к поисковым системам, если первые два метода не дали достаточного набора кандидатов. Однако, возможно

использование и других поисковых систем. Например, в работах [12, 15, 16] для генерации кандидатов используется поисковая система Google. В работе [17] используется Bing. Далее выполняется фильтрация поискового запроса – выбираются только страницы из Википедии.

В данной статье предлагается подход для работы с неразмеченными данными. Подобный подход применяется в работе [18] – авторы предлагают метод для обучения EL-модели без аннотированных данных. В статье используется поверхностное сопоставление (surface matching) между термином и наименованием сущности в базе знаний. При поверхностном сопоставлении для каждого термина выполняется поиск сущности в базе знаний, у которых имя содержит все слова термина. Это отличается от подхода, применяемого в данной статье, где сущности-кандидаты выбираются среди первых n кандидатов, найденных через поисковой запрос по API Википедии или MeSH.

Ранжирование кандидатов. Ранжирование кандидатов — это процесс упорядочения списка потенциальных сущностей-кандидатов по степени их соответствия заданному термину в тексте. Этот этап следует за генерацией кандидатов и используется для выбора наиболее релевантной сущности с учетом контекста. Ранжирование кандидатов подразумевает нахождение близости между термином и сущностью в базе знаний. В качестве меры близости используется скалярное произведение или косинусное сходство между векторными представлениями термина и сущности в базе знаний.

В последнее время все чаще используется механизм внимания [13, 19–21]. В статье [13] все слова в контексте и кандидаты-сущности отображаются в векторное пространство с помощью заранее обученных векторных представлений. Ранжирование сущностей-кандидатов выполняется в два этапа: сначала анализируется локальный контекст термина, а затем результаты уточняются с учётом всех сущностей в документе. Модель использует механизм внимания для выбора значимых слов и учитывает связи между сущностями, чтобы повысить точность предсказаний, что помогает избежать локальных ошибок и обеспечивает согласованность разметки. Подобный двухэтапный подход также используется в работе [21]. Авторы предлагают итеративный подход к ранжированию кандидатов, где сущности связываются последовательно, используя уже определенные термины для уточнения контекста. Рассматриваются два метода выбора сущностей: по наибольшей уверенности (confidence-order) и по порядку в тексте (natural-order). Эксперименты показывают, что первый метод работает лучше, так как позволяет накапливать глобальный контекст для более точного связывания.

В работе [20] ранжирование кандидатов выполняется с помощью кросс-энкодера, который объединяет контекст термина и описание каждой сущности в один вход для

модели BERT. Затем модель оценивает вероятность соответствия каждой сущности данному термину, а итоговый рейтинг формируется на основе выходных логитов. Этот метод значительно повышает точность по сравнению с двунаправленным энкодером, хотя и требует больше вычислительных ресурсов.

Авторы работы [19] предлагают заменить традиционный двухэтапный процесс связывания сущностей сквозным (end-to-end) методом, в котором как термины, так и сами сущности кодируются в одном векторном пространстве. В отличие от предыдущих работ, модель полностью отказывается от заранее подготовленных таблиц псевдонимов. Поиск кандидатов выполняется путем кодирования термина в векторное пространство, после чего выполняется приближенный поиск ближайших соседей (Approximate Nearest Neighbor) среди заранее закодированных сущностей на основе косинусного сходства. Сквозной метод на основе модели BERT также рассматривается в работе [22].

В данной статье для векторного представления термина из текстов на русском языке и сущностей Википедии используется сверточная нейронная сеть (Convolutional Neural Networks, CNN). Подходы, основанные на сверточной сети, используются также в работах [23, 24]. Выбор в пользу CNN в нашей работе был сделан по нескольким причинам. Во-первых, в силу своей природы сверточные нейронные сети хорошо справляются с выявлением локальных зависимостей в данных, что важно для понимания контекста. Во-вторых, CNN часто показывают хорошую производительность даже на относительно небольших объемах данных, что может быть полезно в задачах, где размеченных данных немного, как в нашем случае. В-третьих, хотя современные архитектуры, такие как трансформеры (например, BERT), могут быть более точными, они требуют значительно больше вычислительных ресурсов и времени для обучения и работы. CNN, в этом смысле, менее требовательные к вычислительным ресурсам и более быстрые благодаря возможности параллельной обработки данных.

Особенности связывания медицинских терминов с MeSH. Задача поиска и связывания медицинских терминов с MeSH имеет несколько аспектов сложности. Несмотря на то, что в настоящий момент рубрики довольно хорошо структурированы, на начальном этапе (примерно с середины прошлого века) элементы системы плохо индексировались. Лишь с 2010-х проводилась активная работа по индексации и публикации информации в общий (онлайн) доступ. Поэтому исторически сложилась неравномерность полноты данных: чем старше информация, тем менее полно она структурирована [25]. С этим же связана и проблема изменения данных в динамике, проанализированная по иерархии до 2017 года количественно, а также аналитически и в графическом виде [26]. Очевидно, что при изменении иерархии структуры рубрик

необходимо всем зависимым источникам данных по задачам связывания сущностей или поиска обновить результаты, хотя бы частично использующие изменившийся источник данных. В рамках построения алгоритма связывания с MeSH в настоящей работе сравнение количества рубрик по данным 2024 года с данными 2025 года дало прирост около 200 записей (менее 1%), что говорит о замедлении темпов пополнения терминологии и указывает на то, что система стала достаточно полной и структурированной.

Авторы более узких прикладных задач, таких как связывание сущностей в области генной информации с рубриками MeSH, указывают на несовершенство каждого в отдельности инструмента работы с MeSH: приходится выстраивать pipeline-решения из них и собственного разработанного программного обеспечения, трудности с параллельными вычислениями и тестами, а также неудовлетворительную длительность поиска через различные готовые онлайн инструменты MeSH [27].

Несмотря на предложенные ранее алгоритмы индексации, связывания сущностей и поиска статей при помощи базы знаний MeSH в комбинации с системами MEDLINE и PUBMED (паттерны поиска, классификаторы текстов, включая алгоритм классификации по K-ближайшим соседям, обучение с ранжированием, а также комбинации этих и других подходов), задача все еще остается сложной [25].

3. Описание данных

Для проведения экспериментов была собрана текстовая коллекция из аннотаций к научным статьям, находящимся в открытом доступе. Первая часть коллекции (используемая в экспериментах с Википедией) включает в себя аннотации на русском языке по четырем научным областям: информационные технологии, медицина, психология и лингвистика. Средняя длина текста – 216 слов, общее количество терминов – 367.

Вторая часть только по медицинской тематике (используемая в экспериментах с MeSH) включает в себя аннотации к тем же статьям, которые сами авторы перевели на английский язык. Для связывания с MeSH использовалась коллекция из 48 текстов. Количество терминов в исходных текстах варьировалось от 7 до 60 (в среднем около 25 терминов на текст). Средняя длина текста – 327 слов, общее количество терминов – 1223.

Перед подачей на вход алгоритмам связывания сущностей выполнялась предварительная обработка данных. Схема обработки данных представлена на рисунке 1.

Рисунок 1 – Последовательность обработки данных

Процесс обработки данных проходил в четыре этапа.

1. Разметка сущностей. В тексте научной аннотации выделяются два типа сущностей: TERM и VALUE. Под терминами (сущности типа TERM) понимаются слова или фразы, используемые в определенной предметной области для точного обозначения конкретных понятий, явлений или объектов. Например, в области информационных технологий терминами являются названия методов, архитектур, моделей, языков программирования и другие, в медицине – названия заболеваний, симптомов, препаратов, диагностических процедур и пр. Терминами считаются, в том числе, и аббревиатуры. Сущности типа VALUE – это числовые значения в сочетании с дополнительной информацией (контекстом или единицей измерения), количественные или качественные показатели, используемые для описания конкретных данных, которые можно измерить или оценить.

Каждой сущности сопоставляется уникальный идентификатор. Сущности выделяются в формате ***Термин*** **|** ***Идентификатор*** **|** ***Метка***. Метка TERM означает, что выделенный набор слов является термином; для числовых значений используется метка VALUE.

Первоначально исходные тексты были размечены с помощью модели gpt-4o-mini. Далее выполнялась ручная коррекция (по два аннотатора на каждый текст). Чтобы избежать несогласованной разметки, предварительно была разработана инструкция с описанием и примерами. На заключительном этапе разметки модератор разрешал

оставшиеся неоднозначные случаи в соответствии с этой инструкцией. Согласованность разметки (Inter-Annotator Agreement) вычислялась с помощью стандартной статистической меры – коэффициента каппы Коэна (Cohen's kappa). Для подготовленных данных получено среднее значение 0.73, что свидетельствует о высоком качестве разметки [28].

2. Извлечение терминов. Для удобства дальнейшей обработки термины (сущности типа TERM) обособляются из аннотированного текста и извлекаются в отдельный блок. Далее в алгоритмах связывания рассматриваются только сущности типа TERM; сущности типа VALUE не учитываются.

3. Выявление морфологических кластеров. В тексте может встречаться один и тот же термин несколько раз или в разных словоформах (в разных падежах, единственном и множественном числе). Например, на рисунке 1 термин "*Web-сервисов*" встречается два раза, поэтому после этапа обнаружения терминов будет найдено два термина: ***Web-сервисов | T1 | TERM*** и ***Web-сервисов | T5 | TERM***, у каждого из них – свой уникальный идентификатор. На этапе выявления морфологических кластеров происходит объединение идентификаторов одного термина и разных словоформ этого термина в общий кластер.

4. Корректировка идентификаторов. В аннотированном тексте для дубликатов и всех словоформ одного термина проставляется один идентификатор. В качестве уникального идентификатора выбирается первый из идентификаторов морфологического кластера, построенного на предыдущем шаге. Корректировка идентификаторов позволяет однозначно определить термин.

Данные, прошедшие предварительную обработку, далее подаются на вход алгоритмам связывания сущностей.

4. Связывание сущностей с Википедией

4.1. Описание алгоритма

Предлагаемый алгоритм связывания сущностей состоит из трех шагов:

1. поиск ссылок в Википедии;
2. подсчет семантической близости между кратким описанием содержимого ссылки и термином;
3. ранжирование найденных ссылок.

Далее подробно опишем каждый шаг.

1. Поиск по термину заданного количества ссылок в Википедии. Взаимодействие с Википедией осуществляется через API. Входными данными для поиска являются термин и количество ссылок.

В Википедии поиск реализован с помощью системы Elasticsearch, которая позволяет осуществлять полнотекстовый анализ и учитывать особенности языка. При индексации текста статьи проходит морфологический разбор, выполняется лемматизация (слова приводятся к начальной форме), что позволяет находить нужные статьи независимо от склонений и спряжений. Например, при запросе “*нейронная сеть*” будут найдены страницы, где встречаются формы “*нейронной сети*”, “*нейронными сетями*”, “*нейронную сеть*” и т.д. Кроме того, учитываются заголовки статей, анкорные тексты и первые абзацы, поскольку они имеют больший вес при ранжировании результатов.

Для повышения гибкости поиска используется нечеткое сопоставление — система способна обрабатывать опечатки, неточные формулировки и вариативные формы слов. Это достигается за счёт применения расстояния Левенштейна (fuzzy matching), анализа по n-граммам и специальных алгоритмов подсказок. Ранжирование основано на сочетании факторов: частоты терминов в документе (TF-IDF или BM25), популярности статьи, количества ссылок на неё и положения совпадения в тексте. Таким образом, поиск и ранжирование в русской Википедии сочетают лингвистическую точность и устойчивость к ошибкам пользователя.

Для каждой найденной сущности API возвращает в ответе краткое описание содержимого ссылки, которое имеется в поле snippet (это поле используется для подсчета семантической близости на следующем шаге алгоритма). Пример ответа представлен на рис. 2.


```
{
  "batchcomplete": "",
  "continue": {
    "sroffset": 3,
    "continue": "-||"
  },
  "query": {
    "searchinfo": {
      "totalhits": 106922
    },
    "search": [
      {
        "ns": 0,
        "title": "Экономика",
        "pageid": 10678,
        "size": 16445,
        "wordcount": 804,
                                "snippet": "триллионах долларов США) Отрасль <span class=\"searchmatch\">экономики</span> Народное хозяйство Инновационная <span class=\"searchmatch\">экономика</span> Экономическая социология Электронная <span class=\"searchmatch\">экономика</span> Райзберг Б. А., Лозовский",
        "timestamp": "2024-11-04T19:28:50Z"
```

```
        },
        {
          "ns": 0,
          "title": "Экономика (наука)",
          "pageid": 282590,
          "size": 52258,
          "wordcount": 3136,
          "snippet": "интерэкономика (международная <span class=\"searchmatch\">экономика</span>) и мегаэкономика (мировое хозяйство)[источник не указан 4179 дней]. <span class=\"searchmatch\">Экономика</span> как сфера человеческой деятельности",
          "timestamp": "2025-02-05T20:01:41Z"
        },
        {
          "ns": 0,
          "title": "Экономика России",
          "pageid": 350396,
          "size": 538250,
          "wordcount": 28800,
                    "snippet": "2023 на Wayback Machine. Forbes, 04.03.2022 Цифровая <span class=\"searchmatch\">экономика</span>. Мобильная <span class=\"searchmatch\">экономика</span>. <span class=\"searchmatch\">Экономика</span> данных (неопр.). tass.ru. Дата обращения: 27 июня 2020",
          "timestamp": "2025-02-26T18:10:44Z"
        }
      ]
    }
}
```


Рисунок 2 – Результат работы метода связывания сущностей с Википедией

2. Подсчет семантической близости между кратким описанием содержимого ссылки и термином. Семантическая близость между искомым термином и ссылками-кандидатами, найденными на первом шаге, вычисляется на основе описаний из поля snippet ответа API Википедии с помощью косинусного расстояния. Для этого сам термин и текст из поля snippet представляется в векторном виде. Векторное представление и подсчет семантической близости выполняется с использованием модели ru_core_news_md, предоставляемой библиотекой spaCy. Таким образом, векторизация текста происходит в несколько этапов.

Токенизация текста. Текст разбивается на токены, в результате получается объект Doc – контейнер для лингвистических аннотаций[1,2]. В качестве токенизатора используется базовый токенизатор библиотеки spaCy (spacy.Tokenizer.v1)[3].

Получение векторного представления каждого токена текста. Библиотека spaCy не предоставляет возможности обучать векторные представления, а использует готовые[4]. В

[1] https://spacy.io/api/tokenizer
[2] https://spacy.io/api/doc
[3] https://github.com/explosion/spaCy/blob/master/spacy/language.py

модели ru_core_news_md используется векторные представления из библиотеки Navec[5], которые получены с помощью алгоритма GloVe [29], обученного на художественной литературе, с последующей квантизацией векторных представлений[6].

Получение векторного представления текста. В качестве векторного представления текста используется среднее арифметическое входящих в текст токенов.

3. Ранжирование найденных ссылок по семантической близости. Для каждой найденной ссылки из Википедии алгоритм рассчитывает семантическую близость между векторным представлением научного термина (с его контекстом) и описанием сущности из поля snippet для данной ссылки. По полученному значению семантической близости выполняется ранжирование ссылок из Википедии от наиболее близкой по смыслу к наименее близкой. В результате ранжирования определяется ссылка с самым высоким значением семантической близости, которая и связывается с научным термином.

4.2. Результаты экспериментов

Эксперименты проводились для разных значений ширины контекстного окна и количества ссылок. Шириной контекстного окна является количество слов в тексте слева и справа от термина. Например, если ширина контекстного окна равна 5, то в качестве контекста выбирается 5 слов перед термином, сам термин и 5 слов после термина. Результаты экспериментов представлены в таблице 1.

Таблица 1 – Результаты экспериментов с Википедией. Лучшие значения выделены жирным шрифтом, значения на втором месте выделены наклонным шрифтом.

Количество ссылок	**Ширина контекстного окна**	**Точность, %**	**Полнота, %**	**F1-мера, %**
1	5	35.3896	73.1544	47.7024
1	10	***37.3377***	***77.1812***	***50.3282***
1	100	35.7143	73.8255	48.1400
1	1000	**37.6623**	**77.8523**	**50.7658**
5	5	10.0649	20.8054	13.5668
5	10	10.3896	21.4765	14.0044
5	100	11.0390	22.8188	14.8796
5	1000	11.6883	24.1611	15.7550
10	5	5.1948	10.7383	7.0022

[4] https://spacy.io/usage/embeddings-transformers#static-vectors
[5] https://github.com/natasha/navec
[6] https://natasha.github.io/navec/

Количество ссылок	Ширина контекстного окна	Точность, %	Полнота, %	F1-мера, %
10	10	5.5195	11.4094	7.4398
10	100	3.2468	6.7114	0.043764
10	1000	3.8961	8.0537	0.052516

Можно заметить, что размер контекстного окна оказывает влияние на точность при большом количестве ссылок. Самые высокие результаты получены при использовании одной ссылки. Это можно объяснить тем, что контекст, содержащий термин, чаще всего захватывает лишнюю информацию (создает “шум”), по которой и происходит ранжирование ссылок. Вместе с тем видно, что при использовании одной ссылки увеличение ширины контекстного окна в 100 раз (с 10 до 1000) дает лишь незначительный прирост метрик качества.

Ошибки в работе алгоритма связывания сущностей с Википедией можно разделить на три группы: ошибки поиска, ошибки ранжирования и ошибки разметки.

Ошибки поиска. В категорию ошибок поиска попадают ситуации, когда термин был связан с некорректной сущностью ввиду особенностей работы алгоритма поиска Википедии. При таких ошибках в выдаче запроса отсутствует корректная сущность для связывания. Здесь возможны два случая: в Википедии отсутствует корректная сущность или в Википедии есть корректная сущность, но эта сущность не оказалась в выдаче запроса. В первом случае при корректной работе алгоритм оставит термин несвязанным. Второй случай более интересный. Корректная сущность может не оказаться в выдаче по двум причинам. Первая причина – это особенность работы самого алгоритма поиска в API Википедии. Например, сущность появилась в Википедии недавно, и не была проиндексирована (о работе индекса см. предыдущий раздел). Поэтому такие сущности не будут появляться в поисковой выдаче. В данном случае целесообразно считать правильным, если термин остался несвязанным. Вторая причина – это некорректно составленный запрос. Ошибки такого рода необходимо анализировать отдельно для понимания, каким образом можно корректировать запрос, чтобы в поисковой выдаче получать корректную сущность.

Ошибки ранжирования. В категорию ошибок ранжирования относятся ситуации, когда корректная сущность попала в результаты выдачи поискового запроса, но на этапе ранжирования не получила самую высокую оценку семантической близости с термином.

Ошибки разметки. В данную категорию можно отнести случаи, когда аннотатор неправильно связал термин. Особый интерес представляют термины с неоднозначной

разметкой. Иногда даже эксперты затрудняются выбрать, какая сущность соответствует термину. Когда варианты равнозначны, целесообразно считать работу алгоритма правильной, если алгоритм связывает термин хотя бы с одной из сущностей-кандидатов. Например, термину "*система информационного поиска*" может соответствовать как сущность https://ru.wikipedia.org/wiki/Информационный_поиск, так и сущность https://ru.wikipedia.org/wiki/Поисковая_система. В некоторых случаях аннотатор может быть недостаточно хорошо знаком с предметной областью, чтобы судить о том, насколько корректно связан термин. Например, термину "*врожденной дисфункции коры надпочечников*" может быть сопоставлена сущность https://ru.wikipedia.org/wiki/Врождённая_гиперплазия_коры_надпочечников, однако без экспертных знаний в области медицины иногда сложно оценить полученный результат.

Отметим, что полнота базы знаний влияет на качество результата. Алгоритм связывает термин с сущностью, опираясь на семантическую близость. Поэтому если в базе знаний отсутствует сущность, подходящая для данного термина, то алгоритм свяжет термин с сущностью, которая имеет максимальную семантическую близость с термином. Такой подход может ухудшать точность алгоритма, поскольку термин будет связываться с некорректной сущностью. Одним из решений является добавление условия эмпирически подобранного порогового значения семантической близости: если семантическая близость между термином и сущностью ниже порогового значения, то алгоритм не должен связывать термин с сущностью. В рамках будущих исследований планируется провести эксперименты по подбору порогового значения.

5. Связывание медицинских терминов с MeSH

5.1. Структура MeSH

Medical Subject Headings[7] (MeSH) – словарь медицинских предметных рубрик, созданный и поддерживаемый Национальной медицинской библиотекой США. Основным справочником MeSH является список рубрик (или дескрипторов), имеющий иерархическую структуру. Имя дескриптора и его уникальный идентификатор являются одними из основных понятий, по которому проводится поиск, индексация результатов поиска и связывание сущностей словаря со справочной информацией различного направления и хранения (как в самом словаре, так и в связанных с ним базах данных), а также с научными публикациями в системах PUBMED и MEDLINE. В данной работе

[7] https://www.nlm.nih.gov/mesh/meshhome.html

используется англоязычная версия MeSH как наиболее полная (на 2025 год существует более 30 тысяч рубрик).

Medical Subject Headings Resource Description Framework[8] (MeSH RDF), один из инструментов MeSH – это связанное представление данных биомедицинского словаря. MeSH RDF включает в себя загружаемый файл в формате RDF N-Triples, редактор запросов SPARQL, конечную точку API SPARQL и интерфейс RESTful для извлечения данных MeSH. Одной из основных компонент этой разветвленной системы является файл связанных сущностей в формате XML, который по названию корневого элемента условно можно назвать файл-Descriptor. Файл и связанная с ним функциональность обновляется каждый год и публикуется в открытом для скачивания и обработки доступе на сайте Национальной Библиотеки США. Сущности внутри файла связаны иерархически в нескольких направлениях.

- Непосредственная связь SeeRelatedList, используется древовидная структура, но без типизации.
- Фармакологические свойства PharmalogicalActionList для лекарственных и связанных с ними сущностей.
- Комбинированный тип EntryCombinationList, используемый для расширения полей или структуры как для текущих, так и для будущих, более разветвленных структур, в других файлах и базах данных.
- TreeNumberList – цифровая классификация/подклассификация сущности.
- Тип квалификационных признаков AllowableQualifiersList. Под квалификационным признаком понимается принадлежность сущности к определенной теме верхнего уровня. В настоящий момент поддерживается свыше 75 тем. Сущность может иметь несколько квалификационных признаков (в среднем, от 10 до 20).
- PreviousIndexingList – тип связи для поддержания ранее создаваемых и устаревших связей.
- ConceptList – наиболее сложная связанная с сущностью структура для более полного описания и связывания с терминами, подструктурами, синонимами, концепциями. Для одной сущности может рассматриваться более чем одна структура Concept.

Помимо древовидных связей, описанных выше, можно производить поиск в других полях XML структуры файла: кодированной иерархической информации, датах и неструктурированной текстовой информации.

[8] https://hhs.github.io/meshrdf/

Поиск на сайте MeSH[9] структурирован в соответствии с описанными иерархическими и текстовыми связями. В частности, если знать значение идентификатора дескриптора DescriptorUI, можно быстро перейти к конкретной записи рубрики Descriptor (например, идентификатор D003081 ссылается на рубрику "*Cold Climate*"). Однако, эта опция удобна лишь для частичного тестирования и использовалась для проверки данных.

5.2. Описание алгоритма

Алгоритм связывания терминов из медицинских текстов с сущностями MeSH, предлагаемый в рамках текущей работы, в настоящее время поддерживает первые три структуры из перечисленных в списке выше (SeeRelatedList, PharmalogicalActionList, EntryCombinationList).

Перед началом работы алгоритма выполняется подготовка исходных данных словаря MeSH для связывания. Названия рубрик MeSH, как отдельные сущности, были разбиты на подстроки (до 5 подстрок на название). Минимальный размер подстроки – одно слово. Таким образом, в ходе работы алгоритма термин из обрабатываемого текста сравнивается с сущностью из MeSH или с частью ее названия, полученной на предварительном этапе разбиения. Заметим, что классы сущностей и сами сущности алгоритм обрабатывает единообразно, в силу особенностей связей между ними в структуре MeSH. Определение структуры рубрик реализовано на языке Java при помощи SAX/DOM – библиотек выделения объектов в файловых структурах.

Непосредственно сам алгоритм состоит из следующих этапов:

1. Начальный набор связей формируется путем сопоставления предварительно размеченных текстов (процесс разметки текстов описан в Разделе 3) с подготовленной информацией из структуры названий рубрик MeSH.
2. Связи, выявленные в п. 1, дополняются соответствующими списками дескрипторов (названиями рубрик) через связанные структуры SeeRelatedList, PharmalogicalActionList, EntryCombinationList.
3. Выполняется сопоставление сущностей с идентификационной информацией MeSH непосредственно в тексте. Одна из компонент алгоритма производит постобработку размеченного текста, в результате чего выявленные сущности преобразуются в HTTP ссылки на разделы сайта MeSH с описанием соответствующих сущностей. На рис. 3 приведен пример текста после добавления

[9] https://meshb.nlm.nih.gov/

HTTP ссылок. При переходе по ссылке к сущности словаря “*disseminated intravascular coagulation*” открывается информация на сайте MeSH (см. рис. 4).

Relevance: A [pathological syndrome|T1|TERM] that develops as a result of impaired [outflow of hepatic bile|T2|TERM] through the biliary tract into the intestine due to mechanical obstacles is called [mechanical jaundice|T4|TERM] ([MJ|T5|TERM]). Hemostasis depends on the full function of the liver, since the synthesis of many [coagulation factors|T8|TERM] occurs in [liver cells|T9|TERM], and [activation products|T10|TERM] occur in cells of the [reticuloendothelial system|T11|TERM] of the liver. Violation of hemostasis directly depends on the severity of [hepatocyte dysfunction|T14|TERM]. Such patients may develop disseminated intravascular coagulation ([DIC|T16|TERM]) and

Рисунок 3 – Пример текста после добавления HTTP ссылок на сайт MeSH

Рисунок 4 – На сайте MeSH доступна информация по связанной с текстом рубрике

5.3. Результаты экспериментов

Для связывания с MeSH использовалась коллекция из 48 аннотаций научных статей по медицинской тематике, которые сами авторы статей перевели с русского языка на английский для своих публикаций. На рисунке 5 представлено распределение терминов в текстах.

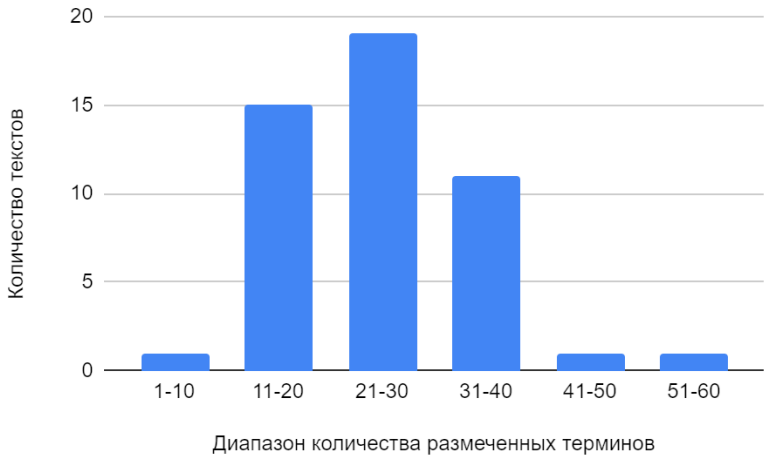

Рисунок 5 – Распределение терминов в исходных текстах

В таблице 2 приведено количество рубрик, полностью или частично соответствующих терминам из текстов, после первого и второго этапов работы алгоритма.

Таблица 2 – Количество терминов из текстов, связанных с рубриками MeSH

Диапазон количества рубрик, связанных с терминами		Количество текстов на первом этапе алгоритма	Количество текстов на втором этапе алгоритма
1-20	1-10	24	16
	11-20	15	
21-40	21-30	5	21
	31-40	3	
41-60	41-50	0	6
	51-60	1	
61-80	–	–	1
80-100	–	–	3
>100	–	–	1
Итого		**48**	**48**

В результате экспериментов по связыванию терминов с MeSH были получены следующие метрик качества: точность — 37.59 %, полнота — 42.85 %, F1-мера — 40.05

%. Основную долю ошибок составляют неоднозначные термины, задаваемые контекстом. Например, термину “*anxiety*” могут быть сопоставлены сущности MeSH “*D001009|Anxiety, Castration*” и “*D001010|Anxiety, Separation*”. В будущем планируется провести эксперименты по влиянию различных мер близости для более релевантного ранжирования.

Кроме того, предполагается добавить обработку точных атрибутов списка ConceptList и подключить логику отношений между выявленными терминами в исходных текстах. Например, через отношение SeeRelatedList рубрика “*D009691|Nucleic Acid Denaturation*” связана с рубрикой “*D020871|RNA Stability*”. Последняя, в свою очередь, через такую же связь соединена с рубрикой “*D059367|RNA Cleavage*”. Рассмотрим случай, когда рубрика “*D009691|Nucleic Acid Denaturation*” найдена среди терминов, размеченных в анализируемом тексте, а две последние рубрики напрямую могут быть там не упомянуты. Однако если в тексте встречается термин “*RNA*”, то его можно связать с одной из рубрик “*D020871|RNA Stability*” или “*D059367|RNA Cleavage*”. В этом случае алгоритм будет не только использовать иерархические связи рубрик (SeeRelatedList, PharmacologicalActionList, EntryCombinationList), которые извлекаются непосредственно из структуры MeSH, но и семантические отношения между терминами в тексте такие как синонимия, гипонимия, меронимия, аппликация и пр.

Следует отметить, что предложенный алгоритм имеет некоторые ограничения. Во-первых, на вход алгоритму предполагается подавать тексты на грамотном английском языке, без ошибок и опечаток. Во-вторых, алгоритм ориентирован на работу с устоявшейся терминологией, и исходя из того, что база знаний MeSH регулярно обновляется, а данные в ней проходят ежегодную ручную верификацию, то есть новые термины будут учтены в работе алгоритма с небольшим опозданием. Тем не менее, разработанный алгоритм для медицинских текстов на английском языке в дальнейшем может быть относительно просто адаптирован для русского языка при наличии довольно полной регулярно обновляемой медицинской терминологической базы.

6. Заключение

В статье исследована задача связывания терминов из научных текстов с сущностями баз знаний. Предложенные оригинальные алгоритмы позволяют автоматически сопоставлять упоминаемые термины в текстах на русском языке с сущностями Википедии и термины в текстах, переведенных на английский язык, с сущностями англоязычной версии MeSH. Связывание с Википедией показало высокую полноту (77.18%) и умеренную точность (37.34%) даже в условиях ограниченного количества данных.

Увеличение контекстного окна не всегда ведет к значительному росту качества, а в некоторых случаях даже вносит шум. Связывание с MeSH дало сбалансированные, но сравнительно низкие метрики (F1 = 40.05%). Анализ ошибок выявил, что основные трудности связаны с многозначностью терминов.

Использование переводов текстов на английский и связывание с англоязычной версией MeSH частично решают поставленную задачу, но подчеркивают необходимость развития специализированных русскоязычных баз знаний по аналогии с MeSH. Перспективным направлением является применение мультиязычных моделей для кросс-лингвистического связывания, что особенно актуально для редких терминов, а также разработка методов автоматического пополнения баз знаний новыми терминологическими записями для отсутствующих сущностей. Полученные результаты могут применяться при создании интеллектуальных систем анализа научных текстов и автоматизированных научных ассистентов, что особенно важно для узких предметных областей.

Список литературы

1. Sevgili Ö., Shelmanov A., Arkhipov M., Panchenko A., Biemann C. Neural entity linking: A survey of models based on deep learning. Semantic Web, 2022. N 13(3), pp. 527–570.
2. Vollmers D., Zahera H., Moussallem D., Ngomo A.-C.N. Contextual Augmentation for Entity Linking using Large Language Models. Proceedings of the 31st International Conference on Computational Linguistics. Association for Computational Linguistics, 2025. pp. 8535–8545.
3. Botha J.A., Shan Z., Gillick D. Entity Linking in 100 Languages. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, 2020. pp. 7833–7845.
4. Orlando R., Cabot P.-L.H., Barba E., Navigli R. ReLiK: Retrieve and LinK, Fast and Accurate Entity Linking and Relation Extraction on an Academic Budget. Findings of the Association for Computational Linguistics: ACL 2024. Association for Computational Linguistics, 2024. pp. 14114–14132.
5. Wu L., Petroni F., Josifoski M., Riedel S., Zettlemoyer L. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. 2020. pp. 6397–6407.
6. Loukachevitch N., Artemova E., Batura T., Braslavski P., Ivanov V., Manandhar S., Pugachev A., Rozhkov I., Shelmanov A., Tutubalina E., Yandutov A. NEREL: a Russian information extraction dataset with rich annotation for nested entities, relations, and

wikidata entity links. Language Resources and Evaluation. 2024. V. 58, pp. 547–583. https://doi.org/10.1007/s10579-023-09674-z

7. Мезенцева А.А., Бручес Е.П., Батура Т.В. Методы и подходы к автоматическому связыванию сущностей на русском языке. Труды Института системного программирования РАН. 2022. Т. 34, № 4. С. 187–200. DOI 10.15514/ISPRAS-2022-34(4)-13. – EDN TEYGLE.
8. Bunescu R., Pasca M. Using Encyclopedic Knowledge for Named Entity Disambiguation. EACL, 2006. pp. 9–16.
9. Cucerzan S. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. EMNLP-CoNLL, 2007. pp. 708–716.
10. Zhang W., Tan C.L., Sim Y.C., Su J. NUS-I2R: Learning a Combined System for Entity Linking. TAC. 2010. https://tac.nist.gov/publications/2010/participant.papers/NUSchime.proceedings.pdf
11. Zheng Z., Li F., Huang M., Zhu X. Learning to link entities with knowledge base. Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. 2010. pp. 483–491.
12. Han X., Zhao J. NLPR_KBP in TAC 2009 KBP Track: A Two-Stage Method to Entity Linking. TAC, 2009. https://tac.nist.gov/publications/2009/participant.papers/NLPR_KBP.proceedings.pdf
13. Ganea O. E., Hofmann T. Deep Joint Entity Disambiguation with Local Neural Attention. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 2017. pp. 2619–2629.
14. Fang Z., Cao Y., Li R., Zhang Z., Liu Y., Wang S. High quality candidate generation and sequential graph attention network for entity linking. Proceedings of The Web Conference 2020. 2020. pp. 640–650.
15. Monahan S., Lehmann J., Nyberg T., Plymale J., Jung A. Cross-Lingual Cross-Document Coreference with Entity Linking. TAC, 2011. https://tac.nist.gov/publications/2011/participant.papers/lcc.proceedings.pdf
16. Dredze M., McNamee P., Rao D., Gerber A., Finin T. Entity disambiguation for knowledge base population. Proceedings of the 23rd international conference on computational linguistics. 2010. pp. 277–285.
17. Cornolti M., Ferragina P., Ciaramita M., Schütze H., Rüd S. The SMAPH system for query entity recognition and disambiguation //Proceedings of the first international workshop on Entity recognition & disambiguation. 2014. pp. 25–30.

18. Le P., Titov I. Distant Learning for Entity Linking with Automatic Noise Detection. 57th Annual Meeting of the Association for Computational Linguistics. ACL Anthology, 2019. pp. 4081–4090.
19. Logeswaran L., Chang M.W., Lee K., Toutanova K., Devlin J., Lee H. Zero-Shot Entity Linking by Reading Entity Descriptions. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2019. pp. 3449–3460.
20. Wu L., Petroni F., Josifoski M., Riedel S., Zettlemoyer L. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, 2020. pp. 6397–6407.
21. Kolitsas N., Ganea O. E., Hofmann T. End-to-End Neural Entity Linking. Proceedings of the 22nd Conference on Computational Natural Language Learning. Association for Computational Linguistics, 2018. pp. 519–529.
22. Ayoola T., Tyagi Sh., Fisher J., Christodoulopoulos Ch., Pierleoni A. ReFinED: An Efficient Zero-shot-capable Approach to End-to-End Entity Linking. Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track. Association for Computational Linguistics, 2022. pp. 209–220.
23. Francis-Landau M., Durrett G., Klein D. Capturing Semantic Similarity for Entity Linking with Convolutional Neural Networks. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016. pp. 1256–1261.
24. Yang X. et al. Learning Dynamic Context Augmentation for Global Entity Linking. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019. pp. 271–281.
25. Mao Yu., Lu Zh. MeSH now: Automatic MeSH indexing at PUBMED scale via learning to rank. Journal of biomedical semantics, 2017. N 8(1), pp. 1–9.
26. Balogh S.G., Zagyva D., Pollner P., Palla G. Time evolution of the hierarchical networks between PUBMED MeSH terms. PLOS ONE, 2019. N 14(8), pp. 1–19.
27. Tsuyuzaki K., Morota G., Ishii M., Nakazato T., Miyazaki S., Nikaido I. MeSH ORA framework: R/bioconductor packages to support MeSH over-representation analysis. BMC bioinformatics, 2015. N 16(1), pp. 1–17.

28. McHugh M. L. Interrater reliability: the kappa statistic. Biochemia medica, 2012. 22(3). pp. 276–282.
29. Pennington J., Socher R., Manning C.D. Glove: Global vectors for word representation. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014. pp. 1532–1543.