Труды Кольского научного центра РАН. Серия: Технические науки. 2023. Т. 14, № 7. С. 5–15.
Transactions of the Kola Science Centre of RAS. Series: Engineering Sciences. 2023. Vol. 14, No. 7. P. 5–15.

Научная статья
УДК 004.853
doi:10.37614/2949-1215.2023.14.7.001

МЕТОД СВЯЗЫВАНИЯ ИМЕНОВАННЫХ СУЩНОСТЕЙ В ТЕКСТЕ С ПОНЯТИЯМИ БАЗЫ ЗНАНИЙ WIKIDATA

Николай Николаевич Тесля[1]✉, Всеволод Дмитриевич Шутюк[2], Владислав Михайлович Жарков[3], Арсений Павлович Витязев[4], Георгий Васильевич Сиповский[5]
[1, 3–5]Санкт-Петербургский федеральный исследовательский центр Российской академии наук, Санкт-Петербург, Россия
[2]ООО «Тинькофф инвестиционные технологии», Санкт-Петербург, Россия
[1]teslya@iias.spb.su✉, https://orcid.org/0000-0003-0619-8620
[2]vsevolod.shutyuk@gmail.com
[3]zharkov201347@gmail.com
[4]vitars2001@yandex.ru
[5]sipovsky2010@yandex.ru

Аннотация

В работе представлен метод для автоматического связывания именованных сущностей в русскоязычных текстах с понятиями из базы знаний Wikidata. В его основе лежит использование инструментов поиска именованных сущностей с последующим семантическим анализом соответствия найденной сущности понятию в базе знаний. Полученные связи в дальнейшем могут быть использованы для формирования связанного корпуса текстов в любой предметной области. Отличием представленного метода от существующих является анализ как самой именованной сущности, так и ее атрибутов, и связанных с ними слов без использования методов машинного обучения. Данный подход позволяет повысить точность поиска соответствующего понятия в базе знаний и снимает необходимость постоянного переобучения нейросетевой модели на распознавание новых сущностей, добавляемых в базу знаний.

Ключевые слова:

именованная сущность, связывание, база знаний, сопоставление

Благодарности:

исследование выполнено в рамках государственного задания Санкт-Петербургского федерального исследовательского центра Российской академии наук FFZF-2023-0001.

Для цитирования:

Тесля Н. Н., Шутюк В. Д., Жарков В. М., Витязев А. П., Сиповский Г. В. Метод связывания именованных сущностей в тексте с понятиями базы знаний Wikidata // Труды Кольского научного центра РАН. Серия: Технические науки. 2023. Т. 14, №. 7. С. 5–15. doi:10.37614/2949-1215.2023.14.7.001.

Original article

METHOD FOR NAMED ENTITIES LINKING WITH CONCEPTS OF THE WIKIDATA KNOWLEDGE BASE

Nikolay N. Teslya[1]✉, Vsevolod D. Shutiuk[2], Vladislav M. Zharkov[3], Arseny P. Vityazev[4], Georgiy V. Sipovsky[5]
[1, 3–5]Saint-Petrsburg Federal research center of the Russian academy of sciences, St. Petersburg, Russia
[2]LLC TINKOFF INVESTMENT TECHNOLOGIES
[1]teslya@iias.spb.su✉, https://orcid.org/0000-0003-0619-8620
[2]vsevolod.shutyuk@gmail.com
[3]Zharkov.V@iias.spb.su
[4]Vityazev.A@iias.spb.su
[5]Sipovskij.G@iias.spb.su

Abstract

The paper presents a method for automatically linking named entities in Russian-language texts with concepts from the Wikidata knowledge base. It is based on the use of named entity search tools with subsequent semantic analysis of the correspondence of the found entity to the concept in the knowledge base. The resulting links can later be used to form a linked corpus of texts in any subject area. The difference between the presented method

and existing ones is the analysis of both the named entity itself, its attributes, as well as associated words without the use of machine learning methods. This approach allows to increase the accuracy of searching for a corresponding concept in the knowledge base and eliminates the need to constantly retrain the neural network model to recognize new entities added to the knowledge base.

Keywords:

named entity, linking, knowledge base, mapping

Acknowledgments:

the study was carried out within the framework of the state research of the St. Petersburg Federal Research Center of the Russian Academy of Sciences FFZF-2023-0001.

For citation:

Teslya N. N., Shutiuk V. D., Zharkov V. M., Vityazev A. P., Sipovskii G. V. Method for named entities linking with concepts of the Wikidata knowledge base // Transactions of the Kola Science Centre of RAS. Series: Engineering Sciences. 2023. Vol. 14, No. 7. P. 5–15. doi:10.37614/2949-1215.2023.14.7.001.

Введение

В настоящий момент многие исследования направлены на разработку различных средств обработки текстов на естественном языке [1]. Одной из таких задач является поиск и классификация определенных объектов — сущностей (Named Entity Recognition — NER). В рамках этой задачи заранее производится выбор заданного набора классов (тегов), по которым классифицируются отдельные слова текста, которые и являются искомыми сущностями. Программные модули, осуществляющие поиск именованных сущностей, используются в различных сферах: от финтех-компаний и банков до медицинских организаций. Более широкой задачей является определение идентичности найденных именованных сущностей. В данном случае говорится о связывании именованных сущностей (Named Entity Linking — NEL) [2], суть которого состоит в том, чтобы не только найти сущность в тексте, но и определить, какому понятию из базы знаний она соответствует.

Обе задачи успешно решены для английского языка, однако специфика русского языка не позволяет напрямую использовать эти решения. Для решения задачи NER существуют правила и модели, которые позволяют с высокой точностью находить в текстах ограниченное количество типов сущностей (обычно людей, мест и организаций). Эти модели чаще всего создаются для одного языка, но есть и многоязычные модели, которые также работают с русским языком. Следует учитывать, что из-за специфики обучения базовая точность мультиязычных моделей ниже, чем моноязычных (до 80 % в мультиязычном варианте, против 90 % в моноязычном). В свою очередь, решения задачи NEL разработаны только для одного языка, требуют значительный объем данных для обучения и не могут работать с сущностями, не представленными в обучающем наборе данных [3].

Целью настоящей работы является разработка метода, осуществляющего поиск именованных сущностей в текстах на русском языке и связывающего найденные сущности с понятиями в базе знаний Wikidata. Результатом работы метода является список именованных сущностей, отметка об их принадлежности определенному классу и ссылки на объекты базы знаний. Апробация проводится на справочных текстах о творчестве А. С. Пушкина, размещенных на ресурсе фундаментальной электронной библиотеки «Русская литература и фольклор» [4].

Обзор связанных работ

Задачу связывания именованных сущностей можно условно разделить на две подзадачи. Первой является поиск именованных сущностей в тексте, а второй — поиск соответствующего объекта в базе знаний. В разделе рассмотрены работы, соответствующие этим подзадачам.

Существует множество подходов к решению проблемы NER. В обзоре представлены наиболее часто используемые [5]: анализ грамматики на основе правил и применение моделей нейронных сетей с глубоким обучением.

Подход к анализу грамматики включает методы, использующие набор правил и словарей, которые создаются экспертами вручную. Методы, основанные на этом подходе, дают наиболее точный результат поиска именованных сущностей. Но в то же время они сложны в реализации, поскольку

требуют больших трудозатрат на создание и поддержание актуальности набора правил и словарей. Среди инструментов, использующих данных подход к анализу грамматики русского языка — Яндекс Томита (tomita-parser) [6] и Yargy-parser (часть проекта Natasha) [7].

Модели нейронной сети с глубоким обучением, способны самостоятельно обнаруживать признаки, идентифицирующие именованную сущность в тексте, и выявлять закономерности для классификации найденных сущностей в заданные классы. Для решения задач обработки естественного языка с применением глубокого обучения была разработана модель BERT со встроенным энкодером и декодером. Эта модель основана на технологии преобразователей (трансформеров), которая в процессе обучения составляет модель зависимости между токенами в предложении. Это помогает снизить влияние контекста обучающей выборки на результат формирования модели. Преимуществами в данном случае являются высокая точность идентификации и классификации именованных сущностей и низкое влияние контекста обучающей выборки на формирование модели. Недостатками являются ее высокая вычислительная сложность, требующая использования соответствующего оборудования, а также сложность отношений, на которых работает модель. Эти связи не могут быть идентифицированы человеком, что усложняет процесс отладки при возникновении ложных срабатываний.

Основными программными библиотеками, предоставляющими функции для решения задачи NER, являются Slovnet [8], DeepPavlov [9] и SpaCy [10]. Все упомянутые библиотеки используют предварительно обученные языковые модели для русского языка на основе BERT, что делает их универсальными. Точность работы с текстом напрямую зависит от используемой модели.

Связывание именованных сущностей (NEL). Для решения задачи NEL также существует несколько подходов [11–13]. В частности, программные библиотеки SpaCy и DeepPavlov, помимо возможности поиска именованных сущностей, предоставляют инструменты для решения задачи NEL. В библиотеку SpaCy для этого включены два компонента:

- EntityLinker обеспечивает поиск именованных сущностей из исходного текста. Для этого требуется использование языковых моделей и моделей поиска именованных сущностей, предоставляемых только библиотекой SpaCy. В качестве базы знаний используется Wikidata. Стоит отметить, что данные из базы знаний получаются не в режиме реального времени, а из снимка данных Wikidata, при этом данные по каждой сущности хранятся только на английском языке. В связи с этим для решения задачи NEL для русского языка потребуется предварительный перевод сущности на английский язык, что может снизить точность полученного результата;
- компонент KnowledgeBase можно использовать для создания собственной базы знаний на основе компонентов Wikidata и использования обучения с учителем для создания классификатора. Однако для этого требуется создание довольно большого набора данных.

Оба компонента в SpaCy требуют предварительно обученной модели для NEL. Модель может быть предварительно обучена пользователем. На момент исследования в репозитарии проекта отсутствовала обученная модель для задачи NEL в русском языке.

В библиотеке DeepPavlov задача NEL является частью программного модуля ответов на вопросы по базе знаний (KBQA) [14], с помощью которого части текста могут быть связаны с понятиями из Wikidata, а также могут формироваться семантические запросы на естественном языке (в том числе и на русском) к базе знаний.

Чтобы эффективно связать именованную сущность с концепцией базы знаний, необходимо решить следующие задачи:

- обеспечить выбор из нескольких вариантов названия или имени именованного объекта. Например, для сущности «А. С. Пушкин» в тексте возможны несколько вариантов представления: Александр Пушкин, Пушкин, Александр Сергеевич Пушкин, А. Пушкин и т. д.;
- решить неоднозначность в определении понятия базы знаний, к которому принадлежит названная сущность. Например, сущность «Пушкин» может относиться к понятиям: «город»», «поэт», «астероид»;
- определить действия в случае отсутствия понятия именованной сущности в базе знаний.

Использование баз знаний. Предлагаемый в данной статье метод основан на поиске понятий в базе знаний для обработки неизвестных сущностей из текста. Поэтому база знаний должна быть универсальной для выполнения связывания. Лучшим решением в этом случае будет использование базы знаний на основе «Википедии», поскольку она содержит максимальное количество информации по многим объектам мира. Крупнейшими базами знаний, основанными на «Википедии», являются Wikidata и DBPedia.

DBPedia — одна из самых популярных баз знаний общего назначения. Она содержит информацию о более чем 5 млн объектов, полученных из Wikipedia. DBPedia хранит и представляет информацию в виде графа знаний RDF. Wikidata, как и DBPedia, хранят данные в виде графа и являются крупнейшей общедоступной базой знаний общего назначения. По состоянию на 2023 г. Wikidata содержит около 106 млн объектов. Поскольку DBPedia основана на RDF-графе, информация о любой сущности представлена как совокупность связей этой сущности с другими сущностями или литералами посредством свойств (предикатов), остальные элементы DBPedia могут предоставить только вторичную информацию об объекте, которую невозможно использовать для решения проблемных задач. В отличие от DBPedia, граф знаний Wikidata имеет свою, более сложную, модель данных, которая не основана на RDF-графе, но в то же время может быть сведена к нему. Каждая сущность в Wikidata, независимо от типа, состоит из следующих основных компонентов:

- название (label) — каноническое имя элемента на разных языках;
- описание (description) — текстовое описание предмета на разных языках;
- псевдоним (alias) — набор вариантов названия или названия предмета на разных языках;
- выражение (statement) — это сложные объекты, содержащие как литералы, так и объекты, описывающие отношения между рассматриваемым элементом и связанными элементами;
- идентификаторы (identificator) — набор идентификаторов элементов в других базах знаний или ресурсах.

Описание метода NEL для связывания с понятиями Wikidata

Предлагаемый метод поиска и связи именованных сущностей с их понятиями в графе знаний Wikidata можно разделить на два этапа:

1) поиск именованных объектов в тексте с использованием предварительно обученной модели NER;
2) связывание найденных названных объектов с их концепциями в Wikidata.

Предварительным этапом работы метода является обработка текста, в ходе которой удаляются некорректные символы и текст разделяется на предложения. Для сегментации на предложения используется библиотека *razdel* [15], в основе которой лежит система правил, разработанная специально для русского языка. Рассмотрим каждый из этапов более подробно.

Поиск именованных объектов для текстов на русском языке. Для каждого предложения модель NER возвращает список токенов и список меток в формате BIO с классом, присвоенным каждому токену. Набор полученных именованных сущностей дополняется соответствующими леммами для обеспечения более высокой эффективности поиска понятий в базе знаний. Для лемматизации именованных сущностей используется библиотека *pymystem3* [16].

Связывание именованных объектов с понятиями графа знаний Wikidata. Алгоритм связывания именованных объектов с понятиями графа знаний Wikidata состоит из трех основных этапов: 1) поиск понятия в базе знаний; 2) фильтрация результатов поиска; 3) выбор концепции для именованного объекта.

Поиск понятия в базе знаний. Алгоритм выполняется последовательно для каждой уникальной леммы именованного объекта (рис. 1). Лемма используется в качестве параметра поискового запроса в графе знаний Wikidata. Поиск осуществляется по тегам и синонимам с использованием API Wikimedia, для этого используется параметр *wbsearcentities*. Поисковый запрос возвращает 10 наиболее релевантных результатов.

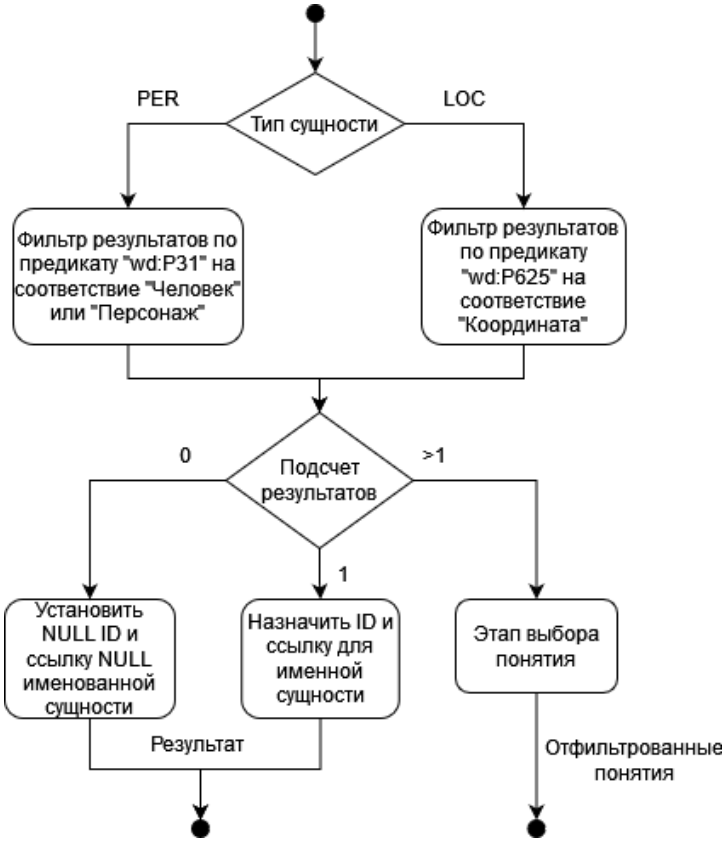

Рис. 1. Поиск понятия в базе знаний Wikidata

Параллельно осуществляется полнотекстовый поиск по всему графу Wikidata с использованием API Wikimedia. Запрос выполняется с параметром *query*. Результаты добавляются в список идентификаторов, полученных из Wikidata. Этап полнотекстового поиска обусловлен недостаточной точностью поиска тегов и синонимов, поскольку их наличие графе знаний Wikidata не гарантируется.

Далее формируется общий список идентификаторов из поиска путем объединения двух списков, полученных на предыдущих шагах, с последующим удалением дубликатов. Данный список проверяется на количество элементов. Если в списке нет элементов и если поисковый запрос представлял собой лемму именованной сущности, тогда считается, что поисковый запрос определяет саму именованную сущность, и все предыдущие шаги повторяются снова. Если лемма именованной сущности определена неверно, то корректный результат может быть получен при попытке поиска по полной строке именованной сущности. В противном случае, если поисковый запрос и так был строкой именованной сущности, а ни один из этапов поиска не вернул результатов, тогда считается, что понятие в базе знаний отсутствует, и сущности присваивается нулевая ссылка.

Если же список содержит только один элемент, то именованной сущности присваивается идентификатор этого элемента и формируется ссылка на Wikidata.

Фильтрация результатов поиска. Если количество элементов в списке больше одного, то в зависимости от класса именованной сущности осуществляется фильтрация (рис. 2). Если именованная сущность принадлежит классу *LOC* («Локация»), то используются фильтры на основе того, имеет ли элемент Wikidata предикат «координаты» (P625). Если же именованная сущность принадлежит классу *PER* («Персоналия»), то генерируется отфильтрованный список идентификаторов сущностей Wikidata. Отфильтрованный список содержит идентификаторы только тех сущностей, значение предиката «частный случай понятия» (P31) у которых имеет хотя бы одну связь с элементом «человек» или «персонаж».

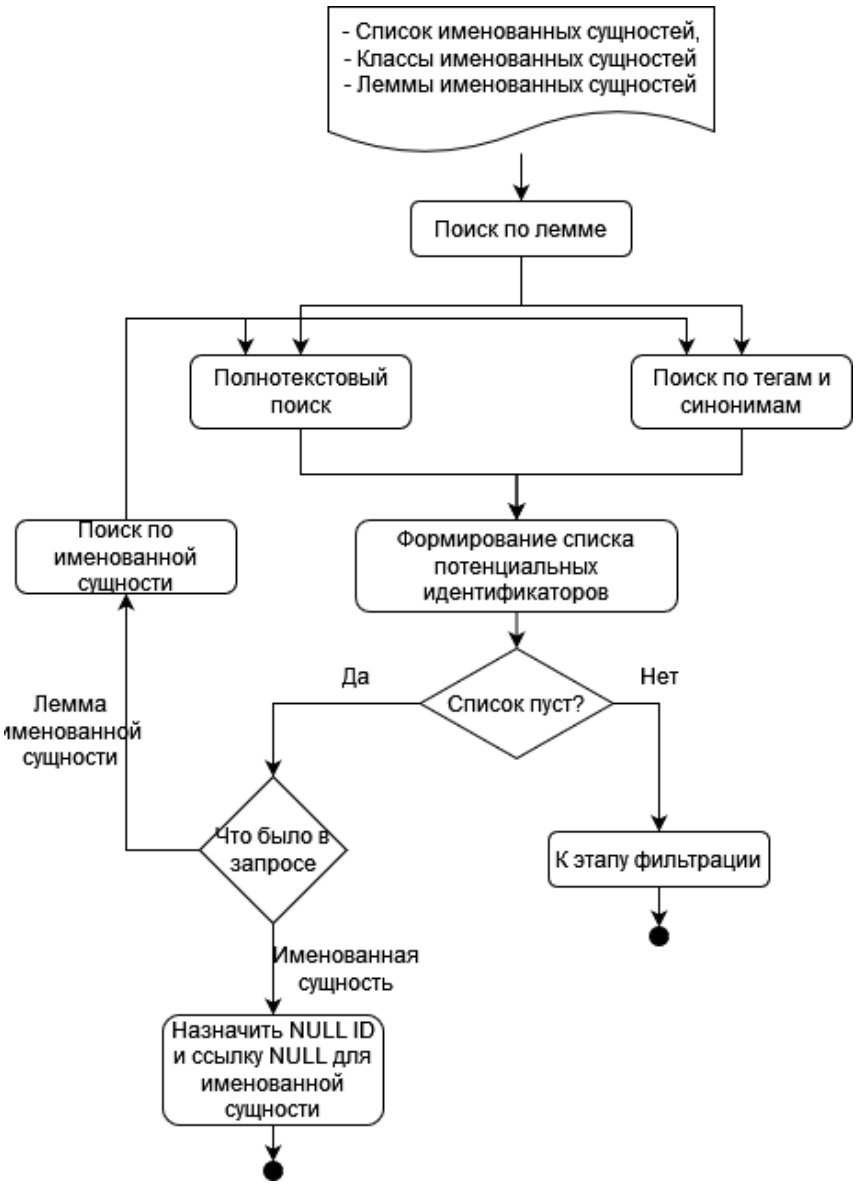

Рис. 2. Этап фильтрации результатов поиска

После получения отфильтрованного списка идентификаторов анализируется количество элементов в этом списке. Если список пуст, именованной сущности присваивается нулевой идентификатор и нулевая ссылка. Если список содержит только один элемент, именованной сущности присваивается его идентификатор и формируется ссылка на Wikidata. Если список содержит более одного элемента, в зависимости от типа именованной сущности алгоритм переходит к этапу выбора понятия для именованной сущности.

Выбор концепции для именованного объекта. На этом этапе используется алгоритм выбора идентификатора (рис. 3). Если именованная сущность принадлежит классу *LOC* («Локация»), то для каждого отфильтрованного идентификатора идентификаторы связанных с ним элементов Wikidata получаются посредством предиката «особый случай понятия» (P31), связанного с географическими объектами такими, как «Страна» (Q6256), «Город» (Q515, Q15253706), «Континент» (Q5107), «Море» (Q165) и др.

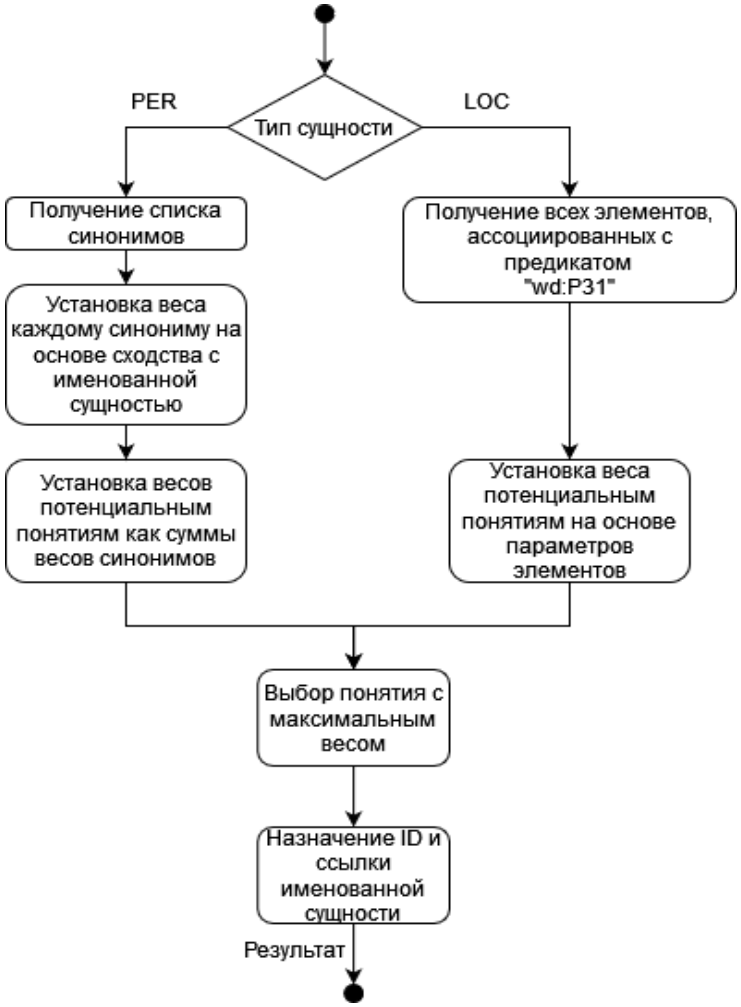

Рис. 3. Этап выбора понятия для именованного объекта

Если именованная сущность принадлежит классу *PER* («Персоналия»), то для каждого фильтруемого идентификатора получаются синонимы на русском языке. Затем для каждого идентификатора определяется вес как сумма весов синонимов. Для каждого синонима рассчитывается его сходство в процентах с поисковым запросом. Для этого используется функция *token_set_ratio* из библиотеки *thefuzz*, которая рассчитывает процент сходства двух строк по токенам. В зависимости от значения сходства вес синонима определяется следующим образом:

- если сходство между поисковым запросом и синонимом меньше или равно 75, вес не присваивается;
- если сходство между поисковым запросом и синонимом больше 75, присваивается вес 1;
- далее в списке определяется первое вхождение элемента с максимальным весом;
- полученный идентификатор присваивается именованной сущности и формируется ссылка на Wikidata.

Оценка метода связывания именованных сущностей с понятиями Wikidata

Разработанный метод был апробирован на выборке текстов различной тематики. В данную выборку вошли как тексты из фундаментальной электронной библиотеки, связанные с Александром Пушкиным, так и тексты на другую тематику, содержащие именованные сущности классов «Персоналия» (*PER*) и «Локация» (*LOC*).

Чтобы представить точность разработанного метода, в табл. 1 показаны результаты тестирования на 5 текстах:

- текст 1: статья И. С. Чистовой по стихотворению А. С. Пушкина из письма Вяземскому [17];
- текст 2: текстовая часть раздела «География» из статьи «Википедии» о Южной Америке;
- текст 3: фрагмент статьи о Российской империи с сайта Государственного исторического музея [18];
- текст 4: статья Э. В. Кардаш по произведению «Актеон» [17];
- текст 5: статья Я. Л. Левкович по произведению «Будрыс и его сыновья» [17].

Для каждого класса рассчитываются следующие показатели:

- количество найденных объектов — общее количество объектов, найденных моделью NER;
- количество ошибок модели NER — количество найденных объектов с неправильным именем. К ошибкам относятся сущности с неправильно присвоенным классом и ложноположительные результаты модели NER;
- количество действительных ссылок — именованные сущности с действительными ссылками на концепции;
- количество недействительных ссылок — количество именованных объектов с найденными недействительными ссылками;
- количество истинно недостающих ссылок — количество именованных сущностей, у которых нет связей, и в базе знаний для них нет понятий;
- количество ошибочно отсутствующих ссылок — количество именованных сущностей, у которых нет ссылок, но при этом база знаний содержит понятия, описывающие их;
- точность связи именованных сущностей — точность разработанного метода в процентах, рассчитываемая как $\frac{TRL+TRML}{EF-E_{\text{NER}}}$, где *TRL* — количество действительных ссылок; *TRML* — количество действительно отсутствующих ссылок; *EF* — количество найденных объектов; E_{NER} — количество ошибок модели NER.

Таблица 1

Результаты тестирования метода связывания именованных сущностей с понятиями Wikidata

Показатель	Текст 1		Текст 2		Текст 3		Текст 4		Текст 5	
	PER	*LOC*	*PER*	*LOC*	*PER*	*LOC*	*PER*	*LOC*	*PER*	*LOC*
Количество ошибок модели NER	3	0	1	2	4	1	2	2	7	0
Количество действительных ссылок	5	6	0	25	11	23	14	1	10	4
Количество недействительных ссылок	2	0	0	3	3	8	1	0	3	0
Количество истинно недостающих ссылок	4	0	0	1	0	0	6	0	3	0
Количество ошибочно отсутствующих ссылок	0	0	0	1	0	0	5	0	0	0
Количество найденных объектов	14	6	1	32	18	32	28	3	23	4
Точность связи именованных сущностей	81%	100%	100%	78%	78%	74%	76%	100%	81%	100%

Для сравнения полученных результатов с существующими, «Текст 1» был обработан модулем DeepPavlov Entity Extractor. Благодаря лучшей модели NER было распознано 48 объектов по типу *PER* и 6 объектов по типу *LOC*. Однако результат NEL для обоих типов объектов равен 50 % (т. е. оценка типа объекта *LOC* составила *TRL* = 3, *TRML* = 0, *EF* = *6*, E_{NER} = 0). Из-за специфики русского языка

было найдено много форм для одних и тех же сущностей, которые не были обработаны при обучении модели (например, «Вяземский», «Вяземского», «Вяземскому»), поэтому модель не могла определить ссылки на сущности, т. к. для модели «Вяземский» и «Вяземскому» рассматриваются как разные сущности.

Заключение

В результате работы реализован метод для поиска связей между именованными сущностями в тексте и понятиями базы знаний Wikidata. Его проверка осуществлялась для классов сущностей *PER* («Персоналия») и *LOC* («Локация»). Связь осуществляется путем присвоения именованной сущности идентификатора понятия в базе знаний и ссылки на страницу этого понятия в Wikidata.

В ходе реализации метода были разработаны алгоритмы поиска именованных сущностей, фильтрации результатов поиска и связи именованных сущностей с понятиями базы знаний. Средняя точность метода на рассматриваемой выборке текстов составляет 90,4 % для типов сущностей *LOC* и 83,2 % для типов сущностей *PER*. Данная оценка превосходит модель NEL, основанную на подходах глубокого обучения. Предлагаемый подход не зависит от языка и может быть масштабирован для других моделей NER и баз знаний. Также не требуется корректировка модели и обучение для расширения списка поддерживаемых объектов.

Список источников

1. Humbel M., Nyhan J., Vlachidis A., Sloan K., Ortolja-Baird A. Named-entity recognition for early modern textual documents: a review of capabilities and challenges with strategies for the future // Journal of Documentation. Emerald Publishing Limited, 2021. Vol. 77, № 6. P. 1223–1247.
2. Al-Moslmi T., Ocaña M. G., Opdahl A. L., Veres C. Named entity extraction for knowledge graphs: A literature overview // IEEE Access. IEEE, 2020. Vol. 8. P. 32862–32881.
3. Shen W., Li Y., Liu Y., Han J., Wang J., Yuan X. Entity linking meets deep learning: Techniques and solutions // IEEE Transactions on Knowledge and Data Engineering. IEEE, 2021.
4. Электронное научное издание «Пушкин» // Фундаментальная электронная библиотека «Русская литература и фольклор». 2002 [Электронный ресурс]. URL: http://feb-web.ru/feb/pushkin/default.asp (дата обращения: 01.10.2023).
5. Nasar Z., Jaffry S.W., Malik M. K. Named entity recognition and relation extraction: State-of-the-art // ACM Computing Surveys (CSUR). ACM New York, NY, USA, 2021. Vol. 54, № 1. P. 1–39.
6. GitHub — yandex/tomita-parser [Электронный ресурс]. URL: https://github.com/yandex/tomita-parser/ (дата обращения: 18.09.2023).
7. GitHub — natasha/yargy: Rule-based facts extraction for Russian language [Электронный ресурс]. URL: https://github.com/natasha/yargy (accessed: 18.09.2023).
8. GitHub — natasha/slovnet: Deep Learning based NLP modeling for Russian language [Электронный ресурс]. URL: https://github.com/natasha/slovnet (дата обращения: 18.09.2023).
9. Burtsev M. et al. Deeppavlov: Open-source library for dialogue systems // Proceedings of ACL 2018, System Demonstrations. 2018. P. 122–127.
10. Honnibal M., Montani I., Van Landeghem S., Boyd A. spaCy: Industrial-strength natural language processing in python. Zenodo, Honolulu, HI, USA, 2020.
11. Hachey B., Radford W., Curran J. R. Graph-based named entity linking with wikipedia // Web Information System Engineering-WISE 2011: 12th International Conference, Sydney, Australia, October 13–14, 2011. Proceedings 12. 2011. P. 213–226.
12. Čuljak M., Spitz A., West R., Arora A. Strong heuristics for named entity linking // arXiv preprint arXiv:2207.02824. 2022.
13. Tamper M., Oksanen A., Tuominen J., Hietanen A., Hyvönen E. Automatic annotation service appi: Named entity linking in legal domain // European Semantic Web Conference. 2020. P. 208–213.
14. Knowledge Base Question Answering (KBQA) — DeepPavlov 1.3.0 documentation [Электронный ресурс]. URL: https://docs.deeppavlov.ai/en/master/features/models/kbqa.html (дата обращения: 18.09.2023).

15. GitHub — natasha/razdel: Rule-based token, sentence segmentation for Russian language [Электронный ресурс]. URL: https://github.com/natasha/razdel (дата обращения: 18.09.2023).
16. Segalovich I. A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine // Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications. 2003. P. 273–280.
17. Пушкинская энциклопедия: Произведения. Вып. 1. А–Д. СПб.: Нестор-История, 2009. 520 с.
18. История России | История Российской империи с древних времен и по наши дни // Государственный исторический музей [Электронный ресурс]. URL: https://shm.ru/articles/istoriya-rossii/#7 (дата обращения: 18.09.2023).

References

1. Humbel M., Nyhan J., Vlachidis A., Sloan K., Ortolja-Baird A. Named-entity recognition for early modern textual documents: a review of capabilities and challenges with strategies for the future. *Journal of Documentation*. Emerald Publishing Limited, 2021, vol. 77, no. 6, pp. 1223–1247.
2. Al-Moslmi T., Ocaña M. G., Opdahl A. L., Veres C. Named entity extraction for knowledge graphs: A literature overview. *IEEE Access.* IEEE, 2020, vol. 8, pp. 32862–32881.
3. Shen W., Li Y., Liu Y., Han J., Wang J., Yuan X. Entity linking meets deep learning: Techniques and solutions. *IEEE Transactions on Knowledge and Data Engineering*. IEEE, 2021.
4. *Elektronnoe nauchnoe izdanie "Pushkin"* [Digital scientific edition “Pushkin”]. *Fundamental'naya elektronnaya biblioteka “Russkaya literatura i fol'klor”* [Fundamental Digital Library. “Russian literature and folklore”], 2002. (In Russ.). Available at: http://feb-web.ru/feb/pushkin/default.asp (accessed: 01.10.2023).
5. Nasar Z., Jaffry S. W., Malik M.K. Named entity recognition and relation extraction: State-of-the-art. *ACM Computing Surveys (CSUR)*. ACM New York, NY, USA, 2021, vol. 54, no. 1, pp. 1–39.
6. GitHub — yandex/tomita-parser. Available at: https://github.com/yandex/tomita-parser/ (accessed: 18.09.2023).
7. GitHub — natasha/yargy: Rule-based facts extraction for Russian language. Available at: https://github.com/natasha/yargy (accessed: 18.09.2023).
8. GitHub — natasha/slovnet: Deep Learning based NLP modeling for Russian language. Available at: https://github.com/natasha/slovnet (accessed: 18.09.2023).
9. Burtsev M. et al. Deeppavlov: Open-source library for dialogue systems. *Proceedings of ACL 2018*, System Demonstrations, 2018, pp. 122–127.
10. Honnibal M., Montani I., Van Landeghem S., Boyd A. spaCy: Industrial-strength natural language processing in python. Zenodo, Honolulu, HI, USA, 2020.
11. Hachey B., Radford W., Curran J. R. Graph-based named entity linking with Wikipedia. *Web Information System Engineering-WISE 2011: 12th International Conference*, Sydney, Australia, October 13–14, 2011. Proceedings 12, 2011, pp. 213–226.
12. Čuljak M., Spitz A., West R., Arora A. Strong heuristics for named entity linking. *arXiv preprint arXiv*:2207.02824, 2022.
13. Tamper M., Oksanen A., Tuominen J., Hietanen A., Hyvönen E. Automatic annotation service appi: Named entity linking in legal domain. *European Semantic Web Conference*, 2020, pp. 208–213.
14. Knowledge Base Question Answering (KBQA) — DeepPavlov 1.3.0 documentation. Available at: https://docs.deeppavlov.ai/en/master/features/models/kbqa.html (accessed: 18.09.2023).
15. GitHub — natasha/razdel: Rule-based token, sentence segmentation for Russian language. Available at: https://github.com/natasha/razdel (accessed: 18.09.2023).
16. Segalovich I. A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications, 2003, pp. 273–280.
17. *Pushkinskaya encyclopedia: Proizvedenya (A–D)* [Pushkin Encyclopedia: Works A–D]. Saint Petersburg, Nestor-Istoria, 2009, vol. 1, 520 p. (In Russ.).

18. Istoriya Rossii | Istoriya Rossijskoj imperii s drevnih vremen i po nashi dni [Russian history | History of the Russian Empire from ancient times to the present day]. *Gosudarstvennyj istoricheskij muzej* [State historical museum]. (In Russ.). Available at: https://shm.ru/articles/istoriya-rossii/#7 (accessed: 18.09.2023).

Информация об авторах

Н. Н. Тесля — кандидат технических наук, старший научный сотрудник;
В. Д. Шутюк — программист;
В. М. Жарков — программист;
А. П. Витязев — программист;
Г. В. Сиповский — младший научный сотрудник.

Information about the authors

N. N. Teslya — Candidate of Science (Tech.), senior researcher;
V. D. Shutiuk — programmer;
V. M. Zharkov — programmer;
A. P. Vityazev — programmer;
G. V. Sipovskii — junior researcher.

Статья поступила в редакцию 13.10.2023; одобрена после рецензирования 01.11.2023; принята к публикации 08.11.2023.
The article was submitted 13.10.2023; approved after reviewing 01.11.2023; accepted for publication 08.11.2023.