Кузьмин А.И. Обзор и сравнение популярных инструментов для обработки естественного языка // Международный журнал информационных технологий и энергоэффективности. 2021– Т. 6 № 4(22) с. 24–33

Международный журнал информационных технологий и энергоэффективности

Сайт журнала:
http://www.openaccessscience.ru/index.php/ijcse/

УДК 681.3.019

ОБЗОР И СРАВНЕНИЕ ПОПУЛЯРНЫХ ИНСТРУМЕНТОВ ДЛЯ ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА

Кузьмин А.И.
Филиал ФГБОУ ВО «Национальный исследовательский университет «МЭИ»в г. Смоленске, Россия,(214013, г. Смоленск, Энергетический проезд), e-mail: timonkyz2@mail.ru

В работе представлено общее сравнение и анализ существующих инструментов для обработки естественного языка. Составлены диаграммы сравнения для более популярных библиотек для работы с естественным языком. Приведены преимущества и недостатки самых популярных инструментов.

Ключевые слова: естественный язык, обработки текста, анализ текста

REVIEW AND COMPARISON OF POPULAR TOOLS FOR NATURAL LANGUAGE PROCESSING

Kuzmin A.I.
Smolensk Branch of the National Research University "Moscow Power Engineering Institute", Smolensk, Russia (214013, Smolensk, Energetichesky proezd, e-mail: timonkyz2@mail.ru

The article presents a general comparison and analysis of existing tools for natural language processing (NLP). Comparison diagrams for the more popular libraries for natural language processing are compiled. The advantages and disadvantages of the most popular tools are given.

Keywords: natural language, text processing, text analysis

Введение

Понятие Data Mining, появившееся в 1978 году, приобрело высокую популярность в современной трактовке примерно с первой половины 1990-х годов. До этого времени обработка и анализ данных осуществлялись в рамках прикладной статистики, при этом в основном решались задачи обработки небольших баз данных.

Data mining (DM) это процесс вычислительного извлечения новой информации из Big Data [1], а различные отрасли генерируют огромные объемы данных, открывая эру "больших данных". Это создает широкие возможности для разработки и внедрения новых алгоритмов интеллектуального анализа. Широкий спектр методов извлечения ценных сведений из различных типов и моделей данных подпадает под понятие "data mining".

Согласно определению в статье "Data clustering: a review", "кластеризация — это классификация шаблонов (наблюдений, элементов данных или векторов признаков) в группы (кластеры) без наблюдения" [2].

Классификация схожа с кластеризацией, поскольку она разделяет данные на группы, называемые классами, но в отличие от кластеризации, анализ классификации требует знания и спецификации того, как определяются эти классы.

Теория статистического обучения стремится "обеспечить основу для изучения проблемы вывода, то есть получения знаний, составления прогнозов, принятия решений или построения моделей на основе набора данных", - утверждает Буске и др. [3].

Заметным переходом, демонстрирующим мощь новых алгоритмов и данных, стало использование подходов интеллектуального анализа данных для изучения не только первичных характеристик, но и характеристик, специфичных для контекста. Например, первоначальные подходы к поиску данных, которые строили одну модель [4]. В отличие от этого, последние подходы изучают множество контекстно-специфических моделей, позволяя строить сети, специфичные для разнообразных процессов [5].

Text mining (TM) — это область интеллектуального анализа данных, целью которой является извлечение новой ценной информации из неструктурированных (или полуструктурированных) источников [6]. Text mining извлекает информацию из документов и агрегирует извлеченные фрагменты по всей коллекции исходных документов получения новой информации. Это предпочтительный взгляд на данную область, который позволяет отличить текстовый майнинг от обработки естественного языка (NLP) [7]. Таким образом, получив на вход набор документов, методы интеллектуального анализа текста стремятся обнаружить новые закономерности, взаимосвязи и тенденции, содержащиеся в документах. В достижении общей цели обнаружения новой информации помогают инструменты NLP, которые варьируются от относительно простых задач обработки текста на лексическом или грамматическом уровнях (таких как токенизация или тегирование части речи) до относительно сложных алгоритмов извлечения информации (таких как распознавание именованных сущностей (NER) для поиска концепций, нормализация для сопоставления их с их уникальными идентификаторами или извлечение отношений и системы анализа настроений). Чем выше сложность задачи, тем больше вероятность интеграции методов интеллектуального анализа данных (таких как классификация или статистическое обучение).

К подобластям text mining, кратко обобщенным, относятся:

- Информационный поиск(IR) занимается проблемой поиска релевантных документов в ответ на конкретную информационную потребность (запрос).
- NER лежит в основе автоматического извлечения информации из текста и занимается проблемой поиска ссылок на объекты (упоминаний) присутствующие в тексте на естественном языке, и их маркировки с указанием местоположения и типа.
- Идентификация именованных сущностей позволяет связать интересующие объекты с информацией, которая не указана в тексте.
- Извлечение ассоциаций - одна из высокоуровневых задач. Она использует результаты предыдущих подзадач для получения списка ассоциаций между различными сущностями, представляющими интерес

Всеобъемлющая проблема анализа текстов заключается в том, чтобы включить многочисленные доступные ресурсы знаний в конвейер NLP. Например, в медицинской области, в отличие от общей области поиска текстов, имеется доступ к большому количеству обширных, хорошо проверенных онтологий и баз знаний. Например, использование онтологий позволило использовать неструктурированные клинические записи для получения

практических данных о безопасности высокоэффективного непатентованного препарата для лечения заболеваний периферических сосудов [8].

Основные этапы обработки естественного языка

Можно выделить общие этапы, которые объединяют обработку текста разными инструментами. Для понимая процесса необходимо подробнее остановиться на первоначальных этапах:

- Стэмминг.
- Лемматизация.
- Тегирование частей речи.

Стэмминг — это процесс сокращения слова до его основы, т.е. корневой формы. Корневая форма не обязательно является словом сама по себе, но она может быть использована для образования слов путем присоединения нужного суффикса.

Например, слова рыба и рыбалка образуются от корня "рыба", что является правильным словом. С другой стороны, слова study, studies и studying превращаются в studi, что не является английским словом.

Чаще всего алгоритмы стемминга (они же стеммеры) основаны на правилах отсечения суффиксов. Наиболее известным примером является стеммер Портера, представленный в 1980-х годах и в настоящее время реализованный в различных языках программирования.

Традиционно поисковые системы и другие приложения применяют стемминг для повышения вероятности совпадения различных форм слова, рассматривая их почти как синонимы, поскольку концептуально они "принадлежат" друг другу.

Цель лемматизации - сгруппировать различные формы слова, называемые леммами. Этот процесс в чем-то схож со стеммингом, поскольку он объединяет несколько слов в один общий корень. Результатом лемматизации является правильное слово, и отсечение суффиксов не даст такого же результата. Например, лемматизатор должен преобразовать gone, going и went в go. Для достижения своей цели лемматизация требует знания контекста слова, поскольку процесс зависит от того, является ли слово существительным, глаголом и т.д. [9]

Метки частей речи (POS) — это процесс отнесения слова к его грамматической категории, чтобы понять его роль в предложении. Традиционными частями речи являются существительные, глаголы, наречия, союзы и т.д. [10].

Теггеры частей речи обычно принимают на вход последовательность слов (т.е. предложение) и выдают на выходе список кортежей, где каждое слово связано с соответствующим тегом.

Тегирование частей речи предоставляет контекстуальную информацию, необходимую лемматизатору для выбора подходящей леммы [11,12].

Сравнение популярных инструментов NLP

Существует множество инструментов и библиотек, предназначенных для решения задач NLP. Краткое сравнение основных инструментов будет представлено далее, но нужно понимать, что все библиотеки, которые рассматриваем, имеют лишь частично пересекающиеся задачи. Поэтому иногда их трудно сравнивать напрямую. Некторые особенности опустим и сравним между собой только те библиотеки, в которых имеется аналогичный функционал.

- *NLTK (Natural Language Toolkit)* используется для решения таких задач, как токенизация, лемматизация, стемминг, синтаксический разбор, POS-тегирование и т.д. В этой библиотеке есть инструменты практически для всех задач NLP.
- *Spacy* является основным конкурентом NLTK. Эти две библиотеки могут использоваться для решения одних и тех же задач.
- *Scikit-learn* предоставляет большую библиотеку для машинного обучения. Здесь также представлены инструменты для предварительной обработки текста.
- *Gensim* - пакет для моделирования тем и векторного пространства, сходства документов.
- Общая задача библиотеки *Pattern* - служить в качестве модуля веб-майнинга. Таким образом, она поддерживает NLP только в качестве побочной задачи.
- *Polyglot* - еще один пакет python для NLP. Он не очень популярен, но также может быть использован для широкого круга задач NLP.
- *UDPipe* - На основании информационных банеов проекта Universal Dependencies создана библиотека UDPipe, позволяющее производить токенизацию, лемматизацию, морфологический анализ, а также строить деревья зависимостей между словами. UDPipe реализовано в виде бесплатных библиотек и пакетов на разных языках программирования, содержащие предварительно обученные языковые модели.

Чтобы сделать сравнение более наглядным, было подготовлена таблица, в которой указаны плюсы и минусы различных инструментов. [13]

Таблица 1 – Преимущества и недостатки библиотек для решения задач NLP

	Преимущества	Недостатки
Natural Language Toolkit	1)Наиболее известная и полная библиотека для работы NLP; 2)Имеется множество сторонних дополнений; 3)Возможность использовать множество подходов к каждой задаче NLP; 4)Быстрая токенизация предложения; 5)Поддержка наибольшего количества языков по сравнению с другими библиотеками;	1)Сложность в изучении и использовании; 2)Довольно медленная обработка по сравнению с другими библиотеками; 3)При токенизации предложений NTLK разбивает текст только на предложения, не анализируя семантику и структуру в целом; 4)Выполняется обработка строк, что не очень характерно для объектно-ориентированного языка Python; 5) Не использует нейронные сети; 6)Отсутствуют интегрированные векторы слов;
spaCy	1)Самый быстрый NLP-фреймворк; 2)Понятный в использовании, потому что для каждой задачи имеется определённый высоко оптимизированный инструмент; 3)Объект процессов более объектно-ориентированный,	1) Менее гибкая по сравнению с NLTK; 2)Токенизация предложений медленнее, чем в NLTK; 3)Поддерживает маленькое количество языков;

	по сравнению с другими библиотеками; 4)Использует нейронные сети для обучения некоторых моделей;	
Scikit-learn NLP toolkit	1) Большое количество алгоритмов для построения моделей; 2) Содержит функции для работы с Bag-of-Words моделью; 3) Имеет подробную документация и интуитивно понятные методы в классах	1)Плохой препроцессинг, что вынуждает использовать ее в связке с другой библиотекой (например, NLTK); 2) Не использует нейронные сети для препроцессинга текста.
Genism	1)Работает с большими датасетами; 2)Поддерживает глубокое обучение; 3)Предоставляет возможность работы с word2vec, tf-idf vectorization, document2vec.	1)Библиотека заточена под модели без учителя; 2)Не содержит достаточного функционала, необходимого для NLP, что вынуждает использовать ее вместе с другими библиотеками;
Pattern	1)Позволяет тегировать части речи, искать n-граммы, анализировать настроения, WordNet, работать с моделью векторного пространства и кластеризацию и SVM; 2)Есть веб-краулер. DOM парсер, некоторые API.	1)Наличие веб-майнера; фреймворк также может быть недостаточно оптимизирован для некоторых специфических задач NLP
Polyglot	1)Поддерживает большое количество языков(16-196 языков для различных задач)	1)Не так популярен, как, например, NLTK или Spacy; 2)Может быть медленным в работе 3)Слабая поддержка сообщества
UDpipe	1)Поддержка более 50 языков 2)Можно обучать собственные модели непосредственно из R 3)Доступны готовые модели для загрузки 4)При тестировании для голландского, французского, испанского, итальянского, португальского языков UDPipe в целом показал себя лучше, чем SpaCy 5)Возможность токенизации тегирования частей речи, тегирования морфологических признаков, лемматизации, парсинга зависимостей	1)Медленее spaCy примерно в 5 раз

Подробное сравнение библиотек UDPipe и spaCy.

Для решение конкретных задач, чаще всего используются две основные библиотеки, которые необходимо рассмотреть подробнее. Для сравнения *UDPipe* и *spaCy* необходимо выделить блоки, по которым можно определить критерии сравнения инструментов [14,15].

Традиционный поток обработки естественного языка состоит из нескольких строительных блоков, которые могут быть использованы для создания на его основе приложения для обработки естественного языка. А именно:

1. Токенизация.
2. Маркировка частей речи.
3. Лемматизация.
4. Маркировка морфологических признаков.
5. Синтаксический разбор зависимостей.
6. Распознавание сущностей.
7. Извлечение смысла слов и предложений.

Можно сравнить данные инструменты по ряду критериев:

- языки, на которых работают инструменты;
- простота использования;
- возможности аннотации;
- точность аннотации моделей;
- скорость аннотации.

Поскольку модели spaCy и UDPipe для испанского, португальского, французского, итальянского и голландского языков были построены на данных из одного и того же древа универсальных зависимостей, можно сравнить точность различных этапов обработки НЛП (токенизация, POS-тегирование, тегирование морфологических признаков, лемматизация, разбор зависимостей)[16].

Оценка традиционно производится путем исключения некоторых предложений из обучающей части и просмотра того, насколько хорошо модель справилась с этими исключенными предложениями, которые были помечены людьми, поэтому их называют "золотыми".

Ниже приведена статистика точности для различных задач NLP с использованием скрипта оценки общих задач Conllu 2021 на удержанных тестовых наборах [17,18]. Эти графики в основном показывают, что:

- UDPipe обеспечивает лучшие результаты для французского, итальянского и португальского языков, равные результаты для испанского языка, менее хорошие результаты для разбора зависимостей и тегов, специфичных для древовидного дерева, для голландского языка, но лучшие результаты для универсальных тегов частей речи.
- Для английского языка можно сравнить только теги XPOS из Penn Treebank, и spaCy показывает менее хорошие результаты, чем мы ожидали, при сравнении с моделью UDPipe.

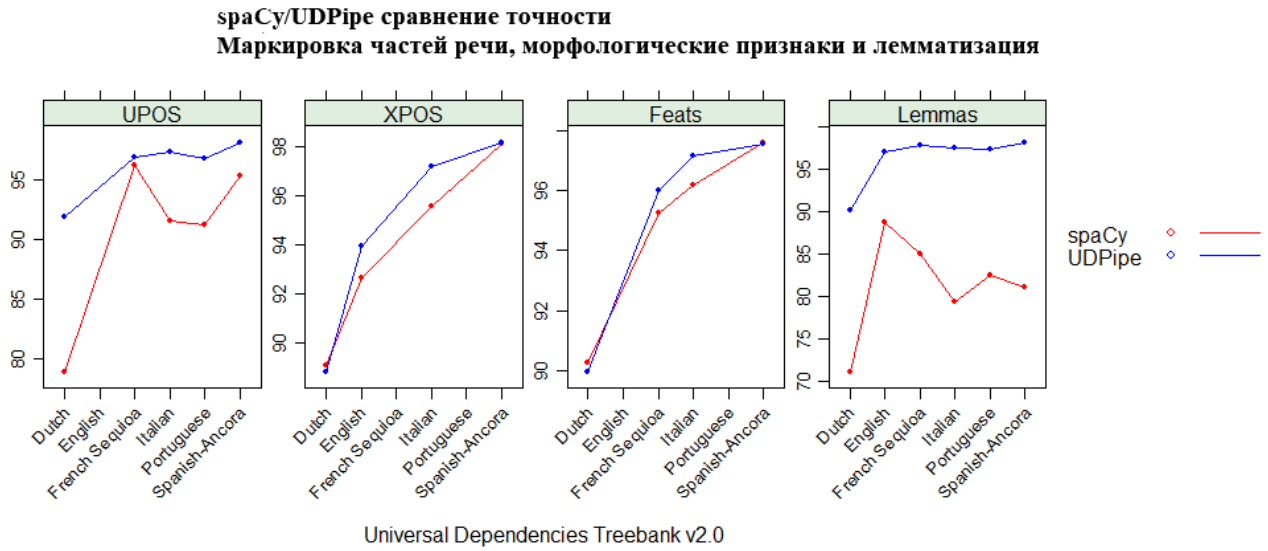

Рисунок 1 – Сравнение точности работы библиотек на разных этапах обработки

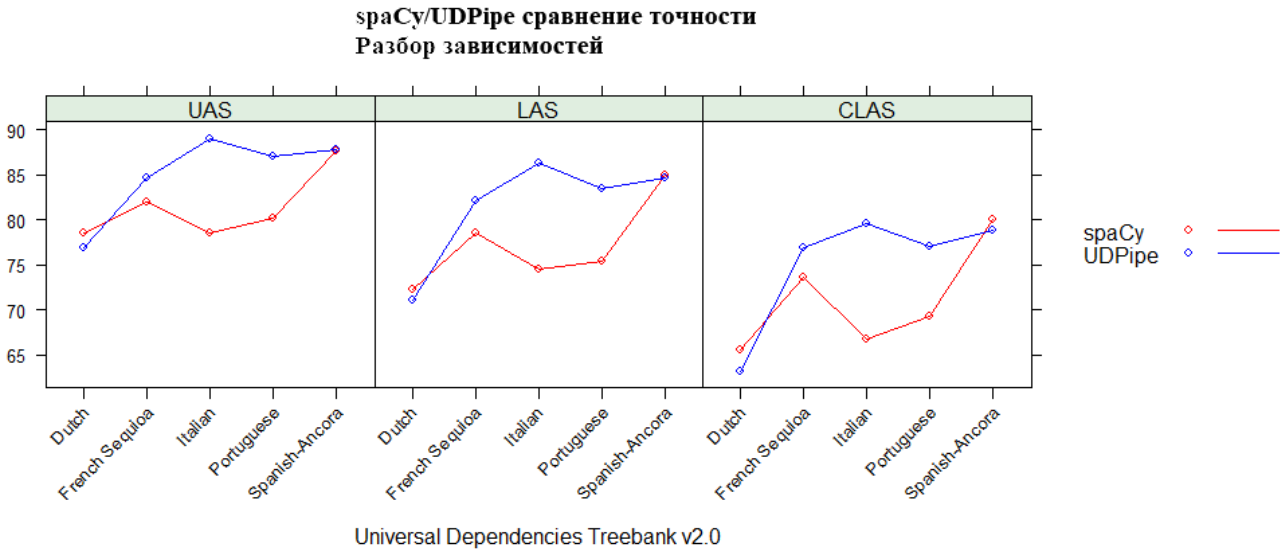

Рисунок 2 – Сравнение точности работы библиотек при разборе зависимостей

Таблица 2 – Сравнение библиотек UDPipe и spaCy [19,20].

	UDPipe	spaCy
Языки	1)Поддержка более 50 языков; 2)Возможность обучить собственные модели в R;	1)Поддерживает 8 языков 2)Сообщество постепенно добавляет поддержку новых языков 3)Обучать модели только через Python
Простота использования	1)Установка в одну команду для R; 2)Имеются встроенные гибкие модели;	1)Установка более сложная по сравнению с UDPipe, нужны дополнительные зависимости для Python
Точность	1)При тестировании для голландского, французского, испанского, итальянского, португальского языков UDPipe в целом показал себя лучше, чем SpaCy	1)На определенных языках и моделях показывает результат лучше чем UDPipe, например, на голландском языке
Возможности аннотации	1)Возможность токенизации тегирования частей речи, тегирования морфологических признаков, лемматизации, парсинга зависимостей	Аналогичные возможности, за исключением того, что spaCy может лемматизировать только англоязычные данные
Скорость работы	1)Медленная обработка по сравнению с spaCy	1)Быстрее примерно в 5 раз чем UDPipe 2)Обширное сообщество, которое оптимизирует работу библиотеки

Определенно, самыми популярными пакетами для NLP сегодня являются UDPipe и Spacy. Они являются основными конкурентами в области NLP. Разница между ними заключается в общей философии подхода к решению задач.

UDPipe более гибкий, также поддерживает большее количество языков. С его помощью можно попробовать различные методы и алгоритмы, комбинировать их и т.д. Spacy же предоставляет одно готовое решение для каждой проблемы, не нужно думать о том, какой метод лучше: авторы Spacy уже позаботились об этом. Кроме того, Spacy работает очень быстро (в несколько раз быстрее, чем UDPipe). Одним из недостатков является ограниченное количество языков, поддерживаемых Spacy. Однако количество поддерживаемых языков постоянно увеличивается.

Вывод.

Таким образом в статье рассмотрены популярные библиотеки для обработки естественного языка, приведены преимущества и недостатки каждой библиотеки, а также проведён их сравнительный анализ. Также было проведено подробное сравнение двух основных библиотек по пяти критериям. В результате сравнительного анализа была выделена библиотека UDPipe, которая несмотря на относительно небольшую скорость работы, обладает большей точностью при работе с различными языковыми группами, и в целом является библиотекой с более гибким и обширным функционалом.

Список литературы

1. Witten I. H., Frank E. Data mining: practical machine learning tools and techniques with Java implementations //Acm Sigmod Record. – 2002. – Т. 31. – №. 1. – С. 76-77.
2. Jain A. K., Murty M. N., Flynn P. J. Data clustering: a review //ACM computing surveys (CSUR). – 1999. – Т. 31. – №. 3. – С. 264-323.
3. Bousquet O., Boucheron S., Lugosi G. Introduction to statistical learning theory //Summer school on machine learning. – Springer, Berlin, Heidelberg, 2003. – С. 169-207.
4. Lee I. et al. A probabilistic functional network of yeast genes //science. – 2004. – Т. 306. – №. 5701. – С. 1555-1558.
5. Myers C. L., Troyanskaya O. G. Context-sensitive data integration and prediction of biological networks //Bioinformatics. – 2007. – Т. 23. – №. 17. – С. 2322-2330.
6. Feldman R. et al. The text mining handbook: advanced approaches in analyzing unstructured data. – Cambridge university press, 2007.
7. Hearst M. A. Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. – 1999.
8. Leeper N. J. et al. Practice-based evidence: profiling the safety of cilostazol by text-mining of clinical notes //PloS one. – 2013. – Т. 8. – №. 5. – С. e63499.
9. Müller T., Schütze H. Robust morphological tagging with word representations //Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. – 2015. – С. 526-536.
10. D. Tunkelang, Stemming and Lemmatization [Site] URL: https://queryunderstanding.com/stemming-and-lemmatization-6c086742fe45 11.01.22
11. Juršic M. et al. Lemmagen: Multilingual lemmatisation with induced ripple-down rules //Journal of Universal Computer Science. – 2010. – Т. 16. – №. 9. – С. 1190-1214.
12. McGillivray B., Passarotti M., Ruffolo P. The Index Thomisticus Treebank Project: Annotation, Parsing and Valency Lexicon //Trait. Autom. des Langues. – 2009. – Т. 50. – №. 2. – С. 103-127.
13. Straka M., Straková J. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe //Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. – 2017. – С. 88-99.
14. Kwartler T. Text mining in practice with R. – John Wiley & Sons, 2017.
15. A comparison between spaCy and UDPipe for Natural Language Processing for R users URL: https://www.bnosac.be/index.php/blog/75-a-comparison-between-spacy-and-udpipe-for-natural-language-processing-for-r-users (дата обращения 12.01.22)
16. Colic N., Rinaldi F. Improving spaCy dependency annotation and PoS tagging web service using independent NER services //Genomics & informatics. – 2019. – Т. 17. – №. 2. Manning C., Schutze H. Foundations of statistical natural language processing. – MIT press, 1999.
17. Kharis M. et al. How to Lemmatize German Words with NLP-Spacy Lemmatizer? //International Seminar on Language, Education, and Culture (ISoLEC 2021). – Atlantis Press, 2021. – С. 189-193..
18. Антропова О. И., Огородникова Е. А. Экстернальная оценка предварительно обученных моделей UDPipe в применении к извлечению гипер-гипонимических словесных пар из словарных определений //AIP Conference Proceedings. – AIP Publishing LLC, 2020. – Т. 2313. – №. 1. – С. 070020.
19. Schmitt X. et al. A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate //2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS). – IEEE, 2019. – С. 338-343.
20. Straka M., Straková J., Hajič J. UDPipe at SIGMORPHON 2019: Contextualized embeddings, regularization with morphological categories, corpora merging //arXiv preprint arXiv:1908.06931. – 2019.

References

1. Witten I. H., Frank E. Data mining: practical machine learning tools and techniques with Java implementations //Acm Sigmod Record. – 2002. – Т. 31. – №. 1. – pp. 76-77.
2. Jain A. K., Murty M. N., Flynn P. J. Data clustering: a review //ACM computing surveys (CSUR). – 1999. – Т. 31. – №. 3. – pp. 264-323.
3. Bousquet O., Boucheron S., Lugosi G. Introduction to statistical learning theory //Summer school on machine learning. – Springer, Berlin, Heidelberg, 2003. – pp. 169-207.
4. Lee I. et al. A probabilistic functional network of yeast genes //science. – 2004. – Т. 306. – №. 5701. – pp. 1555-1558.
5. Myers C. L., Troyanskaya O. G. Context-sensitive data integration and prediction of biological networks //Bioinformatics. – 2007. – Т. 23. – №. 17. – pp. 2322-2330.
6. Feldman R. et al. The text mining handbook: advanced approaches in analyzing unstructured data. – Cambridge university press, 2007.
7. Hearst M. A. Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics. – 1999.
8. Leeper N. J. et al. Practice-based evidence: profiling the safety of cilostazol by text-mining of clinical notes //PloS one. – 2013. – Т. 8. – №. 5. – pp. e63499.
9. Müller T., Schütze H. Robust morphological tagging with word representations //Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. – 2015. – pp. 526-536.
10. D. Tunkelang, Stemming and Lemmatization [Site] URL: https://queryunderstanding.com/stemming-and-lemmatization-6c086742fe45 11.01.22
11. Juršic M. et al. Lemmagen: Multilingual lemmatisation with induced ripple-down rules //Journal of Universal Computer Science. – 2010. – Т. 16. – №. 9. – pp. 1190-1214.
12. McGillivray B., Passarotti M., Ruffolo P. The Index Thomisticus Treebank Project: Annotation, Parsing and Valency Lexicon //Trait. Autom. des Langues. – 2009. – Т. 50. – №. 2. – pp. 103-127.
13. Straka M., Straková J. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe //Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. – 2017. – pp. 88-99.
14. Kwartler T. Text mining in practice with R. – John Wiley & Sons, 2017.
15. A comparison between spaCy and UDPipe for Natural Language Processing for R users URL: https://www.bnosac.be/index.php/blog/75-a-comparison-between-spacy-and-udpipe-for-natural-language-processing-for-r-users (дата обращения 12.01.22)
16. Colic N., Rinaldi F. Improving spaCy dependency annotation and PoS tagging web service using independent NER services //Genomics & informatics. – 2019. – Т. 17. – №. 2. Manning C., Schutze H. Foundations of statistical natural language processing. – MIT press, 1999.
17. Kharis M. et al. How to Lemmatize German Words with NLP-Spacy Lemmatizer? //International Seminar on Language, Education, and Culture (ISoLEC 2021). – Atlantis Press, 2021. – pp. 189-193..
18. Antropova O. I., Ogorodnikova E. A. Extrinsic evaluation of UDPipe pre-trained models in application to hyper-hyponymic verbal pairs extraction from dictionary definitions //AIP Conference Proceedings. – AIP Publishing LLC, 2020. – Т. 2313. – №. 1. – pp. 070020.
19. Schmitt X. et al. A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate //2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS). – IEEE, 2019. – pp. 338-343.
20. Straka M., Straková J., Hajič J. UDPipe at SIGMORPHON 2019: Contextualized embeddings, regularization with morphological categories, corpora merging //arXiv preprint arXiv:1908.06931. – 2019.