UDC 004.89

ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА АВТОМАТИЗАЦИИ ОЦЕНИВАНИЯ ТЕМАТИК ТЕКСТОВОГО КОНТЕНТА

Н.Д. РУДНИЧЕНКО, В.В. ВЫЧУЖАНИН, А.А. ЕГОШИНА, С.М. ВОРОНОЙ, Н.О. ШИБАЕВА

Государственный университет «Одесская политехника», Украина

Поступила в редакцию 25 февраля 2022

Аннотация. В работе приведены результаты разработки и исследования интеллектуальной системы автоматизации оценивания тематик текстового контента на базе применения алгоритмов машинного обучения. Проведен анализ проблематики в задачах обработки естественного языка, обоснована актуальность рассматриваемых задач. Разработана концепция работы системы, описан ее компонентный состав, приведена характеристика собранных данных для анализа, проведены численные эксперименты по обучению тестированию созданных моделей машинного обучения, подтверждающие эффективность использования бустинга и опорных векторов для решения поставленной задачи. Предложены пути дальнейшего совершенствования разработанной системы.

Ключевые слова: обработка естественного языка, машинное обучение, интеллектуальные информационные системы.

Введение

В связи с постоянным ростом информационных источников и технических устройств поддержки процессов ознакомления человека с интересующими данным задача эффективного анализа разнородных текстов большого объема является важной практически в любой современной области человеческой деятельности, где его экспертиза, опыт и знания могут быть изложены в текстовом виде [1]. Появление широкого доступа к сети Интернет привело к бурному росту объемов доступной текстовой информации, находящей в открытом доступе, что значительно ускорило развитие научных направлений в области обработки текста, известной как обработка естественного языка (ОЕЯ) [2]. Спецификой данной сферы научной деятельности является автоматизация процессов обработки и анализа текстовых данных на базе использования методов и подходов искусственного интеллекта (ИИ) и машинного обучения (МО) [3–5].

Важными задачами в сфере ОЕЯ являются анализ тональности текстового контента, соотнесения его к заданным категориям, оценка уникальности концепций и идей контента, уровня согласованности и полноты текста, статистических показателей и характеристик, а также сжатия объемов текста с минимизацией потери содержательной части [6]. Отсутствие универсального и системного подхода в методиках решения поставленных задач приводит к тому, что для существующих программных решений характерным является низкий уровень эффективности и гибкости при решении обозначенных задач [7]. Это связано в том числе с тем, что что полноценная автоматизация всех процессов обработки и анализа текстовых данных возможна только при реализации логики понимания системой текста на уровне человека, с учетом определения сложно формализуемых или нечетких метаданных, что требует создания полноценного ИИ, не являющегося пока возможным и осуществимым технически из-за нехватки знаний о специфике процесса восприятия информации (в частности, в текстовом виде) человеком [8–13].

В связи с этим возникает необходимость разработки и внедрения в процессы анализа текстовых данных гибких, функциональных и удобных интеллектуальных систем (ИС) и

моделей, обеспечивающих как возможности решения бизнес-процессов, так и проведения экспериментальных аналитических исследований, в частности, мониторинга эффективности каждого из имплементированных подходов с возможностью внесения корректировок в его работу, расширение функционала благодаря модульному составу [14–16].

Цель данной работы является исследование возможностей решения задачи оценки и соотнесения текстового контента к сформированному множеству тематических категорий с целью автоматизации обработки и сегментации информационных материалов, размещаемых на различных ресурсах сети Интернет и в системах электронного документа.

Концепция интеллектуальной системы

Концептуально, ИС состоит из 3 отдельных модулей: импорта и предобработки данных, интеллектуального анализа данных, визуализации и интерпретации результатов вычислительных экспериментов.

Выбранными алгоритмами МО для проведения сравнительного исследования работы системы являются: логистическая регрессия, наивный классификатор Байеса, метод опорных векторов, метод ближайшего соседа, случайный лес и метод бустинга.

Ключевым функционалом ИС является: загрузка «сырого» набора данных в виде набора текстовых файлов, содержащих в себе текст статьи и название раздела, к которому она будет принадлежать; преобразования составленного набора данных в структурированный вид, имеющий табличный вид, более удобный для дальнейшего анализа и работы; первичный анализ данных, которая включает в себя подсчет и определение метаданных, в том числе количества текстовых статей в каждой из категорий, а также расчет процента текстов по категориям, размер статьи в формате txt; сохранение набора данных, прошедших первичный анализ в формате *.csv для его удобной обработки программным обеспечением; очистка текстов, включающий в себя удаление специальных символов и знаков препинания, которые не нужны для дальнейшей классификации текстов, объединение слов, написанных с большой и маленькой буквы, несущих одинаковый смысл; лемматизации и стемминг; расчет TF-IDF вектора; создание словарей наиболее значимых слов для проведения оценки текстов; инициирование процесса обучения на базе выбранных алгоритмов МО. Обученные модели сохраняются в формате *.pickle; выбор лучшей модели созданных классификаторов путем сравнения их точности и эффективности по метрикам; представление результатов в виде графиков с использованием двух методов снижения размерности: PCA и t-SNE график; визуализация результатов классификации текстов с указанием точности работы каждой модели МО.

После запуска ИС пользователь осуществляет загрузку данных, которые будут обрабатываться. Выбор наборов данных для обучения возможен путем подключения разных источников (форматы csv, txt, doc, не защищенный pdf, файлы баз данных) или из сети Интернет (ссылки на веб-ресурсы или на прямые текстовые файлы). Все данные хранятся в промежуточном формате txt для упрощения процесса дальнейшей обработки.

Далее модуль программы, отвечающий за обработку данных получает dataset, загруженный в систему, формируя на выходе из всех загруженных наборов текста сводную таблицу, содержащей колонки с названием файла, его содержанием, категорией (тематикой), к которой относится текст. Последний столбец является выходной перемеренной, может быть размечен вручную в рамках данного модуля, в случае наличия этого набора в датасете он может быть отмечен с помощью элементов интерфейса.

На следующем этапе данные, представленные в виде таблицы, проходят первичный анализ, чтобы выяснить, насколько такая выборка является полной для обучения моделей ML и насколько точно набор данных описывает предметную область.

Во время первичного анализа рассчитываются такие показатели как подсчет количества текстов в каждой из выбранных категорий, процент, который составляют тексты с каждой рубрики в соотношении к общему количеству текстов и средний объем текста в каждой из выбранных тематик. В результате длинные тексты удаляются из выборки, что позволяет сбалансировать данные исходя из указанных пользователей параметров и ограничений.

Далее проводится этап очистки данных для перехода к дальнейшему анализу его состава. В это время проводится уменьшение объема текстов, выделяются наиболее значимые

семантические элементы. Очистка текстов выполняются путем изъятия специальных символов и знаков препинания, удаляются часто повторяющиеся слова, не несущие существенного смыслового содержания (союзы, наречия) и притяжательные местоимения, формы единственного и множественного числа в одинаковых словах объединяются в одну, проводится приведение слов к единому регистру. Отдельно выполняются процессы лемматизации и стемминга.

Следующим этапом является создание пакета словарей, состоящих из слов, которые чаще всего встречаются в текстах отдельных рубрик, что позволяет обеспечить семантическую связь между контекстом статей. Обработанные данные и словари сохраняются отдельными файлами для дальнейшего использования в обучении.

Пользователь задает тематики, по которым необходимо провести процесс оценки принадлежности текстов. Далее инициируются вычислительные процессы по разбитию выборок текстовых данных на обучающие и тестовые, создаются объекты классов по каждому из 6 используемых методов ML. Для проведения вычислительных процессов в параллельном режиме на нескольких вычислительных узлах предусмотрена опция подключения Apache Mesos и библиотеки Numba для задействования ресурсов графических аппаратных ускорителей.

На базе полученных результатов вычислений производится расчет метрик оценки качества классификации, главной из которых является точность (отношение правильных ответов к общему числу экземпляров текстов). Еще одной метрикой является полнота классификации, как соотношение правильно положительных ответов ко всем положительным ответам.

После этого полученные данные оценки метрик сохраняются в текстовый файл, агрегируются и передаются на вход графического модуля, на котором осуществляется визуализация графиков анализа точности классификации. Для большей степени интерпретируемости результатов необходимым является использование методов снижения размерности данных. Использованы две методики: анализ основных компонентов, сущность которой заключается в расчете собственных значений и собственных векторов матрицы данных для формирования минимального количества переменных, которые обеспечивают максимальное значение дисперсии в данных. Используется в случае, когда объемы текста не превышают 10 000 знаков; t-SNE, в своей основе содержит вероятностный подход, что удобно при визуализации данных больших размеров (более 10 000 знаков). Позволяет минимизировать расхождение между распределением, измеряющим попарно сходство входящих объектов, и распределением, измеряющим попарно сходство соответствующих низкомерных точек встраивания.

Компонентный состав интеллектуальной системы

Программная имплементация ИС выполнена на основе использования языка программирования Python 3.8, среды разработки PyCharm и ряда функциональных библиотек по обработке и анализу данных, созданию моделей МО. Компонентный состав ИС в виде набора пакетов приведен на рис. 1. Главный компонент программы (MainApp) выполняет функции вызова других компонентов ИС для обработки запросов приложения по обработке данных, выполнению вычислительных процессов по созданию моделей ML, построения графиков и общей бизнес-логики работы системы.

Для удобства проект ИС, представляющий собой модульную структуру, в среде разработки разделен на отдельные пакеты, в каждом из которых содержатся классы, реализующие необходимый функционал исходя из их логической связности между собой. Каждый из которых содержит свой набор программных компонентов.

«Raw dataset» включает в себя необработанные данные в виде файлов и логику реализации импорта.

«Dataset Creation» обеспечивает логику создания структурированного набора данных.

«ExploratoryDataAnalysis» содержит классы, отвечающие за первичный анализ данных.

«Feature Engineering» обеспечивает очистку данных для дальнейшей обработки и выделение признаков.

«Model Training» содержит соответствующие классы, реализующие вызов и создание объектов моделей классификаторов, а также их сериализованные структуры, загружаемые в процессе анализа данных через интерфейс пользователя. Также в данном пакете размещен класс,

содержащий методы для сравнения всех классификаторов и выбора наиболее эффективного для поставленной задачи с учетом расчета метрик.

«ScrapingData» имплементирует логику сбора текстовых данных с указанных источников для дальнейшего анализа.

«UI» реализует функционал графического интерфейса пользователя и вывода визуализаций с элементами управления.

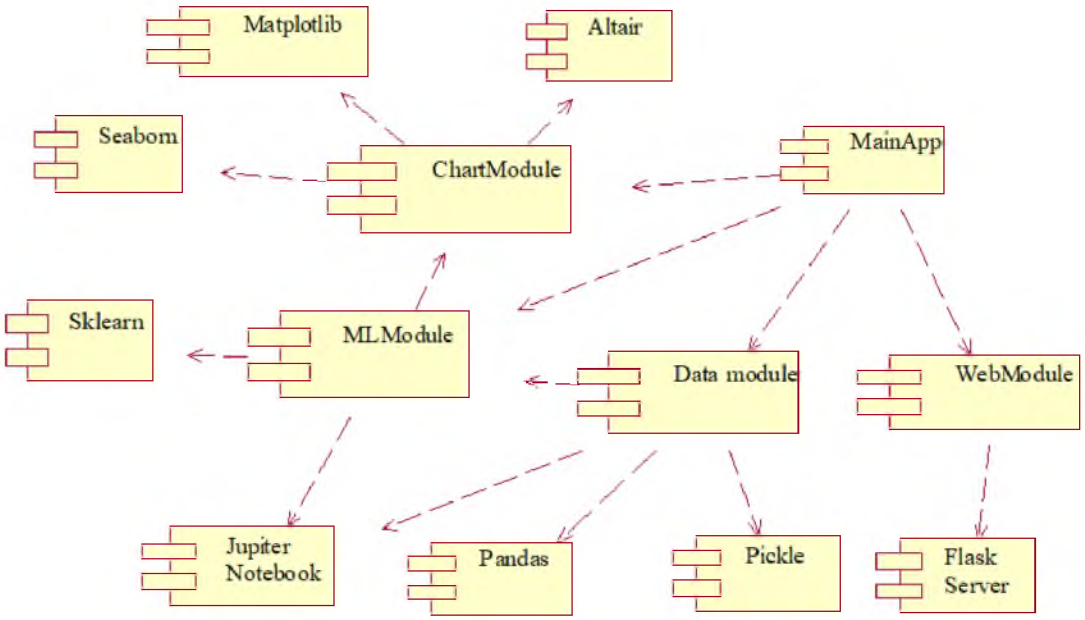

Рис. 1. Диаграмма основных компонентов ИС

Исследование результатов экспериментов

В качестве исходных данных взяты наборы статей из веб-сайтов qz, theguardian, Feedly. Этапами исследования стали сбор и предобработка данных, отбор статей по наиболее популярным рубрикам для обучения моделей МО и оценки их точности с подбором наиболее подходящих значений гиперпараметров для лучшего из выявленных классификаторов.

На базе анализа собранной метаинформации по статьям было подобрано пять основных категорий статей: бизнес, развлечения, политика, IT и здоровье. По результатам подсчета количество текстов в каждой из категорий построена диаграмма (рис. 2), позволяющая нагляднее представить данные (процент каждой из категорий в выборке). Согласно данной визуализации наибольшее число текстов в категории IT, а меньше всего в развлечениях, но несмотря на это данные являются сбалансированными, общая выборка репрезентативна, что позволяет избежать дополнительных манипуляций по их обработке.

В процессе проведения исследования выбрано случайное разбиение набора данных: 75% наблюдений, составляющих тренировочный набор и 25 % наблюдений, составляющих тестовый набор. Процесс настройки гиперпараметров моделей был выполнен с перекрестной валидацией тренировочного набора, чтобы каждая модель МО стала более обобщающей.

Следующим этапом был проведен расчет TF-IDF метрик для определения насколько важно каждое слово в тексте. Формализация процесса заключается в следующем: если слово встречается в любом документе часто, при этом встречаясь редко во всех других документах, то это слово имеет большую значимость для того же документа. Следует отметить, что всего выделено около 500 слов для каждой из тематик.

Формирование словарей из отдельных слов для каждой тематики текстов позволила обеспечить лучшие (чем при обычных словосочетания) соответствие отдельных слов семантическому смыслу в рамках каждой категории.

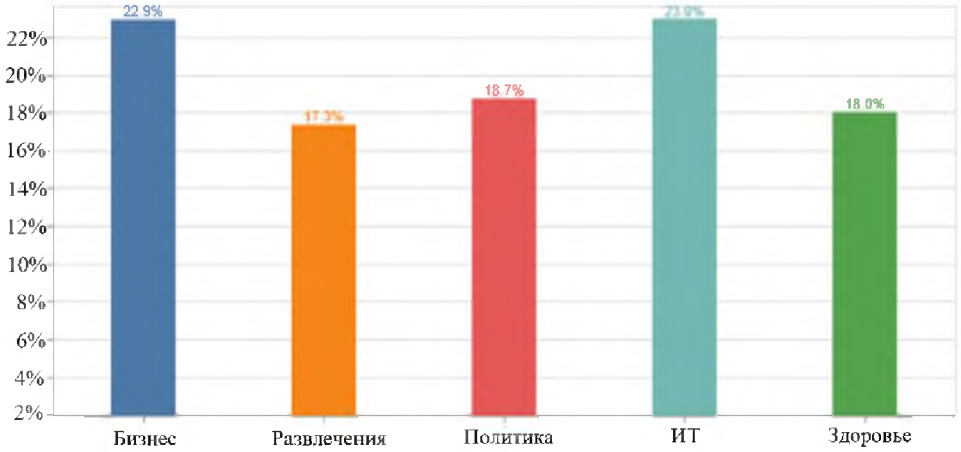

Рис. 2. Диаграмма распределения данных по классам выборки

Результаты оценки созданных моделей МО в рамках ИС приведены в табл. 1.

Табл. 1. **Результаты оценки созданных моделей МО**

№	Название модели	Обучающая выборка			Тестовая выборка		
		Точность	Полнота	Временные затраты, с	Точность	Полнота	Временные затраты, с
1	Бустинг	0,991	0,908	150	0,912	0,813	40
2	Опорные вектора	0,956	0,888	112	0,893	0,811	31
3	Наивный байес	0,713	0,663	43	0,65	0,558	12
4	Ближайший сосед	0,739	0,552	48	0,701	0,615	11
5	Случайный лес	0,885	0,782	140	0,798	0,745	33
6	Логистическая регрессия	0,771	0,699	38	0,743	0,703	9

Согласно табл. 1 следует отметить, что модель бустинга является наиболее точной. Наивный байесовский классификатор и метод ближайших соседей показывают менее значительные результаты, однако их скорость работы достаточно высока.

Высокие показатели точности и полноты характеры для модели опорных векторов, скорость ее обучения и тестирования также выше модели бустинга почти на 30 %, минимальный разброс значений между тестовой и тренировочной выборками также наблюдается для данной модели и для логистической регрессии, являющейся наиболее быстрой, но существенно менее точной.

Выводы

В результате проведения данного исследования была разработана интеллектуальная система автоматизации процесса оценки тематик текстового контента базе использования предложенной концепции предобработки данных и создания разных моделей машинного обучения с успешной апробацией ее использования на реальных наборах данных. Установлено, что наиболее эффективными моделями для соотнесения текстового контента к заданным тематикам являются модели бустинга и опорных векторов. В дальнейшей работе в данном направлении предлагаемыми путями расширения возможностей системы являются: внедрения возможностей итеративного прогона моделей в выбранном диапазоне во многопоточном режиме, что может обеспечит существенное снижение временных затрат на проведение вычислительных экспериментов; расширение методики предобработки данных путем выделения дополнительной метаинформации по статистическим и семантическим параметрам текстов.

INTELLECTUAL SYSTEM OF TEXTUAL CONTENT SUBJECT ESTIMATION AUTOMATIZATION

N.D. RUDNICHENKO, V.V. VYCHUZHANIN, A.A. EGOSHINA,
S.M. VORONOY, N.O. SHIBAEVA

Abstract. This article presents the results of the development and research of an intellectual system of textual content subject estimation automatization based on the application of machine learning algorithms. The analysis of problems in the tasks of natural language processing is carried out, the relevance of the tasks under consideration is substantiated. The concept of system was developed, its component structure was described, the characteristics of the data collected for analysis were described and numerical experiments on training and testing of created models of machine learning were conducted. These experiments confirmed the effectiveness of using boosting and support vectors for solving the task. Ways to further improve the developed system are proposed.

Keywords: natural language processing, machine learning, intelligent information systems.

Список литературы

1. Вычужанин В.В., Рудниченко Н.Д. Методы информационных технологий в диагностике состояния сложных технических систем. Одесса, 2019.
2. Рудниченко Н. и др. // Информационные управляющие системы и технологии. Проблемы и решения. 2019. С. 31–46.
3. Rudnichenko N. [et al.] // Proceedings of the 9th International Conference «Information Control Systems & Technologies». 2020. P. 371–385.
4. Lidong W., Cheryl A. // International Journal of Mathematical, Engineering and Management Sciences. 2016. Vol. 1. P. 52–61.
5. Junfei Q. [et al.] // EURASIP Journal on Advances in Signal Processing. 2016.
7. Bliznyuk B.O. [et al.] // Visnik of Kharkiv National University of the Name of V.N. Karazin. 2017.
8. Yuskov V.S., Barannikova I.V. // Mining information and analytical bulletin. 2017. Vol. 3. P. 272–278.
9. Widiastuti N.I. // IOP Conference Series: Materials Science and Engineering. 2018. Vol. 407.
10. Li H. // National Science Review. 2018. Vol. 5.
11. Noskov D.V. // Bulletin of Science and Education. 2018. Vol. 40. P. 39–41.
12. Jain A., Mandowara J. // International Journal of Computer Application. 2016. Vol. 2.
13. Krasnyansky M.N. [et al.] // Voronezh State University Bulletin. Series: Systems Analysis and Information Technology. 2018. Vol. 3. P. 173–182.
14. Степанов П.А. // Вестник Новосибирского государственного университета. Серия: Информационные технологии. 2013. № 12. С. 106–112.
15. Юсков В.С., Баранникова И.В. // Горный информационно-аналитический бюллетень. 2017. № 3. P. 272–278.
16. Вахтина О.П. // Вестник магистратуры. 2019. № 98. С. 21–24.