УДК 004.42:004.912

ПРОГРАММНОЕ РЕШЕНИЕ ПО ПОСТРОЕНИЮ КЛАССИФИКАТОРОВ ДЛЯ АНАЛИЗА ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Сметанина О.Н.,Сазонова Е.Ю., Сулейманов А.К., Селиванов С.Г., Андрушко Д.Ю., Габдиев Ф.Ф.

ФГБОУ ВО «Уфимский государственный авиационный технический университет», Уфа, e-mail: smoljushka@mail.ru, rassadnikova_ekaterina@mail.ru, azamat-sul2010@yandex.ru, andrewrush@mpo14.ru, faritgabdiev@mail.ru

Авторами статьи предлагается программное решение по построению классификаторов для анализа текстов на естественном языке. В статье представлено современное состояние проблемы построения классификаторов для анализа текстов на естественном языке. Проведенный анализ позволил сделать вывод о необходимости программной реализации в связи со сложностью адаптации существующих программных решений для анализа тональности новостных текстов на башкирском языке. В статье приводится постановка задачи анализа тональности новостных текстов на башкирском языке, для ее решения предлагается гибридный подход, который включает методы, основанные на словарях, и методы, основанные на машинном обучении. Предлагаются следующие этапы для решения задачи анализа тональности текстов на башкирском языке: предварительная обработка текста (приведение к нижнему регистру; удаление символов, не являющихся буквами; удаление стоп-слов), векторное представление слов (метод представления текста в векторном виде Bag-of-words и статистический показатель TF-IDF), классификация (положительная, отрицательная тональность; метод опорных векторов со стохастическим градиентным спуском). Также в статье приводятся входные и выходные данные системы, ее функции и атрибуты, требования к программно-аналитическому комплексу, а именно функциональные, нефункциональные, системные, структурная схема программно-аналитического комплекса. Для реализации были использованы язык программирования Python, большое количество библиотек, таких как Scikit-Learn, NLTK, Gensim, spaCy, NetworkX и Yellowbrick, реализующих методы машинного обучения.

Ключевые слова: машинное обучение, классификаторы, анализ тональности текста, тексты на естественном языке, программно-аналитический комплекс, метод опорных векторов, формализация требований, Python

SOFTWARE SOLUTION ON BUILDING CLASSIFIERS FOR ANALYSIS OF TEXTS IN NATURAL LANGUAGE

Smetanina O.N., Sazonova E.Yu., Suleymanov A.K., Selivanov S.G., Andrushko D.Yu., Gabdiev F.F.

Federal State Budgetary Educational Institution of Higher Education Ufa State Aviation Technical University, Ufa, e-mail: smoljushka@mail.ru, rassadnikova_ekaterina@mail.ru, azamat-sul2010@yandex.ru, andrewrush@mpo14.ru, faritgabdiev@mail.ru

The software solution on constructing classifiers for analyzing texts in natural language is proposed by authors of this article. State of the problem of constructing classifiers for the analysis of texts in natural language is presented. This analysis led to the conclusion about the need for a software implementation due to the complexity of adaptation of existing software solutions for the analysis of news texts tone in Bashkir. The statement of the problem of analyzing the sentiment of news texts in Bashkir is given. Hybrid approach is proposed to solve this problem. This approach includes methods based on dictionaries and methods based on machine learning. The following steps for the solution of texts tonality analysis of the problem in Bashkir: pre-processing of text (leading to lower case; removing characters that are not letters, removal of stop words), word embedding (presentation method of text in vector Bag-of-words and TF-IDF statistic), classification (positive, negative sentiment; support vector machines with stochastic gradient descent). Also input and output data of the system provides in this article, system functions and attributes, requirements for the software-analytical complex, namely, functional, non-functional, system, structural diagram of the software-analytical complex. Python programming language, a large number of libraries such as Scikit-Learn, NLTK, Gensim, spaCy, NetworkX and Yellowbrick that implement machine learning methods have been used to implement.

Keywords: machine learning, classifiers, sentiment analysis, natural language texts, software-analytical complex, support vector machines, requirements formalization, Python

Большие объемы слабоструктурированных данных, возможность использования результатов анализа таких данных для принятия решений потребовали как разработки теоретических основ для проведения анализа, так и их программных реализаций. В частности, как отмечают авторы [1], такие решения основаны на современной инфраструктуре анализа текстов, как то: множество приемов, методов, инструментов для работы со строками; лексические ресурсы; компьютерная лингвистика; алгоритмы машинного обучения и пр. При использовании для анализа данных машинного обучения решения часто реализованы на языке Python, который имеет множество научных и вычислительных библиотек.

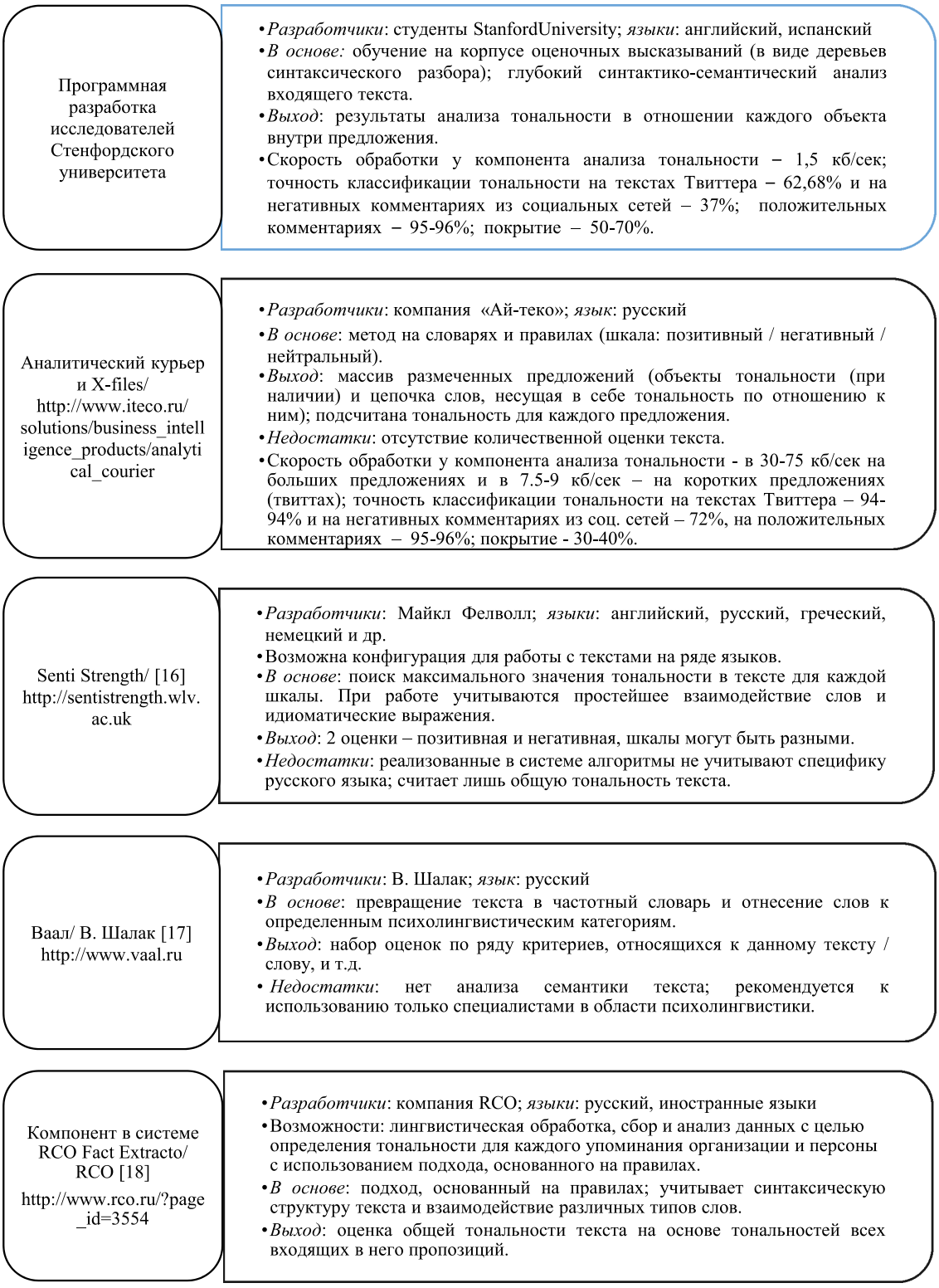

Рис. 1. Известные системы в области анализа тональности в текстах на ЕЯ

Несмотря на то что на данный момент имеются теоретическая база анализа текстов [2–4], в частности модели и методы [5–7], особенности решаемых задач [8–10], специфика данных [11, 12], вопросы автоматизации [13, 14] и ряд программных

решений [15–17], позволяющих обрабатывать и анализировать тексты на естественном языке (ЕЯ), совершенствование теоретических основ, специфика ЕЯ (например, башкирского, когда приходится ограничивать применение методов предобработки из-за особенностей словообразования) и решаемых задач требует разработки нового подхода и, как следствие, программной реализации.

В статье представлены современное состояние проблемы построения классификаторов для анализа текста на ЕЯ, в частности для задачи анализа тональности новостных текстов, и готовые программные решения в этой области, обоснована необходимость новой программной реализации ввиду сложности адаптации готовых решений для анализа тональности новостных текстов на башкирском языке. Приводятся формальные требования к программно-аналитическому комплексу и описывается его структурная схема.

Современное состояние проблемы построения классификаторов для анализа текстов на естественном языке

Развитие фундаментальных положений и информационных технологий в области анализа текста привело к тому, что на рынке появилось множество сервисов, позволяющих обрабатывать и анализировать тексты на ЕЯ. Среди широко известных программных решений для задач автоматического определения тональности текста следует выделить Sentiment140, Библиотека NLTK, Senti Strength [16], RCO FactExtractor [18], «Аналитический курьер» и «X-files» и др. (рис. 1). При проведении сравнительного анализа для каждого программного решения отмечены такие аспекты, как: разработчики; языки, на которых может быть представлен текст для анализа; аппарат, положенный в основу решения; чем представлен выход, а также ряд других особенностей. Кроме указанных программных решений, можно отметить такие инструментальные средства, как текстовый процессор КроЛАН (ООО «ЛАН-ПРОЕКТ»), Quiddi Semantics (ООО «ТомскСофт»), 3i NLP Platform (ООО «ДСС Лаб») и др. (рис. 2).

Также следует выделить программу на основе наивного Байесовского классификатора и нечеткой логики «Гибридный классификатор текстовых документов на естественном языке» (ДГТУ), которая позволяет задать произвольное число категорий для классификации. Особенности языка не всегда позволяют использовать те или иные методы анализа, заложенные в имеющихся программных решениях. Часто информация о методах, положенных в основу программной реализации, отсутствует.

Результаты анализа программных решений для классификации текстов, несмотря на то, что используемые в них словари могут быть созданы на различных языках, продемонстрировали отсутствие возможности их применения для текстов на башкирском языке. Это обусловлено тем, что для решения задачи анализа тональности текстов на башкирском языке недостаточно только составления словаря. Специфика языка не позволяет использовать некоторые методы, заложенные в готовых программных решениях [19]. Поэтому необходима разработка программного решения для построения классификатора текстов на башкирском языке.

Цель исследования обусловлена потребностью решения широкого круга задач и принятия решений на основе анализа тональности текстов. К таким задачам могут быть отнесены, например, оценка уровня лояльности потребителя к товару или услуге, определение взглядов на то или иное событие, оценка новостных текстов с последующей оценкой общественного мнения и пр. В связи с тем, что объемы слабоструктурированных данных, на которых и проводится анализ, постоянно возрастают, необходимо также и повышение эффективности анализа данных.

Целью исследования является построение эффективного классификатора в виде программно-аналитического комплекса. В статье эффективность может быть получена и за счет автоматизации анализа, и за счет того, что построение классификатора осуществляется на основе метода, дающего «лучшее» решение. Выбор метода авторами был осуществлен ранее [19].

Постановка задачи и формализация требований к программному решению

Постановка задачи анализа тональности новостных текстов на башкирском языке предполагает, что можно выделить две категории классов – «положительные» / «отрицательные» (рис. 3). В основе решения задачи лежит гибридный подход, который включает методы, основанные на словарях, и методы, основанные на машинном обучении. Структура решения задачи анализа тональности текстов на башкирском языке (рис. 4) включает следующие этапы: предварительная обработка текста (приведение к нижнему регистру; удаление символов, не являющихся буквами; удаление стоп-слов), векторное представление слов

(метод представления текста в векторном виде Bag-of-words и статистический показатель TF-IDF), классификация (положительная, отрицательная тональность; метод опорных векторов со стохастическим градиентным спуском).

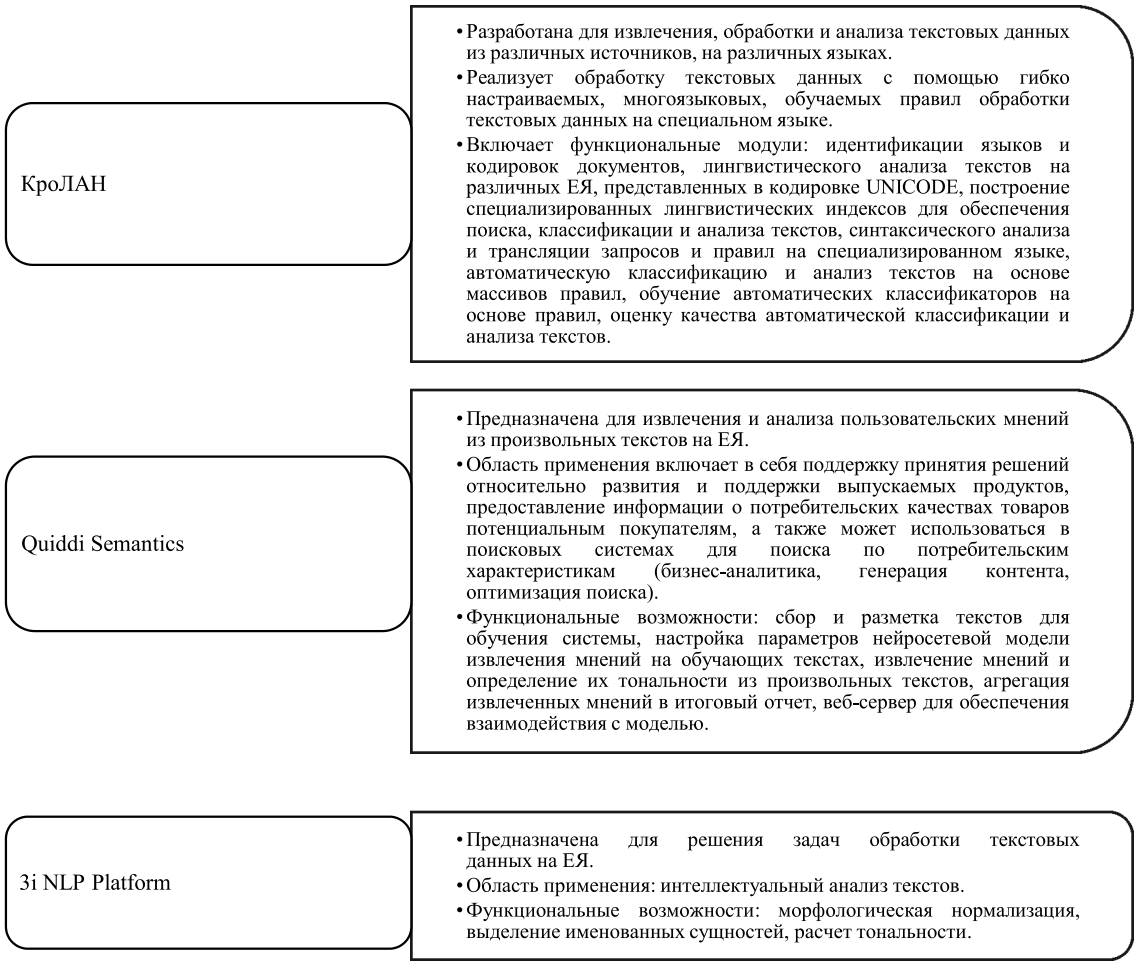

Рис. 2. Программные решения в области анализа тональности текста

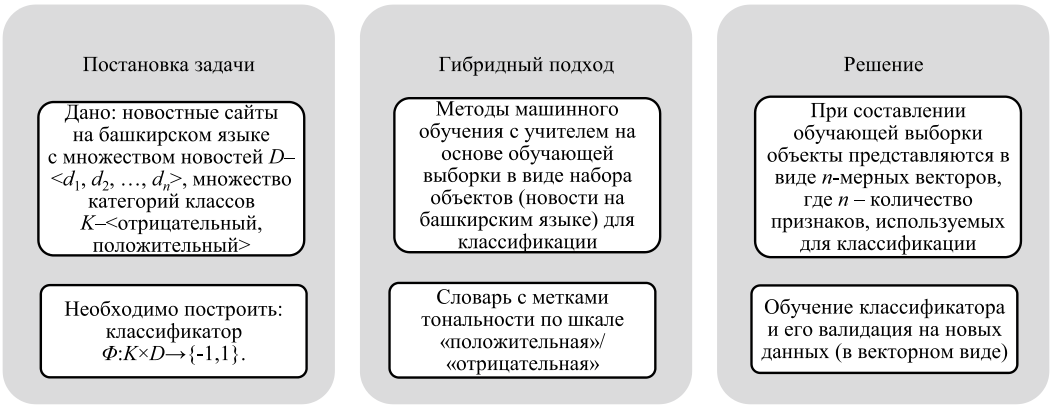

Рис. 3. Краткое представление функционала

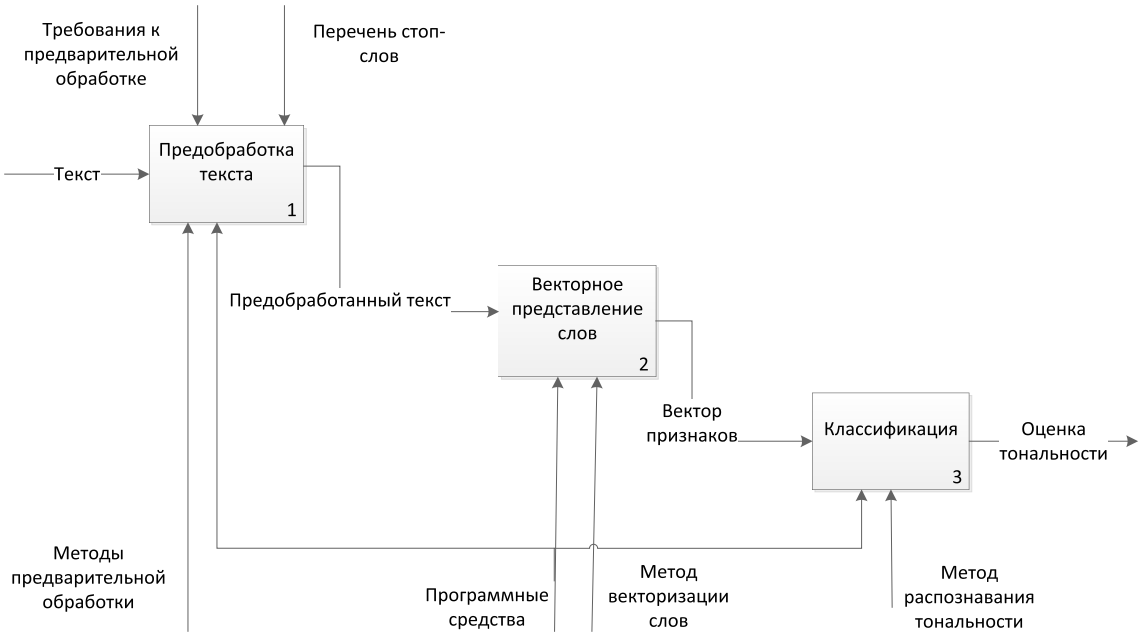

Рис. 4. Структура решения задачи анализа тональности текстов на башкирском языке

По результатам проведенного эксперимента для построения классификатора как наиболее точный был отобран метод опорных векторов со стохастическим градиентным спуском. В его основе лежит построение гиперплоскости для оптимального разделения объектов обучающей выборки $(x_1, y_1), \ldots, (x_k, y_k), x_i \in \mathbb{R}^n$ на два класса: $y_i \in \{-1, 1\}$. Классифицирующая функция: $F(x) = sign(\langle w, x \rangle + b)$, где $\langle w, x \rangle$ – скалярное произведение, w – нормальный вектор к разделяющей плоскости, b – вспомогательный параметр. Один класс – объекты со значением функции $F(x) = 1$, другой класс – объекты с $F(x) = -1$. Любая гиперплоскость задается в виде $\langle w, x \rangle + b = 0$ для некоторых w и b, выбираемых для максимизации расстояния от гиперплоскости до объектов каждого класса $\frac{1}{\|w\|}$. Учитывая, что проблемы нахождения $\max \frac{1}{\|w\|}$ и нахождения $\min \|w\|^2$ аналогичны, можно записать задачу оптимизации: $\begin{cases} \arg\min_{w,b} \|w\|^2, \\ y_i(\langle w, x \rangle + b) \geq 1, i = 1, \ldots, m \end{cases}$, и ее решение с помощью множителей Лагранжа [19].

Таким образом, в основу создаваемого программно-аналитического комплекса легли предлагаемая теоретическая база для предварительной обработки текста и отобранный по результатам проведенного эксперимента метод опорных векторов со стохастическим градиентным спуском [19].

После определения того, что должно быть на входе и выходе системы, ее функций и атрибутов формулируются требования. Комплекс требований к программно-аналитическому комплексу включает: функциональные, нефункциональные, системные требования (рис. 5). Функциональные требования представлены UML-диаграммой вариантов использования: ввод текста, просмотр результата, определение тональности текста (предварительная обработка текста, векторное представление текста, анализ тональности), вывод результатов анализа (рис. 6). Диаграмма последовательностей демонстрирует взаимодействие объектов (компонентов программно-аналитического комплекса и пользователя) в динамике (рис. 7). Диаграмма размещения представляет общую конфигурацию и топологию распределенного программно-аналитического комплекса и содержит распределение компонентов по отдельным узлам системы.

Архитектура приложения представляет собой классическую архитектуру – «клиент – сервер».

К системным требованиям данного программно-аналитического комплекса относятся характеристики сервера и клиента, возможность переносимости комплекса и пр. (рис. 8).

Функциональные требования	• Охватывают предполагаемое поведение системы с определением действий, которые система должна выполнять, а также накладываемых ограничений. • Разработчику необходимо подготовить данные для обучения и обучить классификатор. • Пользователь вводит текст. • Модуль предобработки производит обработку текста. • Обработанный текст поступает на вход классификатора, который определяет тональность текста. • Затем система вывода подготавливает результаты анализа и отправляет их пользователю.
Нефункциональные требования	• Понятие «нефункциональные требования» связано с требованиями к характеру поведения системы. • Для разрабатываемого программного решения: легкость и простота использования приложения; эффективность и устойчивость к сбоям; наличие проверки корректности передаваемых данных.
Системные требования	• Системные требования – это определения элементарных операций, которые должна иметь система, а также различных условий, которым она может удовлетворять.

Рис. 5. Требования к программно-аналитическому комплексу

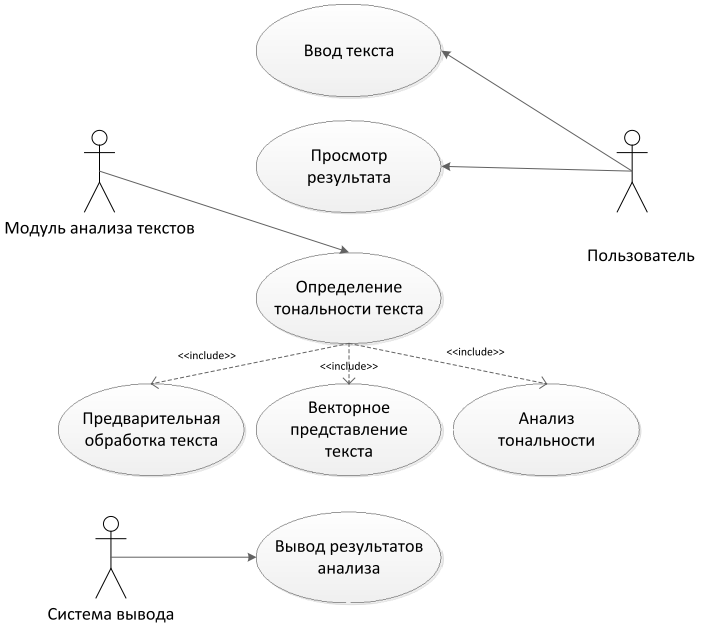

Рис. 6. Диаграмма вариантов использования

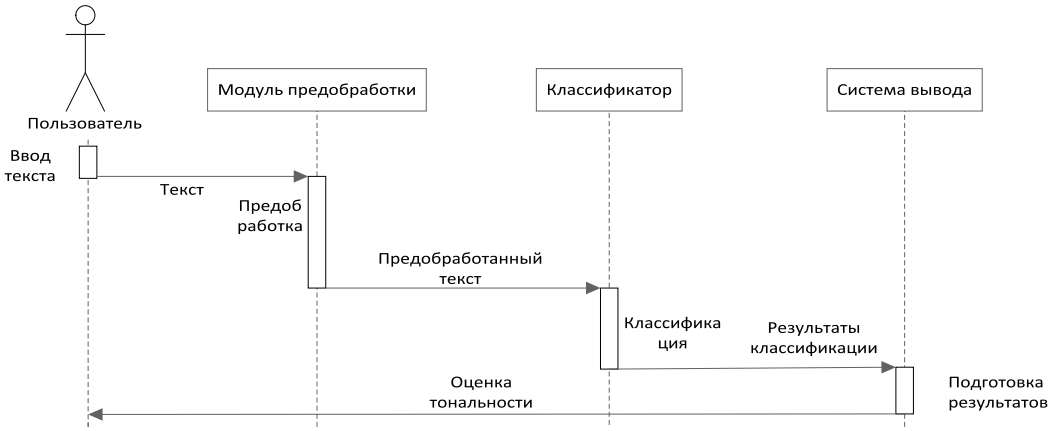

Рис. 7. Диаграмма последовательностей

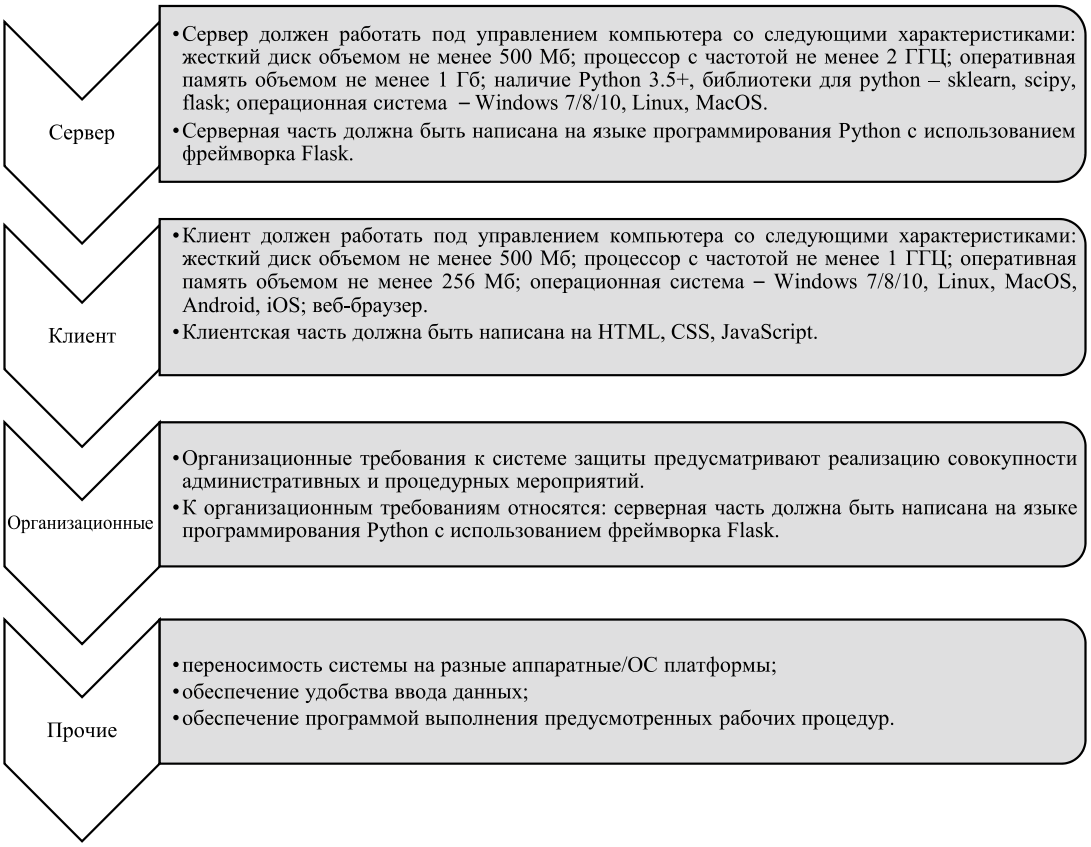

Рис. 8. Детализация системных требований

Язык Python обеспечивает довольно короткие сроки написания программ, также используется большое количество библиотек, таких как Scikit-Learn, NLTK, Gensim, spaCy, NetworkX и Yellowbrick, реализующих методы машинного обучения.

Структурная схема программно-аналитического комплекса

Структурная схема программно-аналитического комплекса включает такие базовые модули, как: модуль web-приложения и модуль анализа тональности, а также несколько вспомогательных (рис. 9). Согласно функционалу, в структурной схеме некоторые вспомогательные модули вошли в состав базовых.

Модуль web-приложения позволяет: задавать конфигурацию приложения, импортировать пакеты, отслеживать обращения по адресам / и */index*, реагировать на отправку формы ввода текста для анализа вызовом метода predict модуля анализа, обеспечивать безопасность клиентских сессий, для генерации формы ввода текста для анализа использовать класс AnalyzeForm; содержит папки с перечнем html-страницы для web-приложения, базовой страницей и страницей, ее расширяющей, формой для ввода текста и графические элементы, папки со статическими объектами, файлы со списком необходимых библиотек для работы приложения с указанием версий для каждой библиотеки.

Модуль анализа тональности: позволяет получить по обученной модели оценку тональности, очистить текст. В данный модуль включен блок обучения модели (программно реализован как вспомогательный). Вспомогательные модули реализуют следующий функционал: реализация методов парсинга, методов для автоматической разметки текстов, загрузки словаря тональности, методов предварительной обработки текста, метод для удаления стоп-слов.

Выводы

Результаты анализа современного состояния проблемы построения классификаторов для анализа текста на ЕЯ показали как наличие теоретической базы для разработки инструментальных средств для анализа, так и широкий спектр готовых программных решений: от «универсальных» до специальных, созданных под конкретную задачу.

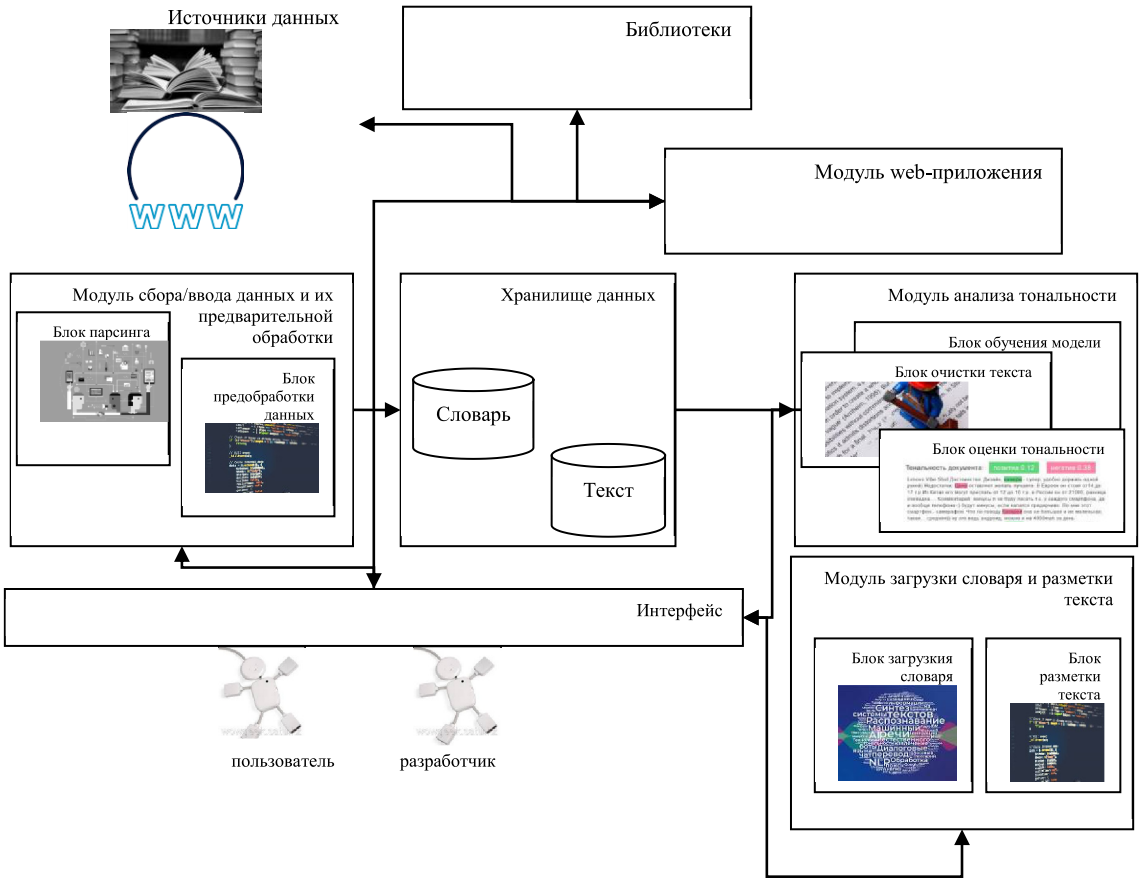

Рис. 9. Структурная схема программно-аналитического комплекса

Специфика словообразования в башкирском языке и отсутствие готовых корпусов текстов обусловили необходимость разработки программно-аналитического комплекса для анализа тональности новостных текстов, представленных на башкирском языке. Программная реализация основана на ранее описанном гибридном подходе [19]. Формальные требования к программно-аналитическому комплексу включают функциональные, нефункциональные и системные требования. Использование языка Python дало возможность написать программу в довольно короткие сроки.

Результаты исследований, приведенные в статье, получены в рамках выполнения грантов РФФИ 18-07-00193, 19-07-00709 и государственного задания № FEUE-2020-0007.

Список литературы

1. Бенгфорт Бенджамин, Билбро Ребекка, Охеда Тони. Прикладной анализ текстовых данных на Python. Машинное обучение и создание приложений обработки естественного языка. СПб.: Питер, 2019. 368 с.

2. Maite Taboada Sentiment Sentiment Analysis: An Overview from Linguistics // Annual Review of Linguistics. 2016. P. 54. [Electronic resource]. URL: https://www.researchgate.net/publication/283954600_Sentiment_Analysis_An_Overview_from_Linguistics (date of access: 04.12.2020).

3. Liu B. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies. 2012. V. 5. № 1. P. 1–167.

4. Лукашевич Н.В., Четвёркин И.И. Комбинирование тезаурусных и корпусных знаний для извлечения оценочных слов // Системы и средства информатики. 2015. Т. 25. № 1. С. 20–33.

5. Аббаси М.М., Бельтюков А.П. Анализ эмоций из текста на русском языке с использованием синтаксических методов // Информационные технологии и системы. 2019. С. 137–142.

6. Романов А.С., Васильева М.И., Куртукова А.В., Мещеряков Р.В. Анализ тональности текста с использованием методов машинного обучения // Proceedings. Сер. «CEUR-WS Workshop Proceedings» St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Herzen State Pedagogical University of Russia. Издательство: Creative Commons CCO, 2018. С. 86–95.

7. Чиркин Е.С., Лопатин Д.В. Подходы к нечеткому поиску нежелательного контента на веб-странице // Вестник Тамбовского университета. Серия Естественные и технические науки. 2016. Т. 21. № 6. С. 2358–2365.

8. Potapova R., Komalova L. Multimodal perception of aggressive behavior. International Conference on Speech and Computer. Springer, Cham. 2016. P. 499–506.

9. Ананьева М.И., Кобозева М.В., Соловьев Ф.Н., Поляков И.В., Чеповский А.М. О проблеме выявления экстремистской направленности в текстах // Вестник новосибирского государственного университета: Информационные технологии. 2016. Т. 14. № 4. С. 5–13.

10. Wiebe, J.M., Wilson, T., Cardie, C. Annotating expressions of opinions and emotions in language. Language Resources and Evaluation. 2005. Vol. 39. Iss. 2–3. P. 165–210.

11. Бодрунова С.С. Кросс-культурный тональный анализ пользовательских текстов в Твиттере // Вестник Московского университета. Журналистика. 2018. № 6. С. 191–212.

12. Воронина И.Е. Анализ эмоциональной окраски сообщений в социальных сетях (на примере сети «вконтакте // Вестник ВГУ: Системный анализ и информационные технологии. 2015. № 4. С. 151–158.

13. Гаршина В.В., Калабухов К.С., Степанцов В.А., Смотров С.В. Разработка системы анализа тональности текстовой информации // Вестник ВГУ: Системный анализ и информационные технологии. 2017. № 3. С. 185–194.

14. Лукашевич Н.В. Автоматический анализ тональности текстов по отношению к заданному объекту и его характеристикам // Электронные библиотеки. 2015. Т. 18. № 3–4. С. 88–119.

15. Ведущий российский системный интегратор Айтеко. [Электронный ресурс]. URL: http://www.iteco.ru/solutions/business_intelligence_products/analytical_courier (дата обращения: 04.12.2020).

16. SentiStrength. [Электронный ресурс]. URL: http://sentistrength.wlv.ac.uk (дата обращения: 04.12.2020).

17. Проект ВААЛ. [Электронный ресурс]. URL: http://www.vaal.ru (дата обращения: 04.12.2020).

18. RCO Fact Extractor SDK. [Электронный ресурс]. URL: http://www.rco.ru/?page_id=3554 (дата обращения: 04.12.2020).

19. Сулейманов А.К., Шарипова М.А., Сметанина О.Н., Сазонова Е.Ю., Миронов К.В. Модели и методы анализа тональности в текстах на башкирском языке // Моделирование, оптимизация и информационные технологии. 2020. Т. 8(3). [Электронный ресурс]. URL: https://moit.vivt.ru/wp-content/uploads/2020/08/SuleimanovSoavtors_3_20_1.pdf (дата обращения: 04.12.2020).