ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

DOI 10.21672/2074-1707.2021.53.1.009-017
УДК 004.4

СТРУКТУРА ПРОГРАММНОГО ПРОДУКТА ДЛЯ СЕМАНТИЧЕСКОГО АНАЛИЗА ТЕКСТОВОЙ ИНФОРМАЦИИ

Статья поступила в редакцию 15.09.2020, в окончательном варианте – 13.01.2021.

Ажмухамедов Искандар Маратович, Астраханский государственный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

доктор технических наук, декан факультета цифровых технологий и кибербезопасности, профессор кафедры информационной безопасности, ORCID https://orcid.org/0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

Зорин Кирилл Андреевич, Астраханский государственный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

ассистент кафедры информационной безопасности и цифровых технологий, ORCID https://orcid.org/0000-0003-1614-8168, e-mail: kirocan95@gmail.com

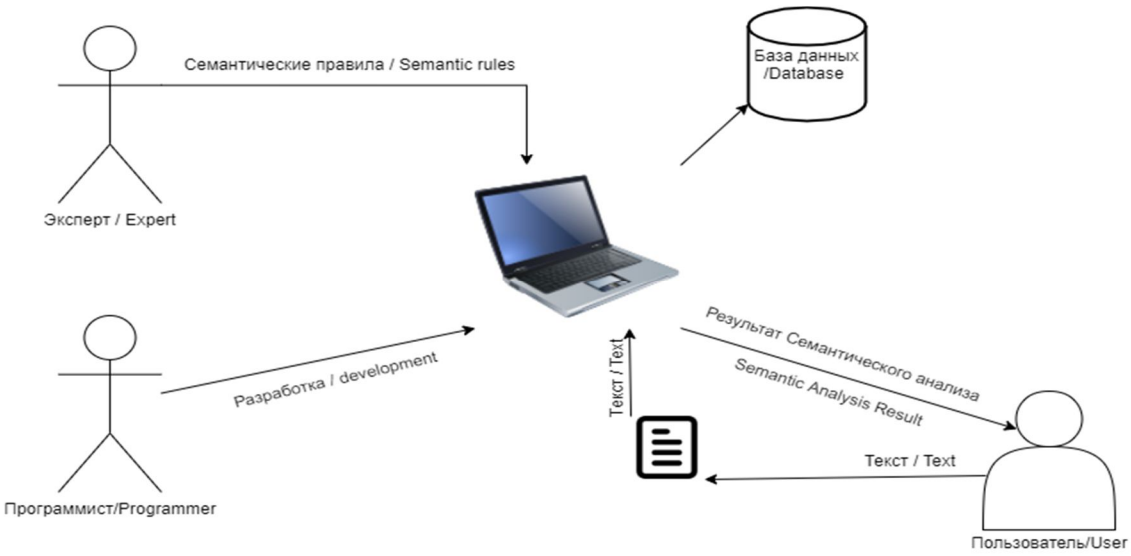
Кузнецова Валентина Юрьевна, Астраханский государственный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

ассистент кафедры информационной безопасности и цифровых технологий, ORCID https://orcid.org/0000-0002-6954-5020, e-mail: arhelia@bk.ru

Статья посвящена методике семантического анализа текста и реализующего ее программного обеспечения. Описывается общая логика работы разработанного web-приложения, которое осуществляет классификацию текстовой информации по заданным экспертом семантическим категориям. Для иллюстрации функционирования программного продукта приведены диаграммы использования, развертывания, а также общая схема работы алгоритма. Применение разработанного программного продукта позволяет обрабатывать текстовую информацию любого содержания, а также проводить семантический анализ текста. Реализованный метод актуален во многих сферах деятельности, где приходится работать с текстовой информацией: правоохранительная деятельность по выявлению незаконного контента, маркетинг, бренд-менеджмент, скоринг.

Ключевые слова: семантический анализ, алгоритм, методы классификации, программное обеспечение, оптимизация, текстовая информация, web-сервис

Графическая аннотация (Graphical annotation)



STRUCTURE OF A SOFTWARE PRODUCT FOR SEMANTIC ANALYSIS OF TEXTUAL INFORMATION

The article was received by the editorial board on 15.09.2020, in the final version – 13.01.2021.

Azhmukhamedov Iskandar M., Astrakhan State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

Doct. Sci. (Engineering), Dean of the Faculty of Digital Technologies and Cybersecurity, Professor of the Department of Information Security and Digital Technologies, ORCID https://orcid.org/0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

Zorin Kirill A., Astrakhan State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

assistant of the Department of Information Security and Digital Technologies, ORCID https://orcid.org/0000-0003-1614-8168, e-mail: kirocan95@gmail.com

Kuznetsova Valentina Yu., Astrakhan State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

assistant of the Department of Information Security and Digital Technologies, ORCID https://orcid.org/0000-0002-6954-5020, e-mail: arhelia@bk.ru

The article is devoted to the development of software for semantic text analysis. Describes the General logic of the developed web application, which implements the classification of textual information by previously pre-defined semantic categories. This article provides usage and deployment diagrams, as well as a General diagram of the algorithm. The methods implemented in the app allow you to increase the speed of processing text information. The use of the developed software product allows you to quickly process text information of any content, as well as conduct a deeper semantic analysis of the text. The implemented method is relevant in many areas of activity where you have to work with text information: monitoring of emotional mood towards the company, marketing, social network analysis.

Keywords: semantic analysis, algorithm, classification methods, software, optimization, text information, web service

Введение. Процесс обмена информации в социальной среде является одним из основополагающих процессов, который предопределяет развитие общества. Без постоянной передачи знаний прогресс в сфере науки, бизнеса, искусства, медицины практически невозможен. Передача информации, как и ранее, является основным фактором нормального функционирования и развития общества. В подавляющем большинстве случаев такой обмен происходит посредством применения текстового формата. Поэтому автоматизированный анализ текстовой информации является актуальной на сегодняшний день технологией для решения различных прикладных задач – в банковской, туристической, рекламной, правоохранительной и других сферах жизни общества.

Существуют различные методы анализа текста на выявление его семантической направленности. В статье Д.Ю. Турдакова «Семантический анализ текстов с использованием системы Texterra» [9] приводится метод анализа текста с использованием баз знаний, в частности Википедии. Также описан процесс построения семантической модели в Texterra, где на основе данных из открытых источников система обнаруживает термины и устойчивые выражения в тексте, что позволяет автоматически наполнять базу данных и улучшать качество для семантического распознавания текста. А.С. Епрев в статье «Автоматическая классификация текстовых документов» приводит обзор методов, которые возможно применить для классификации текстовых документов [3]. Т.В. Батура в статье «Методы автоматической классификации текстов» [2] приводит сравнение современных подходов решения задачи классификации текстов, показывает тенденцию развития данного направления, а также выбор наилучших алгоритмов для применения в исследовательских и коммерческих задачах. В работе также произведены анализ и сравнение качества работы различных методов классификации по таким характеристикам, как точность, полнота, время работы алгоритма, возможность осуществлять каждый анализ документа без выполнения полного цикла вычислений, количество предварительной информации, необходимой для классификации, независимость от языка. Однако методы, описанные в вышеуказанных статьях, не позволяют определить семантическую направленность текстов, а лишь классифицируют документы, отличая их по частоте вхождения слов или иным признакам. Текстовая информация, представленная на русском языке, отличается сложностью анализа, так как в отличие от английского языка порядок слов в русском языке не определяет смысл предложения. Также присутствуют категории рода, падежи, что напрямую влияет на представленную в тексте информацию. Предложенные методы показывают недостаточную эффективность при больших объемах текстовой информации, поэтому задача семантического анализа русскоязычного текста является не до конца решеной.

Предлагаемая методика. В рамках решения задачи была предложена автоматизация методики анализа русскоязычной текстовой информации путем выделения семантических единиц-индикаторов и их поиска в исследуемом тексте (на примере материалов экстремистской направленности). Для этого была подробно проанализирована и описана последовательность действий эксперта при проведении лингвистической экспертизы, что позволило выявить основные маркеры, которые позволяют сделать обоснованные предположения о принадлежности текста к той или иной смысловой категории. После выделения групп слов и присвоения весов Фишберна в рамках каждой из групп производится анализ текста [1].

Описанная в работе [1] методика была реализована в виде web-приложения. Данный подход к реализации дает возможность пользователю применять указанное программное обеспечение с любого устройства, которое имеет доступ к интернету. Приложение является многопользовательским. Обновлять web-приложение необходимо только на серверной части, что упрощает обслуживание и избавляет пользователя от установочных процедур на рабочей станции. Разработанный программный продукт позволяет обработать большое количество текстовых фрагментов за достаточно небольшой промежуток времени и на выходе получить результат в виде числа, которое отражает вероятность отнесения текста к определенной «смысловой» категории, например, категории текстов, связанных с политикой, религией, насилием и другие [1].

В основе семантического анализа текста лежит процесс определения принадлежности текстового фрагмента к выделенным смысловым группам. Каждая группа содержит в себе набор слов и устойчивых выражений, схожих по семантике.

В словесной форме алгоритм семантического анализа текста состоит из следующих этапов:

1) создание базы знаний, состоящей из слов-токенов;
2) определение семантических характеристик токенов;
3) создание групп – категорий на основе семантического значения;
4) расчет численного значения – веса, каждого токена в категории;
5) в зависимости от заданных экспертом правил на основании рассчитанных весов устанавливается семантическая направленность текста.

База слов составляется экспертами – лингвистами на основе известных публикаций, а также общих правил семантического анализа. Каждое слово может существовать в различных формах, с разными окончаниями, в различных падежах. Для того чтобы привести слово к первоначальному словарному значению, применяется метод лемматизации слова. Лемматизация – метод морфологического анализа, который сводится к приведению слова к первоначальной словарной форме (лемме) [5].

Каждому лемматизированному слову – токену эксперт ставит в соответствие численное значение его веса в категории от 0 до 1. Вес определяет значимость данного токена в категории. Чем выше вес – тем значимее употребление данного слова в тексте для категории, в которой присутствует данный токен. Токен может использоваться в различных категориях, однако вес токена в каждой категории будет разный, в зависимости от семантического значения, т.е. одно и то же слово может по-разному влиять на конечный результат семантического анализа.

Описание структуры разработанного программного продукта.

Пользователи системы. В системе предусмотрено 3 вида пользователей:

- Администратор системы;
- Эксперт;
- Пользователь.

Администратор системы осуществляет полный контроль над системой: подтверждает регистрацию новых пользователей, регистрирует новых экспертов. Имеет возможность редактировать категории, в том числе и удалять их. Зарегистрированный администратором эксперт имеет возможность создавать новые категории для анализа текстовой информации, добавлять новые токены в общий словарь, добавлять или обновлять список токенов в различных категориях, настраивать веса токенов в категориях. Диаграмма вариантов использования приведена на рисунке 1.

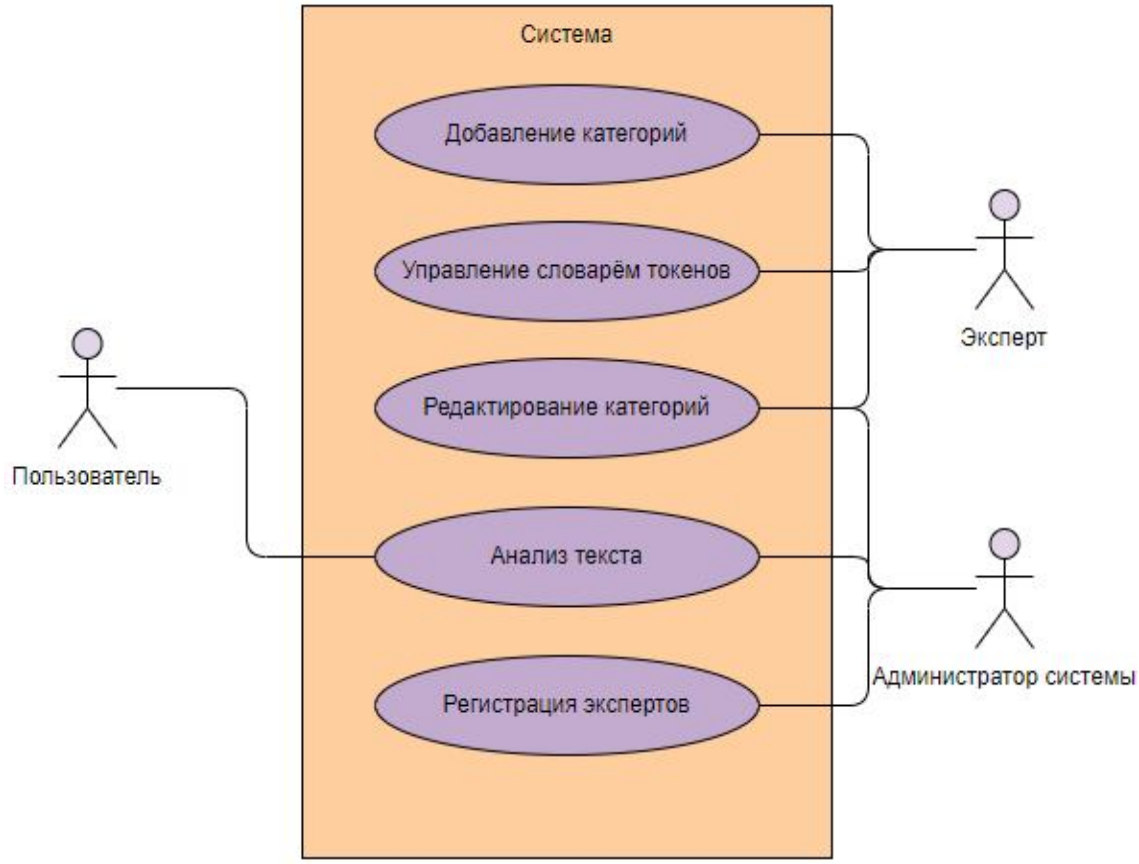


Рисунок 1 – Диаграмма вариантов использования

Пользователь может работать в системе только после подтверждения его учетной записи администратором системы. Пользователю доступно просматривать категории слов и веса токенов. Главный раздел для пользователя – анализ текстовой информации по доступным категориям, а также получение результатов.

Общий алгоритм работы программы при семантическом анализе текста. Эксперт составляет базу слов-токенов. После того как база токенов готова, эксперту необходимо создать категории. В каждую категорию эксперт добавляет токены с заданными весами. После того как базы данных заполнены – можно приступать к анализу текста.

Анализ текста включает следующие этапы:

1. Удаление знаков препинания из текстовой информации.
2. Лемматизация всех слов в тексте.
3. Поиск полученных токенов в категориях.
4. Анализ и вычисление результатов.
5. Вывод результата работы алгоритма семантического анализа текста.

Рассмотрим этапы предложенного алгоритма более подробно.

1. Удаление знаков препинания из текстовой информации. Удаление пробелов, запятых, точек и т.д. из исходного текста необходимо для сокращения объема анализируемого текста, а также правильного распознавания словосочетаний в тексте. Слова в тексте могут быть представлены в различной форме.

2. Лемматизация слова. Для приведения слова в словарную форму используется процесс лемматизации. Лемматизация слова позволяет привести все слова в тексте в начальную форму, после чего производить быстрый поиск необходимых слов – токенов во всех категориях. Для процесса лемматизации слова используется нейронная сеть, которая построена и обучена на фреймворке tensorflow [8]. В качестве датасета используется словарь, полученный в результате выполнения проекта по созданию размеченного текста – OpenCorpora. Словарь доступен бесплатно и в полном объёме по ссылке [7]. Он представляет набор морфологически, синтаксически и семантически размеченных текстов на русском языке.

3. Поиск токена слова в категориях. После процесса лемматизации система начинает поиск токенов слов в заранее сформированной экспертами базе данных категорий и токенов. Реляционная база данных представляет собой набор данных, организованных в виде таблиц, которые состоят из строк и столбцов. Общая схема алгоритма поиска токенов представлена на рисунке 2.

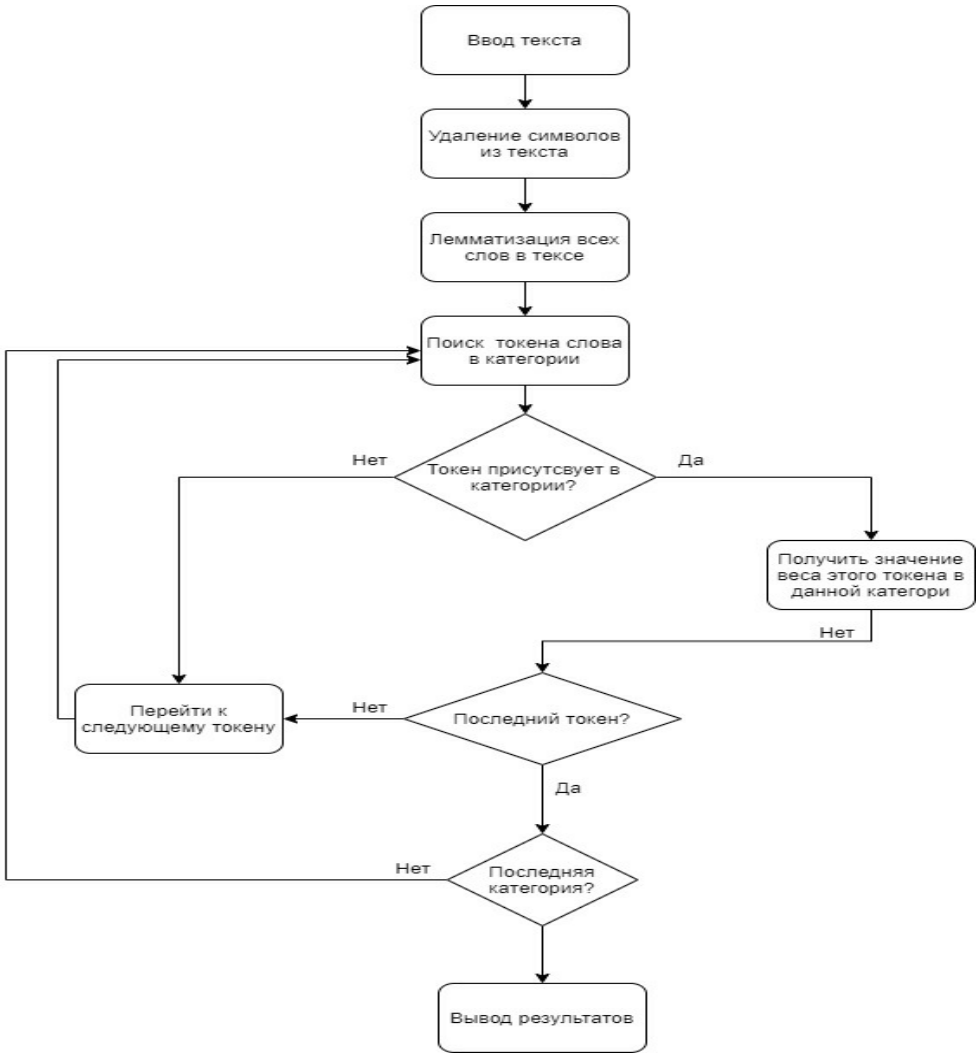


Рисунок 2 – Схема алгоритма для поиска токенов в категориях

4. Анализ и вычисление результатов. Для каждого найденного *i*-го токена в категории вычисляется нормированное значение (*V*) веса с применением метода Фишберна по следующей формуле [1]:

$$V_i = \frac{B_i}{\sum_{k=1}^{z} n_k B_k},$$

где n – количество слов в категории; B_i – оценка существенности слова в категории, так как одно и то же слово может иметь разный вес в различных категориях; k – индекс категории слов.

После вычисления нормированного значения веса для каждого найденного токена в категории необходимо рассчитать параметр γ_m (принадлежность текста к категории m), который характеризует принадлежность текстовой информации к m-й категории по следующей формуле:

$$\gamma_m = \sum_{i=1}^{n} s_i V_i,$$

где s_i – частота повтора *i*-го слова в тексте; m – номер категории, в которой имеется рассматриваемый токен.

Описание структуры разработанного приложения. Приложение имеет серверную часть, написанную на языке программирование C# 8.0 с использованием платформы .net core 3.1. Одно из основных преимуществ данной платформы – возможность использовать любую операционную систему для установки приложения. Для работы с базой данных используется Entity Framework, который позволяет абстрагироваться от используемой базы данных и работать с моделями данных вне зависимости

от типа хранилища. Таким образом мы получаем гибкую информационную систему, которая позволяет запустить приложение с использованием различных баз данных и операционных систем.

Для разработки клиентской части приложения использовались технологии Razor и библиотека Bootstrap. Razor легко интегрируется в среду разработки и позволяет гибко использовать данные, полученные от серверной части приложения. В качестве компонентов веб-интерфейса используются элементы библиотеки Bootstrap.

Схема взаимодействия компонентов разработанной системы представлена на рисунке 3.

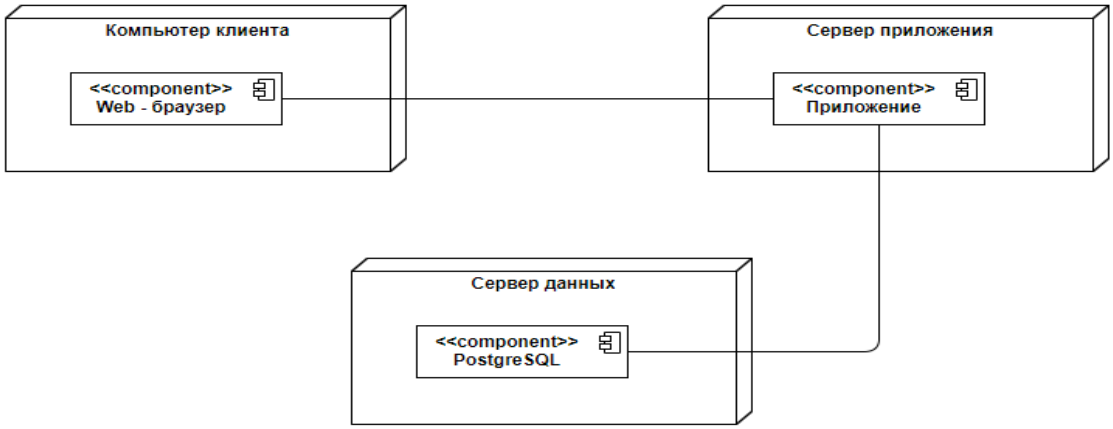


Рисунок 3 – Схема взаимодействия компонентов

В качестве хранилища данных была выбрана свободная объектно-реляционная система управления базами данных PostgreSQL.

Структура базы данных. Разработанное приложение использует реляционную базу данных, преимущество использования которой заключается в том, что при доработке функционала системы изменения в структуре базы данных сводятся к минимуму и не затрагивают хранимые данные. Данные создаются и обновляются с помощью структурированных запросов – sql. Запросы могут извлекать из базы как информацию из одной таблицы, так и связанные сущности из разных таблиц. Такой подход позволяет быстро обрабатывать результаты выборки, что для семантического анализа текста является важной составляющей. Модель данных приведена на рисунке 4.

Для работы программы эксперту необходимо создать категории, а также добавить в них токены, по которым будет производиться анализ текста. В схеме базы данных присутствует таблица токенов, которая связана с таблицей категорий, благодаря чему один токен может принадлежать к разным категориям.

Эксперт добавляет слова в любой форме, после чего при помощи нейронной сети определяется лемма данного слова – токен. При добавлении токена в категорию необходимо задать его вес в категории.

Вес токена в категории представляет собой целочисленное значение, которое указывает на то, насколько наличие слова-токена в тексте относит его к категории m. Чем выше это значение, тем выше вероятность употребления слова в рассматриваемой категории. Добавление новых токенов происходит на вкладке «Управление словарем». Добавляя новое слово, пользователю необходимо добавить краткое описание, которое может состоять как из морфологического значения, так и примеров употребления данного токена в тексте.

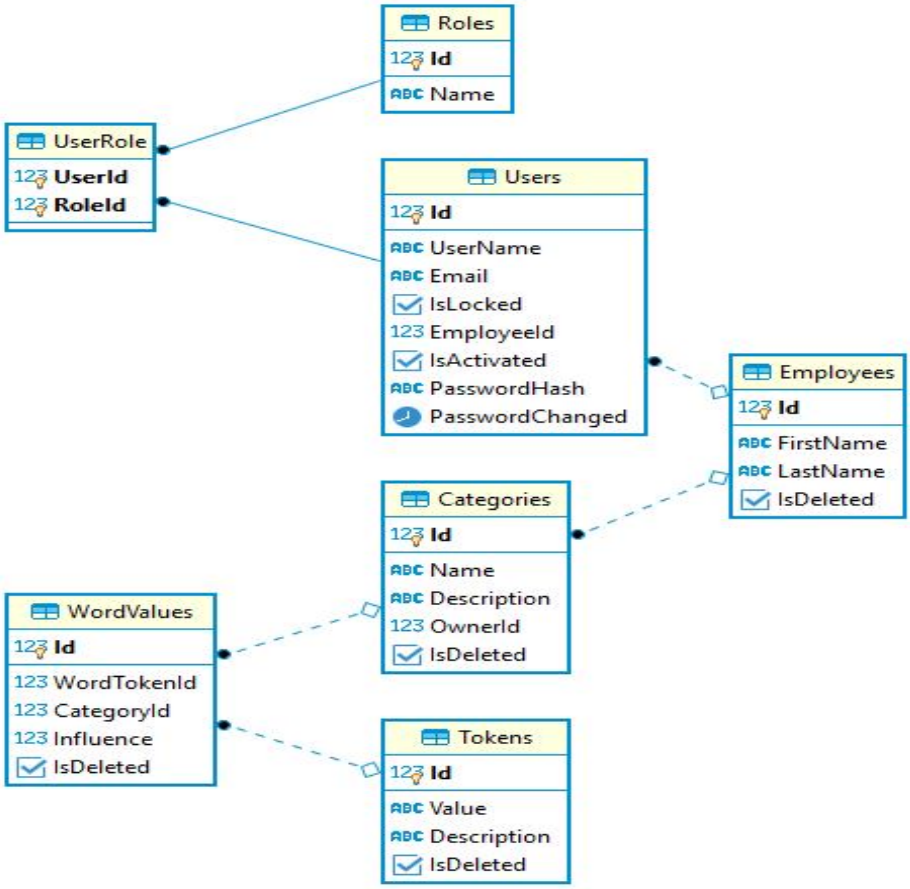


Рисунок 4 – Модель данных

Существует два способа добавления новой категории в систему:

1) можно на вкладке «Управление категориями» добавить название и описание новой категории. После этого при помощи поиска добавить в нее новые слова и установить значение веса для каждого слова в данной категории;

2) можно заполнить заранее структурированный файл в формате *JSON (рис. 5), в который эксперт добавляет новую категорию вместе с набором слов-токенов, которые ее характеризуют.

```
{
  "categoryName": "Экспрессивность",
  "categoryDescription": "Возможны негативные эмоции значительной интенсивности",
  "words": [
    {
        "tokenValue" : "Избавиться",
        "tokenDescription" : "Поможет избавиться",
        "tokenInfluence" : 3
    },
    {
        "tokenValue" : "Захлебнуться",
        "tokenDescription" : "Мир захлебнулся в крови",
        "tokenInfluence" : 2
    },
    {
        "tokenValue" : "Распоряжаться",
        "tokenDescription" : "Распоряжаться судьбами других",
        "tokenInfluence" : 1
    },
```

Рисунок 5 – Структура файла импорта категории

Для проведения анализа по существующим категориям пользователь на главной странице приложения вводит текст любого объема в соответствующее поле. После чего необходимо нажать на кнопку «Анализировать». Будет запущен процесс для семантического анализа текстовой информации. После проведения анализа пользователь получит результат анализа.

Результатом работы программного продукта является список категорий с указанием параметра γ_m (принадлежность текста к категории), который характеризует принадлежность текстовой информации к определенной категории.

Пример работы приложения. Для использования приложения пользователю необходимо выполнить вход в систему с помощью подтвержденной администратором учетной записи. На главной странице web-сервиса присутствует поле для ввода текстовой информации, куда пользователю нужно ввести текстовую информацию. После нажатия кнопки «Анализировать» запустится процесс анализа. Скриншот работы программного продукта приведен на рисунке 6.

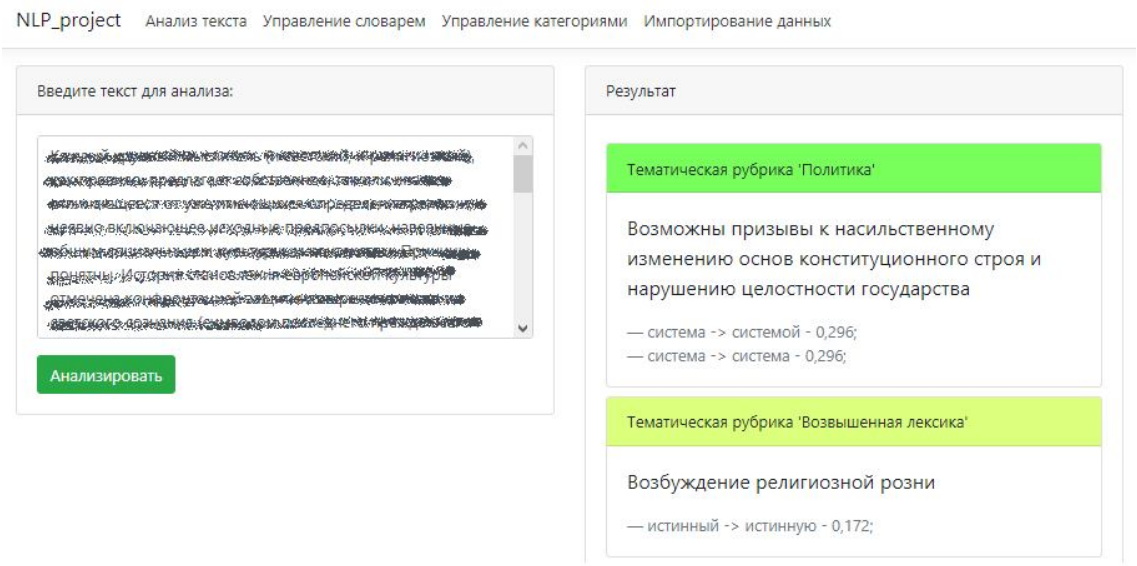


Рисунок 6 – Скриншот результатов работы программного продукта для семантического анализа текстовой информации

Особенность данного визуального представления результатов дает возможность анализирующему увидеть, наличие какого именно слова в тексте способствовало тому, что он был отнесен к той или иной смысловой категории, и самостоятельно принять решение о том, есть ли необходимость более детальной проверки подозрительного текста.

Заключение. Реализация предложенной ранее методики в виде web-сервиса дает возможность анализировать текстовую информацию на русском языке для определения ее семантической направленности. Автоматизация анализа текстовой информации позволяет экономить время на проведение сложной работы по определению семантической направленности полученной информации. Так как данное решение можно использовать для анализа любой текстовой информации, используя различные подготовленные экспертами категории, это позволит сэкономить бюджет различных компаний и государственных организаций за счет сокращения времени и затрат на услуги экспертов-лингвистов.

Библиографический список

1. Ажмухамедов И. М. Методы автоматизации анализа текстовой информации на русском языке с целью выявления ее семантической направленности / И. М. Ажмухамедов, Е. Е. Завьялова, В. Ю. Кузнецова // Прикаспийский журнал: управление и высокие технологии. – 2020. – № 2. – С. 118–126.

2. Батура Т. В. Методы автоматической классификации текстов / Т. В. Батура // Программные продукты и системы. – 2017. – № 1. – С. 3–6.

3. Епрев А. С. Автоматическая классификация текстовых документов / А. С. Епрев // Математические структуры и моделирование. – 2010. – № 21. – С. 65–81.

4. Ерхов Р. В. Преимущества разработки веб-приложения на платформе asp.net core / Р. В. Ерхов // Новые информационные технологии в научных исследованиях. – Рязань : Рязанский гос. радиотехнический ун-т, 2017. – С. 128–130.

5. Многофункциональный многоязычный словарь «Викисловарь». – Режим доступа: https://ru.wiktionary.org, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 02.08.2020).

6. Ожегов С. И. Толковый словарь русского языка: 100 000 слов, терминов и выражений / С. И. Ожегов ; под общ. ред. Л. И. Скворцова. – Москва : Мир и образование, 2015. – 1375 с.

7. Проект по созданию размеченного корпуса текстов «OpenCorpora». – Режим доступа: http://opencorpora.org, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 02.08.2020).

8. Сервис с открытым исходным для машинного обучения. – Режим доступа: https://www.tensorflow.org, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 12.08.2020).

9. Турдаков Д. Ю. Семантический анализ текстов с использованием системы Texterra / Д. Ю. Турдаков, И. А. Андрианов, Н. А. Астраханцев, В. Д. Майоров, Я. Р. Недумов, А. А. Сысоев, Д. Г. Федоренко. – Режим доступа: http://www.dialog-21.ru/digests/dialog2014/materials/pdf/TurdakovDY.pdf, свободный. – Заглавие с экрана. – Яз. рус. (дата обращения: 10.08.2020).

10. Шарапов Н. Р. Архитектура технологии разработки веб-приложений asp.net core mvc / Н. Р. Шарапов // Вопросы науки и образования. – 2018. – № 13 – С. 30–31.

11. Юркин С. Ю. Авторизация и аутентификация пользователей в веб-приложениях на основе asp.net core identity / С. Ю. Юркин, Д. Е. Турчин // Сборник материалов IX Всероссийской научно-практической конференции молодых ученых с международным участием «Россия молодая» / Кузбасский гос. тех. ун-т имени Т.Ф. Горбачева. – Кемерово, 2017. – 420 с.

References

1. Azhmukhamedov I. M., Zavyalova Ye. Ye., Kuznetsova V. Yu. *Metody avtomatizatsii analiza tekstovoy informatsii na russkom yazyke s tselyu vyyavleniya ee semanticheskoy napravlennosti* [Methods for automating the analysis of textual information in Russian in order to identify its semantic orientation]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Control and High Technologies], 2020, no. 2, pp. 118–126.

2. Batura T. V. Metody avtomaticheskoy klassifikatsii tekstov [Methods for automatic classification of texts]. *Programmnye produkty i sistemy* [Software products and systems], 2017, no. 1, pp. 3–6.

3. Eprev A. S. Avtomaticheskaya klassifikatsiya tekstovykh dokumentov [Automatic classification of text documents]. *Matematicheskie struktury i modelirovanie* [Mathematical structures and modeling], 2010, no. 21, pp. 65–81.

4. Erkhov R. V. Preimushchestva razrabotki veb-prilozheniya na platforme asp.net core [Advantages of developing a web application on the ASP.NET CORE platform]. *Novye informatsionnye tekhnologii v nauchnykh issledovaniyakh* [New information technologies in scientific research]. Ryazan, Ryazan State Radioengineering University, 2017, pp. 128–130.

5. *Mnogofunktsionalnyy mnogoyazychnyyslovar «Vikislovar»* [Multifunctional multilingual dictionary "Wiktionary"]. Available at: https://ru.wiktionary.org (accessed 02.08.2020).

6. Ozhegov S. I., Skvortsova L. I. (ed.) *Tolkovyy slovar russkogo yazyka: 100 000 slov, terminov i vyrazheniy* [Explanatory dictionary of the Russian language: 100,000 words, terms and expressions]. Moscow, Mir i obrazovanie Publ., 2015. 1375 p.

7. *Proekt po sozdaniyu razmechennogo korpusa tekstov «OpenCorpora»* [Project for the creation of a marked-up text corpus "Opensorpore"]. Available at: http://opencorpora.org (accessed 02.08.2020).

8. *Servis s otkrytym iskhodnym dlya mashinnogo obucheniya* [Open-source service for machine learning]. Available at: https://www.tensorflow.org (accessed 12.08.2020).

9. Turdakov D. Yu., Andrianov I. A., Astrakhantsev N. A., Mayorov V. D., Nedumov Ya. R., Sysoev A. A., Fedorenko D. G. *Semanticheskiy analiz tekstov s ispolzovaniem sistemy Texterra* [Semantic analysis of texts using the Texterra system]. Available at: http://www.dialog-21.ru/digests/dialog2014/materials/pdf/TurdakovDY.pdf (accessed 10.08.2020).

10. Sharapov N. R. Arkhitektura tekhnologii razrabotki veb-prilozheniy asp.net core mvc [Architecture of web application development technology asp.net mvc core]. *Voprosy nauki i obrazovaniya* [Issues of Science and Education], 2018, no. 13, pp. 30–31.

11. Yurkin S. Yu., Turchin D. E. Avtorizatsiya i autentifikatsiya polzovateley v veb-prilozheniyakh na osnove asp.net core identity [User authorization and authentication in web-based applications asp.net core identity]. *Sbornik materialov IX Vserossiyskoy nauchno-prakticheskoy konferentsii molodykh uchenykh s mezhdunarodnym uchastiem "Rossiya molodaya"* [Collection of Materials of the IX All-Russian Scientific and Practical Conference of Young Scientists with the International Participation "Rossiya molodaya"]. Kemerovo, 2017. 420 p.