**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**Отчёт о выполнении заданий**

Дисциплина: «Статистические методы»

Студент: Леонов Владислав Денисович

Учебная группа: 224-322

Преподаватель 1: Филиппович Юрий Николаевич

Преподаватель 2: Воробьев Никита Григорьевич

**Москва 2024**

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc164950580)

[**ПРАКТИЧЕСКАЯ РАБОТА 1** 4](#_Toc164950581)

[**ПРАКТИЧЕСКАЯ РАБОТА 2** 5](#_Toc164950582)

[**ЛАБОРАТОРНАЯ РАБОТА 1** 6](#_Toc164950583)

[**ЛАБОРАТОРНАЯ РАБОТА 2** 8](#_Toc164950584)

[**ЛАБОРАТОРНАЯ РАБОТА 3** 10](#_Toc164950585)

[**ЛАБОРАТОРНАЯ РАБОТА 4** 11](#_Toc164950586)

[**ВЫВОДЫ** 15](#_Toc164950587)

[**ЗАКЛЮЧЕНИЕ** 16](#_Toc164950588)

[**ЛИТЕРАТУРА** 17](#_Toc164950589)

# **ВВЕДЕНИЕ**

В рамках разработки предметной области по созданию чат-бота и по работе с анализом больших объемов данных из открытых источников, были выбраны статьи, которые связаны с темой ВКР «Адаптивный интерфейс САПР на основе нейросетевого анализа пользовательских логов в задачах предсказания команд».

Для данной ВКР потребуются как статьи с пониманием инструментов, которые требуются для решения задачи, так и информация о работе используемых систем. Потому основным набором статей стали те, которые описывают работу ВКР.

Для прохождения курса необходимо выполнить задачи:

1. Собрать ЕЯ текстовые описания ПО.
2. Составить набор данных для обучения векторной модели.
3. Обучить модель Word2Vec, используя собранные данные.
4. Провести рубрицирование ЕЯ текстовых данных описания ПО.
5. Составить набор справочных данных, описывающих рубрики исследуемой ПО.
6. Создать автоматическую систему, предлагающую пользователю справочную информацию при запросе темы с использованием ЕЯ.

# **ПРАКТИЧЕСКАЯ РАБОТА 1**

При сборе ЕЯ текстовых описаний ПО были выполнены следующие работы:

1. Были найдены 30 статей, соответствующих теме ВКР.
2. Был сохранен список названия статей и ссылок на них.
3. Было проведено ручное рубрицирование текстов.

При проведении ручного рубрицирования текстов было изучено их содержание. Далее для каждого текста были выделены 5-10 ключевых слов/словосочетаний, с помощью которых можно было составить поисковые образы. После этого статьи были разделены на 5 категорий.

Все данные ручного рубрицирования были собраны в виде таблицы со столбцами со следующими названиями «№», «Название статьи», «Ссылка на статью», «Дата обращения», «Ключевые слова», «Рубрика» (Рис. 1).



Рисунок 1 – Задание 1

# **ПРАКТИЧЕСКАЯ РАБОТА 2**

При создании обучающего датасета были выполнены следующие работы:

* + 1. Была извлечена текстовая информация из найденных в первом задании статей и сохранена в формате docx.
    2. Файлы docx были собраны в единый файл .txt.
    3. Полученный файл был конвертирован в формат .tsv.

При извлечении текстовой информации из найденных в [Задании 1](file:///G:\User%20Folders\Downloads\Telegram%20Desktop\321%20Шальнов%20курсовой%20проект.docx#_2_ЗАДАНИЕ_1) статей файлы были удалены лишние разделы.

В результате выполнения данного пункта был получен файл «\*.tsv», являющийся обучающим датасетом (Рис. 2).

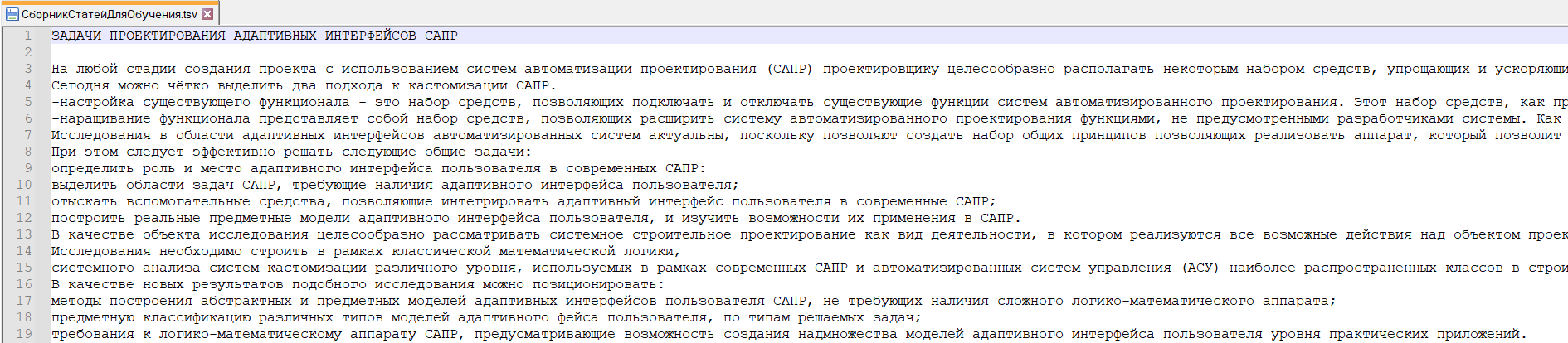


Рисунок 2 – Задание 2

# **ЛАБОРАТОРНАЯ РАБОТА 1**

Для обучения модели Word2Vec был получен шаблон у преподавателя. Далее были установлены следующие библиотеки, используемые в шаблоне.

Сначала происходит импорт необходимых библиотек. Далее происходит загрузка данных из файла с датасетом «\*.tsv» с предобработкой.

После обучения модели происходит вывод векторного представления слова «технологий» для проверки того, что обучения модели произошло корректно. Далее модель сохраняется в файл «\*.model» (Рис. 3).

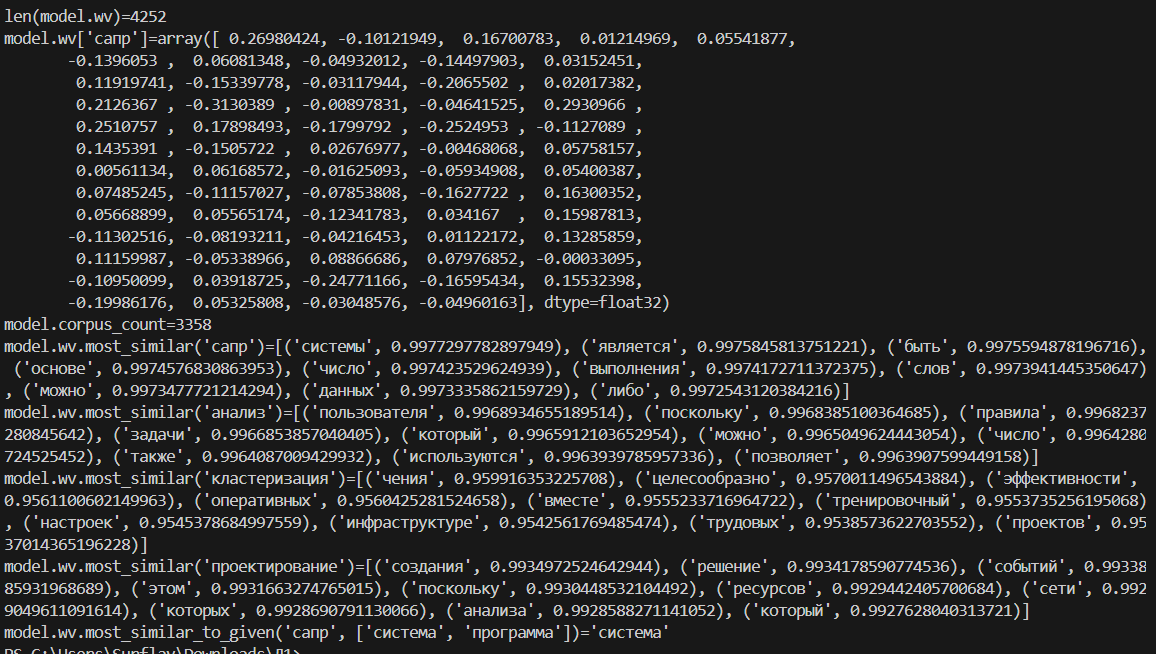


Рисунок 3 – Задание 3

Обучение модели Word2Vec происходило со следующими параметрами:

* «min\_count» = 5 (слова, которые встречаются менее 5 раз, будут игнорироваться, причина столь низкого значения заключается в проблеме с текстами. В них встречается обилие слов, которые не относятся к темам работ, а сами же темы упоминаются не так частно. Потому возникла проблема с небольшим количеством упоминаний слов.);
* «window» = 2 (размер контекстного окна для обучения, максимальное расстояние между текущим словом и целевым словом);
* «vector\_size» = 64 (размерность векторного представления слов);
* «alpha» = 0.03 (начальная скорость обучения);
* «negative» = 1 (количество отрицательных образцов, которые нужно сгенерировать для каждого положительного образца, при увеличении параметра происходило обильное просачивание прилагательных и служебных слов в финальную работу);
* «min\_alpha» = 0.0007 (конечная скорость обучения);
* «sample» = 6e-5 (порог для снижения частоты частотных слов).

# **ЛАБОРАТОРНАЯ РАБОТА 2**

Для автоматического рубрицирования датасет был разделен по текстам. На рисунке 4 представлен результат разделения датасета.

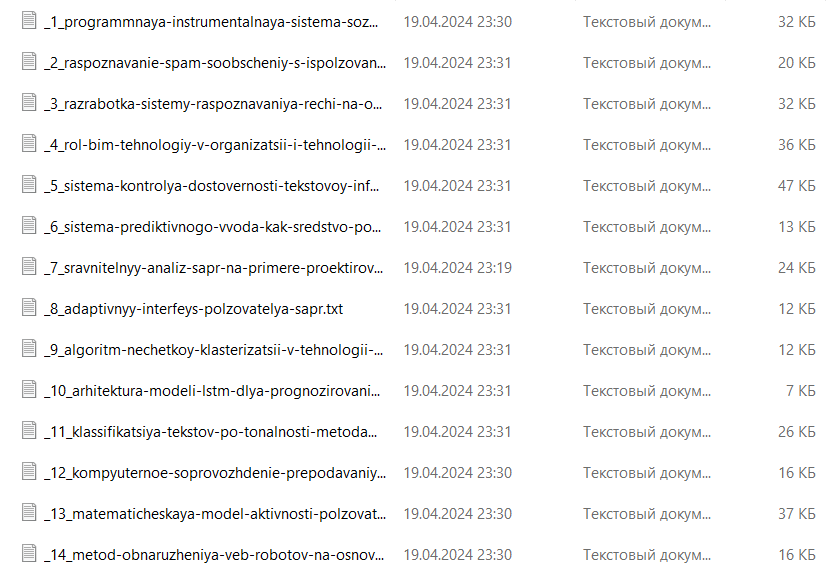


Рисунок 4 – Результат разделения датасета по текстам

Далее был разработан код, который производит автоматическое рубрицирование текстов. На рисунке 5 представлен результат работы кода.

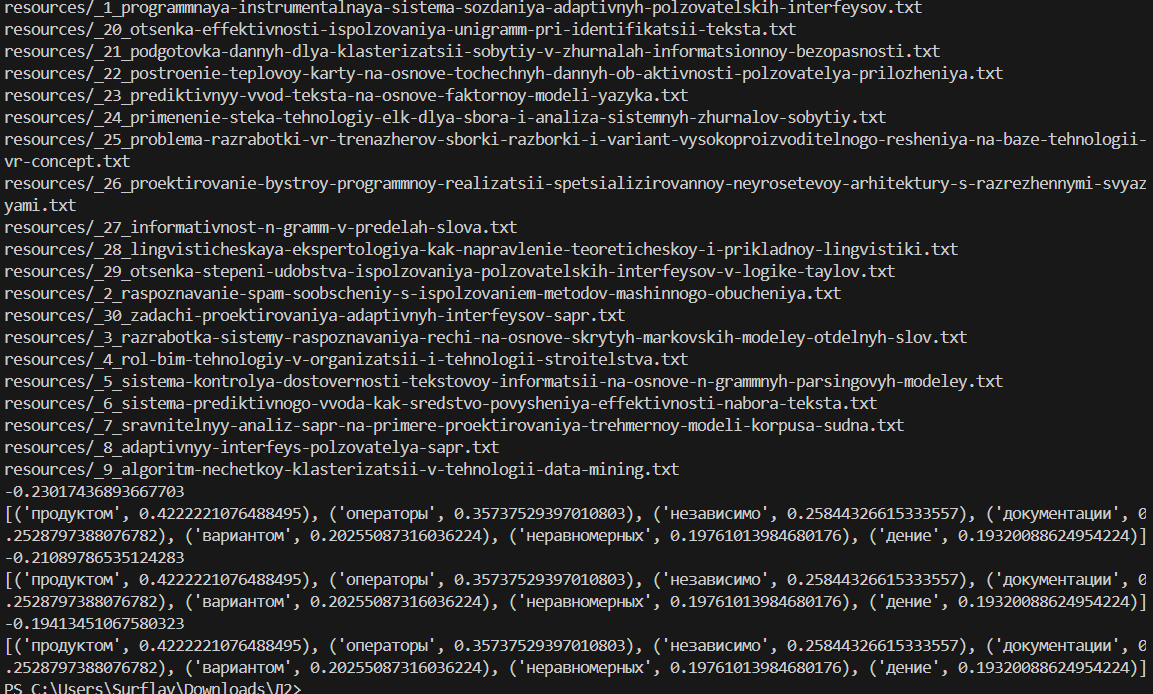


Рисунок 5 – Результаты автоматического рубрицирования

# **ЛАБОРАТОРНАЯ РАБОТА 3**

Для выполнения задания выбрал описание тем, фугирующих в статья, входящих в категорию. Таблица состоит из вектора, категории, выделенной из статей, а также описания категории (Рис. 6).

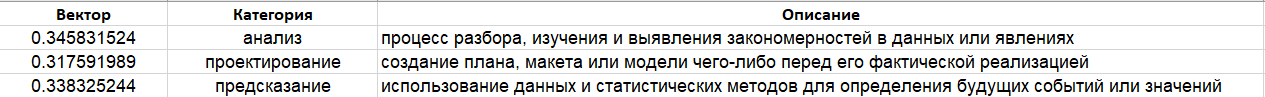


Рисунок 6 – Задание 5

# **ЛАБОРАТОРНАЯ РАБОТА 4**

В ходе выполнения задания 6 получилось 2 таблицы со связью один ко многим. На основе таблиц была создана база данных, структура базы данных представлена на рисунке 7.

A black line with a white background

Description automatically generated

Рисунок 7 – Структура базы данных

База состоит из двух таблиц, в одной хранятся наименования рубрик и их векторные значения (таблица rubrics), и таблицы rubrics\_word, которая хранит в себе основные определения каждой из рубрик. Заполнение таблиц rubrics и rubrics\_word представлено на рисунках 8 и 9.

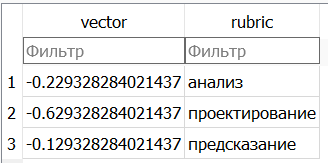


Рисунок 8 – Заполнение таблицы rubrics



Рисунок 9 – Заполнение таблицы rubrics\_word

С использованием созданной таблицы был реализован алгоритм, который выполняет действия:

1. Пользователь вводит запрос;
2. Система разбивает запрос на массив слов;
3. Для каждого отдельного слова создается нормализованный вектор из обученной модели;
4. Рассчитывается среднее значение вектора запроса;
5. Из таблицы rubric выбирается поле с наиболее близким числовым значением категории;
6. Пользователю выводится информация о теме, которое система смогла найти в его запросе, если же определение не было найдено, то выводится случайная информация по теме.
7. Система запрашивает у пользователя подтверждение правильности выбора рубрики
8. Если рубрика определена верно, то пользователю предлагается вновь сделать запрос
9. Если же рубрика была определена неверно, то к значению вектора рубрики, указанной пользователем как правильная, прибавляется или вычитается, в зависимости от значения вектора введенного запроса, число равное корню квадратному модуля разности вектора запроса и вектора рубрики.

На рисунке 10 представлен интерфейс программы для взаимодействия с чат-ботом.

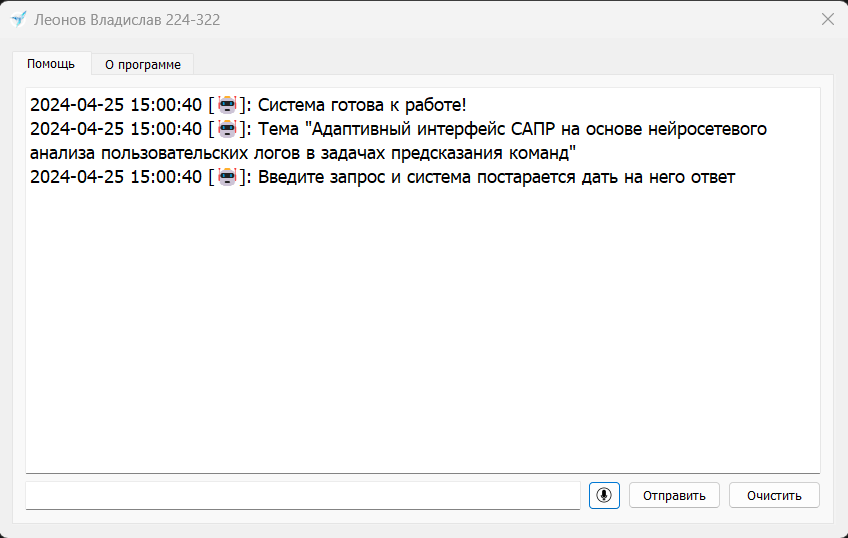


Рисунок 10 – Интерфейс программы

На рисунке 11 представлен ответ на вопрос пользоватяля.

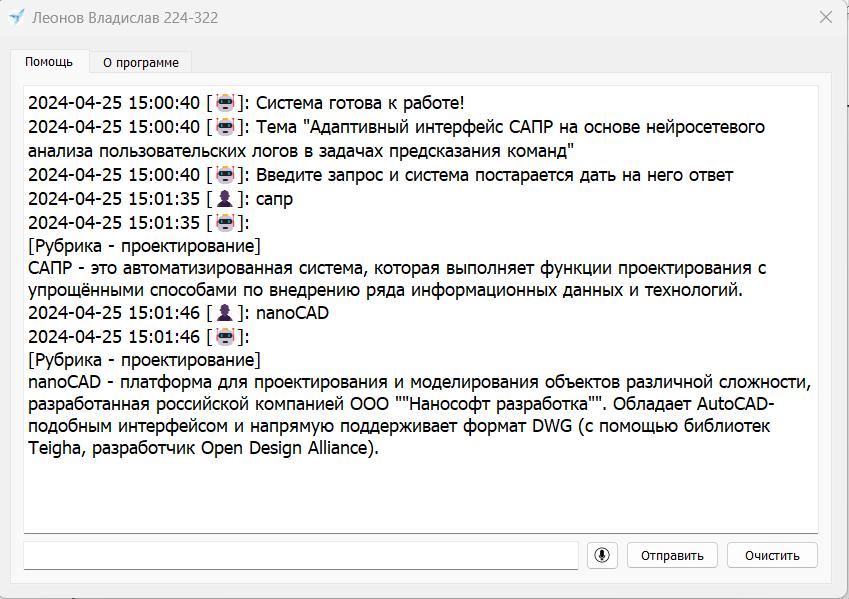


Рисунок 11 – Ответ на вопрос пользователя

# **ВЫВОДЫ**

Рассматривая применение векторного решения для рубрицирования текстов можно прийти к выводу, что подобный подход является слишком примитивным для рубрицирования российских текстов. Проблема в том, что русский язык пестрит большим количеством средств выразительности, что приводит к увеличению работы по очищению текста до работы с ним векторной модели. Также играет роль и то, как люди пишут научные работы. Так, например, довольно редко упоминается сам объект исследования, что приводит к проблеме с его выделением по вектору без снижения значения min\_count. Тем самым, более выгодным решением для подобного решения будет многослойная нейронная сеть.

# **ЗАКЛЮЧЕНИЕ**

Были успешно достигнуты поставленные цели и задачи, связанные с обработкой и использованием естественно-языковых текстовых данных. В процессе проекта были выполнены следующие ключевые этапы:

* изучение материалов и освоение необходимых методов и технологий, связанных с анализом и обработкой текстовых данных. Это включало в себя изучение лекционных курсов, выполнение практических заданий и использование специального программного обеспечения;
* сбор текстовых описаний программного обеспечения (ПО) на естественном языке, которые послужили основой для дальнейших исследований;
* создание датасета для обучения векторной модели, что позволило улучшить качество анализа текстов;
* обучение модели word2vec на основе собранных данных, что сделало возможным получение векторных представлений для слов и текстов, улучшая качество анализа текстовых данных;
* рубрицирование текстовых описаний ПО, что позволило организовать информацию в систему категорий и облегчило поиск и доступ к данным;
* создание автоматической системы, предоставляющей пользователям справочную информацию на основе запросов с использованием естественно-языковых текстовых данных.

Таким образом, были приобретены навыки по работе с текстовыми данными и их автоматическому рубрицированию.

# **ЛИТЕРАТУРА**

1. Автоматическая рубрикация текстов: методы и проблемы – URL: https://cyberleninka.ru/article/n/avtomaticheskaya-rubrikatsiya-tekstov-metody-i-problemy (Дата обращения 15.04.2024);
2. Автоматическое тематическое рубрицирование сообщений средств массовой информации на основе применения технологии нейронных сетей – URL: http://engineering-science.ru/doc/56926.html (Дата обращения 05.04.2024);
3. Методы и алгоритмы рубрикации текстов – URL: https://cyberleninka.ru/article/n/metody-i-algoritmy-rubrikatsii-tekstov (Дата обращения 15.04.2024);
4. Vector Space Model для семантической классификации текстов – URL: https://habr.com/ru/sandbox/18635/ (15.04.2024);
5. Векторное представление слов – URL: <https://neerc.ifmo.ru/wiki/index.php?title=%D0%92%D0%B5%D0%BA%D1%82%D0%BE%D1%80%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%B5%D0%B4%D1%81%D1%82%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D1%81%D0%BB%D0%BE%D0%B2> (Дата обращения 15.04.2024);
6. Word2vec в картинках – URL: https://habr.com/ru/articles/446530/ (Дата обращения 15.04.2024);
7. Word2Vec: как работать с векторными представлениями слов – URL:<https://neurohive.io/ru/osnovy-data-science/word2vec-vektornye-predstavlenija-slov-dlja-mashinnogo-obuchenija/> (Дата обращения 15.04.2024);
8. Семантика и технология Word2Vec – URL: https://habr.com/ru/articles/585838/ (Дата обращения 15.04.2024).