Министерство науки и высшего образования Российской федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Факультет «Электроники и вычислительной техники»

Кафедра «Системы автоматизированного проектирования и поискового конструирования»

КОНТРОЛЬНАЯ РАБОТА

по дисциплине: «Компьютерная лингвистика»

Выполнили:

студенты 3 курса, гр. ИВТ-363

Сороко М.А.

Панасенко С.С.

Яньшина Д.Д.

Проверил:

доц. Коробкин Д.М.

Волгоград, 2021

Содержание

[Первое задание 3](#_Toc74789642)

[База данных 3](#_Toc74789643)

[Парсер сайта 4](#_Toc74789644)

[Второе задание 7](#_Toc74789645)

[Томита-парсер 7](#_Toc74789646)

[Word2Vec 8](#_Toc74789647)

[Natasha 9](#_Toc74789648)

[Третье задание 10](#_Toc74789649)

[Анализ тональности 10](#_Toc74789650)

# Первое задание

## База данных

Используется СУБД Mongodb.

Описание коллекций базы данных:

Vpravda – коллекция, в которой содержатся новостные статьи

VpravdaSentence – коллекция, в которой содержатся предложения в которых упоминались персоны и места

Objects – коллекция, в которой содержатся персоны и места, а также их идентификаторы

TonalitySentence – коллекция, в которой содержаться предложения с упоминанием персон и мест и тональность предложения

## Парсер сайта

Парсер (parser\_site.py) работает в несколько этапов:

1) Получение ссылки на сайт

2) Получение ответов от сервера

3) Получение ссылок с сайта

4) Получение информации с ссылки путем краулинга

5) После получения информации о новостной статье, идет поиск этой новости в коллекции

6) Если новость есть, то текст новости обновляется, если новости нет, то она записывается в коллекцию

Большой парсер (big\_parser\_site.py) работает по такому же принципу, но проходит по сайту 25 раз, чтобы собрать 1000 новостных статей для дальнейшего обучения модели.

Скриншот коллекции заполненной парсерами базы данных:



Рисунок 1. Коллекция Vpravda

UI-интерфейс

UI-интерфейс содержит в себе строки и столбцы с параметрами и документами коллекции в базе данных

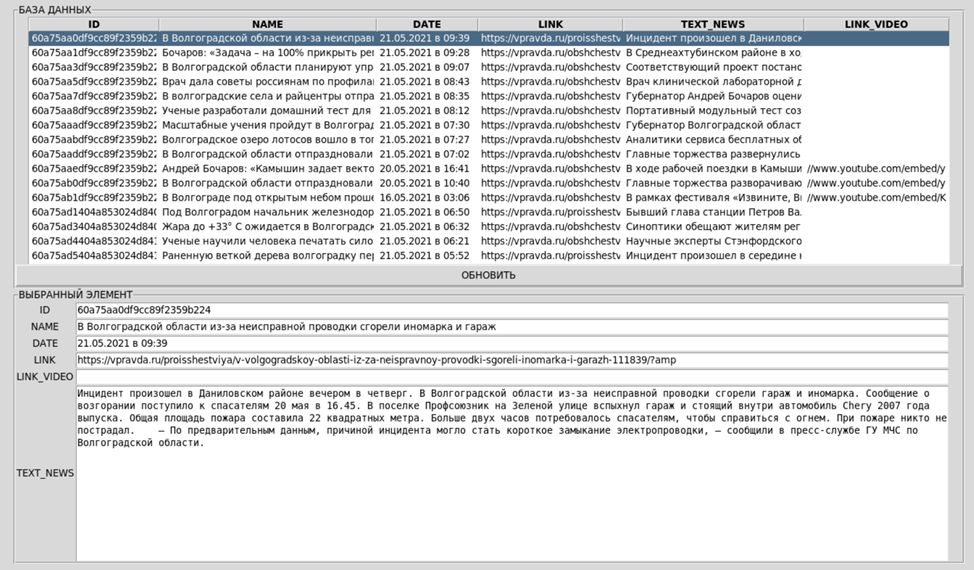


Рисунок 2. Интерфейс страницы с базой данных

# Второе задание

## Томита-парсер

Грамматики для томита-парсера генерировались на основе данных полученных с сайтов: <https://global-volgograd.ru/person> и <https://avolgograd.com/sights?obl=vgg>.

Томита-парсер работает в несколько этапов:

1) Перед анализом текста, каждая новостная статья записывается в отдельный текстовый документ

2) Далее текст одной новости записывается в input.txt

3) Запускается томита-парсер, который ищет персон и места из файлов persons.txt и places.txt

4) После завершения поиска нужных предложений, запускается парсер файла pretty.html

5) Предложения с упоминаниями записываются в коллекцию

6) Предложения с упоминаниями обрабатываются, имена и названия персон и мест изменяются на идентификаторы.

7) Измененные предложения записываются в отдельный текстовый файл

8) Измененные предложения заменяют предложения с упоминаниями в тексте новости

9) Новый текст новости перезаписывается в файл, где находился старый текст новости

## Word2Vec

Модель word2vec была обучена на измененных новостных статьях из текстовых документоа, объем которой составлял примерно ~1000 статей. Модель находится в папке /home/vagrant/tomita-parser/build/bin/packs\_text\_news/models. Запуск модуля осуществляется запуском скрипта word2vec[.py](https://github.com/asuka1211/ComputationalLinguistics/blob/master/word2vec/sinonim.py). В UI выбирается персона или место и по уже обученной модели подбираются синонимы.

Выбор: “Бочаров Андрей”

Синонимы:

губернатор

область

принять

волгоградский

обозначить

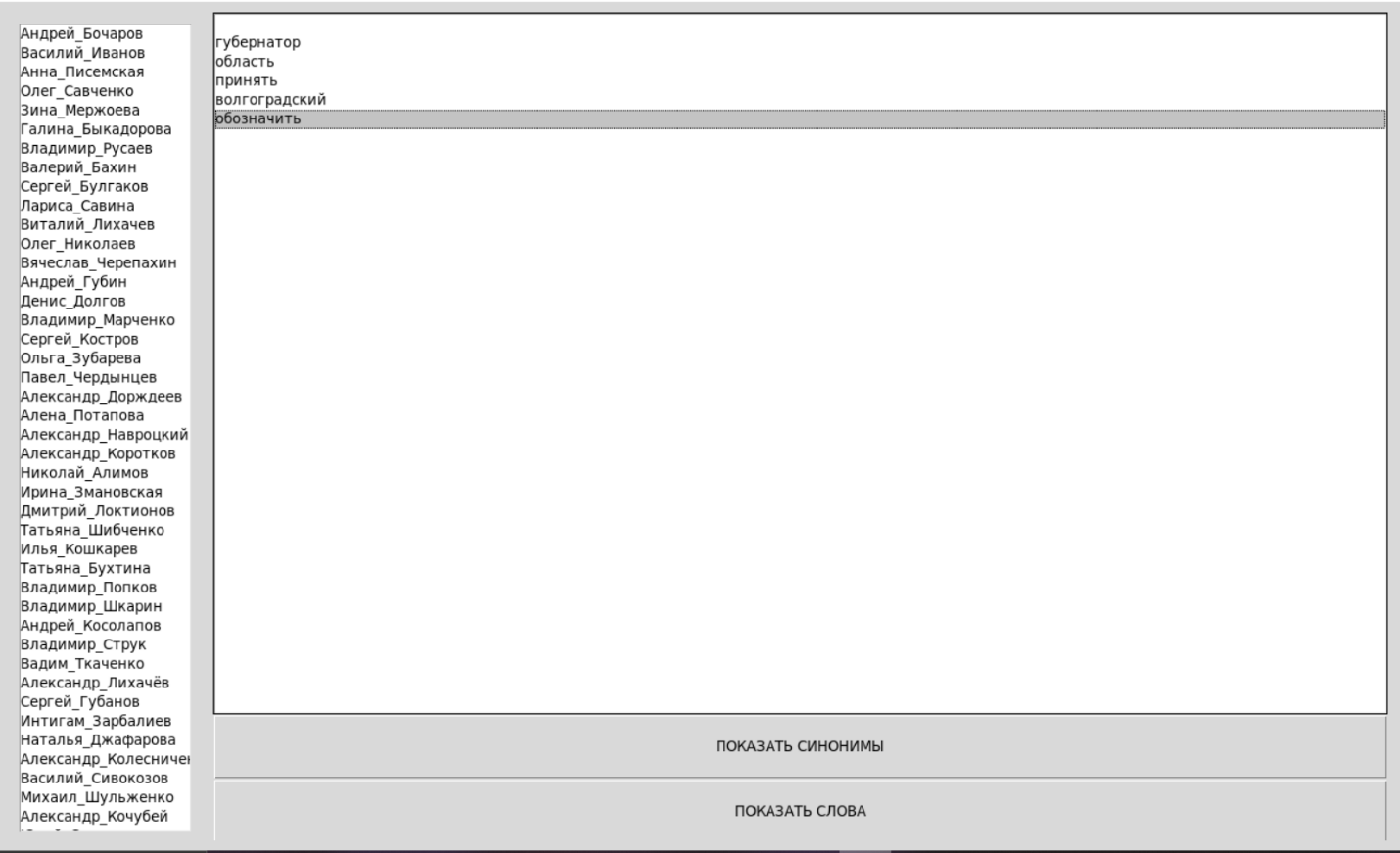


Рисунок 3. Страница контекстных синонимов

## Natasha

Для работы модуля Natasha потребуется установить библиотеку Natasha

Natasha работает в несколько этапов:

1) Измененные предложения с упоминанием персон и мест подвергаются обработке

2) Выявляются связи между словами в предложениях

3) В UI выбирается персона или места и для выбора пользователя в скрипте ищутся связи с другими словами

Выбор: “Бочаров Андрей”

Слова:

семья

москвичев

возглавить

проект

…

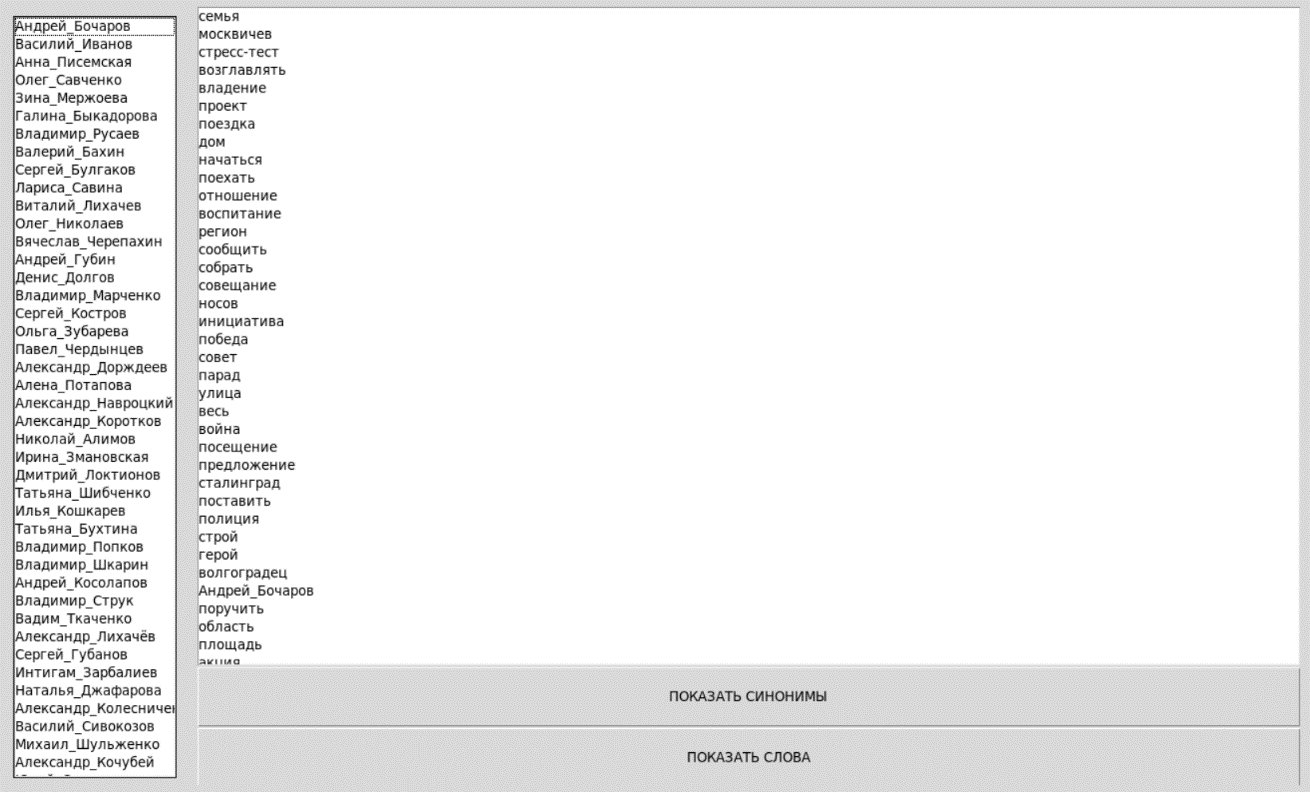


Рисунок 4. Страница контекстных синонимов

# Третье задание

## Анализ тональности

Анализ тональности производится у предложений, которые были определены томита-парсером на втором этапе работы. Для определения тональностей предложений была использована библиотека nltk. Также были использованы дата-сеты positive.csv и negative.csv. После обработки этих дата-сетов данные разделяются на обучающую выборку и тестовую выборку. После того, как модель обучилась, то в модель поступают предложения с упоминаниями персон или мест, проверяются на тональность. После проверки, предложение и его тональность записывается в коллекцию.

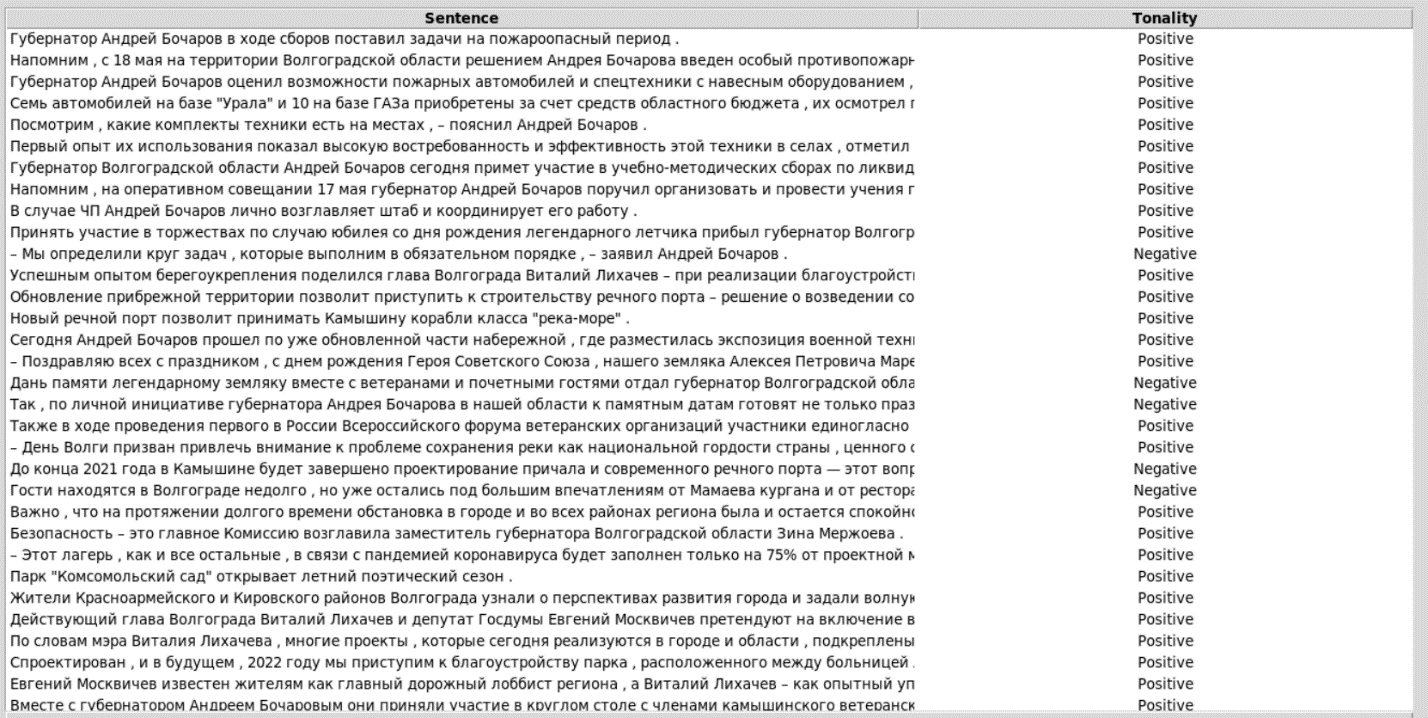


Рисунок 5. Интерфейс предложений и их тональности

Коллекция TonalitySentence:

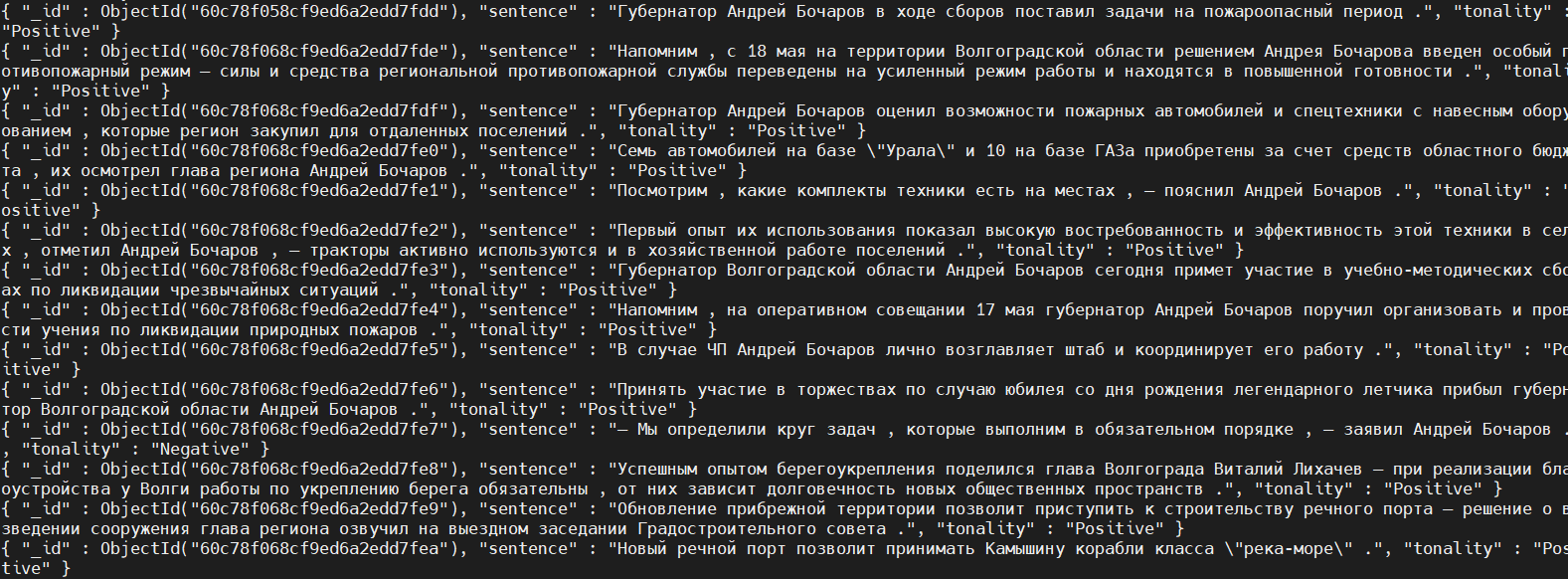


Рисунок 6. TonalitySentence