Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра интеллектуальных информационных технологий

Отчёт

по курсу «**Естественно-языковой интерфейс интеллектуальных систем**»

Лабораторная работа №2

«Разработка текстового корпуса, его менеджера»

|  |  |
| --- | --- |
| Выполнили:  студенты гр. 221701: | Крупский А. В. Глёза Е. Д. |
| Проверил: | Крапивин Ю.Б. |

Минск, 2025

# **Цель работы:**

Изучить принципы построения корпусов текстов, виды разметки и способы аннотирования, инструменты работы с корпусами текстов. Построить корпус текстов и разработать корпусный менеджер.

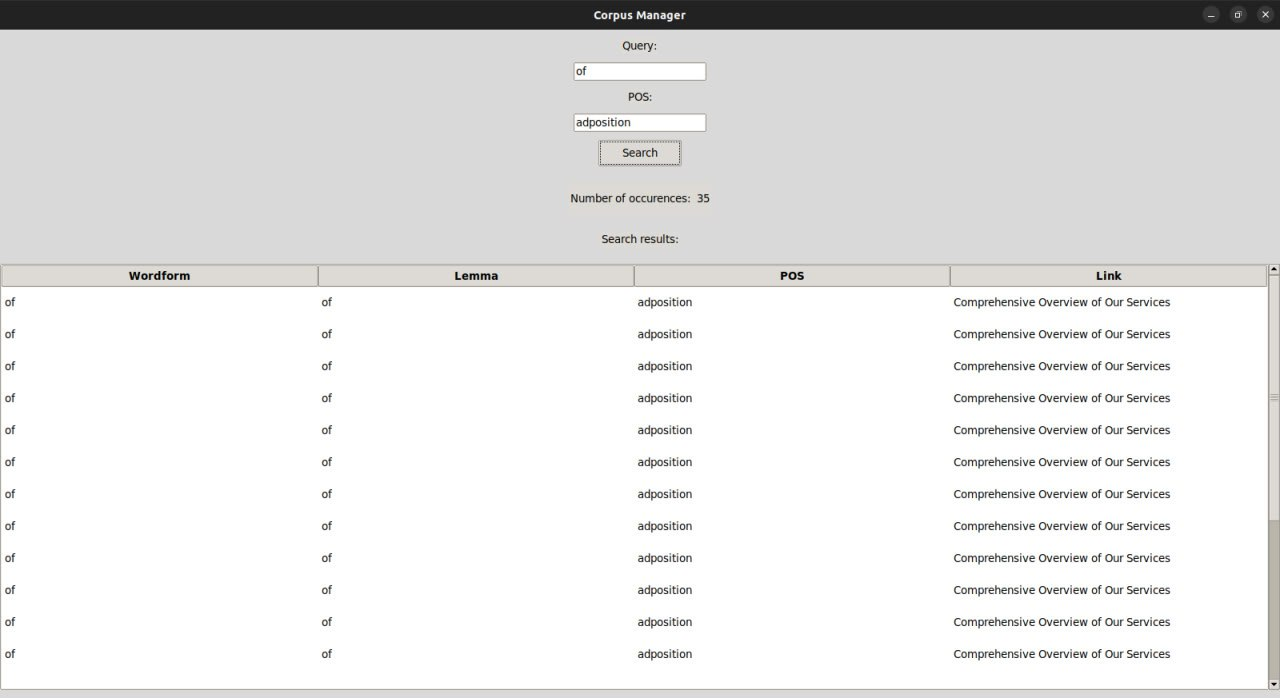
# **Задание:**

Сформировать электронный корпус текстов по выбранной предметной области (предметная область “услуги”, язык текста английский). Разработать корпусный менеджер, обеспечивающий базовую функциональность работы с созданным корпусом текстов.

**Используемые бибилиотеки:**

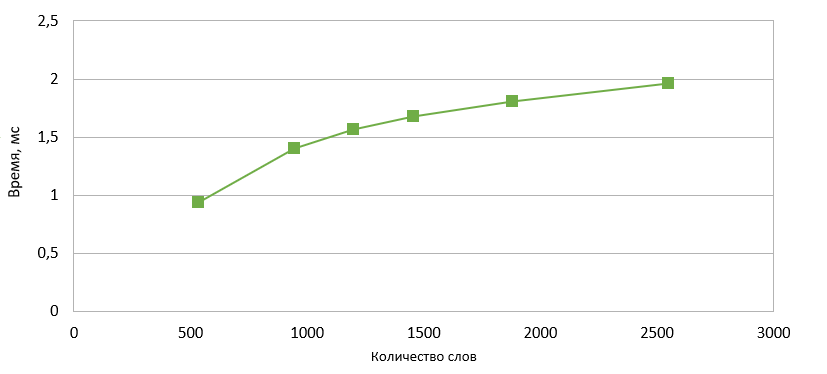
1. **Pydantic**  
   **Описание:** Библиотека для валидации и сериализации данных на основе аннотаций типов Python.  
   **Использование:** Создание моделей данных с автоматической проверкой типов, преобразование входных данных, валидация конфигураций и API-запросов.
2. **pydantic-settings**  
   **Описание:** Надстройка над Pydantic для удобной работы с конфигурацией приложений.  
   **Использование:** Загрузка и валидация конфигурации из переменных окружения, файлов и других источников.
3. **spaCy**  
   **Описание:** Высокопроизводительная библиотека для обработки естественного языка (NLP).  
   **Использование:** Токенизация, лемматизация, синтаксический разбор, извлечение сущностей и построение NLP-пайплайнов на множестве языков (включая английский и русский).
4. **SQLAlchemy**  
   **Описание:** Библиотека для работы с базами данных в Python, реализующая ORM (Object-Relational Mapping) и Core-интерфейс для SQL-запросов.  
   **Использование:** Определение моделей данных, выполнение SQL-запросов, взаимодействие с различными СУБД через единый интерфейс.
5. **typing**  
   **Описание:** Стандартная библиотека Python для поддержки статической типизации.  
   **Использование:** Аннотация типов переменных, функций и классов (например, List, Dict, Optional), повышение читаемости и качества кода, использование в инструментах проверки типов.

**Интерфейс:**

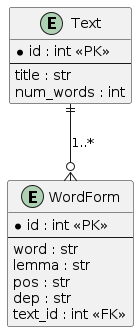


Окно программы

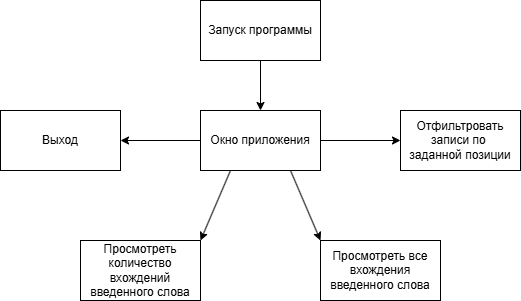
**Тест производительности:**



## Структура хранения данных



## Структурная схема приложения



**Описание алгоритмов:**

*Разбиение текста на словоформы и получение их морфологической информации:*

1. **Начало**
2. Получаем файл формата TXT/RTF
3. Считываем текст файла
4. Получаем все словоформы и сохраняем в БД
5. **Конец**

*Поиск словоформ по подстроке:*

1. **Начало**
2. Пользователь вводит строку
3. Получаем из БД список всех словоформ, которые имеют такую подстроку
4. Отображаем все словоформы и морфологическую информацию о ней в таблице
5. **Конец**

**Вывод:**

В результате выполнения данной лабораторной работы мы изучили понятие текстового корпуса и его менеджера. Текстовый корпус представляет собой набор текстовых документов, собранных для анализа и обработки в рамках определенной задачи или исследования. Он является важным инструментом в области обработки естественного языка, машинного обучения и лингвистического анализа. Данная программа имеет перспективы в области анализа текстов и книг для подведения различных статистик.