Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления

Кафедра Интеллектуальных информационных технологий

**ОТЧЁТ**

**По лабораторной работе № 4**

по дисциплине «Естественно-языковой интерфейс интеллектуальных систем»

по теме: “ Семантико-синтаксический анализ текстов естественного языка”

Выполнил: Самута Д.В.

Семенов Е. Г.

гр. 221703

Проверил: Крапивин Ю. Б.

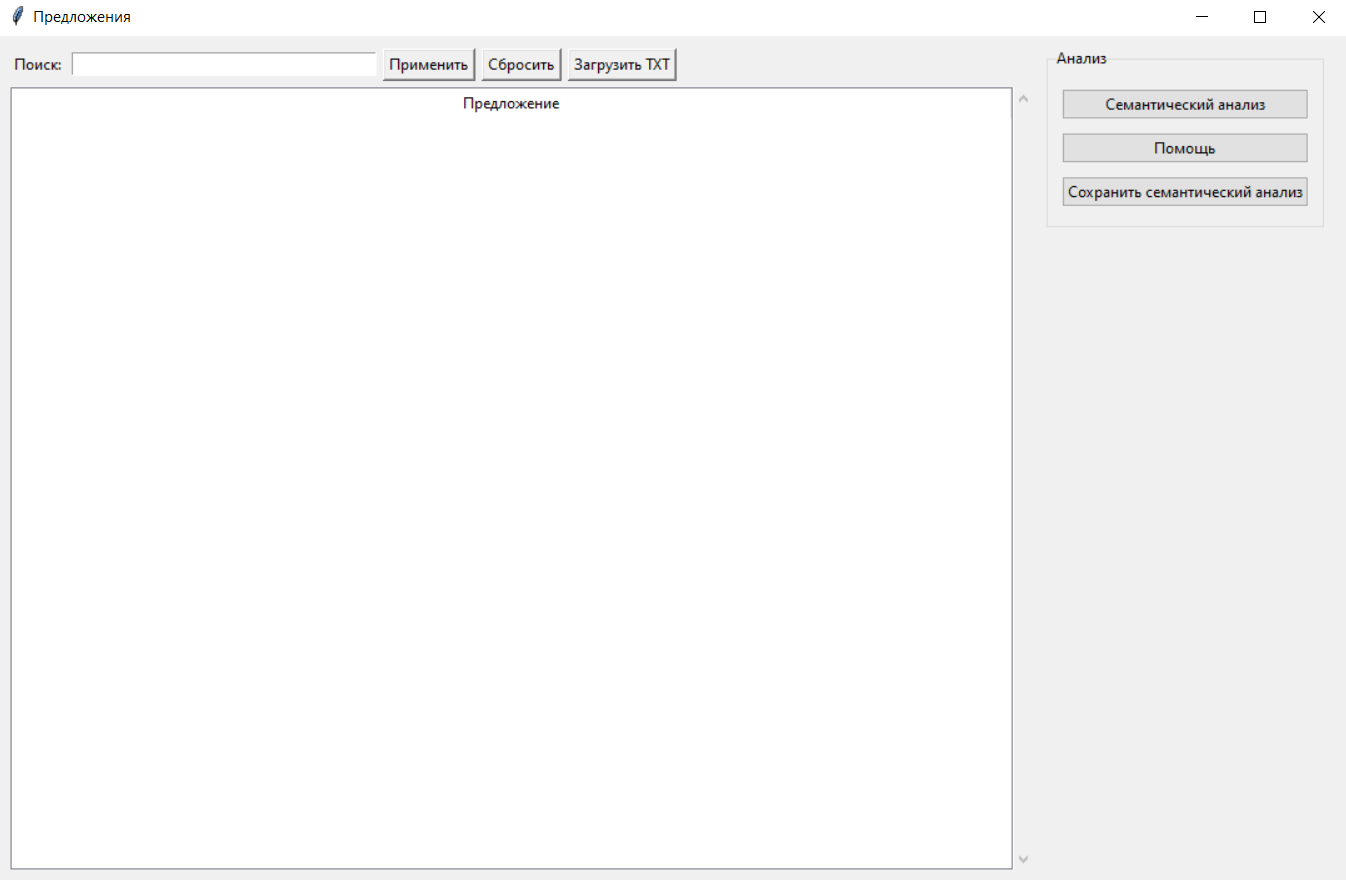
Минск 2025

**Цель работы:** освоить принципы разработки прикладных сервисных программ для решения задачи автоматического семантико-синтаксического анализа текста естественного языка.

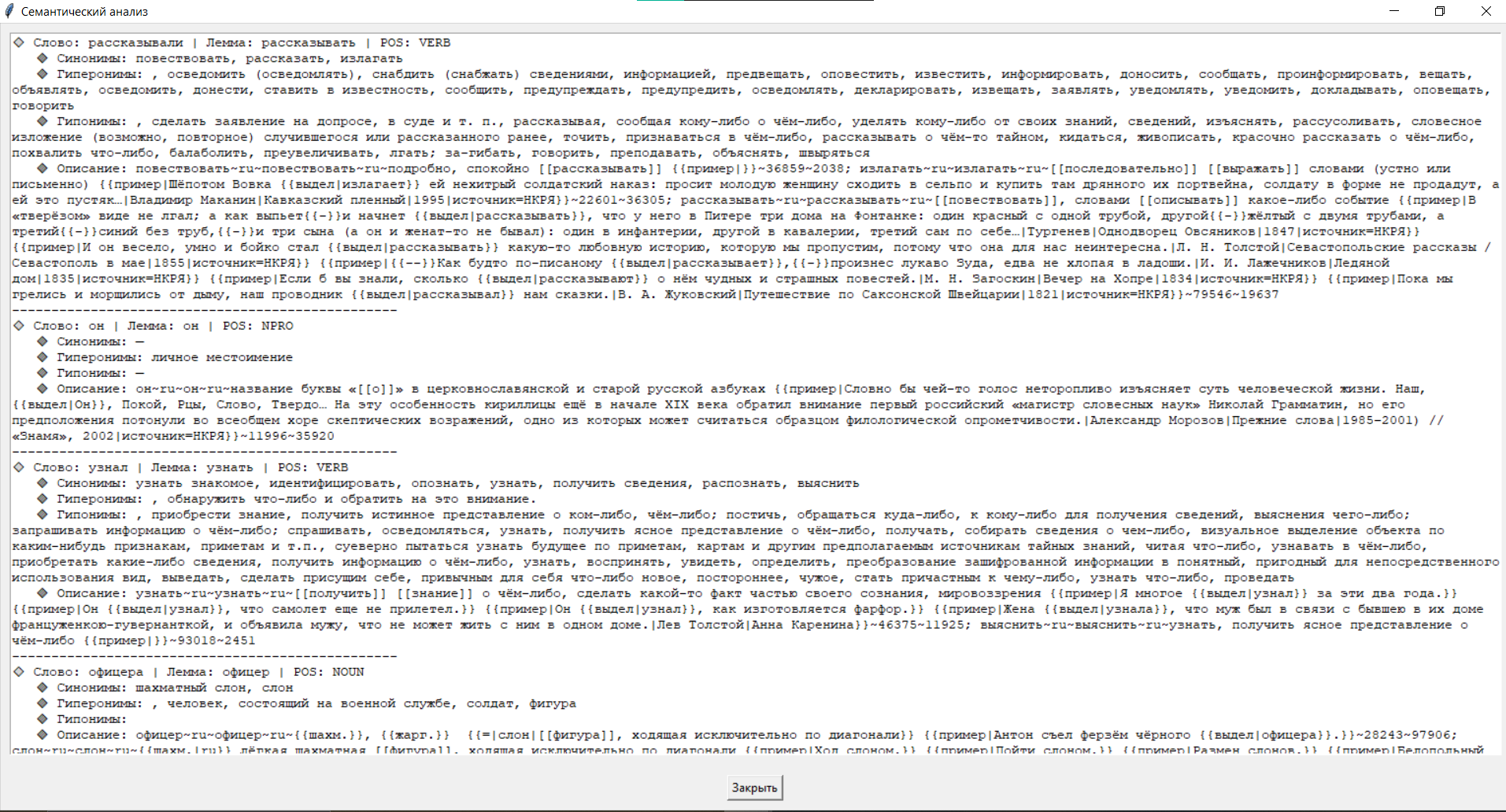
**Задачи лабораторной работы: ­**

1. познакомиться с назначением, структурой и функциональностью, предоставляемой базовым ЛП для решения задачи автоматического семантико-синтаксического анализа ТЕЯ; ­
2. закрепить навыки программирования при решении задач автоматической обработки ТЕЯ

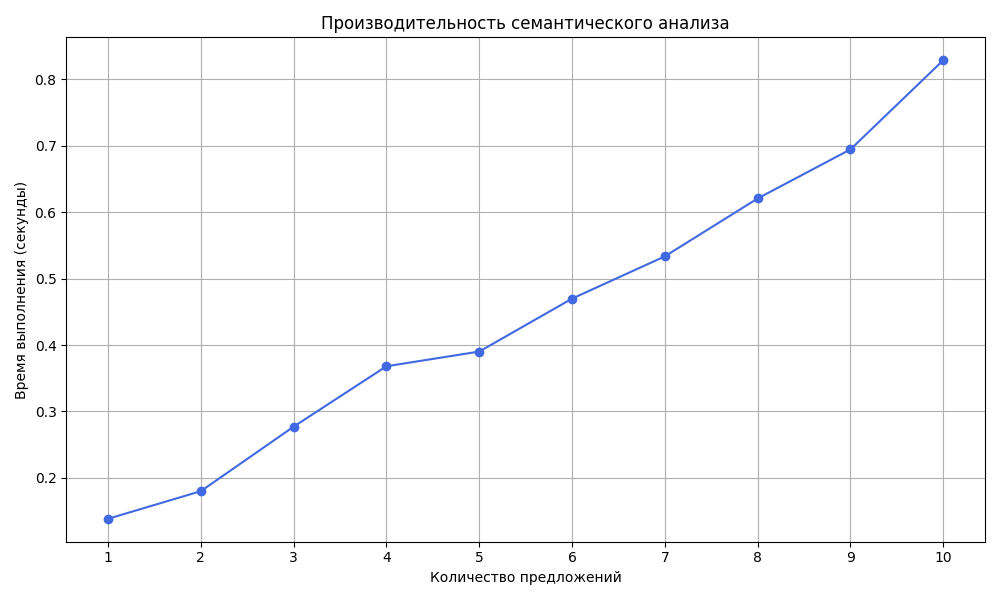
**Интерфейс**



**Интерфейс семантического анализа предложения**

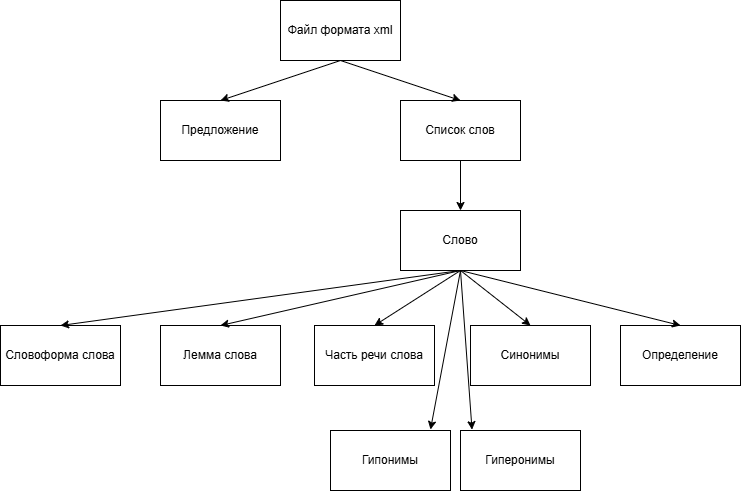
****

**Тест производительности:**

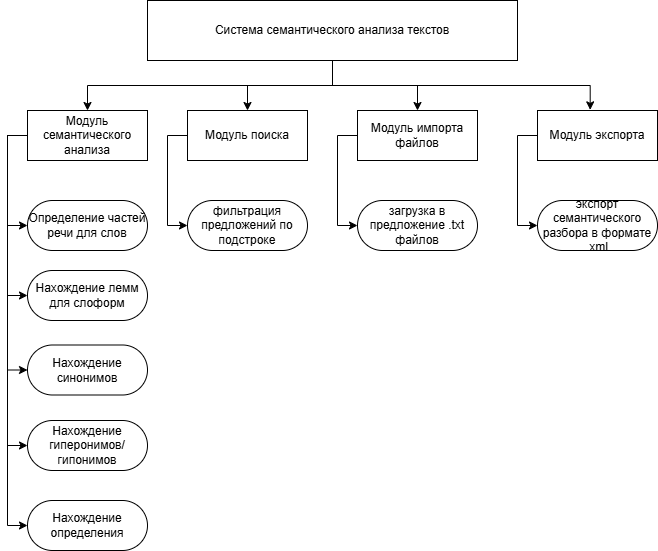
****

На данном графике показана зависимость времени семантического анализа предложений от кол-ва предложений. Видно, что с увеличением кол-ва предложений время обработки также возрастает. Это указывает на **линейную зависимость** между количеством предложений и временем обработки. Общая тенденция — увеличение времени обработки при увеличении кол-ва предложений.

**Структура хранения данных**

****

**Структурная схема приложения**

****

**Использованные библиотеки:**

**re** — библиотека для обработки текста с использованием регулярных выражений.

**tkinter** — библиотека для создания графических пользовательских интерфейсов (GUI).

**xml.etree.ElementTree** — библиотека для работы с XML-документами: создание, парсинг и сохранение.

**tkinter.ttk** — модуль tkinter, предоставляющий расширенные виджеты с поддержкой тем оформления.

**tkinter.filedialog** — модуль tkinter, позволяющий открывать диалоговые окна для выбора файлов.

**tkinter.messagebox** — модуль tkinter, используемый для отображения всплывающих окон с сообщениями.

**nltk** — библиотека для обработки и анализа естественного языка (Natural Language Processing).

**pymorphy2** — морфологический анализатор для русского языка, используется для получения нормальной формы и грамматической информации о словах.

**ruwordnet** — библиотека для работы с русскоязычной версией WordNet, используемой для поиска семантических связей между словами.

**wiki\_ru\_wordnet** — библиотека для доступа к WikiWordnet — русскоязычному тезаурусу на основе Википедии, позволяет извлекать синонимы, гиперонимы, гипонимы и определения.

**Описание алгоритмов:**

**1. Алгоритм построения списка предложений для текста**

1. **Начало.**
2. **Выбор файла пользователем:**
   * Через диалоговое окно пользователь выбирает текстовый файл.
3. **Очистка текста (при необходимости):**
   * Удаляются лишние пробелы, управляющие символы, заменяются переносы строк.
4. **Разбиение текста на предложения:**
   * Регулярное выражение, например re.split(r'(?<=[.!?])\s)+', text).
5. **Формирование списка предложений:**
   * Каждое предложение сохраняется как отдельный элемент списка.
6. **Конец.**

**3. Алгоритм семантического анализа предложения:**

1. **Начало.**
2. **Выбор предложения для анализа:**
   * Из таблицы (дерева tree) выбирается одно предложение, выделенное пользователем.
   * Если предложение не выбрано — анализ не выполняется.
3. **Токенизация и морфологический разбор:**
   * С помощью nltk.word\_tokenize() выполняется разбиение предложения на слова.
   * Для каждого слова вызывается морфологический анализ с помощью pymorphy2.
4. **Фильтрация служебных слов:**
   * Исключаются слова, являющиеся предлогами, союзами, частицами, междометиями, наречиями, прилагательными или не имеющие части речи (None).
5. **Приведение слов к леммам:**
   * Для каждого подходящего слова определяется **лемма** — нормальная форма слова.
6. **Поиск семантических связей:**
   * Если для леммы найдены синсеты в wikiwordnet, выполняется:
     + Получение **синонимов** из ruwordnet.
     + Получение **гиперонимов** (понятий более высокого уровня) из wikiwordnet.
     + Получение **гипонимов** (частных случаев) из wikiwordnet.
     + Получение **определения** леммы из wikiwordnet.
7. **Формирование текстового результата:**
   * Для каждой леммы собирается текстовая информация: слово, лемма, часть речи, синонимы, гиперонимы, гипонимы и определение.
   * Результат форматируется и добавляется в общий вывод.
8. **Отображение результата:**
   * Метод show\_semantic\_result открывает окно с подробным результатом семантического анализа в текстовом формате.
9. **Завершение работы:**
   * После анализа возможно закрытие окна или выбор нового предложения для повторного анализа.
10. **Конец.**

**Вывод:** в ходе выполнения лабораторной работы был реализован семантический анализ текста с использованием библиотек NLTK, PyMorphy2, RuWordNet и WikiWordnet для русского языка. Анализ включал этапы токенизации, морфологического разбора и лемматизации слов, фильтрацию служебных частей речи, а также извлечение семантических характеристик — синонимов, гиперонимов, гипонимов и определений. Результаты анализа сохранялись в XML-формате и отображались в удобном текстовом виде. Реализация позволила получить представление о значении и семантических связях слов в предложении, что расширяет возможности для глубокой обработки и понимания текстовых данных.