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 № 6

по дисциплине

Вариант 21

Естественно-языковой интерфейс интеллектуальных систем

“Семантико-синтаксический анализ текстов естественного языка”

|  |  |
| --- | --- |
| Выполнили | студенты группы 221702  Хлуд А.Н.  Набешко Д.С.  Чаюк Т.А. |
| Проверил | Крапивин Ю. Б. |
|  |  |

Минск 2025

**Цель работы:**

* освоить принципы проектирования и разработки
* естественно-языковых интерфейсов с пользователем в диалоговых системах.

**Задачи лабораторной работы:**

* ­ Изучить основы моделирования процесса общения.
* ­ Приобрести навыки проектирования элементов диалоговых систем.

**Язык текста:** английский.

**Предметная область:** Музыка

**Используемые средства разработки:**

* **Python:** Основной язык разработки приложения. Python выбран за его простоту, богатую экосистему библиотек и удобство работы с текстовыми данными.
* **Flask:** Легковесный веб-фреймворк для Python, используемый для создания HTTP-приложения. Flask позволяет легко управлять маршрутами, обрабатывать запросы и рендерить HTML-шаблоны.
* **Sentence Transformers** - для работы с векторными представлениями текста
* **FAISS (Facebook AI Similarity Search)** - для эффективного поиска похожих векторов
* **spaCy + NLTK** - для лингвистического анализа текста
* **Ollama** (с моделью Gemma3:1b) - локальная LLM для генерации ответов
* **LangChain** - для управления цепочками взаимодействия с LLM

**Используемые структуры для хранения и алгоритмы их обработки:**

Для реализации семантического поиска была реализована векторная база данных на основе json-файла, содержащего 500+ записей по предметной области.

Каждая запись в JSON содержит:

* + text (описание трека/исполнителя/теории/инструмента),
  + tags (ключевые слова),
  + category (жанр).
* Перед индексацией эти поля объединяются в одну строку
* Модель Sentence Transformers (all-MiniLM-L6-v2) преобразует текст в 384-мерный вектор чисел (эмбеддинг).
* Все векторы из json добавляются в индекс FAISS (используется IndexFlatL2 для поиска по евклидову расстоянию).

**Алгоритмы обработки данных.**

**Алгоритм семантического поиска:**

1. Начало алгоритма
2. Пользователь вводит запрос
3. Запрос векторизуется в 384-мерный вектор.
4. FAISS ищет k ближайших векторов из индекса (по расстоянию L2).
5. Возвращаются записи из music\_db.json, соответствующие найденным векторам.
6. Если расстояние между векторами слишком большое (>1.0), результат отбрасывается.
7. Конец алгоритма

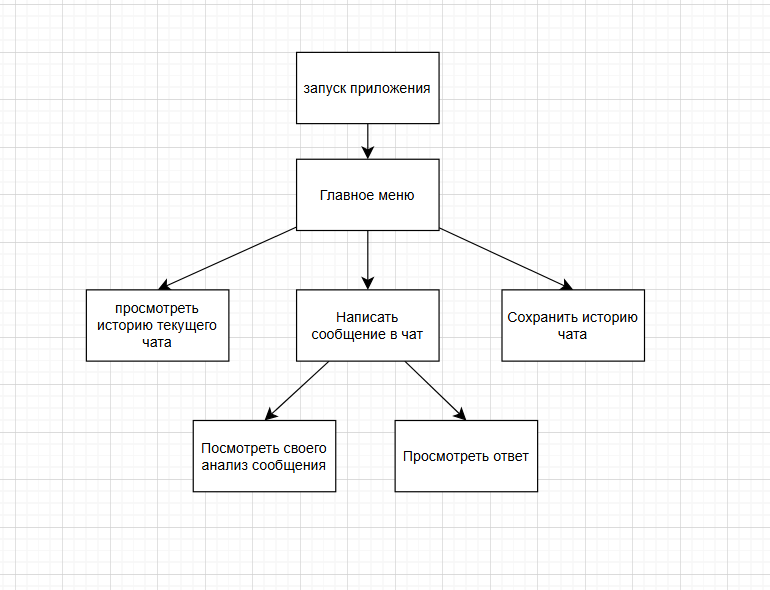
**Алгоритм анализа запроса пользователя с помощью spacy + NLTK**

1. Начало алгоритма
2. Текст приводится к нижнему регистру (text.lower()), удаляются лишние пробелы.
3. Текст разбивается на токены (слова/знаки препинания).
4. Каждое слово приводится к базовой форме (лемме).
5. Отфильтровываются предлоги, союзы (is, the), пунктуация и пробелы.
6. Оценивается эмоциональная окраска текста.
7. Выделяются имена исполнителей, групп, стран.
8. Полученные данные передаются далее по цепи.
9. Конец алгоритма

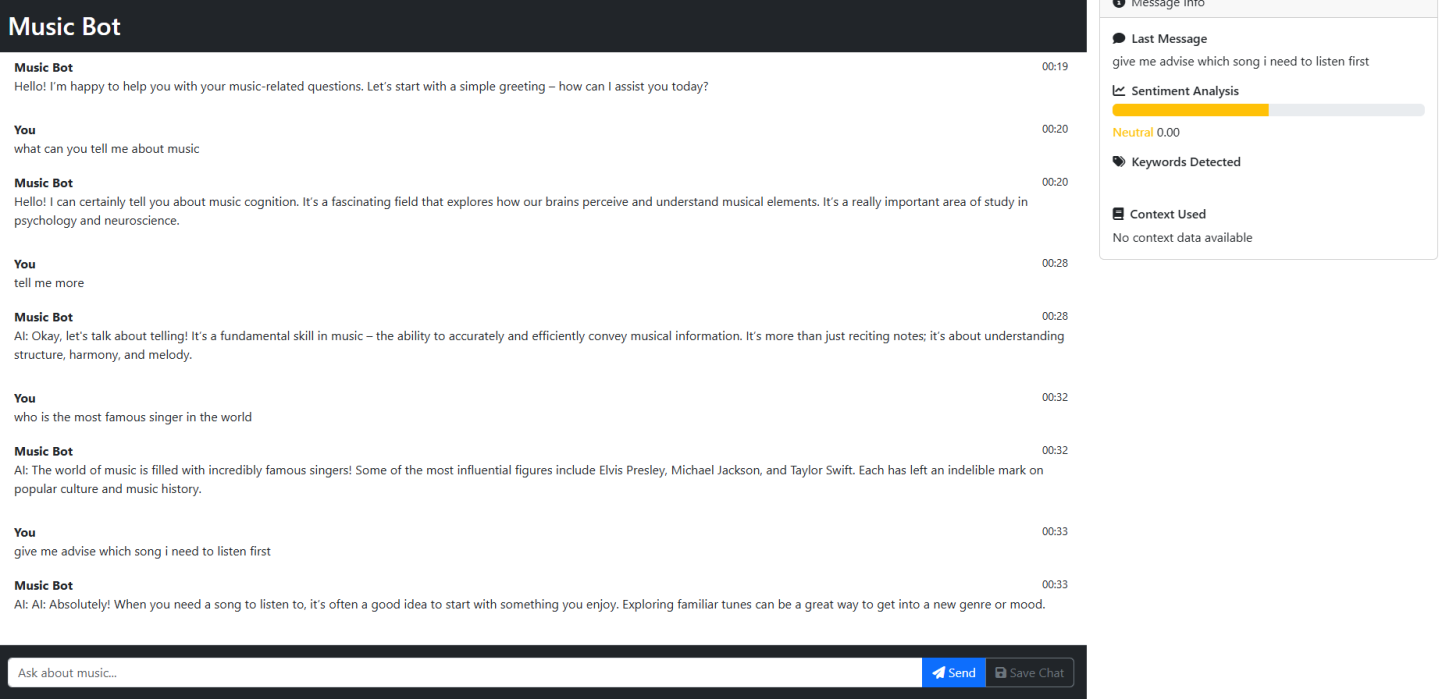
**Алгоритм получения ответа:**

1. Начало алгоритма (получен вопрос пользователя)
2. Анализ текста (spaCy + NLTK)
3. Семантический поиск (FAISS + Sentence Transformers)
4. Подготовка контекста
5. Генерация ответа (Ollama + LangChain)
6. Сохранение результатов в историю
7. Возврат ответа пользователю
8. Конец алгоритма

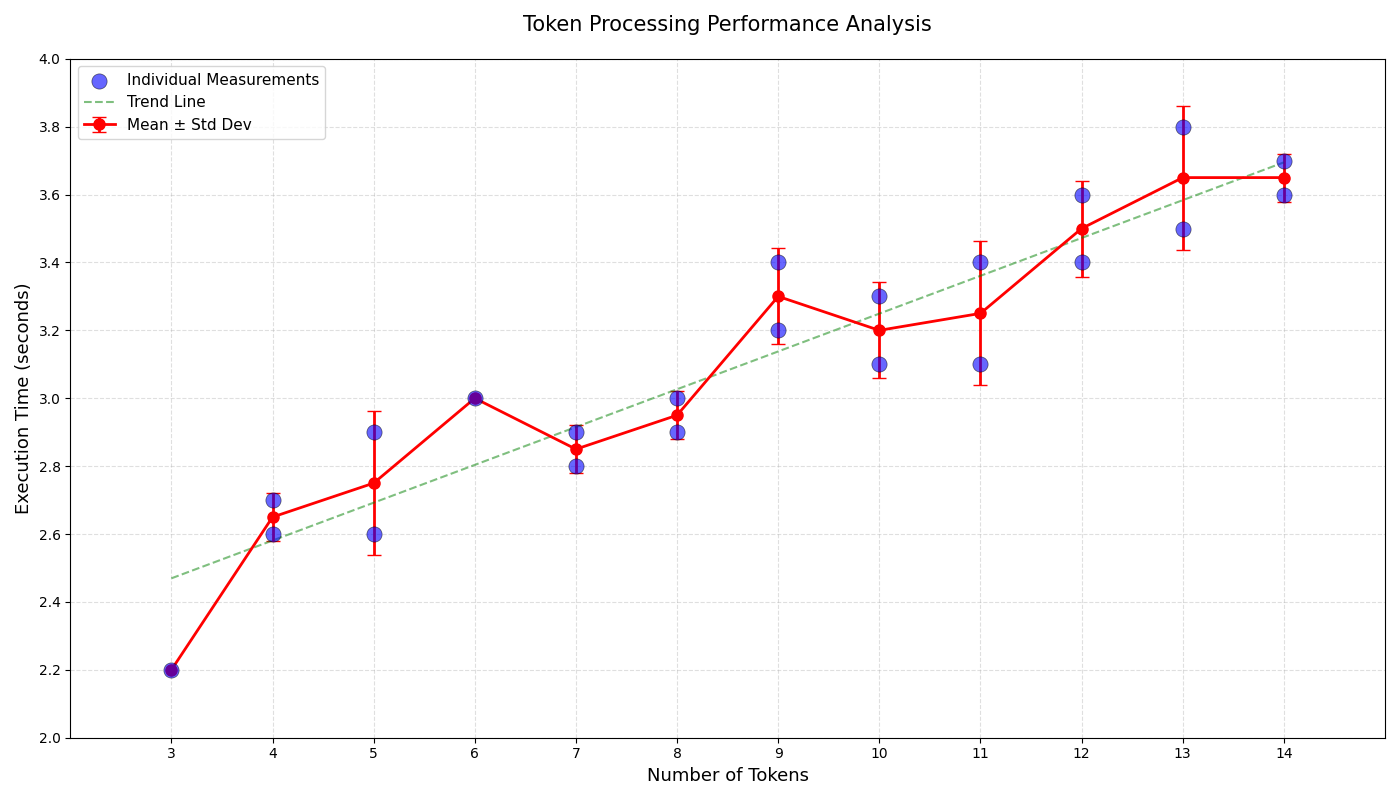
**Структурно-функциональная схема приложения:**

****

**Интерфейс:**

****

**Оценка быстродействия приложения:**

****

**Вывод:** Разработанная система музыкальных рекомендаций на основе RAG-архитектуры успешно сочетает семантический поиск и генерацию ответов, обеспечивая релевантные результаты для сложных запросов.

Однако ограничения включают зависимость качества от объёма базы и возможности локальной LLM. Перспективы включают интеграцию со стриминговыми сервисами. Для масштабирования потребуется оптимизация FAISS для больших данных и эксперименты с более мощными открытыми моделями (Llama 3, Mistral).