Академическая гимназия имени Д. К. Фаддеева

Санкт-Петербургского государственного университета

Разработка быстродействующего бота в телеграмме

**"Телеграм-бот, отвечающий на вопросы, связанные с предпринимательской деятельностью"**

Выполнили:

Ученики 11И и 11К классов

Дивеев Даниил Андреевич

Попов Алексей Михайлович

Гритченко Георгий Сергеевич

Ямпольский Александр Владимирович

Руководитель:

Преподаватель АГ СПбГУ

Тайницкий Владислав Александрович

Санкт-Петербург 2022

**Содержание**

Введение……………………………………………………....3

1. Глава 1. Архитектура..……………………………………4
   1. Архитектура………………………....………..…........4
   2. Запросы……………….………………...……………..4
   3. Получение данных о пользователе..............................4
   4. Парсинг…...…………..…………………………...…..5
   5. Предварительная обработка данных….……….….....5
   6. Поиск наиболее релевантной части текста………….6
   7. Поиск ответа на вопрос по контексту………………..7
   8. Внесение данных о запросе в базу данных………….7
   9. Получение ответа и отправка в чат…………………..8

2. Заключение………………………………………………...9

3. Список литературы…………………………………….....10

**Введение**

Зачастую у людей, работающих в предпринимательской сфере, возникают юридические вопросы, связанные с их деятельностью, поиск ответа на которые может отнять время. Что с этим делать?

**Актуальность**

Почему наш проект будет востребован и важен? Во-первых, чтение большого количества текста и отсеивание нужной информации занимает много времени, из-за чего не получается переходить к следующей задаче, но наш бот сможет выполнить запрос в кратчайшие сроки. Во-вторых, телеграм сейчас есть практически у всех и он чрезвычайно прост в освоении и использовании.

**Цели**

Целью является создание телеграм-бота, который сможет ускорить и упростить поиск юридической информации, которая может потребоваться в предпринимательской деятельности. Благодаря этому люди смогут получать нужные сведения кратко, полно и быстро.

**Задачи**

1.

2.

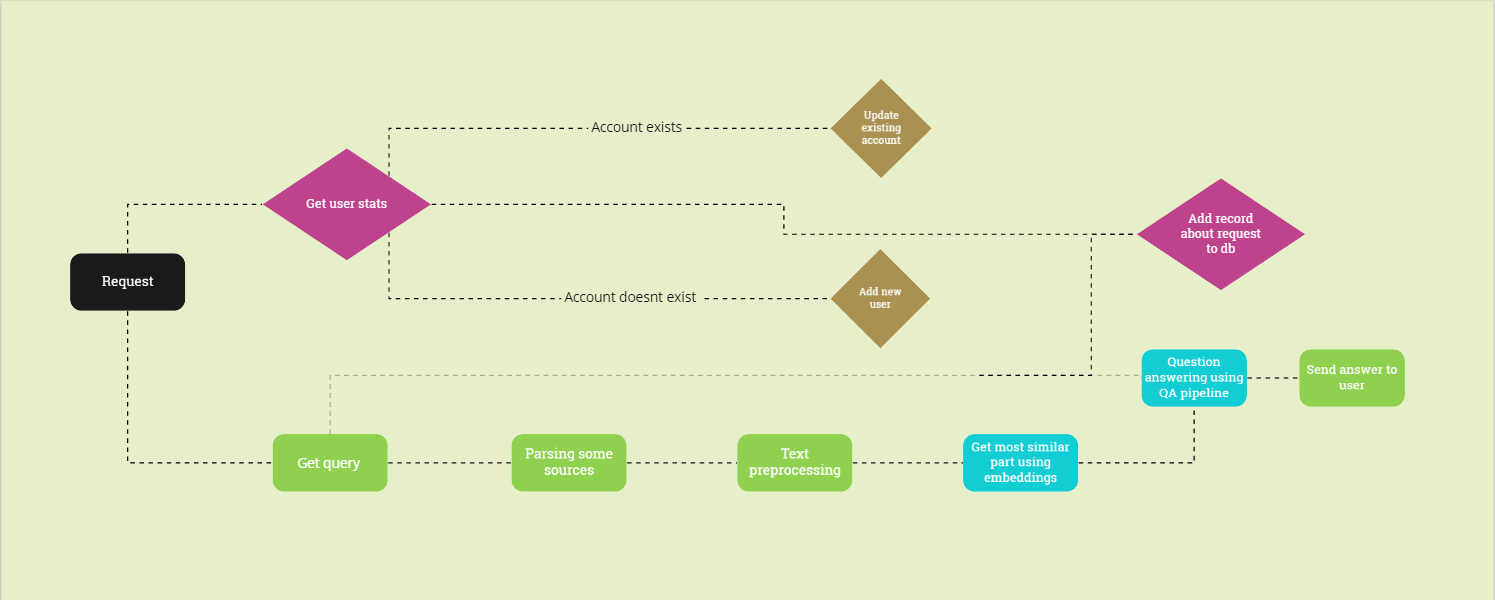
3.

4.

**Глава 1. Архитектура**

**Архитектура**

Телеграм-бот – это программа, выполняющая различные действия, автоматически или по определенной команде.



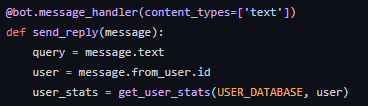
**Запросы**

Запрос – текст, вводимый пользователем в чат с ботом. Эта текстовая информация будет обрабатываться скриптом программы для сбора информации с целью последующего поиска данных.



**Получение данных о пользователе**

Получение данных о пользователе происходит благодаря скрипту, вложенному в бота. После получения данных они либо, если такой пользователь существует, вносятся в базу данных, либо сначала происходит создание нового пользователя с последующим внесением в БД.



*Скрипт получения информации*

**Парсинг**

Парсинг (англ.: Parsing) — сбор и систематизация информации, размещенной на определенных сайтах, с помощью специальных программ, автоматизирующих процесс.



*Пример парсера сайта nalog.ru из скрипта бота.*

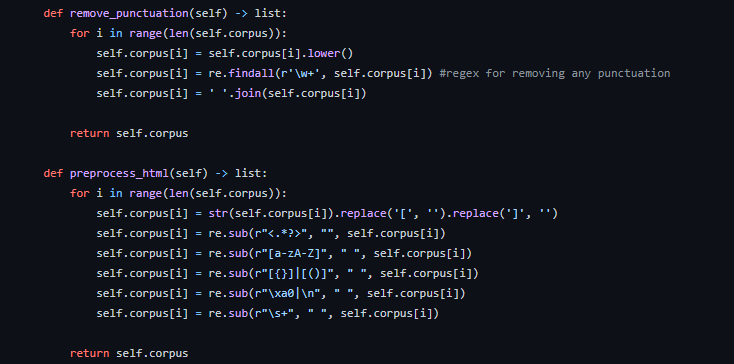
**Предварительная обработка данных**

Предварительная обработка данных (англ.: Preprocessing) – автоматизированная подготовка данных для приведения их к списку требований, необходимых для специфики решаемой задачи.

Препроцессинг делится на 2 стадии: очистка и оптимизация.

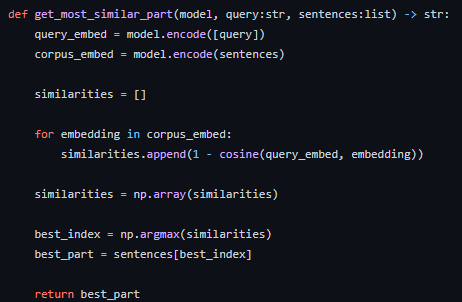
Очистка производится с целью повышения качества данных для более корректной работы алгоритмов. Она включает обработку повторяющихся данных, противоречий и фиктивных значений, восстановление и заполнение пропусков. Также во время очистки восстанавливается нарушенная структура информации и некорректные форматы данных.

Оптимизация заключает в себя оптимизация формы, наиболее удобную для анализа. Проблемы, решаемые при оптимизации, адаптируют данные к конкретной задаче и повышают эффективность их анализа.

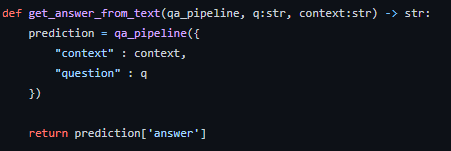


*Пример кода, в котором текст проходит очистку и оптимизацию*

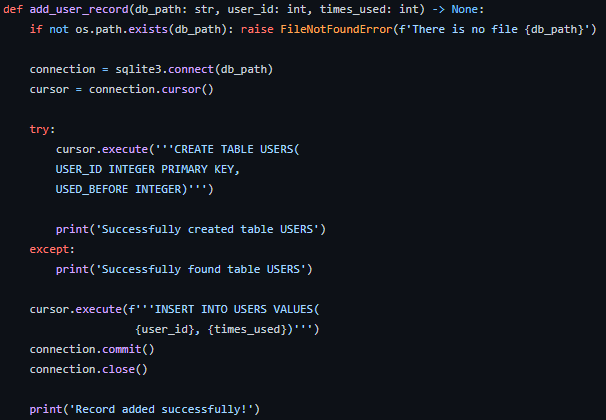
**Поиск наиболее релевантной части текста**

Поиск наиболее релевантной части текста проходит при помощи сравнения эмбединга полученного вопроса и эмбедингов найденных текстов, косинусным расстоянием.

**Поиск ответа на вопрос по контексту**

После получения релевантной части текста при помощи модели QA BERT, в полученном куске текста ищется ответ на заданный вопрос. 

**Внесение данных о запросе в базу данных**

После прохождения всех шагов и получения ответа на заданный вопрос, в базу данных о вопросах добавляется новая запись: 

*Добавление данных о пользовательском запросе в базу данных*



**Получение ответа и отправка в чат**

Полученный ответ бот отправляет в чат пользователю. 

**Заключение**

Наш проект помимо своей практичности несёт также и социальную роль, так как при использовании нашего продукта, люди будут меньше ошибаться, тем самым допуская меньше правонарушений. Также стоит заметить, что наш проект с продуктовой точки зрения является жизнеспособным продуктом, а при наличии должных инвестиций может перерасти в достойный стартап.

**Список использованных источников и литературы**

**Документы и источники**

Предпроцессинг: <https://wiki.loginom.ru/articles/data-preprocessing.html>

Парсинг: <https://blog.calltouch.ru/chto-takoe-parsing/>

Код: <https://github.com/sexozavr/entrepreneur-helper>