

Автоматический FAQ на примере чат-бота Telegram

Выполнил: Фролов Олег СКБ201

Используемые библиотеки

- pandas - для извлечения данных из csv таблицы
- re - для использования регулярных выражений
- sklearn - для распознавания текста, это основа нашей "машинки"
- sentence_transformers - нужен для использования уже подготовленной Bert модели для текстов, позволяет найти ближайшее предложение для запроса (наиболее подходящее, подсчитывает его схожесть с запросом)
- nltk - используем для импорта стоп-слов из русского языка (союзы, предлоги, местоимения, частицы)
- oauth2client, oauthlib, httplib2, googleapiclient - для взаимодействия с гугл таблицей, откуда будем формировать таблицу csv для работы программы
- telebot - для взаимодействия с телеграм-ботом
- Установку производить через pip.

Формирование таблицы с вопросами и ответами!

Для начала разберем, как формировать базу для работы программы: первый столбец в первой строке таблицы - questions, . Второй столбец в первой строке таблицы - "answers". Далее под этими названиями перечисляются соответственно вопросы и ответы в формате: "вопрос" , | "ответ"(горизонтальная палочка олицетворяет разделение между ячейками). В формировании вопросов и ответов настоятельно следует избегать использования двойных кавычек, потому что они ведут к ошибкам при считывании таблицы, такого формата стоит придерживаться в строгом порядке, чтобы избежать возникновения ошибок!

AutoFAQ.py

Разберем функции по порядку.

1. `cleanRequest(sentence)` - используется для предварительной обработки поступающего запроса(`sentence`): удаление стоп-слов.
2. `cleanedQuestions(df)` - используется для предварительной обработки массива вопросов из файла-таблицы `csv` (`df`).
3. `autoFAQ(question_embedding, sentence_embeddings, FAQdf, sentences, min_similarity)` - используется для поиска подходящего ответа по методу косинусового сходства на заданный запрос (сначала отыскивается максимально схожий вопрос, а затем уже выдергивается ответ на этот вопрос).

GoogleSheets.py

Разберем функции по порядку.

1. `get_service_sacc(filename)` - для соединения с сервисным аккаунтом Google для дальнейшего взаимодействия с гугл-таблицей, использует крендициалсы из файла `filename`.
2. `prepare_file(filename, gtoken)` - для парсинга гугл-таблицы с идентификатором `gtoken` и сохранения данных в локальный файл-таблицу `csv`, вызывает `get_service_sacc(filename)`.

ChatBot.py

Разберем функции по порядку.

1. `prepare_token(filename)` - достает токен из `filename` (токен должен лежать первой строкой, только токен).
2. `start(message)` - вызывается, когда пользователь пишет Telegram чат-боту сообщение `"/start` устанавливает кнопки для сообщений (вопрос, сообщение тех. поддержке).
3. `message_listener(message)` - вызывается при любом другом сообщении Telegram чат-боту от пользователя, на основе характера сообщения определяет пользователя в группу тех, кто задал вопрос, или в группу тех, кто хочет связаться с тех. поддержкой. Далее, в соответствии с категорией пользователя, бот или выдает максимально подходящий на его усмотрение ответ на вопрос, или передает информацию в тех. поддержку, ту информацию, которую ему предоставил пользователь.