К. С. Герасимов, А. А. Редикульцев, В. Е. Сидоров,
Т. Д. Низамов, М. С. Воробьева
Тюменский государственный университет, г. Тюмень
УДК 004.9

РАЗРАБОТКА ПАНЕЛИ АДМИНИСТРАТОРА ВИРТУАЛЬНОГО ПОМОЩНИКА СТУДЕНТА ТюмГУ

Аннотация. В статье представлена разработка панели администратора виртуального помощника студента ТюмГУ в виде веб-приложения с использованием паттерна MVP. Для выделения тематических групп вопросов пользователей, на которые ответ не получен или неудовлетворителен, произведен кластерный анализ, выделены ключевые слова. Такой функционал позволит администратору понимать, документов с какой информацией не хватает в корпоративном хранилище.

Ключевые слова: панель администратора, паттерн MVP, кластерный анализ, выделение ключевых слов.

Введение. В Тюменском государственном университете (ТюмГУ) внедрен виртуальный помощник студента в виде чат-ботов VK и Telegram, которые отвечают на вопросы пользователей с использованием большой языковой модели, интегрированной с корпоративным хранилищем документов. За наполнение и актуализацию инструкций и регламентов в хранилище отвечает управление по сопровождению студентов «Единый деканат». Однако не всегда имеющихся документов хватает для генерации чат-ботами ответов на вопросы, поэтому необходимо анализировать, на вопросы по каким тематикам ответы не получены или оценены пользователями виртуального помощника как неудовлетворительные.

Для улучшения работы различных сервисов часто применяются панели администраторов. Например, студентами Санкт-Петербургского политехнического университета разработана панель администратора для веб-сервиса по передержке домашних животных PetCher[1]. В Белорусском государственном университете реализована панель администратора для управления процессами сервиса автомастерских[2].

Проблема исследования. Пользователи виртуального помощника студента ежедневно задают вопросы, некоторые из которых остаются без ответа. Сотрудники Единого деканата не обладают инструментом для анализа вопросов пользователей виртуального помощника студента, чтобы принимать решения по дополнению корпоративного хранилища документов ТюмГУ.

Цель работы: разработать веб-приложение для администратора виртуального помощника студента с возможностью объединения вопросов пользователей, на которые ответ не получен, либо неудовлетворителен, в тематические группы. Для выделения тематических групп необходимо провести кластерный анализ вопросов, для обобщения информации выделить ключевые слова. При этом для каждого кластера требуется отобразить: набор ключевых слов, набор вопросов и дату самого свежего вопроса. Такой функционал позволит администратору понимать, документов с какой информацией не хватает в корпоративном хранилище.

[1] URL: https://elib.spbstu.ru/dl/3/2019/vr/vr19-1025.pdf/info.
[2] URL: https://elib.bsu.by/handle/123456789/302189.

Методы и материалы. Для разработки панели администратора использован паттерн MVP [1], который разделяет приложение на модель (model), отображение (view) и представитель (presenter). Модель управляет данными, отображение отвечает за представление данных и получение ввода пользователя, а представитель выступает посредником между моделью и представлением, обрабатывая логику и обновляя их в соответствии с действиями пользователя.

Для реализации веб-приложения панели администратора на языке программирования Python 3.10 выбран микрофреймворк Flask, представляющий собой открытый инструмент с большим набором внешних библиотек, в том числе SQLAlchemy, реализующей ORM для взаимодействия с данными (моделью). Пользовательский интерфейс разработан с помощью веб-фреймворка Bootstrap 5 и шаблонизатора Jinja2.

При реализации модуля кластерного анализа вопросов и выделения ключевых слов (далее — модуля анализа вопросов) для предобработки текстов, подразумевающей удаление специальных символов и строк, не являющихся словами, использована библиотека PyMorphy2 [2].

Рассмотрено несколько типов векторизации: TF-IDF [3], doc2vec [4] и Sentence Transformer [5]. В качестве Sentence Transformer использована модель Sentence RuBERT, дообученная на вопросах к корпоративным документам ТюмГУ[1].

Для решения задачи кластеризации рассмотрены такие методы, как иерархический [6], k-means [7], DBSCAN [8] и спектральный [9]. Оценка качества кластерного анализа осуществлена с помощью коэффициента силуэта [10], который показывает, на сколько каждый объект близок к объектам того же кластера и на сколько он далеко от объектов ближайшего кластера в пространстве признаков.

Среди способов выделения ключевых слов рассмотрены TF-IDF, Rake [11], YAKE! [12], KeyBERT [13], RuTermExtract[2]. Оценка качества выделения ключевых слов произведена с помощью BERTScore [14], который показывает семантическую близость исходного текста и сгенерированного.

В ходе тестирования методов кластеризации и выделения ключевых слов использовано 279 вопросов, оставшихся без ответа, от 76 пользователей виртуального помощника студента ТюмГУ в период с 6 февраля по 16 марта 2024 года.

Результаты. На основе паттерна MVP разработано веб-приложение панели администратора в рамках архитектуры виртуального помощника студента ТюмГУ (см. рис. 1). Приложение включает в себя модули:

1) авторизации администратора через корпоративную учетную запись;

2) организации массовых рассылок сообщений пользователям;

3) переиндексации документов из корпоративной вики-системы для ответов на вопросы;

4) анализа вопросов за определенный период, ответы на которые не получены или оценены пользователями как неудовлетворительные.

[1] URL: https://hf.co/nizamovtimur/rubert-tiny2-wikiutmn.
[2] URL: https://github.com/igor-shevchenko/rutermextract.

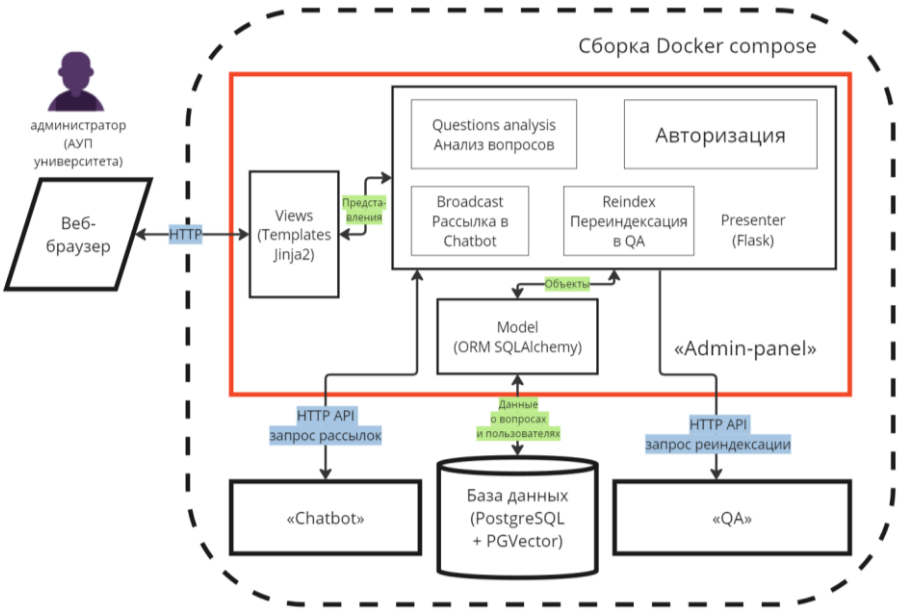

Рис. 1. Схема архитектуры панели администратора виртуального помощника

Для реализации модуля анализа вопросов оценены методы векторизации, кластеризации, выделения ключевых слов. Тестирование методов проводилось на базе виртуального сервера с Intel(R) Xeon(R) CPU @ 2.20GHz и 13 ГБ оперативной памяти.

При использовании метода векторизации TF-IDF минимальная частота встречаемости леммы min_df=2; для doc2vec размерность векторов vector_size=150, использовано окно с размером 5, минимальная частота встречаемости слов min_count=1.

Для иерархической кластеризации используется метод связи (method) «complete» (полная), мера (metric) «cosine» (косинусная); для k-means количество кластеров увеличивается от двух до количества элементов, пока значения силуэтов для следующих пяти не будет ниже данного; для DBSCAN количество объектов в кластере n_neighbors=3; для спектральной кластеризации выбрана стратегия присваивания меток (assign_labels) «discretize», количество кластеров аналогично k-means.

Для каждой рассмотренной комбинации методов векторизации и кластеризации оценены коэффициент силуэта и время выполнения в среднем за 7 запусков (табл. 1).

Таблица 1

Результаты сравнения методов кластерного анализа

Метод векторизации	*Метод кластеризации*	*Коэффициент силуэта*	*Среднее время выполнения*	*Количество кластеров*
TF-IDF	Иерархическая	**0.59**	2.73 s ± 286 ms	69
	k-means	0.54	5.15 s ± 440 ms	2
	DBSCAN	0.24	2.73 s ± 196 ms	9
	Спектральная	0.12	48.6 s ± 416 ms	2
Doc2vec	Иерархическая	0.13	**346 ms ± 7.73 ms**	98
	k-means	0.01	32.9 s ± 635 ms	2
	DBSCAN	-0.06	495 ms ± 106 ms	1
	Спектральная	0.04	1min 28s ± 2.58 s	2

Окончание табл. 1

Метод векторизации	*Метод кластеризации*	*Коэффициент силуэта*	*Среднее время выполнения*	*Количество кластеров*
Sentence Transformer	Иерархическая	0.27	533 ms ± 61	49
	k-means	0.17	50.2 s ± 1.46 s	46
	DBSCAN	0.01	445 ms ± 11.3 ms	1
	Спектральная	0.06	3.69 s ± 541 ms	2

Наибольший коэффициент силуэта показал пайплайн с векторизацией с помощью TF-IDF и иерархической кластеризацией.

При выделении ключевых слов для TF-IDF значение min_df=1. Для метода KeyBERT использована предобученная модель Sentence BERT[1]. Для остальных рассмотренных методов применены настройки с использованием русского языка. При оценке алгоритмов поиска ключевых слов оценены метрики BERTScore и время работы алгоритма в среднем за 7 запусков (табл. 2).

Таблица 2

Результаты тестирования методов выделения ключевых слов

Метод	*BERTScore*			*Среднее время выполнения*
	Precision	*Recall*	*F1*	
TF-IDF	0.55	0.68	0.61	2.97 s ± 351 ms
Rake	0.71	**0.78**	**0.74**	**31 ms ± 10.9 ms**
YAKE!	0.71	0.73	0.72	1.87 s ± 520 ms
KeyBERT	**0.72**	0.76	**0.74**	2 min 2s ± 1.61 s
RuTermExtract	0.65	0.75	0.69	644 ms ± 154 ms

Наибольший показатель F1-меры BERTScore показали алгоритмы Rake и KeyBERT, при этом Rake выполняется за 31 мс, а KeyBERT более, чем за 2 мин, поэтому в дальнейшей работе использован Rake.

По результатам рассмотрения методов и проведенных замеров сформирован пайплайн анализа вопросов, используемый в разработанной панели администратора (рис. 2).

Рис. 2. Пайплайн анализа вопросов

Пример работы модуля анализа вопросов пользователей, ответы на которые не получены или оценены неудовлетворительно, за период с 6 февраля по 16 марта 2024 года, представлен на рис. 3.

[1] URL: https://hf.co/sentence-transformers/distilbert-base-nli-mean-tokens.

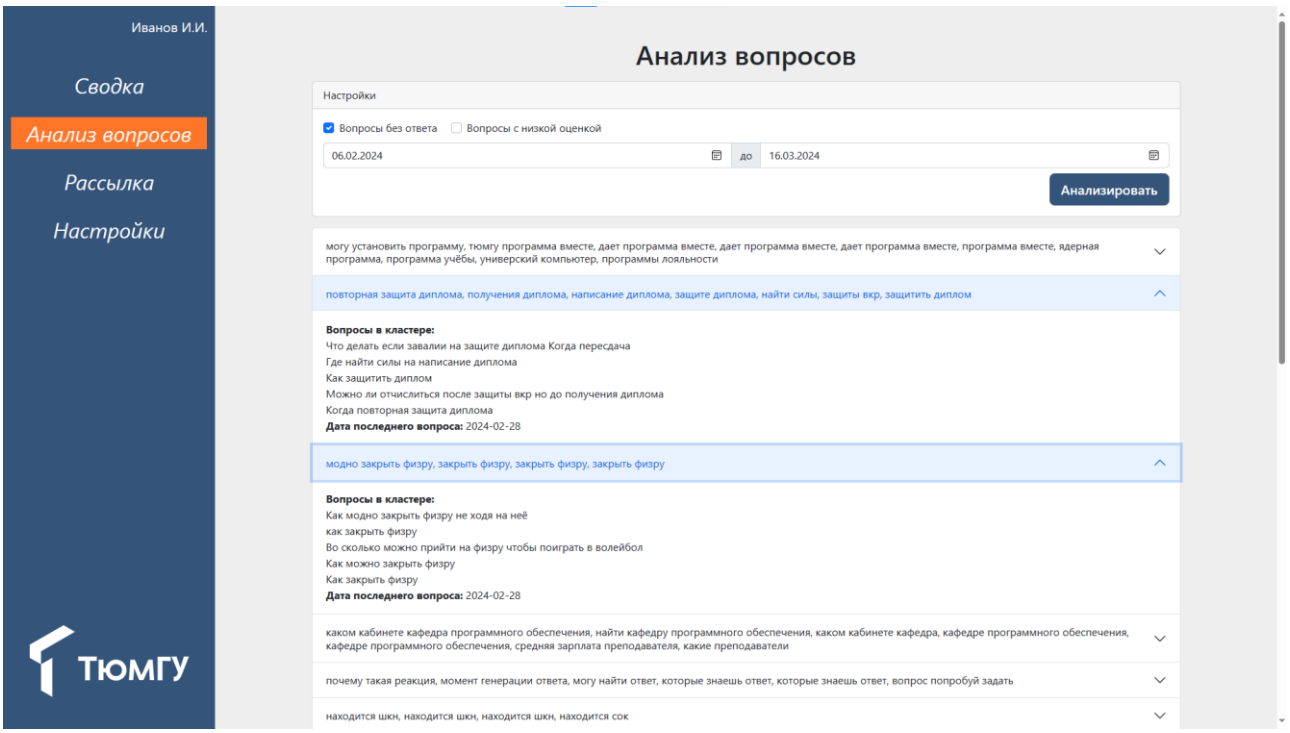

Рис. 3. Страница с результатами анализа вопросов пользователей

Заключение. Разработана панель администратора виртуального помощника студента ТюмГУ на основе паттерна MVP с возможностью анализа вопросов, ответы на которые не получены или оценены пользователями как неудовлетворительные, с использованием методов кластеризации текстов и выделения ключевых слов.

В дальнейшем планируется развертывание разработанной панели администратора на production-сервере виртуального помощника студента в ИТ-контуре Тюменского государственного университета и расширение функционала за счет визуализации сводки использования виртуального помощника студента и выявления вопросов, на которые следует обратить внимание центру тьюторского сопровождения ТюмГУ.

СПИСОК ЛИТЕРАТУРЫ

1. Елисеев Б.П., Тарасенко А.В, Горбачев О.А., Лю Джонда Программный паттерн проектирование структурного каркаса Model-View-Controller при разработке веб-приложений систем мониторинга спецтранспорта аэропорта // Научный вестник МГТУ ГА. — 2015. — № 2 (20). — С. 139-140.
2. Korobov M. Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts. — 2015. — Vol. 542. — Pp. 320-332.
3. Sparck Jones K. A statistical interpretation of term specificity and its application in retrieval //Journal of documentation. — 1972. — Vol. 28, № 1. — P. 11-21. DOI:10.1108/00220410410560573
4. Le Q., Mikolov T. Distributed representations of sentences and documents // International conference on machine learning. — PMLR, 2014. — P. 1188-1196. DOI: 10.48550/arXiv.1405.4053.
5. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. — Association for Computational Linguistics. — 2019. DOI:10.48550/arXiv.1908.10084.
6. Johnson S. C. Hierarchical clustering schemes // Psychometrika. — 1967. — Vol. 32, № 3. — P. 241-254. DOI: 10.1007/BF02289588.

7. Lloyd, S. P. Least squares quantization in PCM // Technical Report RR-5497, Bell Lab, September 1957. DOI:10.1109/TIT.1982.1056489.
8. Ester M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise // kdd. — 1996. — Vol. 96, № 34. — P. 226-231. DOI:10.5555/3001460.3001507.
9. Ng A., Jordan M., Weiss Y. On spectral clustering: Analysis and an algorithm // Advances in neural information processing systems. — 2001. — Vol. 14.
10. Глазырин А.Г. Численное моделирование и оценка характеристик плотности распределения модельных кластеров на евклидовой плоскости // Процессы управления и устойчивость. — 2020. — Т. 7, № 1. — С. 225-229.
11. Stuart R., Dave E., Nick C., Wendy Cowley, Automatic Keyword Extraction from Individual Documents. March 2010. In book: Text Mining: Applications and Theory. P. 1–20. DOI: 10.1002/9780470689646.ch1
12. Campos R., Mangaravite V., Pasquali A., Jorge A., Nunes C., Jatowt A. YAKE! keyword extraction from single documents using multiple local features // Information Sciences. — 2020. — № 509. — P. 257–289. DOI: 10.1016/j.ins.2019.09.013.
13. Sharma P., Li Y. Self-Supervised Contextual Keyword and Keyphrase Retrieval with Self-Labelling. Preprints 2019, 2019080073. — URL: https://doi.org/10.20944/preprints201908.0073.v1.
14. Zhang T., Kishore V., Wu F., Weinberger K.Q., Artzi Y. BERTScore: Evaluating text generation with BERT. — 2020. — arXiv: 1904.09675.