FAQ-Based Question Answering Systems with Query-Question and Query-Answer Similarity

Vijay Kumari[1], Miloni Mittal[1], Yashvardhan Sharma[1] and Lavika Goel[2]
[1]*Birla Institute of Technology and Science, Pilani, Rajasthan, India*
[2]*Malaviya National Institute of Technology, Jaipur, Rajasthan, India*

Keywords: FAQ Retrieval, Question-Answering System, Information Retrieval.

Abstract: A Frequently Asked Question (FAQ) Answering System maximizes knowledge access by enabling users to request a natural language question using the FAQ database. Retrieving FAQs is challenging due to the linguistic difference between a query and a question-answer pair. This work explores methods to improve on this linguistic gap in FAQ retrieval of the Question Answering System. The task is to retrieve frequently asked question-answer pairs (FAQ pairs) from the database that are related to the user's query, thus providing answers to the user. We do so by leveraging natural language processing models like BERT and SBERT and ranking functions like BM25. The best results are obtained when BERT is trained in a triplet fashion (question, paraphrase, non-matching question) and combined with the BM25 model, which compares query with FAQ question answer concatenation.

1 INTRODUCTION

Building a Question Answering (QA) System is an important problem statement in the field of Natural Language Processing (NLP). It involves extracting the most relevant information from abundant information, which may be classified into relevant, useful, or irrelevant. Due to the information overload with respect to quantity and categories, searching for the relevant answer to the posed query is of utmost importance. A question-answering system can be built for two domains: 1) Open Domain: There is no boundary on the content category for extracting the answer. Example: Google search engine, Yahoo search engine. 2) Closed Domain: There is a certain boundary on what queries this type of system can answer. These systems restrict a particular category of questions and sometimes even answers by referring to a document—for example, QA services on various business websites and solution providers for school textbooks. The speed of deriving the answers to the user query plays a major role in such systems, which demands a need to find methods to extract relevant answers to the queries in the fastest manner possible. One way to do this is to look into the database of already asked and answered questions, or FAQs, and provide the most relevant answers from that list.

The Frequently Asked Questions (FAQ) database is made up of manually generated question-answer pairs. Large-scale service providers tend to use FAQ lists to present easily accessible information to their clients. These collections focused on the closed domain of interest of QA system. Users can browse FAQ databases on their own, using a faster and more efficient FAQ retrieval model. A FAQ retrieval system offers a natural language interface for making queries about the FAQ list. The model generates a series of question-answer pairs ranked by their relevancy according to a user's query. A method for answering frequently asked questions (FAQs) gives users quick access to internal FAQ databases, which improves service quality and efficiency (Karan and Šnajder, 2018).

Many questions have usually been answered in the FAQ section of a question-answering system. However, the vastness of such a database might make it difficult for a user to search for a particular relevant query. Thus, answers to a user query posed to a closed domain QAS can be derived using the existing FAQ database. By checking if a similar question exists in the database, a faster and more efficient QAS can be built that does not need to access the original database for every query. In this work, we retrieve the FAQ pairs from the database that are pertinent to the user's query, thus providing answers to the user. Initially, this problem statement was solved only by referring to the similarity between the FAQ question

Kumari, V., Mittal, M., Sharma, Y. and Goel, L.
FAQ-Based Question Answering Systems with Query-Question and Query-Answer Similarity.
DOI: 10.5220/0012454800003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 1189-1196
ISBN: 978-989-758-680-4; ISSN: 2184-433X

(Q henceforth) and user query (q henceforth). But in recent times, researchers are leaning towards more robust modeling involving FAQ answers (referred to as A henceforth) and similarity comparison (Mass et al., 2020). This method tends to give better results as the other can compensate for lexical gaps in either comparison (q-Q or q-A).

Labeled data is essential for training a model to predict the relationship between user queries and FAQ questions. A dataset of this kind is often created manually or collected through query logs (Mass et al., 2020). The FAQIR dataset specifically contains pairs of questions and answers without any labeling to query, but a relevance score is given to each query.

The proposed system's goal is to train the different models for the task of retrieving frequently asked questions (FAQs) and assessing their performance to provide the most accurate model. This comprehensive approach aims to enhance the effectiveness and precision of the system in handling user queries and retrieving relevant FAQs. The significant contributions of the paper are as follows:

1. We developed a FAQ retrieval model and experimented with various ranking techniques [weighted measures, re-ranking after initial retrieval] to rank top FAQ pairs. Explored and implemented various techniques for generating embeddings in order to find query-question and query-answer similarity.
2. We trained and evaluated the outcomes of various models in the context of the FAQ retrieval task to provide the most accurate model.
3. Built a website using HTML, CSS, JavaScript, and Flask and integrated our final model [BM25 q(Q+A) + BERT qQ training] with it. Created an end-to-end website that gives top answers based on FAQ from the FAQIR dataset and 5 FAQ pairs that are similar to that category.

The paper structure comprises a review of FAQ Question Answering System-related work in Section 2, an overview of task techniques in Section 3, experiments and results in Section 4, and a conclusion in Section 5.

2 RELATED WORK

FAQ models simply need to extract the FAQ pairs instead of the complete context-specific answer. These FAQ pairs are made up of a question and an answer. The correspondence between the query and the FAQ pairs is determined by comparing the query to either the questions, answers, or the concatenation of both.

The appropriate class label must be present for supervised learning to rank the FAQ pairs. Recent approaches shown in Table 1 utilize both supervised and unsupervised techniques for the FAQ retrieval task. Unsupervised methods can act more effectively as they require no labeling of the data. (Sakata et al., 2019) proposes a supervised technique for FAQ retrieval. It leverages the TSUBAKI model for retrieving the q-Q similarity and BERT for q-A matching (Sakata et al., 2019). A novel technique that generates question paraphrases compensates for the lack of a query-question matching training data (Mass et al., 2020). For the re-ranking, it uses elastic search, passage re-ranking, and finally ranks on the basis of query-answer and query-question similarity. This model uses BERT for training query-question and query-answer similarity.

(Piwowarski et al., 2019) uses an attention mechanism for FAQ Retrieval. It compares various aggregation methods to effectively represent query, question, and answer information. It is observed that attention mechanisms are consistently the most effective way to aggregate the inputs for ranking. Attentive matching in FAQ retrieval eliminates the need for feature engineering to effectively combine query-answer and query-question embeddings. (Jeon et al., 2005) assumed that if answers demonstrate semantic resemblance, their associated questions will also possess a comparable level of similarity. The author employed different similarity metrics, including cosine similarity with TF-IDF weights, LM-Score, and a symmetric version of the LM-Score. LM-Score measures semantic similarity by converting answers into queries and using query likelihood language modeling for retrieval. However, its resulting scores are not symmetric. The measure gauges the semantic similarity between answers, with higher scores indicating stronger semantic connections. To address the non-symmetry issue, a modification is introduced known as Symmetric LM-Score which employs a harmonic mean of ranks for a balanced assessment in question-answering systems. It uses the rank method instead of scores, where the similarity between answers A and B is determined by the reverse harmonic mean based on their respective ranks.

3 DATASETS USED

3.1 FAQIR Dataset

We used the FAQIR (Karan and Šnajder, 2016) dataset for evaluation, which is derived from the "maintenance & repair" domain of the Yahoo! An-

Table 1: Existing State-of-the-art FAQ retrieval models.

Method	Dataset	Additional experiments and findings
Two BERT models are trained for query-answer and query-question matching using unsupervised FAQ retrieval and then augmented with the BM25 similarity measure for effective re-ranking according to user queries (Mass et al., 2020).	FAQIR	The unsupervised model competes with or outperforms existing supervised approaches. However, entire dataset findings are omitted; alternative models built for query-answer and query-question similarity give opportunities for capturing semantic details.
The research compared three aggregation techniques—Deep Matching Network (DMN), Multihop Attention Network (MAN), and Symmetrical Bilateral Multi-Perspective Matching—and ranked them using cosine similarity (Gupta and Carvalho, 2019)."	SemEval-CQA task3, Tax-Domain QA	Using an attention technique during the aggregation step improves performance.
The unsupervised technique TSUBAKI analyses query-question similarity, BERT tests query-answer similarity, and the suggested method picks the top 10 BERT question-answer pairings, followed by questions with the highest TSUBAKI score based on the OKAPI BM25 similarity measure (Sakata et al., 2019).	LocalgovFAQSet-in Japanese language, StackExchange	Retrieval can be combined with the query-question and query-answer matching score also
The study considers that the semantic similarity among answers indicates similar queries and employs three metrics for symmetry: Cosine + TFIDF for symmetry, LM-Score for query conversion with non-symmetric probability, and a symmetric variation of LM-Score(Jeon et al., 2005).	5200 question-answer pairs from NHN Corp.	The approach recovers comparable question pairs through answer matching across varied collections, utilizing provided similarity metrics for clustering, with potential applications in automating FAQ development and improving question and answer retrieval system performance.

swers community website. The dataset contains 4313 FAQ pairs and 1233 queries with corresponding manually annotated relevance judgments. The judgments are described as 1- relevant, 2- useful, 3- useless, and 4- irrelevant. Each query has at least one FAQ pair annotated as "relevant". However, it is possible for a FAQ-pair to be irrelevant for all queries. We utilized the FAQIR dataset for the FAQ task as it also has answers to the questions.

3.2 Quora Question Pairs (QQP)

Each logically distinct question is represented by a single question page in the dataset, enhancing knowledge-sharing efficiency. For instance, queries such as "What are the symptoms of influenza?" and "How can I identify if I have the flu?" should not be present as separate entities on Quora, as they share identical intent. The purpose of the data set is to facilitate a study on whether question text pairings match semantically identical queries. The dataset facilitates the development and evaluation of models of semantic equivalence using authentic Quora data, which holds 363,871 question pairs. Each line includes a binary value (0/1) indicating if the line contains a duplicate pair, the complete text of each question, and the IDs for each question in the pair (Chen et al., 2017). The reason for not utilizing this dataset is that it doesn't contain data on the answer aspect of the FAQ; it only contains question pairs.

3.3 Other Datasets

There are other datasets, such as StackFAQ and COUGH Dataset (Zhang et al., 2020), which provide FAQ questions and answers along with the queries. StackFAQ holds some ambiguity with respect to what is to be treated as a FAQ pair and what is to be treated as a query. The COUGH dataset is a multilingual dataset and can be explored for the multilingual task.

4 METHOLOGY

This section will elaborate on the model adopted for the FAQ retrieval system, followed by the details regarding the web interface.

The FAQ section of a QA system typically contains answers to many commonly asked questions. An FAQ comprises sets of questions and their corresponding answers. The FAQ retrieval task entails the process of ranking question-answer pairs based on a provided user query. The developed model provides responses to user queries within a closed domain question-answering system (QAS) by leveraging the information stored in the pre-existing FAQ database,

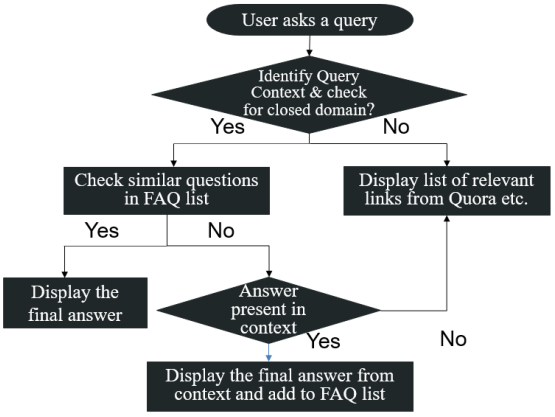

Figure 1: Flowchart for the Developed Question-Answering Model.

achieved through the process of verifying whether a similar question is present within the database. Upon receiving a user query, the model first utilizes an SVM (support vector machine) classifier(Cortes and Vapnik, 1995)to determine the question's context, checking if it aligns with the domain. If the query relates to the domain, the model conducts a search in the FAQ database. Conversely, if the query is not domain-related, the model provides relevant links from other systems based on the question's context. The question-answering model, as depicted in Figure 1, initially conducts a search for the user's query within the FAQ database. In the event that the query is not located in the database, the model proceeds to derive the answer from the context (paragraph). Following the extraction of the answer from the context, the question and answer pairs are subsequently stored within the FAQ database. We employed the transformer-based Bidirectional Encoder Representations from Transformers (BERT) model (Devlin et al., 2018) to extract answers from the context. We conducted experiments with various FAQ retrieval techniques to optimize the provision of answers to user queries. The completed model suggests the most pertinent list of FAQ pairs to the user, along with the response to their query. The details pertaining to the techniques employed and the training of FAQ models are presented as follows:

4.1 Data Preprocessing

Data Preprocessing is a very important step to fine-tune the dataset. The major sub-tasks we performed during data preprocessing include:

1. Made all the FAQ pairs and queries lowercase
2. Removing punctuations
3. Removing stopwords
4. Removing question numbers

The Sentence BERT (SBERT) and DistilBERT models used pre-processing steps 1, 2, and 4. The BM25 models used pre-processing steps 1, 2, 3 and 4. The BERT models used pre-processing steps 1 and 2.

4.1.1 Building the Training Data for the Query-Answer (q-A) Model

The original FAQIR dataset is composed of question-answer pairs sourced from the FAQ database. Additionally, it incorporates queries, each accompanied by a list of questions from the FAQ database that corresponds to the query. The final q-A model should be able to give a similarity score of whether the given answer matches the query or not. To build it, we need to fine-tune the model with a dataset that contains:

- (Q, A) the matching FAQ pair
- (Q, A') an FAQ question with a non-matching answer

This is done by randomly selecting A' for every question. The (Q, A') pairs were labeled as 0, whereas the (Q, A) pairs were labeled as 1, as in Figure 2

FaqQuestion	Answer	Label
how do you seal leaking dorme	fixing a leaky roof especially somethi	1
how do i connect a us water filt	lots of elbow grease wash down the	0
how do i install front door spea	carefully remove all the screws from t	1
how do i change rear brakes or	couldnt find exactly your vehicle but	0
how do i grease my slip yoke g	i own a repair shop and the clunk yo	1
how do you replace a 3 way sw	go to your local hardware store hom	1
how do i fix a broiler in my gas	water pumps last less than 100k miles	0
how do you get to the actual sp	check with the owners manual also g	1
how to wire hbl insulgrip twist l	i m not sure what you are trying to wi	1

Figure 2: A sample of the dataset used for training the q-A model.

4.1.2 Building the Training Dataset for the Query-Question(q-Q) Model

The model built with this dataset should be able to give a similarity score of whether an FAQ question matches the query. The training dataset was derived from the model proposed in (Mass et al., 2020). Label 1 was assigned to the corresponding question-paraphrase pairs. The label 0 was assigned to the question pairs that did not match. The second half of the dataset was built by random selection of a question from the FAQ database.

4.2 FAQ Models

4.2.1 BERTScore

BERTScore was introduced as a metric for evaluating language generation (Zhang et al., 2019). The central

idea is that, given a candidate sentence and a reference sentence, contextual embeddings are used to represent tokens (in the sentences) and compute cosine similarity. To compute the cosine similarity between tokens, greedy matching is used for matching of every token in a sentence to the most similar token in the other sentence, thus maximizing the aggregated matching score. We used BERTScore to compute the similarity score of a given input query with the FAQ pairs provided in the dataset for FAQ retrieval. The intuition here is that a relevant FAQ pair to a query will contain words similar to those in the query, thus yielding a higher similarity score. Using this method, we can retrieve FAQ pairs in a completely unsupervised and scalable manner.

We compute similarity scores as shown in equation 1 for each input query q with every FAQ pair p provided in the dataset. Here, p represents the contextual embedding of the concatenation of both the question and answer for the given FAQ pair, and q represents the contextual embedding of the input query generated by the BERT model.

$$similarity_{score} = Cosine_{contextual\_embeddings}(q_i, p_j) \quad (1)$$

Using the bert-score library, recall, precision, and F1 scores can be obtained for each such query and FAQ pair. Using these candidates, we then use the top-k FAQ pairs based on the F1 score as our retrieved FAQ pairs and compute our evaluation metrics (P@5, MAP, MRR).

4.2.2 SBERT

Sentence-BERT (Reimers and Gurevych, 2019) is an adapted version of the pre-trained BERT model. It employs siamese and triplet network structures to generate semantically meaningful sentence embeddings, allowing for comparison using cosine similarity. We trained SBERT for query-answer (qA) comparison in two ways: 1) Taking 1:1 ratios in A vs A' ratio for the dataset 2) Taking 1:5 ratios in A vs A' ratio for the dataset. We tried two variants of SBERT for query-question (qQ) comparison. SBERT encoding is directly used to obtain the similarity score for query and FAQ questions. Fine-tuned SBERT is built by training the SBERT model using the dataset described in the previous section. For the query-answer (qA) comparison, a bert-base uncased model is used.

4.2.3 DistilBERT

DistilBERT is a rapid, cost-effective, and lightweight transformer-based model derived from the BERT architecture. The model is obtained using knowledge distillation during the pre-training stage to decrease its size. The model has fewer parameters than the original BERT model but preserves over 95 % of BERT's performance. (Sanh et al., 2019).

For the query-answer (qA) model, distilbert base-uncased is used, and for the query-question (q-Q) comparison model, distillery-bert-uncased is used to obtain the sentence embeddings.

4.2.4 BM25 qQ

The corpus is built using the pre-processing methods applied to the FAQ questions. Then, the BM25 model is applied using the rank-bm25 library. The top 100 results are retrieved, and the performance metrics are calculated for them (Robertson et al., 2009).

4.2.5 BM25 q(Q+A)

Each corresponding FAQ question and answer is concatenated and is represented as Q+A. The corpus is built using the pre-processing on the FAQ Q+A. Then, the BM25 model is applied using the rank-bm25 library. The top 100 results are retrieved, and the performance metrics are calculated for them.

4.2.6 BM25 q(Q+A) + BERT qA

The BERT model is trained on triplets (question, corresponding answer, non-corresponding answer) to understand the intricacies of matching and non-matching answers. The learning rate is 2e-5, and the number of epochs is 3.

Example of triplets used in training BERT qA model

Question - How do you change an alternator?
Answer - Depending on what model car you have it will require different steps most libraries have manuals for these operations chilton's is probably the best if you can't find one at the library I'm sure you can buy one for your car online good luck
AnswerDash - You need to flush out the water heater with a garden hose it is probably filled with little rocks the inlet is probably at the bottom of the tank

Figure 3: Training example of BERT qA model.

The top 100 FAQ pairs are picked using BM25 Q+A. The encoding of the answers of these 100 FAQ pairs is found and compared with the query encoding using cosine similarity. The FAQ pairs are re-ranked based on these cosine similarity scores. The relevance of the retrieved FAQ pairs is cross-checked with the relevance score in the dataset, and the performance metrics are calculated accordingly. An example for training BERT qA is given in Figure 3.

Example of triplets used in training BERT qQ model

Question - How to get rid of garbage disposal odor?
Paraphrase - How do I clean the disposal and how do I get rid of the smell of paraffinic garbage?
QuestionDash - How do you fix a heater for on a van?

Figure 4: Training example of BERT qQ model.

4.2.7 BM25 q(Q+A) + BERT qQ

The BERT (Devlin et al., 2018) model is trained on triplets (question, paraphrase, non-matching question) to understand the intricacies of matching and non-matching questions. The learning rate is 2e-5, and the number of epochs is 3. Top 100 FAQ pairs are picked using BM25 Q+A. The encoding of the questions of these 100 FAQ pairs is found and compared with the query encoding using cosine similarity. The FAQ pairs are re-ranked based on these cosine similarity scores. The relevance of the retrieved FAQ pairs is cross-checked with the relevance score in the dataset and the performance metrics are calculated accordingly. An example for training BERT qQ is given in Figure 4.

5 EXPERIMENTS AND RESULTS

The experiments have been executed employing methods for FAQ retrieval, and subsequent comparisons of results have been conducted. Various models and ranking techniques were explored. The following performance metrics have been used for the retrieval:

1. **Mean Precision at 5 (P@5)** is the measure of a number of relevant documents within the first five retrieved documents. It helps to determine how many relevant documents are ranked in the top 5. The more documents in the top 5, the better the information retrieval system is.
2. **Mean Average Precision (MAP)** is a measure of whether all of the relevant documents get ranked highly or not. It is needed because a relevant document being retrieved but present lower in the list would not be very useful for a user entering his/her query.
3. **Mean Reciprocal Rank (MRR)** is a measure of the position at which the first relevant document occurs within the retrieved documents.

5.1 Evaluation on FAQIR Dataset

We compared the results with the original full dataset and the filtered dataset. In our studies, training the model on the filtered dataset resulted in greater accuracy than training on the complete dataset. Initial retrieval for the filtered FAQIR dataset is performed on a subset of 789 FAQ pairings pertinent to at least one user query (Mass et al., 2020). The metrics obtained for different methods on the filtered dataset have been displayed in Tables 2. Using the BertScore, we retrieved the FAQ pairs completely in an unsupervised way with comparable accuracy. The model with BM25 q(Q+A)+BERT qQ gives better results than others. While the filtered dataset led to higher accuracy, it is essential to acknowledge potential limitations, such as reduced diversity and the risk of introducing biases. Utilizing the complete dataset, we intend to capture the full range of variances and complexity included in the data, allowing for a more thorough study. The metrics obtained for different methods on the full dataset have been displayed in Tables 3. An example of the top 5 retrieved answers with relevance scores is shown in Figure 6.

We observed that the BM25 q(Q+A) + BERT qQ model gives the best results among all other models. This is because the BM25 model focuses on lexicons in the corpus, and the BERT model focuses on semantic meaning. Hence, they complement each other. BM25 q(Q+A) works better than BM25 qQ because concatenating the answer with the question provides more scope for matching lexicons. Words present in the query may be absent in the FAQ question but present in the FAQ answer. The BM25 q(Q+A) + BERT qA model does not work well in comparison to the BM25 q(Q+A) + BERT qQ model. This is because the semantics of a query and answer are usually very different. And hence, BERT qA does not perform that well. It is also observed that on adding the BERT qA model to the BM25 q(Q+A) model, the performance worsens.

5.2 Evaluation on QQP Dataset

We rigorously evaluated the top-performing models on the QQP dataset, adjusting both the number of examples and the threshold for determining question similarity based on the cosine similarity score. Our evaluation methodology involved varying the number of examples and introducing thresholds (random values with a higher similarity score) to discern question similarity. For instance, with a threshold set at 0.8, we selectively retrieved examples with scores surpassing this threshold, calculating accuracy based

Table 2: Performance Comparison on FAQIR Dataset (on Filtered Queries).

Model	Ranking Method	P@5	MAP	MRR
BM25 q(Q+A)	(top qA, sort qQ)	0.48	0.44	0.74
BM25 q(Q + A) + BERT qQ	(All pairs)	0.423	0.515	0.681
BM25 q(Q + A) + BERT qQ	(Only {1, 2} pairs)	0.499	0.751	0.788
BertScore	(all pairs)	0.215	0.552	0.554

Table 3: Performance Comparison on FAQIR Dataset (on the full dataset).

Model	Ranking Method	P@5	MAP	MRR
1:1 qA training SBERT	(top qA, sort qQ)	0.14	0.32	0.33
1:5 qA training SBERT	(top qA, sort qQ)	0.19	0.35	0.37
1:5 qA + qQ training DistilBERT	(0.2*qA score+0.8*qQ score)	0.18	0.30	0.40
BM25 qQ training	(top 100 FAQ Q)	0.30	0.38	0.57
BM25 q(Q+A) training	(top 100 FAQ Q+A)	0.34	0.39	0.60
BM25 q(Q+A) + BERT qA training	(BM25 top 100 + rerank qA)	0.27	0.32	0.52
BM25 q(Q+A) + BERT qQ training	BM25 top 100 + rerank qQ)	0.42	0.51	0.69

on this subset of retrieved examples. This meticulous approach allowed us to comprehensively assess the models' performance under different conditions, providing valuable insights into their effectiveness in capturing question similarity nuances. The accuracy metrics for the experiments are shown in Table 4.

Table 4: Performance comparison on QQP dataset.

Threshold	0.8	0.85	0.9	0.95	Example count
BERT	61.6	62.8	63.3	62.4	1000
SBERT	69.3	70.02	71.6	69.5	
BERT	63.28	64.12	63.98	62.62	10,000
SBERT	70.33	72.82	72.61	69.60	
BERT	65.38	67.12	69.98	65.62	100,000
SBERT	73.33	74.82	75.61	74.60	

We noticed that SBERT consistently outperforms BERT over all the different numbers of training samples. The accuracy peak occurs at a threshold value of 0.9 for 1000 examples, 0.85 for 10,000 examples, and 0.9 for 100,000 examples. The maximum accuracy achieved is 75.61 for SBERT on 100,000 examples of the QQP dataset.

5.3 Web Interface

The web interface is built using HTML, CSS, JavaScript, and Flask. The BM25 q(Q+A) + BERT qQ model is implemented at the front end. The front end consists of a home page (Figure 5), where the user enters the query to get the responses. The result has two aspects: an answer and a "People also asked" section, which lists five pairs of FAQs that are similar to the query. Users can directly access these recommended question-answer pairs according to the query.

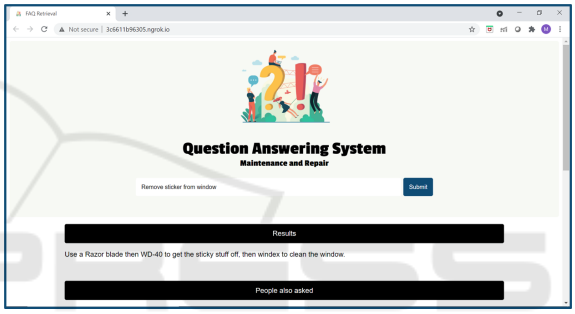

Figure 5: Web Interface.

6 CONCLUSION

FAQ retrieval is a common natural language processing task where the goal is to match a user's query to a relevant, frequently asked question (FAQ) from a dataset. We trained and evaluated different unsupervised techniques for the task of FAQ retrieval using the FAQIR dataset. Starting from the vanilla SBERT model with an MRR of 0.33, we present the final BERT qQ + BM25 q(Q+A) model with P@5, MAP, and MRR of 0.69 on the full FAQIR dataset. We observe that SBERT consistently outperforms BERT over all the different numbers of training samples on the QQP dataset by matching the query with FAQ questions. Our final model is a fusion of triplet-trained BERT and BM25 ranking functions. The current model limitation involves reliance on contextual or external sources for unanswerable questions, prompting a need for innovative solutions. Future work may extend model evaluations to diverse datasets to comprehensively assess generalizability.

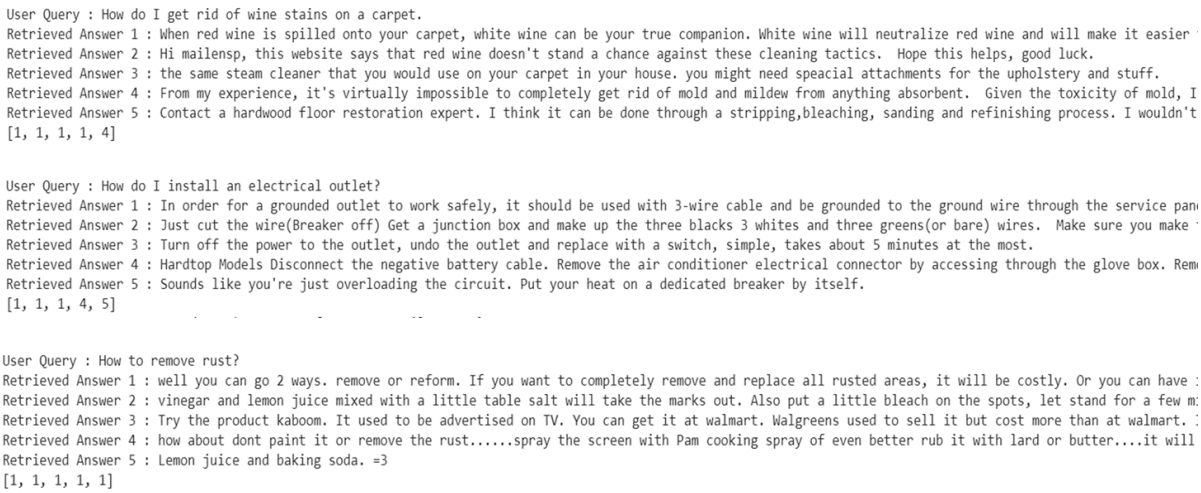

Figure 6: List of some generated answers by the FAQ model.

ACKNOWLEDGMENT

The authors extend sincere gratitude to the Department of Science and Technology (DST/ICPS/CLUSTER/DataScience/2018/Proposal 16:(T-856)) for financial support at CSIS, BITS Pilani, India.

REFERENCES

Chen, Z., Zhang, H., Zhang, X., and Zhao, L. (2017). Quora question pairs.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gupta, S. and Carvalho, V. R. (2019). Faq retrieval using attentive matching. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 929–932.

Jeon, J., Croft, W. B., and Lee, J. H. (2005). Finding semantically similar questions based on their answers. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 617–618.

Karan, M. and Šnajder, J. (2016). Faqir–a frequently asked questions retrieval test collection. In *Text, Speech, and Dialogue: 19th International Conference, TSD 2016, Brno, Czech Republic, September 12-16, 2016, Proceedings 19*, pages 74–81. Springer.

Karan, M. and Šnajder, J. (2018). Paraphrase-focused learning to rank for domain-specific frequently asked questions retrieval. *Expert Systems with Applications*, 91:418–433.

Mass, Y., Carmeli, B., Roitman, H., and Konopnicki, D. (2020). Unsupervised faq retrieval with question generation and bert. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 807–812.

Piwowarski, B., Chevalier, M., and Gaussier, É. (2019). Sigir'19: Proceedings of the 42nd international acm sigir conference on research and development in information retrieval. In *42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019)*. ACM: Association for Computing Machinery.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Robertson, S., Zaragoza, H., et al. (2009). The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Sakata, W., Shibata, T., Tanaka, R., and Kurohashi, S. (2019). Faq retrieval using query-question similarity and bert-based query-answer relevance. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1113–1116.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2019). Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zhang, X. F., Sun, H., Yue, X., Lin, S., and Sun, H. (2020). Cough: A challenge dataset and models for covid-19 faq retrieval. *arXiv preprint arXiv:2010.12800*.