Optimized Transformer Models for FAQ Answering

Sonam Damani(✉), Kedhar Nath Narahari, Ankush Chatterjee, Manish Gupta, and Puneet Agrawal

Microsoft, Hyderabad, India
{sodamani,kedharn,anchatte,gmanish,punagr}@microsoft.com

Abstract. Informational chatbots provide a highly effective medium for improving operational efficiency in answering customer queries for any enterprise. Chatbots are also preferred by users/customers since unlike other alternatives like calling customer care or browsing over FAQ pages, chatbots provide instant responses, are easy to use, are less invasive and are always available. In this paper, we discuss the problem of FAQ answering which is central to designing a retrieval-based informational chatbot. Given a set of FAQ pages s for an enterprise, and a user query, we need to find the best matching question-answer pairs from s. Building such a semantic ranking system that works well across domains for large QA databases with low runtime and model size is challenging. Previous work based on feature engineering or recurrent neural models either provides low accuracy or incurs high runtime costs. We experiment with multiple transformer based deep learning models, and also propose a novel MT-DNN (Multi-task Deep Neural Network)-based architecture, which we call Masked MT-DNN (or MMT-DNN). MMT-DNN significantly outperforms other state-of-the-art transformer models for the FAQ answering task. Further, we propose an improved knowledge distillation component to achieve ∼2.4x reduction in model-size and ∼7x reduction in runtime while maintaining similar accuracy. On a small benchmark dataset from SemEval 2017 CQA Task 3, we show that our approach provides an NDCG@1 of 83.1. On another large dataset of ∼281K instances corresponding to ∼30K queries from diverse domains, our distilled 174 MB model provides an NDCG@1 of 75.08 with a CPU runtime of mere 31 ms establishing a new state-of-the-art for FAQ answering.

1 Introduction

Reducing agent costs in the call center is typically high on the list of priorities of call center managers in any enterprise. Enterprises put up frequently asked questions (FAQ) pages to satisfy users' frequent information needs so as to avoid such calls. But often such pages are too large and not very well structured for users to read. The difficulties faced by users in interacting with the FAQ pages are multi-fold – (1) User has to scan through a long list of QA pairs. (2) FAQs in a list may be poorly organized and not semantically grouped. (3) Multiple FAQs may answer the query, and the user must look out for a QA pair that answers

H. W. Lauw et al. (Eds.): PAKDD 2020, LNAI 12084, pp. 235–248, 2020.
https://doi.org/10.1007/978-3-030-47426-3_19

the question with the right level of specificity. (4) An FAQ list may sometimes be scattered over several documents.

In addition, a poorly managed call center or mismatching working hours for global customers, could lead to long wait times for customers who may then move over to other competitive businesses. Alternatively, users pose such queries on community question answering (cQA) forums, or contact businesses over slow media like emails or phone calls. In 2014, Quora, a popular cQA forum, claimed that 10% of U.S. population uses its service every month[1] contributing to a total of 61M questions with 108M answers[2]. Such popularity of cQA forums at least partially indicates the difficulty faced by users in interacting with FAQ pages to obtain answers.

To provide correct information instantly at much lower operating costs, retrieval-based chatbots that can match user queries with content on FAQ pages are highly desirable. In this paper, we discuss the problem of FAQ answering which is central to designing a retrieval-based information chatbot. Let D denote the set of question-answer pairs extracted from a set of FAQ pages s for an enterprise. Given D and a user query q, our goal is to rank question-answer pairs in D. Top K QA pairs with high scores are returned to the user. Figure 1 shows possible system snapshots using two user interfaces – web search as well as a chatbot. In case of web search interface (left of Fig. 1), K is set to 4, while $K = 1$ for the chatbot interface (right of Fig. 1).

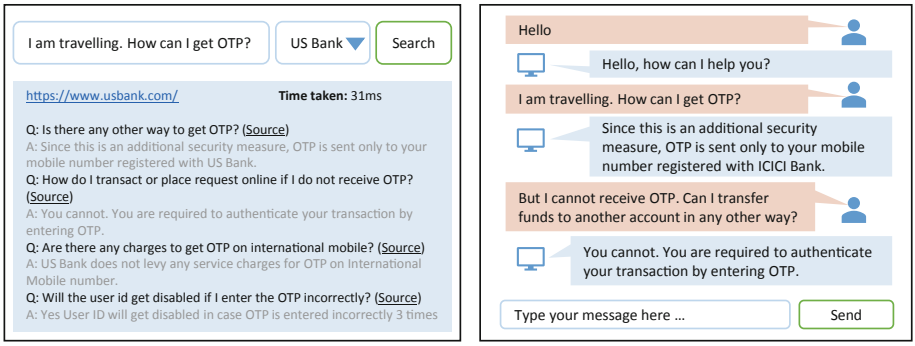

Fig. 1. Web Search interface (left), Chatbot interface (right);

Note that this problem is similar to the problem of automatically answering questions on cQA forums by matching existing question-answer pages. However, there are some major differences as follows: (1) Queries on cQA forums are much longer than queries and questions on FAQ pages. In fact, often times, cQA queries have a subject and a body [29]. (2) cQA forums have a user network. Thus every

[1] https://venturebeat.com/2015/12/21/quora-claims-10-of-u-s-population-uses-its-service-every-month/.

[2] https://www.quora.com/How-many-questions-have-been-asked-on-Quora-1.

QA pair is associated with a set of users. Unlike that, when ranking QA pairs from FAQ pages, we cannot exploit signals from any user network. (3) cQA pages typically have a question but multiple user-voted answers. FAQ pages have no user-voting, and only one answer per question. (4) On cQA forums, different answers may apply based on user context. On FAQ pages, every question has a unique answer.

FAQ answering is a challenging task. Solving the problem needs prediction of query-question semantic similarity and query-answer relevance, in a joint manner. Also, building a general system that works across domains implies that we cannot resort to any domain specific heuristics. Finally, although recent deep learning based systems provide high accuracy across multiple NLP tasks, building a deep learning based system for FAQ Answering for large QA databases with low runtimes and model size brings in more challenges.

Previous work on answering a question, given FAQ pages, was based on feature engineering (FAQ-Finder [8], Auto-FAQ [28], [11,22]) or typical attention-based recurrent neural network models (SymBiMPM [7]). Recently, transformer based networks [24] have shown significant gains across many natural language processing tasks. In this paper, we propose the use of transformer network based methods like Bidirectional Encoder Representations from Transformers (BERT) [6] and Multi-task Deep Neural Network (MT-DNN) [15]. Further, we propose a novel architecture, MMT-DNN, based on a masking trick specifically applicable to input in the form of (query, question, answer) triples. To make such models practically usable, we need to reduce the model size as well as the execution time. Hence, we propose an improved knowledge distillation method for our MMT-DNN model. Our experiments with two datasets show that the proposed model outperforms all the baselines by significant margins. Also, our distilled 174MB MT-DNN-3 model provides a runtime of mere 31 ms making it usable in real-time chatbot scenarios. We make the following main contributions in this paper.

- We propose the use of transformer based models like BERT and MT-DNN for solving the FAQ Answering task, and also present a novel architecture, MMT-DNN, that achieves better accuracy.
- We propose and experiment with an improved knowledge distillation method to reduce the model size and model runtime.
- On two real world datasets, our proposed MMT-DNN establishes a new state-of-the-art for FAQ answering.

2 Related Work

Data Mining for FAQ Web Pages. Research on FAQ web pages has focused on three sub-areas: (1) FAQ mining using list detection algorithms [11,14], (2) answering questions using FAQ web pages [8,11,22,28], (3) navigational interface for Frequently Asked Question (FAQ) pages [20], and (4) Completeness

of FAQ pages [3]. In this paper, we focus on the FAQ answering task. Previous work on answering a question given FAQ pages (FAQ-Finder [8], AutoFAQ [28], [2,11,13,21,23]) was based on traditional feature engineering for surfacing statistical/semantic similarities between query and questions. Most of these works considered similarity between query and questions, very few considered query-answer similarity. We use transformer based deep learning methods for jointly considering query-question and query-answer similarity.

Recently deep learning based methods have been proposed for FAQ Answering. Wu et al. [29] propose the attention-based Question Condensing Networks (QCN) to align a question-answer pair where the question is composed of a subject and a body. To suit our problem setting, we experiment by substituting query for the subject, and question for the body. Gupta et al. [7] propose SymBiMPM (BiLSTMs with multi-perspective matching blocks) for computing query-QA match. Recently, transformer network models have emerged as state-of-the-art across multiple NLP tasks. Hence, unlike previous deep learning works, we use transformer networks for FAQ answering.

Applications of Transformer Models. After the original Transformer work by Vaswani et al. [24], several architectures have been proposed like BERT [6], MT-DNN [15] etc. The GLUE [26] and the SuperGLUE [25] dashboards tell us that such models have outperformed previously proposed methods across complex NLP tasks like text classification, textual entailment, machine translation, word sense disambiguation, etc. We present the first work to investigate application of transformers to FAQ answering task.

Model Compression. Existing deep neural network models are computationally expensive and memory intensive, hindering their deployment in devices with low memory resources or in applications with strict latency requirements. Chatbots expect near realtime responses. Thus, transformer based models need to be compressed and accelerated. In the past few years, multiple techniques have been proposed for model optimization including pruning, quantization, knowledge distillation, and low rank factorization. Cheng et al. [5] provide a good survey of such methods. In this paper, we explore different variations of knowledge distillation and present a novel architecture that provides best results for the FAQ answering task.

3 Approach

Given a question-answer database, when a user query q arrives, we first compute a list of candidate QA pairs which have high BM25 score [19] with respect to the query. Given the latency constraints, we use computationally cheap BM25 match, however, understandably, BM25 may have missed semantically similar but syntactically different QA pairs. If q uses synonyms of the words in the ideal QA pair, it is possible that the pair would not be selected based on BM25 score. These candidate QA pairs, along with the original query, are scored using various methods described in this section. Top K QA pairs with high scores are returned to the user.

We first discuss baseline methods like BiLSTMs with attention and SymBiMPM [7]. Next, we discuss our proposed transformer based methods. All of these methods take query (q), question (Q), and answer (A) as input, and output one of the three classes: Good, Average or Bad indicating the degree of match between q and the (Q, A) pair. Figure 2 illustrates architectures of various methods discussed in this section.

3.1 Baselines

BiLSTMs. As illustrated in Fig. 2(A), in this approach, the query, question and answer are processed using three two-row bidirectional LSTMs [10]. The query and question BiLSTMs share weights. We use BiLSTMs with attention. The final output from the last hidden layer of each of the BiLSTMs is concatenated and fed into a fully connected neural network (MLP). The output layer has three neurons (one for each of the three classes) across all the architectures. The network is trained using Adam optimizer [12] with cross entropy loss.

SymBiMPM. Symmetric Bilateral Multi-Perspective Matching Block (SymBiMPM) is the method proposed by Gupta et al. [7]. This model uses a multi-perspective matching block [27] to compare two sequences and generate the matched representations for both these sequences. This block has four different matching mechanisms that are used on the input sequences. Matching is applied in both the directions, i.e. if P and Q are the two inputs, then the output is a matched representation of P obtained by attending to Q, and a matched representation of Q obtained by attending to P. All the BiLSTMs share weights. Also, both the match blocks share weights. As illustrated in Fig. 2(B), Multi-perspective matching blocks are used for query-question and query-answer matching followed by attention layer and fully connected layers to get the final class label.

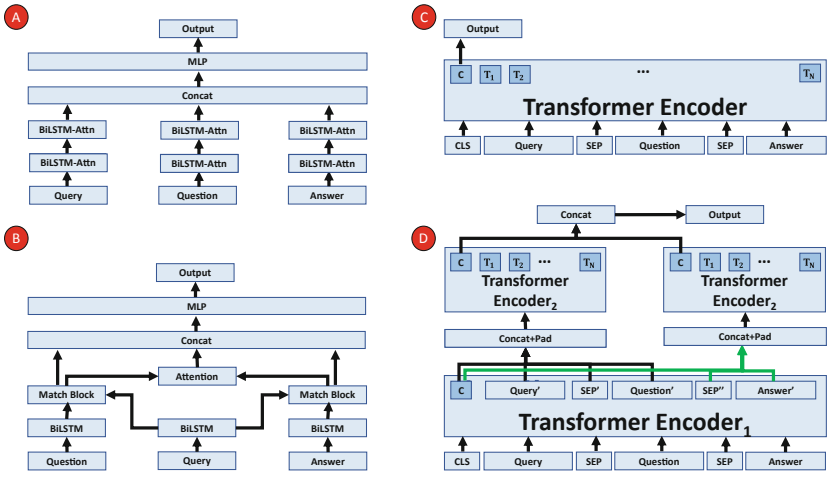

Fig. 2. Architectures of various methods: (A) BiLSTMs with attention (B) SymBiMPM (adapted from [7]) (C) BERT/MT-DNN (D) MMT-DNN

3.2 Proposed Methods

Transformer networks proposed by Vaswani et al. [24] follow a non-recurrent architecture with stacked self-attention and fully connected layers for both the encoder and decoder, each with 6 layers. BERT and MT-DNN are two most popular extensions of the Transformer encoder network. Broadly this architecture is illustrated in Fig. 2(C).

BERT. BERT [6] essentially is a transformer encoder with 12 layers. We used the pre-trained model which has been trained on Books Corpus and Wikipedia using the MLM (masked language model) and the next sentence prediction (NSP) loss functions. The query, question and answer are concatenated into a sequence and are separated with a special "SEP" token. The sequence is prepended with a "CLS" token. The representation C for the "CLS" token from the last encoder layer is used for classification by connecting it to an output softmax layer. Optionally, we can finetune the pre-trained model using labeled training data for the FAQ answering task.

MT-DNN. The MT-DNN architecture [15] extends BERT by further pre-training it with large amounts of cross-task data. Specifically, the MT-DNN is a 12 layer transformer encoder where the BERT model has been further pre-trained using single sentence classification, text similarity, pairwise text classification and relevance ranking tasks. The representation C for the "CLS" token from the last encoder layer is used for classification by connecting it to an output softmax layer. Optionally, we can finetune the pre-trained model using labeled training data for the FAQ answering task.

MMT-DNN. The proposed Masked MT-DNN method modifies the MT-DNN architecture, as illustrated in Fig. 2(D). The transformer encoder is divided into two parts: encoder_1 and encoder_2. Encoder_1 consists of l encoder layers, while encoder_2 contains 12-l layers. l is a hyper-parameter tuned on validation data. The input sequence (query, question, answer) is first processed by encoder_1 to get a transformed sequence (query', question', answer'). Intuitively, (query, question) pair is more homogeneous compared to (query, answer) pair. Hence, we explore disjoint encoding of the two pairs using separate encoder_2 blocks for query-question and query-answer matching. Both encoder_2 blocks share weights. Specifically, the first encoder_2 block receives the concatenated string of the CLS token, query and question as input, where the answer is masked by replacing answer tokens by zeros. Similarly, the second encoder_2 block receives the concatenated string of the CLS token, query and answer as input, where the question is masked by replacing the question tokens by zeros. The C token from both these encoder_2 blocks are concatenated and connected to an output softmax layer.

Knowledge Distillation. The proposed MMT-DNN model, like other Transformer models, is very large and also incurs a large number of computations at prediction time. Hence, we use knowledge distillation strategies [1] to compress the model and reduce latency while retaining accuracy. Figure 3 shows our improved knowledge distillation component.

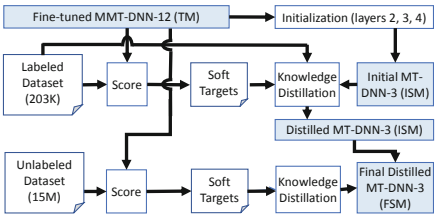

Fig. 3. Knowledge distillation for FAQ answering

Table 1. Dataset statistics (train/dev/test)

Dataset	SemEval-2017	FSD
#Queries	266/72/70	20242/1966/7478
#Question-Answer pairs	6711/1575/2313	1630/477/649
#Data points	9977/1851/2767	202969/22549/55751
Avg length of queries	41.2/37.9/43.7	7.2/9.3/9.5
Avg length of questions	50.4/47.2/49.1	7.7/9.8/7.9
Avg length of answers	48.9/45.1/46.3	61.4/55.3/57.9

We use student-teacher networks for knowledge distillation [9] by considering the fine-tuned MMT-DNN-12 model as a teacher model (TM) for knowledge distillation. Layers 2, 3, and 4 of the fine-tuned MMT-DNN-12 are used to initialize a MT-DNN-3 model which is the initial student model (ISM) for knowledge distillation. Note that the student model is a MT-DNN and not a MMT-DNN. A combination of hard targets from the labeled dataset and soft targets from the fine-tuned MMT-DNN-12 TM is used to define the loss for training the MT-DNN-3 model to obtain the distilled student model (DSM). Although not shown in the figure (due to lack of space), in order to facilitate gradual transfer of knowledge, the distillation from MMT-DNN-12 to MT-DNN-3 is done in a chain of steps where MMT-DNN-12 is first distilled to a MT-DNN-9, then to MT-DNN-6 and finally to an MT-DNN-3 student model (DSM) [17]. We also have access to a much larger (15 million sized) unlabeled dataset of queries which lead to clicks to FAQ pages. These are scored against the TM to generate soft targets. These soft targets are then used to further distill the DSM, followed by TVM compiler optimizations [4] to obtain the final distilled MT-DNN-3 student model (FSM).

4 Experiments

4.1 Datasets

Table 1 presents basic statistics about the two datasets.

SemEval-2017. This dataset[3] was intended for community question answering (cQA) originally, but the task 3 data had the QA pairs grouped by search query terms, which facilitated the transformation of this data into FAQ Retrieval format where FAQs are ranked for a query and are awarded ranks as Good, Average or Bad (as in original dataset). We used standard train, dev, test splits provided by the task organizers. Although this dataset is small, we experiment with it since this is the only publicly available dataset.

[3] http://alt.qcri.org/semeval2017/task3/.

FAQ Search Dataset (FSD). This dataset was created using ∼30K queries (from a popular search engine's query log) leading to clicks to FAQ pages. We took only those queries which resulted into at least 5 clicks to some FAQ page. The query was then compared to all the QA pairs extracted from the clicked FAQ pages using BM25 score [19] to extract a max of top 15 QA pairs. We then got these (query, QA) instances judged into 3 classes (Good, Average or Bad) using a crowdsourcing platform with three-way redundancy. The queries and FAQ pages were carefully chosen such that (1) they belong to multiple domains like airports, banks, supermarkets, tourism and administrative bodies, (2) queries and QA pairs of various sizes are considered, and (3) FAQ pages with varying number of QA pairs are included. Note that this dataset is ∼20x larger than the SemEval-2017 dataset.

4.2 Accuracy Comparison

The query, question and answer are all represented using GloVe [18] embeddings for all the baseline methods. Transformer based methods use WordPiece [30] embeddings. All experiments were done on a machine with 4 Tesla V100-SXM2-32GB GPUs. We use the popular ranking metric, Normal Discounted Cumulative Gain (NDCG)@K to compare various methods. For BiLSTMs in all baseline methods, the hidden layer size was 300. For transformer based methods, the embedding size was fixed to 30522 and the input sequence length was fixed to 512 tokens.

Table 2 shows accuracy comparison across various methods on both the datasets. Block A shows results for baseline methods. Surprisingly, Sym-BiMPM [7] performs worse than BiLSTMs. SemEval-2017 dataset has labels for query-question pair, for query-answer pair, as well as query-answer pair. For SymBiMPM [7], the authors used query-question label as the label for a (query, question, answer) triple. As a refinement, we first considered only those QA pairs where question-answer label is "good", and then used the label for query-answer similarity as the label for a (query, question, answer) triple. Also, queries in the SemEval-2017 set have a subject as well as a body. Gupta et al. [7] simply used the query subject and ignored the query body. We experiment with just the query subject as well as with query subject + body. Table 2 shows that using query subject + body usually provides better accuracy, sometimes with a large margin. We also experimented with QCN [29] but the results were worse than even BiLSTMs. This is expected due to mismatch in the problem setting, as discussed in Sect. 2. Further, Block B shows results for the proposed methods. As the table shows, our proposed methods outperform existing methods by a significant margin. All results are obtained as a median of 5 runs. Both BERT and MT-DNN benefit from finetuning across the two datasets. Also, MMT-DNN outperforms all other methods by a significant margin ($p < 0.05$ using McNemar Test [16]), establishing a new state-of-the-art for FAQ answering task.

Table 2. Accuracy comparison across various methods. For SemEval-2017 dataset, results are for two settings: (using just the query subject/using query subject + body).

		SemEval-2017			FSD		
	Model	NDCG@1	NDCG@5	NDCG@10	NDCG@1	NDCG@5	NDCG@10
A	BiLSTMs	36.62/38.83	38.43/43.17	41.76/46.3	55.70	63.02	69.34
	SymBiMPM [7]	34.21/34.00	38.55/38.59	40.86/44.71	54.03	61.21	68.11
B	BERT (pre-trained)	63.38/65.39	61.85/68.41	62.87/68.44	71.77	76.17	78.47
	MT-DNN (pre-trained)	68.01/60.97	64.92/61.85	64.67/62.83	70.29	75.08	77.65
	BERT (finetune)	68.01/69.22	65.19/68.61	67.46/71.12	73.97	78.29	79.79
	MT-DNN (finetune)	70.22/82.49	67.06/81.79	67.72/81.99	73.75	78.14	79.79
	MMT-DNN	71.03/84.71	70.67/82.59	71.51/82.18	75.38	78.59	80.24

Figure 4 shows NDCG@K for $K = 1$ to 10 for the MMT-DNN approach for both the datasets. With increase in K, while the accuracy improvement for the FSD is intuitive, the result is not very intuitive for the SemEval-2017 dataset. This is mainly because of the small size of the dataset because of which usually there are very few good answers matching any query.

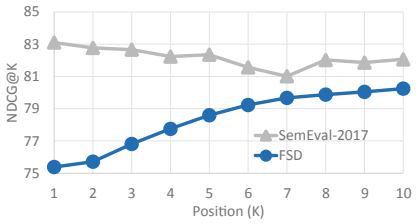

Fig. 4. NDCG@K for the MMT-DNN approach for both the datasets

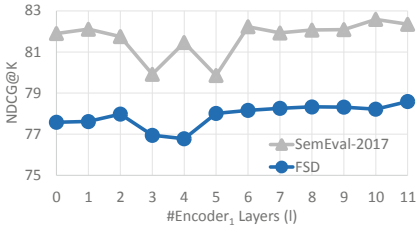

Fig. 5. NDCG@5 with varying number of Encoder$_1$ layers (l) for MMT-DNN for the two datasets

q_1 from SemEval-2017 dataset: working permit ... 1- do i need working permit since i have residence visa in qatar n under husband sponsor? 2- without working permit expat's wife could not work in qatar? ...

Q_1: Work permit for husband? I am thinking of sponsoring my husband to live in Qatar. I heard that if he gets a job; he will need to get a work permit. Are husbands able to get a work permit? ...

A_1: If he is on a family visa he needs to find a job first so that the company who will hire him will be the one to process his work permit. ...

Q_2: Work permit for husband? I am thinking of sponsoring my husband to live in Qatar. I heard that if he gets a job; he will need to get a work permit. Are husbands able to get a work permit? ...

A_2: if you get over 7k you can sponser him with fam visa. once he is here and under your visa; ... he will still remain under your sponsorship. ...

Q_3: Wife/Husband with family sponsorship to work. If your wife / husband under family sponsorship of your sponsor wants to work do they have to transfer the sponsorship to new sponsor or they can work without sponsorship change?

A_3: they don't have to transfer their sponsorship under the Company; unless they either want to or the Company requires their transfer. ...

q_2 from FSD: i've paid for my parking but my flight is delayed

Q_1: What happens if I exit the car park prior to my confirmed booking time?

A_1: If for whatever reason you cannot exit the car park in your confirmed booking time (e.g., you haven't returned due a cancelled flight), the credit card or debit card that you use to exit the car park (i.e. your nominated card) will be debited with the cost of the additional time, based on the rates displayed at the entry to the car park.

Q_2: What happens if I enter the car park prior to my confirmed booking time?

A_2: If you enter the car park before your confirmed booking time, or exit the car park later than your confirmed booking time, the credit card or debit card that you use to exit the car park (i.e. your nominated card) will be debited with the cost of the additional time, based on the rates displayed at the entry to the car park.

Q_3: How do I amend or cancel my booking?

A_3: You may cancel your Booking, for any reason at any time up to 24 hours before the start of the Booking Period. To do this, ...

Fig. 6. Top 3 QA pairs returned by MMT-DNN for two queries (one from each dataset)

Figure 5 shows the NDCG@5 for MMT-DNN across the two datasets with varying number of Encoder$_1$ layers (l). As expected, the accuracy is better at larger values of l. This means that it is useful to allow attention across question and answer in the first few layers but let the query-question and query-answer attention be learned separately at higher layers. Note that $l = 0$ corresponds to not having Encoder$_1$ at all, and processing (query, question) and (query, answer) separately throughout the network.

Next, we show two queries with top three QA pairs ranked by our MMT-DNN system in Fig. 6. For query q_1, BiLSTMs had this question as the second result: "Work Permit How many days does it take to finish the processing of a Work Permit?". Similarly, SymBiMPM leads to unrelated questions within top 3 like "Hepatitis C (HCV) - Work permit I have Hepatitis C (HCV); Can i get work permit?".

Similarly, for q_2, baselines lead to the following unrelated question in top 3: "Can I get motorbike parking?", "What do I do if I take a ticket on arrival to the car park when I should have entered my credit card?".

4.3 Attention Visualization for MMT-DNN

In Fig. 7, we visualize the heads from the multi-head self attention module of the last encoder for our best approach. This visualization helps us understand what pairs of words in the (query, question, answer) have high self-attention weights. The results are very intuitive showing that query, question and answer are jointly enhancing each other's representations to contribute to high accuracy. Figure 7

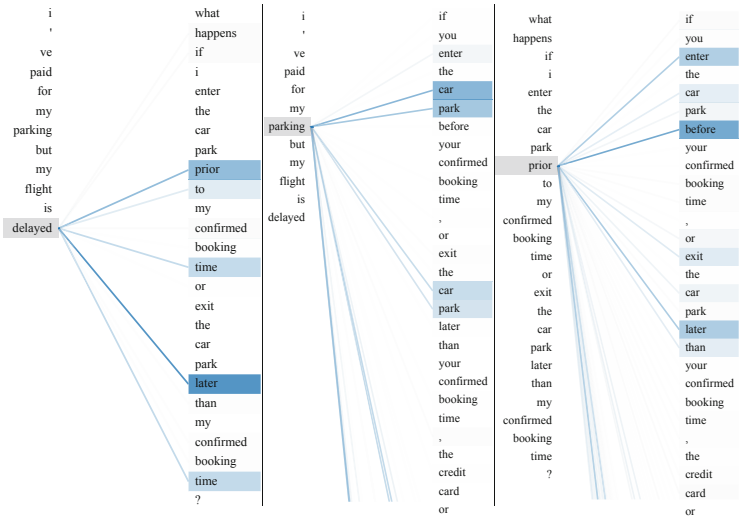

Fig. 7. Visualization of a few heads for various examples for the last encoder layer of our best approach. (left): query-question, (middle): query-answer, (right): question-answer

(left) shows the token "delayed" in the query has high attention weights for the tokens "prior", "time" and "later" in the question. Figure 7 (middle) shows the token "parking" in the query has high attention weights for the tokens "car" and "park" in the answer. Figure 7 (right) shows the token "prior" in the question has high attention weights for the tokens "before" and "later" in the answer.

4.4 Error Analysis

We analyzed error patterns for our best method (MMT-DNN). The most confusing category is the "Average class" with lowest precision and recall. Fortunately, this does not impact ranking significantly especially in cases where there are enough "good" QA pairs for a query. Further, we look at a few examples to do more detailed error analysis by manually assigning error categories to 60 (query, question, answer) triples incorrectly classified by MMT-DNN method. Table 3 shows percentages contributed by each error pattern and a few examples. Verbose Match errors accounted for more than half of the errors, which is in line with our expectations.

4.5 Knowledge Distillation (KD)

Table 4 shows the NDCG obtained using the proposed architectures for KD. Even with small labeled data, distilled MT-DNN-3 provides accuracy comparable to the teacher model. Further distillation using large unlabeled data leads to better results. Note that we fixed the hard versus soft loss balancing parameter α as 0.01. Overall, the final model TVM-optimized MT-DNN-3 provides NDCG@1 of 75.08 on FSD dataset with a model size of 174MB and a CPU/GPU runtime of 31.4/5.18 ms per instance.

Table 3. Analysis of various types of errors with examples

Category	Meaning	%	Examples
Entity mismatch	q and Q/A refer to a different main entity	29	q:"What is best mall in Doha to buy good **furniture**?", Q:"where to buy good **abhaya** in doha"
Generalization	q and Q/A have entities with "is a" relationship	7	q: "Any aquapark in **Doha**?", Q:"any water theme park in **qatar**?"
Intent mismatch	q and Q/A have different intents	5	q:"What is best mall in Doha to buy **good** furniture? ... showrooms ...", Q:"Where to buy **used** furniture? .. cheap ..."
Negation	q and Q/A have opposite intents	7	q:"Is there any Carrefour which is open?", Q:"any other good supermarkets **apart from** Carrefour"
Verbose match	q and Q/A match on unimportant parts	52	q:"Is it good offer? Hi Frds;i QA supervisor with 8 years exp in pharmaceutical have got job offer from Qatar pharma company; Salary which they have offered to me is 5000QAR...", Q:"Is it a good offer? Dear all; I need your help please:) ; i got an offer from Habtoor leighton group for Planning Engineer position. They are offering 10K ..."

Table 4. Accuracy vs size and runtime latency comparison across various models for the knowledge distillation experiments (on FSD)

Model	Size	CPU runtime	NDCG@1	NDCG@5	NDCG@10
MMT-DNN-12	417 MB	225 ms	75.38	78.59	80.24
MT-DNN-9	336 MB	210 ms	76.28	78.83	80.48
MT-DNN-6	255 MB	143 ms	74.55	77.88	79.76
MT-DNN-3	174 MB	68.9 ms	70.56	75.47	77.91
MT-DNN-3 (unlabeled data)	174 MB	68.9 ms	75.08	78.28	80.00
MT-DNN-3 (unlabeled data + TVM)	174 MB	31.4 ms	75.08	78.28	80.00

We tried various ways of initialization of the student model for knowledge distillation as shown in Table 5. Initialization using some layers of the teacher model (usually the first few layers) is clearly better than random initialization.

Table 5. Initialization for knowledge distillation for MT-DNN-3 model using MMT-DNN-12 layers or Random (on FSD)

Initialization	NDCG@1	NDCG@5	NDCG@10
Layers 1, 2, 3	69.20	74.76	77.24
Layers 2, 3, 4	70.56	75.47	77.91
Layers 4, 5, 6	65.83	71.35	75.03
Layers 7, 8, 9	68.57	73.87	76.66
Layers 10, 11, 12	59.83	67.16	71.87
Random	52.92	60.49	67.55

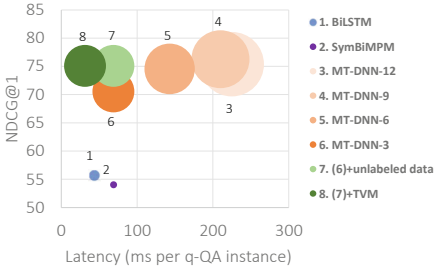

Fig. 8. Accuracy, runtime, model size comparison for various models (best viewed in color) (Color figure online)

Figure 8 shows the accuracy versus runtime trade-off for various models. The radius of the circle corresponds to the model size. Compared to all other approaches, the distilled MT-DNN-3 models are better than others, and among them the best one is the TVM-optimized MT-DNN-3 which also used unlabeled data during distillation.

5 Conclusion

We proposed the use of transformer based models like BERT and MT-DNN for solving the FAQ Answering task. We also proposed a novel MT-DNN architecture with masking, MMT-DNN, which establishes a new state-of-the-art for FAQ answering, as evaluated on two real world datasets. Further, we propose and experiment with an improved knowledge distillation strategy to reduce the model size and model runtime. Overall the proposed techniques lead to models with high accuracy, and small runtime and model size.

References

1. Ba, J., Caruana, R.: Do deep nets really need to be deep? In: NIPS, pp. 2654–2662 (2014)
2. Berger, A., Caruana, R., Cohn, D., Freitag, D., Mittal, V.: Bridging the lexical chasm: statistical approaches to answer-finding. In: SIGIR, pp. 192–199 (2000)
3. Chatterjee, A., Gupta, M., Agrawal, P.: FAQaugmenter: suggesting questions for enterprise FAQ pages. In: WSDM, pp. 829–832 (2020)
4. Chen, T., et al. *TVM*: an automated end-to-end optimizing compiler for deep learning. In: OSDI, pp. 578–594 (2018)
5. Cheng, Y., Wang, D., Zhou, P., Zhang, T.: A survey of model compression and acceleration for deep neural networks. arXiv preprint arXiv:1710.09282 (2017)
6. Devlin, J., Chang, M., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Gupta, S., Carvalho, V.: FAQ retrieval using attentive matching. In: SIGIR, pp. 929–932 (2019)
8. Hammond, K., Burke, R., Martin, C., Lytinen, S.: FAQ finder: a case-based approach to knowledge navigation. In: Conference on AI for applications, vol. 114 (1995)
9. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
10. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
11. Jijkoun, V., de Rijke, M.: Retrieving answers from frequently asked questions pages on the web. In: CIKM, pp. 76–83 (2005)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
13. Kothari, G., Negi, S., Faruquie, T.A., Chakaravarthy, V.T., Subramaniam, L.V.: SMS based interface for FAQ retrieval. In ACL, pp. 852–860 (2009)
14. Lai, Y., Fung, K., Wu, C.: FAQ mining via list detection. In: Multilingual Summarization and Question Answering, pp. 1–7 (2002)
15. Liu, X., He, P., Chen, W., Gao, J.: Multi-task deep neural networks for natural language understanding. arXiv preprint arXiv:1901.11504 (2019)
16. McNemar, Q.: Psychological Statistics. Wiley, New York (1969)
17. Mirzadeh, S., Farajtabar, M., Li, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant: bridging the gap between student and teacher. arXiv preprint arXiv:1902.03393 (2019)
18. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: EMNLP, pp. 1532–1543 (2014)
19. Robertson, S., Zaragoza, H., et al.: The probabilistic relevance framework: BM25 and beyond. FnTIR **3**(4), 333–389 (2009)
20. Schmonsees, R.J.: Dynamic frequently asked questions (FAQ) system. US Patent 5,842,221, November 1998
21. Sneiders, E.: Automated FAQ answering: continued experience with shallow language understanding. In: Question Answering Systems. Papers from the 1999 AAAI Fall Symposium, pp. 97–107 (1999)
22. Sneiders, E.: Automated FAQ answering with question-specific knowledge representation for web self-service. In: Human System Interactions, pp. 298–305 (2009)

23. Song, W., Feng, M., Gu, N., Wenyin, L.: Question similarity calculation for FAQ answering. In: Semantics, Knowledge and Grid, pp. 298–301. IEEE (2007)
24. Vaswani, A., et al.: Attention is all you need. In: NIPS, pp. 5998–6008 (2017)
25. Wang, A., et al.:. SuperGLUE: a stickier benchmark for general-purpose language understanding systems. arXiv preprint arXiv:1905.00537 (2019)
26. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: GLUE: a multi-task benchmark and analysis platform for natural language understanding. In: ICLR (2019)
27. Wang, Z., Hamza, W., Florian, R.: Bilateral multi-perspective matching for natural language sentences. arXiv preprint arXiv:1702.03814 (2017)
28. Whitehead, S.D.: Auto-faq: an experiment in cyberspace leveraging. Comput. Netw. ISDN Syst. **28**(1–2), 137–146 (1995)
29. Wu, W., Sun, X., Wang, H.: Question condensing networks for answer selection in community question answering. In: ACL, pp. 1746–1755 (2018)
30. Wu, Y., et al.: Google's neural machine translation system: bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 (2016)