DTAFA: Decoupled Training Architecture for Efficient FAQ Retrieval

Haytham Assem
Huawei Research, Ireland
haytham.assem@huawei.com

Sourav Dutta
Huawei Research, Ireland
sourav.dutta2@huawei.com

Edward Burgin
Huawei Research, Ireland
edwardburgin@huawei.com

Abstract

Automated Frequently Asked Question (FAQ) retrieval provides an effective procedure to provide prompt responses to natural language based queries, providing an efficient platform for large-scale service-providing companies for presenting readily available information pertaining to customers' questions. We propose *DTAFA*, a novel *multi-lingual* FAQ retrieval system that aims at improving the top-1 retrieval accuracy with the least number of parameters. We propose two decoupled deep learning architectures trained for (i) candidate generation via text classification for a user question, and (ii) learning fine-grained semantic similarity between user questions and the FAQ repository for candidate refinement. We validate our system using real-life enterprise data as well as open source dataset. Empirically we show that *DTAFA* achieves better accuracy compared to existing state-of-the-art while requiring nearly $30\times$ lesser number of training parameters.

1 Introduction

FAQ retrieval system provides a natural language interface for querying FAQ collection and is increasingly becoming popular with large-scale service-providing companies. Further, with the advent of personal assistants (like XiaoIce, Siri, Alexa, Google Assistant, etc.), these "virtual agents" can provide answers and help users solve routine tasks by an additional interface to FAQs, hotlines and forums – enabling a natural interaction with users (Lommatzsch and Katins, 2019).

FAQ retrieval is a challenging task, majorly attributed to the fact that question-answer texts are short, making it harder to bridge the *lexical and semantic gap* between a user query and FAQ questions due to limited context (Karan and Šnajder, 2018; Lee et al., 2008). Further, in certain cases, precise understanding of the user questions might be difficult due to informal representations, domain-specificity, abbreviations, and formal-colloquial term mismatches (Lommatzsch and Katins, 2019).

In addition, FAQ retrieval systems should be able to handle both keyword as well as *short span* "natural language" questions. Given the predominantly "customer-centric" nature, such systems generally demand higher precision and interpretability compared to traditional information retrieval methods.

Challenges. In modern interactive applications, the fluidity of natural language based human-computer interactions provides an additional metric to capture quality of user experience. For example, consider a voice-based FAQ platform interfaced via a personal assistive system. In such cases, providing the user with the top-k "matching" results (from the FAQ platform) to choose from, impedes natural fluidity of interaction. An intelligent system should be able to automatically understand and/or infer the context, meaning and relevance to provide the best matching FAQ to address the user's concern. Hence, in such scenarios the *top-1* or "one-best" accuracy tends to precisely capture the Quality-of-Service. Further, note that modern enterprises have global footprints with diverse product and service portfolios, and hence such FAQ systems should also be able to handle the challenge of *multi-lingual* customer base associated with globalization. Unfortunately, "multi-linguality", particularly in FAQ retrieval systems, has been under-addressed in the literature; although being crucial to organizations for faster scaling of operations to geographically distributed markets. In this work, we propose the *Decoupled Training Architecture for FAQ Retrieval* (DTAFA) framework geared towards *enhanced "one-best" accuracy to alleviate the above challenges in modern interactive application settings*.

Problem Statement. FAQ Retrieval engines attempt to understand the underlying *intent* of

Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 423–430
July 29–31, 2021. ©2021 Association for Computational Linguistics

user questions and retrieve the most related documents or answers that may contain correct information (Kothari et al., 2009). Formally, consider FAQ $= \{(Q_1, A_1), \cdots, (Q_n, A_n)\}$ to be a pre-curated collection (or repository) of question-answer pairs, where Q denotes a question related to the domain, and A represents the corresponding answer. Given a user query q, the task then is to return $\{(Q^q_1, A^q_1), \cdots, (Q^q_n, A^q_n)\}$, a ranking of (Q, A) pairs $\in$ FAQ; such that $\rho[q, (Q^q_i, A^q_i)] \geq \rho[q, (Q^q_j, A^q_j)] \mid \forall i \leq j$, where $\rho[q, (Q, A)]$ captures the relevance score (i.e., semantic and intent similarity) of the question-answer pair (Q, A) with respect to the query q. This work aims at developing an FAQ retrieval system that maximizes the accuracy at rank 1, i.e., the relevant (Q,A) pair to the query q should be represented by (Q^q_1, A^q_1).

Without loss of generality, we assume that each question Q_i in the FAQ collection is re-phrased into different possible lexico-syntactic variants, but conveying the same semantic meaning. For example, the question "`How to delete my account?`" can be reformulated as "`Process to close account?`" with the same intent. Let, Q'_i represent the set of re-phrased questions associated with $Q_i \in$ FAQ. In the remainder of the paper, we refer to the original question Q_i as "Questions (QU)", while its paraphrased formulations (Q'_i) are denoted as "Extended Questions (EQ)". Observe, that for a (Q_i, A_i) pair, both Q_i and Q'_i are mapped to the same answer A_i; and a small set of paraphrasings is constructed either manually or via automated systems (Kumar et al., 2019, 2020).

Related Work and Contributions *DTAFA* provides a novel learning framework for *Multilingual FAQ retrieval* with enhanced top-1 recommendation accuracy (or "one-best" accuracy), geared towards improving the overall quality of interactive automated customer experience. As shown in Figure 1(b), DTAFA leverages two "decoupled" deep learning architectures trained independently. The main fundamental intuition behind DTAFA is simple but yet found to be effective; to decrease the search space first via a simple classification module which does not take into account the semantics of the label and then aiming to select from the reduced search space the most semantic similar to the label context give the label has enough context.

Prior art focuses mainly in dealing with the FAQ retrieval problem as either text classification or semantic textual similarity problem. For text classification, we have seen set of large-scale Transformer-based Pre-trained Language Models (PLMs) such as (Devlin et al., 2019), RoBERTA (Liu et al., 2019), and XLM (Lample and Conneau, 2019). These PLMs are fine-tuned using task-specific labels and created new state of the art in many downstream natural language processing (NLP) tasks including FAQ Retrival Problems or more broadly text classification (Jiang et al., 2019). On the other side, there have been several prior work that relies in measuring semantic similarities for FAQ-based QA such as MatchPyramid (Pang et al., 2016), IWAN (Shen et al., 2017), and Pair2vec (Joshi et al., 2018) and more recently using Q-to-a matching using an unsupervised way, and further introducing a second unsupervised BERT model for Q-to-q matching (Santos et al., 2020).

However, adapting PLM text classification based approaches do not take label textual semantics into account which they have have some useful lexical information that can be used for improving the system accuracy. In addition, these architectures impacts the inference time when deployed in production due to the huge number of model parameters. Semantic Textual Similarity based methods usually do not scale when the number of FAQ pairs increases as there will be a need for performing matching to every pair to extract the corresponding answer. In that sense, we propose DTAFA with an aim to solve such challenges relying on two decoupled deep learning architectures trying to leverage the advantages of each of the above approaches in a hybrid approach yielding to more practical implementation. Our contributions, in a nutshell, are:
(i) We propose *DTAFA*, a novel framework for multi-lingual FAQ retrieval that captures lexical and semantic similarities and relationships among user queries, FAQ questions and their paraphrased versions to understand fine-grained differences to provide enhanced "one-best" accuracy;
(ii) We exhibit that DTAFA using two trained decoupled architectures achieves better accuracy for both monolingual and multi-lingual setup compared to existing techniques;
(iii) Empirically we observe DTAFA to require significantly less model parameters compared to existing deep learning architectures (e.g., PLMs like BERT, RoBERTa, etc.), an important factor for deployment in industrial settings having a direct impact on inference times;
(iv) DTAFA shows better results on *zero-shot learn-*

ing especially for distant languages.

2 *DTAFA* Framework

We next describe the detailed architecture and working of the different components in *DTAFA* shown in Figure 1(a). *DTAFA* hinges on two decoupled deep learning architecture based modules. The first module is trained to learn *latent lexical relationships* between the FAQ questions (QU) and their paraphrased variants (EQ) for generating candidate top-k most relevant or similar questions within the FAQ collections. The top-k candidates are then fed to the second module, a probabilistic Siamese LSTM-based architecture, to capture *fine-grained differences in semantic context* between the questions and their possible variants (proxies for real user queries) for further improving the accuracy of the final top-1 recommended result.

To support multi-linguality and zero-shot learning for scaling to other languages, both modules in DTAFA are based on LASER sentence embeddings (Artetxe and Schwenk, 2019) which are language-independent representations – similar sentences are mapped onto nearby vector spaces (in terms of cosine distance), regardless of the input language. However, instead of training using only one language and performing zero-shot learning on the others (the default setting (Pires et al., 2019)), we use three languages, namely English, Spanish and Chinese, for training across the components.

We present DTAFA in the context of FAQ retrieval, observe that it can easily be extended to other classification problems, where the textual labels contain enough semantic information.

2.1 EQ-EQ Classification Module

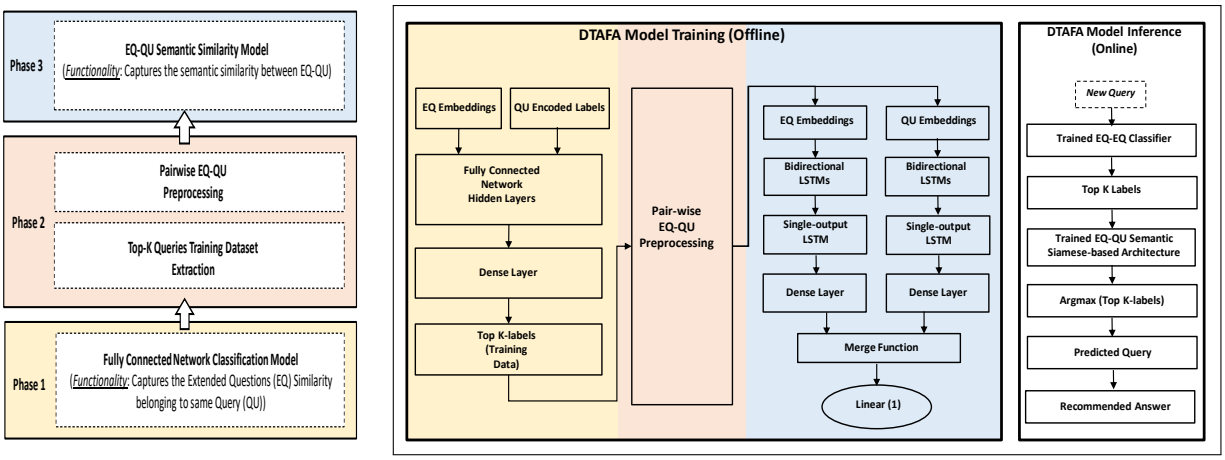
This module constitutes the Phase 1 of our *DTAFA* framework as shown in Figure 1(b) (Yellow part). This stage attempts to model the latent lexical and semantic similarities between the re-formulated extended questions (EQ) and the original questions in FAQ (QU). Intuitively, different paraphrased versions of a question capture the same underlying *intent* in diverse lexical formulations, providing our system with a generalized view as to how different users might express the same intent or query. Thus, in the first phase, DTAFA learns to map the extended questions to their corresponding original question, formulated as a classification task based on the semantic similarities between EQ and QU. Specifically, we trained a full connected neural network with two hidden layers with the extended questions (in embedded vector representation) as inputs and the original questions (encoded as class labels) as outputs.

The resulting input matrix $\mathbb{R}^{m \times n}$, where m is the number of samples in the dataset and $n = 1024$ is the vector length of LASER embeddings, is passed through a fully connected neural network with two hidden layers of 700 units each and an activation function of ReLU. The final layer employs a softmax activation function to output a classification probability corresponding to the different intent/question categories (QU labels), as annotated in the datasets. We use 0.5 as dropout, 32 batch size, 400 epochs, categorical cross-entropy loss function, ADAM as an optimizer. The full architecture has 1.5 million trainable parameters.

We also used a 0.5 dropout factor across all layers. The EQ-EQ classification module was trained for 400 epochs using a batch size of 32, the learning rate was reduced by a factor of 0.5 and a patience of 40 epochs for the validation loss was used. We considered sparse categorical cross-entropy as the loss function and ADAM as the model optimizer. The total number of trainable parameters was found to be nearly 1.5 million.

2.2 Pairwise EQ-QU Preprocessing Module

The above trained EQ-EQ classification model is next used by DTAFA to generate the top-k candidate intents or questions (QU) for the extended questions (EQ). The vector representations of the paraphrased questions, EQ, are again fed to the classifier trained in Phase 1, to obtain the top-k QU labels for each of the EQ, along with the classification probability score. For this phase, since the input to the model is, in fact, the exact data on which it had been used for training. However, the aim of this stage is to identify different classes of user questions (or intents) that are semantically very close. Intuitively, these top-k identified similar candidates contribute to the "confusion" for learning architectures. Thus, we aim to identify fine-grained difference among these categories using a Siamese Bidirectional LSTM-based architecture in Phase 3 of DTAFA (Figure 1(a)). Further, in our experimental evaluations presented later, we found this module to be useful as it acts as a label smoothing mechanism, preventing the model from over-fitting and consequently improving performance and generalizability across domains and



(a) (b)

Figure 1: Architectural Overview of DTAFA for (a) High Level Working – Phase 1: QU Classification Model; Phase 2: Data Preparation; Phase 3: EQ-QU Siamese Based Network Architecture, and (b) Model Training and Inference.

languages.

Formally, for each extended question EQ_i (in the training dataset), DTAFA generates $\mathcal{Q}^i$, the set of top-k queries (QU) returned by the EQ-EQ classifier as possible matching candidate questions (or intents). Let $\mathcal{P}^i$ represent the classification probabilities associated with the candidate questions, $\mathcal{Q}^i$. Thus, for each EQ_i, we construct a set of k 3-tuples, $\mathcal{T} = \{\langle EQ_i, \mathcal{Q}^i_j, \mathcal{P}^i_j \rangle\}$ $(j \in [1, k])$, where $\mathcal{Q}^i_j$ is the j^{th} element in $\mathcal{Q}^i$ and its associated classification probability is given by $\mathcal{P}^i_j$.

In other words, the 3-tuple $\langle EQ_i, \mathcal{Q}^i_j, \mathcal{P}^i_j \rangle$ represents that the question $\mathcal{Q}^i_j$ in the FAQ collection (QU) was identified by the EQ-EQ classifier as a possible matching candidate (for the extended question EQ_i) with a classification score of $\mathcal{P}^i_j$. The set of 3-tuples, $\mathcal{T}$ for all the pairwise EQ-QU candidates extracted from the FAQ collection is constructed and forms the input to the next stage.

2.3 EQ-QU Semantic Similarity Module

The final phase of DTAFA consists of a Siamese-network based architecture with Long Short-Term Memory (LSTM) to assess semantic similarities and learn fine-grained differences among the above identified candidates. Hence for a candidate 3-tuple, $\langle EQ_i, \mathcal{Q}^i_j, \mathcal{P}^i_j \rangle \in \mathcal{T}$, the vector representation (using LASER) of EQ-QU question pair $(EQ_i, \mathcal{Q}^i_j)$ is given as input and the network is trained as a regression model with the associated probability score $\mathcal{P}^i_j$ treated as output.

As shown in Figure 1(b) (blue part), the Siamese network comprises two branches, each with a masking layer followed by Bidirectional-LSTM layers. Incorporating the intermediate representations across the branches enables increased context flow between them, positively impacting the overall parameter updation process. We further employ some multiplication and subtraction layers between the outputs of the branches from the BiLSTM layers to capture more variations between the paired sentences, intuitively "fine-tuning" the semantic similarity captured by the pretrained language model. We found such intermediate layers before the concatenation layer to help avoid the gradient vanishing problem by allowing more gradient to flow. Finally, a concatenation layer followed by one hidden layer with ReLU activation function was employed. The output layer consists of a linear activation function on the concatenated representation for the regression based prediction task; concluding the training setup.

2.4 Inference Module

Given a new user query q, the DTAFA framework retrieves the most relevant answer (to q) from the FAQ collection, based on the trained architecture as described above. The inference module (the online interactive component) follows a similar flow to that of the training process as shown in Figure 1(b). The user query q is initially represented in a high-dimensional vector space using multilingual LASER embeddings, and is subsequently fed to the pre-trained EQ-EQ classification module, which extracts the top-k best matching questions (QU) from the FAQ repository along with their classification scores. The query q, the candidate similar questions identified, along with their classification scores are used to generate the list of 3-tuples as described in Section 2.3. The 3-tuples are fed to the pre-trained EQ-QU similarity module, and the candidate question with the highest output score

is considered as the best matching and most relevant FAQ to the user concern. The corresponding answer to the matched question (from the FAQ) is then returned to the user. The overall architecture of *DTAFA* is presented in Figure 1(b).

3 Experimental Setup

In this section, we describe the experimental setup for comparing the performance of DTAFA against state-of-the-art approaches. We consider the "one-best" accuracy, measured in terms of *Precision-at-Rank-1*. All models trained using NVIDIA Titan RTX GPU.

3.1 Dataset

We validate our framework using the following datasets: ***(a) Enterprise Dataset***: A real-life enterprise data containing customer queries in 13 different languages related to mobile services. Our dataset comprises 336 unique queries (QU) representing different user intents. Each of the queries have subsequently been paraphrased, by human annotators, to an average of 15 different formulations to form the extended questions (EQ). It is worth noting that the dataset is anonymized and all identifiers have been irreversibly removed and data subjects are no longer identifiable in any way. ***(b) StackExchange FAQ Dataset***: We processed the data[1] by labeling each class with a random picked question belong to such class so we include more semantics in the label. We have machine translated the English data to the other 12 languages to test with same languages to the Enterprise dataset.

3.2 Baselines

We benchmark the performance of DTAFA against the following baselines, spanning across context-free and contextualized language model embeddings based similarities, as well as other learning approaches geared towards understanding textual semantic similarities. We also consider multilingual settings and different variants of DTAFA for ablation studies. We construct our baselines having (A) monolingual setup using English only and (B) multi-lingual setup with zero-shot learning as described next.

A. Monolingual Baselines: In this setting, we evaluate the performance of DTAFA when trained and evaluated using only one language, English, using pre-trained language models. We categorize the competing approaches into three types:

• ***Context-free language models*:** A FCN with 3 hidden layers of 700 units each, ReLU activation functions, cross-entropy loss and softmax output function. Epochs are set to 150 and batch size to 32. We consider the following embeddings: ***TF-IDF*** (Jing et al., 2002), ***Word2Vec*** (Mikolov et al., 2013), ***GloVe*** (Pennington et al., 2014), and ***FastText*** (Bojanowski et al., 2017).

• ***Contextualized language models*:** We fine-tuned pretrained contextualized language models architectures with two added feed-forward layers and a softmax normalization to predict the QU by framing the FAQ retrieval problem as a classification problem. We adapted the following pretrained architectures: ***ULMFiT*** (Howard and Ruder, 2018), ***Flair*** (Akbik et al., 2018), ***ELMo*** (Peters et al., 2018), ***BERT*** (Devlin et al., 2019), ***XLM*** (Lample and Conneau, 2019), ***XLNet*** (Yang et al., 2019), and ***RoBERTA*** (Liu et al., 2019).

• ***Semantic-based Similarity Architectures*:** The objectives of these architectures is to train models to learn the pairwise EQ-QU (described in Section 2.3) semantic similarity, and the most similar QU to a user query (or test set) is extracted. We used the following two baselines as they were found quite standard and proved across various NLP tasks; ***SBERT*** (Reimers and Gurevych, 2019) and ***MaLSTM*** (Mueller and Thyagarajan, 2016).

B. Multilingual Baselines: This baseline setup explores the possibility of using a single language model pre-trained on the concatenation of corpora comprising different languages, i.e., the performance of possible "zero-shot cross-lingual transfer learning" for FAQ retrieval systems. Such frameworks are of prime interest in enterprise settings, given the dual advantages of (i) enable enterprises to easily expand their consumer outreach globally by supporting a larger set of languages, and (ii) faster launch cycles with zero-shot learning eliminating the need for annotated training data for each language. We use ***M-BERT*** (Pires et al., 2019) as a solid baseline for comparing DTAFA in the multlingial context in which we fine-tune the whole architecture using three languages of English, Spanish, and Chinese to make it fairly comparable to DTAFA-ML discussed next.

C. DTAFA Variations: We also perform ablation tests across different variations of DTAFA architecture to study the impact of different com-

[1] obtained from www.takelab.fer.hr/data/StackFAQ/

Table 1: P@1 Results on Monolingual dataset (using English only).

Models Category	Approach	Ent. Data	Stk. Data
Semantic-based Similarity models	MaLSTM	61.98	83.29
	SBERT	62.87	83.21
Context-free language models	TF-IDF	66.25	82.21
	Word2Vec	66.76	83.99
	GloVe	66.79	83.43
	FastText	66.93	84.92
Contextualized language models	ULMFiT	67.67	85.34
	Flair	66.68	86.01
	ELMo	67.70	88.92
	XLNet	68.71	90.01
	XLM	67.72	90.33
	BERT	71.71	93.45
	RoBERTa	72.82	94.91
DTAFA Variations	DTAFA-C1	67.63	85.66
	DTAFA-C2	63.46	87.31
	DTAFA-EN	**73.87**	**95.89**

ponents of our framework.

DTAFA-ML – full multi-lingual DTAFA architecture as described in Section 2.

DTAFA-EN – full proposed architecture trained only on English and tested on multi-lingual data to assess zero-shot capabilities compared to using 3 languages in training.

DTAFA-C{X} – the individual DTAFA architectural components performance are studied – DTAFA-C1 refers to the *EQ-EQ Classification Module* alone, while DTAFA-C2 refers to the *EQ-QU Semantic Similarity Module* only.

4 Empirical Results

This section reports the empirical results obtained for DTAFA (both monolingual and multi-lingual settings) as compared to the competing approaches described previously. To capture "one-best"' accuracy, we report the *Precision-at-Rank-1* (P@1) performance, which captures the fraction of the top-1 answer retrieved by the system that are relevant to the user query. This indirectly captures the quality-of-service for speech-based assistive platforms. DTAFA is currently in pre-deployment phase in our organization.

4.0.1 Monolingual Results

The performance results obtained in the monolingual setting (i.e., training and testing both using English only) for the competing algorithms are presented in Table 1. We observe the *Semantic-based Similarity* approaches (i.e., MaLSTM and SBERT) to perform the worst on the Enterprise Dataset. This can be attributed to the specific nature of our dataset – containing a large number of categories (336 classes) compared to the StackExchange dataset.

Among the *context-free language models*, TF-IDF attained the worst accuracy on both datasets, Word2Vec and GloVe showed similar performances with FastText being marginally better than GloVe with $\sim$ 0.12% improvement for the Enterprise dataset and $\sim$ 1% improvement for the StackExchange dataset. These results follow the natural evolution of the techniques to better learn the occurrence context of words for better representations.

RoBERTa outperforms other *contextualized language model* techniques, and being a fine-tuned version of BERT architecture, marginally outperformed BERT with $\sim$ 1% improvement. The proposed *DTAFA-EN* framework was seen to outperform all the competing baselines, achieving $\sim 73.87\%$ and $\sim 95.89\%$ accuracy as compared to the best result for existing approaches (72.82% and 94.91% obtained by RoBERTa) for the enterprise and StackExchange datsets respectively.

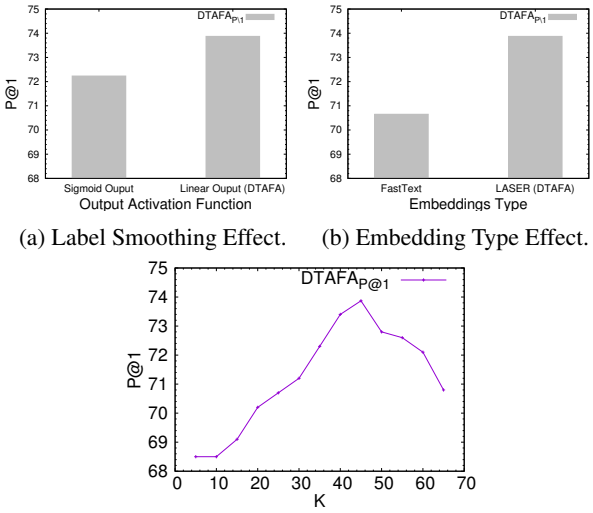
We observe nearly $\sim$ 1% performance improvement over state-of-the-art baselines for monolingual setting. However, DTAFA enjoys a major advantage in terms of *model complexity*, requiring only $4.2M$ trainable parameters compared to $125M$ parameters in RoBERTa giving more advantage to DTAFA to be deployed in practice. The $\mathbf{30}\times$ lesser number of parameters play a crucial role in (i) training time, (ii) amount of annotated training data necessary, and (iii) inference time – vital factors for development, deployment, and scalability for enterprises.

4.0.2 Multi-lingual Results

From Table 2, we observe that DTAFA-ML provides substantial performance improvement (based on zero-shot learning), outperforming M-BERT on all languages with an average gain of $\sim$ $\mathbf{30}\%$ for the Enterprise Data and $\sim$ $\mathbf{40}\%$ on StackExchange Data. We can clearly notice that training using the 3 languages (DTAFA-ML) compared to using English only (DTAFA-EN) brought an additional boost in the performance not only on the trained used languages (English, Chinese, Spanish) but more significantly on the zero-shot tested languages with an average boost in performance of $\sim$ $\mathbf{7}\%$ on the rest of the 10 languages for the Enterprise Dataset and almost $\sim$ $\mathbf{9}\%$ for the StackExchange Dataset. We believe from the results that training using more than one language to boost the performance on other languages using zero-

Table 2: "Zero-shot" Multilingual Results with English, Chinese & Spanish for training.

Datasets	Approach	Languages Tested (P@1 (%))												
		English	Chinese	Spanish	Italian	French	Portuguese	German	Catalan	Romanian	Russian	Japanese	Turkish	Arabic
Enter. Dataset	M-BERT	71.61	79.59	71.21	54.10	51.23	50.94	40.21	52.55	35.15	30.22	30.51	18.26	15.64
	DTAFA-EN	73.87	68.19	62.09	60.28	62.98	63.88	60.09	64.87	62.87	56.87	55.78	53.98	60.76
	DTAFA-ML	**74.12**	**78.26**	**72.43**	**69.63**	**70.51**	**69.46**	**67.42**	**69.22**	**68.41**	**65.41**	**63.48**	**61.32**	**66.42**
StackE. Dataset	M-BERT	92.44	91.53	91.92	48.24	49.12	47.32	43.21	50.21	42.12	28.12	29.10	15.19	14.87
	DTAFA-EN	95.89	72.45	75.12	73.18	72.90	70.57	68.87	70.80	72.98	76.69	72.78	70.11	67.69
	DTAFA-ML	**97.32**	**96.12**	**96.82**	**90.12**	**89.30**	**91.28**	**87.78**	**94.34**	**92.10**	**87.79**	**86.76**	**85.48**	**69.35**



(a) Label Smoothing Effect. (b) Embedding Type Effect.

(c) Impact of Top-k on training EQ-QU Similarity module.

Figure 2: DTAFA Finetuning Parameters.

shot should become the norm to scale to more languages with more reliable performance. To the best of our knowledge, this was not enough discussed nor experimented in the literature. From our experiments, we found that choosing the languages to train DTAFA depends on the languages we want to achieve best performance when applying zero-shot. For instance, we found that choosing Spanish as one of the languages used in training allowed us to achieve better performance when applying zero-shot to languages such as Portuguese, Catalan, and Romanian. Interestingly, as an example, the performance on Arabic improved in this case, even with less points, due to lesser lexical and semantic gap between the trained languages and Arabic. Based on this, we believe that choosing the training languages in DTAFA should be use-case dependent.

4.0.3 DTAFA Parameters Impact

Finally, we discuss the empirically guided parameter setting for DTAFA used in the above evaluations. We show such evaluation on the Enterprise dataset as we found the same intuition is applicable on the StackFAQ dataset. Compared to the traditional approach of using binary outputs with *Sigmoid function*, we gain $\sim 1.5\%$ in performance by using *linear activation function* as shown in Figure 2a– possibly due to some "label smoothing" for the output layer. We replaced the input embeddings in the EQ-EQ Classification Module from LASER to FastText. However, LASER was seen to obtain $\sim 1.5\%$ better performance compared to FastText, as shown in Figure 2b. EQ-QU Semantic Similarity module in DTAFA generates the top-k best matched QU candidates for each question in EQ during training. Figure 2c illustrates the impact of varying the value of k. We observe that as k increases, the overall performance of DTAFA improves until $k = 45$. Further increase in the value of k was found to degrade the efficacy of our framework, as large values of k potentially results in dissimilar samples with low classification score also being considered as potential candidates. We set $k = 45$ for training DTAFA.

5 Conclusion

We propose a novel multi-lingual FAQ retrieval framework (*DTAFA*) for improving the accuracy of top-1 results ("one-best" performance). Our framework combines the advantages of both classification and semantic textual similarity approaches in one single framework and hence, improves FAQ retrieval problem accuracy while keeping number of parameters less compared to other state-of-the-art approaches making it more practical approach in an industrial context. Experiments on real enterprise data as well as open source dataset across 13 languages demonstrate the efficacy of our system over existing tradional approaches, both in monolingual and multi-lingual settings. We show DTAFA to robustly generalize to multiple languages based on "zero-shot" transfer learning, providing upto 40% accuracy improvement on distant languages along with $30\times$ lesser number of trainable model parameters.

References

A. Akbik, D. Blythe, and R. Vollgraf. 2018. Contextual String Embeddings for Sequence Labeling. In *COLING*, pages 1638–1649.

M. Artetxe and H. Schwenk. 2019. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. 2019. BERT: Pre-training of Deep Bidir. Transformers for Lang. Understanding. In *NAACL-HLT*, pages 4171–4186.

J. Howard and S. Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *Annual Meeting of the Association for Computational Linguistics*, pages 328–339.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 2019. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. *arXiv preprint arXiv:1911.03437*.

Li-Ping Jing, Hou-Kuan Huang, and Hong-Bo Shi. 2002. Improved feature selection approach tfidf in text mining. In *Proceedings. International Conference on Machine Learning and Cybernetics*, volume 2, pages 944–946. IEEE.

Mandar Joshi, Eunsol Choi, Omer Levy, Daniel S Weld, and Luke Zettlemoyer. 2018. pair2vec: Compositional word-pair embeddings for cross-sentence inference. *arXiv preprint arXiv:1810.08854*.

M. Karan and J. Šnajder. 2018. Paraphrase-focused Learning to Rank for Domain-specific FAQ Retrieval. *Expert Systems With Applications*, 91:418–433.

G. Kothari, S. Negi, T. A. Faruquie, V. T. Chakaravarthy, and L. V. Subramaniam. 2009. SMS Based Interface for FAQ Retrieval. In *ACL-IJCNLP*, pages 852–860.

A. Kumar, K. Ahuja, R. Vadapalli, and P. Talukdar. 2020. Syntax-Guided Controlled Generation of Paraphrases. In *ACL*.

A. Kumar, S. Bhattamishra, M. Bhandari, and P. Talukdar. 2019. Submodular Optimization-based Diverse Paraphrasing and its Effectiveness in Data Augmentation. In *NAACL*, pages 3609–3619.

G. Lample and A. Conneau. 2019. Cross-lingual Language Model Pretraining. In *Advances in Neural Information Processing Systems (NIPS)*.

J. T. Lee, S. B. Kim, Y. I. Song, and H. C. Rim. 2008. Bridging Lexical Gaps between Queries and Questions on Large Online Q&A Collections with Compact Translation Models. In *EMNLP*, pages 410–418.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.

A. Lommatzsch and J. Katins. 2019. An Information Retrieval-based Approach for Building Intuitive Chatbots for Large Knowledge Bases. In *LWDA*, pages 343–352.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed Represt. of Words and Phrases and their Compositionality. In *NIPS*, pages 3111–3119.

J. Mueller and A. Thyagarajan. 2016. Siamese Recurrent Architectures for Learning Sentence Similarity. In *AAAI*, pages 2786–2792.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

J. Pennington, R. Socher, and C. D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

T. Pires, E. Schlinger, and D. Garrett. 2019. How multilingual is Multilingual BERT? arxiv.org/abs/1906.01502.

N. Reimers and I. Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*, pages 3982–3992.

José Santos, Ana Alves, and Hugo Gonçalo Oliveira. 2020. Leveraging on semantic textual similarity for developing a portuguese dialogue system. In *International Conference on Computational Processing of the Portuguese Language*, pages 131–142. Springer.

Gehui Shen, Yunlun Yang, and Zhi-Hong Deng. 2017. Inter-weighted alignment network for sentence pair modeling. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 1179–1189.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *NIPS*, pages 5753–5763.