Improve DistilBERT-based Question Answering model performance on out-of-domain datasets by Mixing Right Experts

Stanford CS224N Default Project: Robust QA track

Kaikai Sheng
Department of Computer Science
Stanford University
kaikais@stanford.edu

Abstract

While large language representation model like BERT [1] shows its great potential to improve the performance of Question Answering (QA) system, such QA systems fail to extend its performance beyond in-domain datasets. This project aims to improve the performance of DistilBERT-based QA model trained on in-domain datasets in out-of-domain datasets by only using provided datasets. We did extensive analysis cross in-domain and out-of-domain datasets to build essential insights, e.g. domain-finetuning may cause model performance to degenerate. By using such insights to mix expert models that are carefully chosen, we achieved **F1** score of **61.7** (ranked 6th out of 74 in test leaderboard) and **EM** score of **44.4** (ranked 2nd out of 74 in test leaderboard) in out-of-domain test datasets as of March 19, 2021.

1 Introduction

Large pre-trained language representation model, e.g. BERT [1], serves a nice base for a variety of downstream natural language understanding tasks to fine-tune upon. One of such example is QA systems, aka. reading compression (RC). In a typical QA task, the model takes a span of context and one question as inputs and predicts the start and end position of the answer in the context. To fine-tune a pre-trained model for QA task, we simply add a adapter-like layer in front of BERT outputs to adapts them into two outputs, one representing the logits of start position of answer span in context and another representing the logits of end position.

The primary purpose of such fine-tuning is to adapt a BERT-based model to perform on QA tasks. As a side effect, QA ability of such models is restrained on the domain it fine-tunes on and fails to generalize other domains that has different distribution from fine-tuning domains as shown in previous work [2] [3] [4].

In this work, we mainly explores Mixture-of-Experts (MOE) technique to improve QA task performance (measuing by **F1** score) of DistilBERT-based model on out-of-domain datasets by only using given datasets. DistilBERT [5] is a small-sized BERT model but have comparable performance. We use DistilBERT as our base model since its small size allows to load multiple models into GPU memory. To enable DistilBERT-based QA model to perform on out-of-domain QA tasks, we first fine-tuned it in a variety of combinations of three large in-domain QA datasets (we will call it task-fine-tuning stage for the rest of context since its primary purpose is to adapt DistilBERT to QA task). We then continue fine-tune such models in different small out-of-domain datasets (we will call it domain-fine-tuning stage for the rest of context since its primary purpose is to adapt task-fine-tuned model to perform out-of-domain QA tasks). We finally evaluate models against out-of-domain validation datasets for development purposes and also evaluates them against test datasets, results of which are submitted in the test leaderboard.

The main contributions of this paper is as follows

- We did extensive analysis across datasets to build data insights about domain correlations. The most important one is that task-fine-tuning a domain-fine-tuned a model on small out-of-domain datasets may reduce its out-of-domain performance, i.e. task-fine-tuning does not necessarily translate into better out-of-domain performance.
- We experimented with different hyperparameters, i.e. learning rate (LR), number of epochs but found they did little help to improve out-of-domain performance.
- By leveraging data insights above, we mixed right task-fine-tuned expert models together to achieve a gain of 6 **F1** score compared to the baseline on out-of-domain validation dataset and achieve **F1** score of **61.7** (ranked 6th out of 74 in test leaderboard) and **EM** score of **44.4** (ranked 2nd out of 74 in test leaderboard) in out-of-domain test datasets.

2 Related Work

We will briefly review previous related work to give readers a bigger context.

2.1 Question Answering Models

In a QA task, a model will take a paragraph, i.e. a span of context and a question as inputs and predicts the answer span in the input paragraph by outputting start and end position of the answer. In pre-BERT era, Bi-Directional Attention Flow (BiDAF) [6] achieved a **EM/F1** score of **68.0/77.3** on SQuAD v1.1 dataset by building on top of serveral bidirectional LSTMs.

In post-BERT era, many QA models are built based on BERT or variant of BERT. BERT-based models are built on top of Transformers [7]. The original BERT paper [1] achieves **EM/F1** score of **80.8/88.5** for small-sized model and **EM/F1** score of **84.1/90.9** for large-sized model on SQuAD v1.1 dataset. SpanBERT [8], a variant of BERT model, achieves a **F1** score of **94.6** on SQuAD v1.1 dataset. The baseline model we use in this work is DistilBERT [5], which achieves **EM/F1** score of **77.7/85.8** on SQuAD v1.1 dataset, which is slightly lower than original small-sized BERT but reduces its size by 40%.

2.2 QA System in domain shifted QA tasks

Previous work [2] [3] [4] shows that QA models that are well-task-finetuned perform much worse in other QA domains. As shown in [3] the difference of **F1** and **EM** score between in-domain datasets and out-domain datasets can be as large as 20. Many solution are proposed to improve QA models in domain shifted QA tasks.

Data augmentation is an effective technique to preserve the invariances of task-fine-tuned QA model by generating more data points only based on existing training data and unlabeled text. Zhang et al. proposed [9] to replace a word with its synonym. Wang et al. proposed [10] to find nearest word in word embedding space to replace the original word instead of using its synonym. Wei et al. [11] brings it to a higher level by introducing a NLP package Easy Data Augmentation (EDA) that consists of four text editing operations including replacement. Instead of performing operations on words, Kafle et al. [12] proposed to generate additional text by using recurrent neural networks. Following the similar spirit, Yu et al. [13] proposed to use back-translation to generate paraphrases instead of add, removing or replacing simple words.

In this paper, we mainly focus on applying Mixture-of-Experts (MOE) technique to improve out-of-domain performance. First introduced in [14], MOE add one more learn-able layer on top of a diverse set of expert models. During domain-fine-tuning stage, this layer learns the weights to combine outputs from expert models to produce final output. MOE differs from ensemble in that weights assigned to each expert model is learned by training on the small out-of-domain dataset while weights in ensemble is assigned by humans.

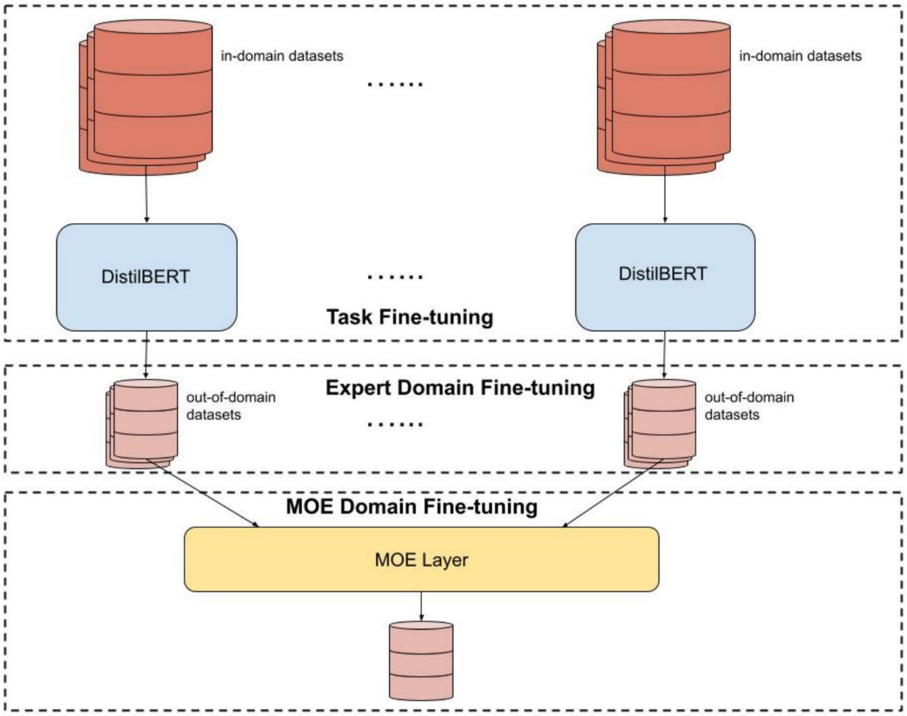

Figure 1: The QA model training diagram for a out-of-domain dataset. We task-fine-tuned 7 DistilBERT-based QA models on different combinations of in-domain datasets. We then domain-fine-tuned on different combinations of out-of-domain datasets for each expert model. We fine-tuned all expert models with a MOE layer on the target out-of-domain dataset to produce the final model.

3 Approach

In this section, we will talk about how we trained our models to perform out-of-domain datasets and why we did such. We will also talk about the baseline model we uses for training and performance metrics we use.

3.1 DistilBERT QA model

We use pretrained DistilBERT QA model as our baseline as a requirement of final project instructions [15]. There are other benefits to use DistilBERT instead of original BERT

- DistilBERT is 60% faster than original BERT model while retaining 97% of it language understanding capabilities as described in [5]. We can finishing task-fine-tuning in three large in-domain datasets within 4 hours on a server equipped with one NVIDIA Tesla V100 GPU. It is a good variant for us to explore BERT model in QA task.
- DistilBERT is 40% smaller in size than original BERT. It is possible to load multiple task-fine-tuned models into GPU memory (our NVIDIA Tesla V100 GPU has $16GB$ memory) to fine-tune a MOE model.

3.2 MOE model

We proposed to apply data augmentation technique to improve model out-of-domain performance in our project proposal. However, after reading best poster [16] of 2019 and best report [17] of 2020, we realized that MOE technique has more headroom to gain than data augmentation. Since the goal of project is to train a model that performs best in out-of-domain test dataset, we take a different route than we first proposed.

To make MOE model work, it is crucial to find a set of right expert to put together. We built QA model training pipeline as shown in Figure 1 to explore different settings of experts. We trained a separate MOE model for each out-of-domain dataset since out-of-domain datasets have different distributions from each other.

Step 1 We task-fine-tuned 7 DistilBERT-based QA models to mix with. We trained each expert model on one combination of large in-domain datasets. We will talk about how we choose combination of in-domain datasets to feed into model in later sections. We set the number of expert models to be 7 since this setting performs best as shown in previous CS224n report [16]. QA models are trained in two dimensions, one by using different seed value and the other by fine-tuning on different combination of in-domain datasets. We will elaborate on this more in later sections.

Step 2 We domain-fine-tuned each expert model on one combination of small out-of-domain datasets. We will talk about how we choose combination of out-of-domain datasets in later sections.

Step 3 We put 7 domain-fine-tuned expert models together with a MOE layer to fine-tune a MOE on target out-of-domain dataset to produce the final MOE model.

We also explored different hyper-parameters, mainly leaning rate (LR) and number of epochs for both task-fine-tuning and domain-fine-tuning. We did not explore batch size too much since larger batch size may cause GPU out-of-memory error.

4 Experiments

4.1 Data

Pretrained DistilBERT models are task-fine-tuned in a variety of combinations of three large in-domain QA datasets, i.e. SQuAD [18], NewsQA [19], Natural Questions [20] (we will call it NatQA for short), each of which has 50000 training data points [15]. As shown in Figure 1, task-fine-tuned models are further domain-fine-tuned in three small out-of-domain datasets, i.e. DuoRC [21], RACE [22] , RelationExtraction [23] (we will call it REQA for short), each of which has only 127 training data points.

Since out-of-domain datasets are very small, we create larger out-of-domain training datasets by merging training datasets and validation datasets together after discussing with CS224n TAs. We call such train datasets as DuoRC_merge, RelationExtraction_merge and RACE_merge for the rest of context.

4.2 Evaluation method

We use the two standard metrics of Exact Match (EM) and F1 Score (F1) to measure QA model performance. **EM** is a binary measure of whether predicted answer span matches the exact human-provided groudtruth answer span. **F1** is the harmonic mean of precision and recall of the predicted answer span. During evaluation or test stage, the final EM and F1 is the maximum ones after evaluating predicted answer span against three human-provided groudtruth answer spans.

4.3 Experimental details

Throughout the entire section, we use a default hyper-parameter setting of $num\_epochs = 3$, $batch\_size = 16$, $seed = 42$, $lr = 3e - 5$ and train on a server equipped with NVIDIA Tesla V100 GPU unless otherwise specified.

4.3.1 Correlations across In-domains and Out-of-domains

We first studied how the models **only** task-finetuned on different combinations of in-domain datasets perform on out-of-domain datasets as shown in table 1 and how models both task-finetuned on different combinations of in-domain datasets and domain-finetuned on target out-of-domain dataset as shown in table 2. The purpose of the study is

- Understand the performance gap between in-domain validation datasets and out-of-domain validation dataset.

- Understand the impact of in-domain train datasets on out-of-domain performance.
- Get a sense on how out-of-domain performance varies across different out-of-domain datasets.
- Get a sense on how much out-of-domain performance improved after training on the target out-of-domain datasets.

When reading two tables together, we have a some important findings.

- Domain-fine-tuning on target out-of-domain dataset may reduce task-fine-tuned model out-of-domain performance, e.g. row "SQuAD,NatQA,NewsQA" column "DuoRC", "RACE" in table 1 and table 2.
- Only REQA dataset benefits from domain-fine-tuning, especially for models task-fine-tuned on SQuAD,NatQA datasets.

We think READ benefit from domain-fine-tuning because READ dataset has similar distribution as SQuAD and NatQA since data source of all three datasets are from Wikipedia. However, if domain-fine-tuning on a small dataset that has quite different distribution than in-domain training dataset, the model get consufued and it will hurt its performance.

In-domain Train	**In-domain Val**		**RACE**		**REQA**		**DuoRC**		**Overall**	
	F1	**EM**	**F1**	**EM**	**F1**	**EM**	**F1**	**EM**	**F1**	**EM**
SQuAD	76.0	61.4	29.2	14.8	63.0	39.8	31.9	24.6	41.6	26.6
NewsQA	55.4	37.9	35.4	21.9	52.1	34.4	40.7	29.4	43.0	28.7
NatQA	66.4	50.2	23.3	12.5	63.1	36.0	28.8	17.5	38.6	22.1
SQuAD,NatQA	72.3	56.6	30.9	14.8	66.9	42.2	33.8	27.0	44.1	28.1
SQuAD,NewsQA	71.0	55.3	32.1	18.8	65.1	39.8	37.8	30.2	45.2	29.7
NatQA,NewsQA	64.4	47.8	33.6	21.1	67.0	40.6	36.0	29.4	45.8	30.5
SQuAD,NatQA,NewsQA	70.3	54.3	35.7	21.9	66.1	39.1	43.3	34.1	48.6	31.9

Table 1: Comparison of out-of-domain performance for models **only** task-fine-tuned in different combinations of in-domain datasets

In-domain Train	**RACE**		**REQA**		**DuoRC**		**Overall**	
	F1	**EM**	**F1**	**EM**	**F1**	**EM**	**F1**	**EM**
SQuAD	28.3	14.8	71.1	52.3	28.3	22.2	42.7	30.0
NewsQA	30.5	15.6	69.8	48.4	31.4	23.8	44.1	29.4
NatQA	22.7	12.5	73.5	53.9	33.9	24.6	43.6	30.5
SQuAD,NatQA	30.2	15.6	76.3	55.5	34.9	25.4	47.4	32.3
SQuAD,NewsQA	33.4	19.5	72.7	53.1	32.0	23.0	46.3	32.1
NatQA,NewsQA	26.3	14.8	71.6	51.6	39.3	29.4	46.0	32.1
SQuAD,NatQA,NewsQA	31.8	16.4	72.4	52.3	36.9	27.0	47.3	32.1

Table 2: Comparison of out-of-domain performance for models task-fine-tuned in different combinations of in-domain datasets and domain-fine-tuned in target out-of-domain dataset

4.3.2 Out-of-domains Performance Stability

We also studied the out-of-domain performance stability. We changed the training seed in task-fine-tuning stage and tested model performance on all three out-of-domain datasets. The results are shown in table 3. By doing this experiment, we can

- Get a sense on how stable the model perform on each out-of-domain datasets.
- Get a of set of expert models training on different seeds.

As we can see from the table 3, out-of-domain performance of models vary a lot for RACE and DuoRC but not for REQA. This is may be due to the face that REQA has similar distribution as in-domain datasets since data source of SQuAD, NatQA and REQA is Wikipedia.

Seed	In-domain Val		RACE		REQA		DuoRC		Overall	
	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM
42(default)	70.3	54.3	35.7	21.9	66.1	39.1	43.3	34.1	48.6	31.9
12	70.6	54.7	34.5	21.9	69.7	48.4	38.3	29.4	47.7	33.4
22	69.6	53.8	41.0	26.6	67.7	40.6	40.7	34.9	50.1	34.2
32	70.3	54.1	31.2	16.4	68.0	43.8	40.9	31.8	46.9	30.8
52	70.9	54.9	33.7	19.5	68.2	45.3	41.1	35.7	47.9	30.8
62	70.2	54.2	37.3	23.4	66.7	46.1	42.5	34.1	49.0	34.7
72	70.4	54.8	35.2	23.4	69.8	43.8	41.6	33.3	49.1	33.7

Table 3: The stability of out-of-domain performance for models **only** task-fine-tuned on SQuAD, NewsQA and NatQA

4.3.3 Combination of Out-of-domain Training Datasets

We also studied on how model task-fine-tuned on the same in-domain datasets perform different if domain-fine-tuned in different combination of out-of-domain datasets as shown in table 4. Surprisingly, we found out that model performs best if finetuned only on REQA dataset regardless of their target out-of-domain dataset. But we think that such surprising results may be due to the fact that both out-of-domain training datasets and validation datasets are too small to draw any conclusions. So we did not leverage such findings when building MOE model.

4.3.4 Hyper-parameter tuning

We did hyper-parameter tuning for both task-fine-tuning and domain-fine-tuning.

We only tune number of epochs for task-fine-tuning since

- Task-fine-tuning is time-consuming and expensive. It takes 4 hours to finish one run and cost about $\$20$.
- One of observation in original BERT paper [1] is that large data sets are far less sensitive to hyperparameter choice than small datasets. Our experiments agree with this observation as shown in Figure 2.

In Figure 2, we task-fine-tuned model using different seed and different number of epochs. By using different seed, we can observe the stability of model, i.e. how resistant the model is to randomness. We task-fine-tuned the model with $num\_epochs = 1, 2, 3, 5, 7, 10$ and $seed = 30, 42, 50, 60, 70$. As we can see from the figure, the model performance did not change too much if number of epochs is larger than 2. Nevertheless, it looks like 3 is a sweet spot for number of epochs in terms of both **F1** score and stability.

We also tune hyper-parameter for domain-fine-tuning. We did find out the model performance is sensitive to hyperparameter, mostly likely due to out-of-domain train dataset size.

Out-of-domain Train	RACE		REQA		DuoRC	
	F1	EM	F1	EM	F1	EM
RACE	31.8	16.4	64.2	38.0	32.9	25.1
REQA	33.0	19.2	72.4	52.3	37.9	31.9
DuoRC	30.8	15.3	64.0	37.0	36.9	27.0
RACE,REQA	31.4	16.0	70.8	49.5	32.9	25.0
RE,DuoRC	30.4	16.0	71.9	50.5	36.5	26.9
RACE,DuoRC	31.9	16.5	63.6	36.5	35.4	25.5
RACE,REQA,DuoRC	31.7	16.7	71.9	49.4	34.6	25.2

Table 4: Comparison of out-of-domain performance for models domain-fine-tuned on SQuAD, NewsQA and NatQA but task-fine-tuned in different combinations of out-of-domain datasets

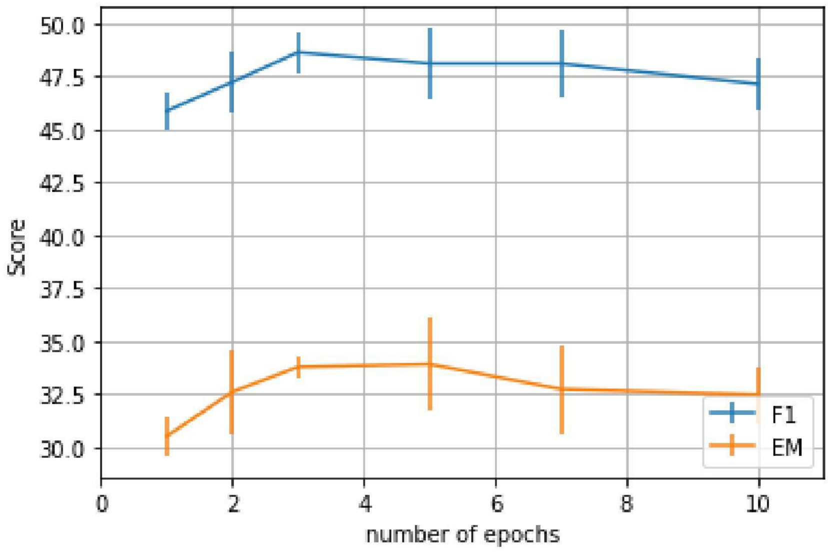

Figure 2: The **F1** and **EM** score of model trained using different number of epochs. The horizon error line shows the maximum and minimum score for different seed.

RACE		REQA		DuoRC		Overall	Test
MOE	**F1/EM**	**MOE**	**F1/EM**	**MOE**	**F1/EM**	**F1/EM**	**F1/EM**
7*B	39.8/25.8	7*{B,REQA}	76.3/57.8	7*B	45.0/37.3	53.1/40.3	**61.7**/44.4
7*B	39.8/25.8	7*{SQuAD+ NatQA,REQA}	78.0/60.9	7*B	45.0/37.3	54.33/41.4	61.3/44.3
7*B	39.8/25.8	4*{SQuAD+NatQA, REQA}+3*{B,REQA}	77.2/58.3	7*B	45.0/37.3	53.4/40.5	-
7*B	39.8/25.8	7*{SQuAD+ NatQA,REQA}	78.0/60.9	7*{NewsQA+ NatQA,DuoRC}	41.7/31.0	53.2/39.2	-
7*B	39.8/25.8	7*{SQuAD+ NatQA,REQA}	78.0/60.9	4*B+3*{NewsQA+ NatQA,DuoRC}	45.6/35.7	54.5/40.8	61.1/44.0
7*B	39.8/25.8	7*{SQuAD+NatQA, REQA_merge}	-	7*B	-	-	61.5/**44.7**

Table 5: Performance of differnt MOE models on dev and test datasets. "B" stands for model that only task-fine-tuned on SQuAD,NatQA,NewsQA. "7*B" means is combined with seven such models but with different seeds. "7*{SQuAD+NatQA,REQA}" means model task-fine-tuned on SQuAD,NatQA datasets and domain-fine-tuned on REQA dataset.

4.4 Results

We get important data insights from previous data analysis

- Domain-fine-tuning on REQA dataset improve performance of models task-fine-tuned on SQuAD, NewsQA and NatQA, and even more on SQuAD and NatQA. However, it is less true for other two out-of-domain datasets. For DuoRC, it only improves if the model is task-fine-tuned on NewsQA and NatQA.
- Models domain-fine-tuned on a small target out-of-domain train dataset may reduce its performance on target out-of-domain validation dataset and test dataset. It is not a bad option to use expert model without any domain-fine-tuning.

Based on such data insights, we build MOE model by mixing right experts. We did not many trials for RACE dataset because 1) From table 1 and 2, we know that model task-fine-tuned on all combinations of in-domain datasets does not improve performance (or not too much) by domain-fine-tuning on

RACE dataset. 2) Only less than 10% of data points in test out-of-domain dataset comes from RACE. Improvement on RACE dataset have much less impact on overall improvement compared to other two datasets.

As shown in table 5, we have 4 submissions to test leaderboard. The MOE model that mixs 7 models task-fine-tuned on SQuAD, NewsQA and NatQA and domain-fine-tuned only for REQA dataset achieved the best results of **F1 61.7** and **EM 44.4**, ranked at $6th$ on test leaderboard. Our model The similar one but domain-fine-tuned on a REQA_merge dataset achieved best **EM** score of **44.7**, ranked at $2nd$ in test leaderboard in terms of **EM** score.

5 Analysis

In the following example, the QA model predict the rough position of answer but add a non-funcional word into its predicted answer span. This is may be due to how the model tokenize the context since some tokenizer will ignore non-functional word like "a", "the".

- **Question**: What color was the background for ABC's 1977 ID sequence?
- **Context**: The 1970s and 1980s saw the emergence ... Among the "ABC Circle" logo's many variants was a 1977 ID sequence that featured a bubble on a black background representing the circle with glossy gold letters, ... card to have a three-dimensional appearance.
- **Answer**: black background
- **Prediction**: a black

We also found a frequent error pattern (the following is one of such examples) where QA model get lost if it is required to predict a date but there are multiple dates in the context. For such questions, the groudtruth answer span is pretty short. The QA model either completely miss it or only predicted answer span barely cover it which cause $F1$ score to be low.

- **Question**: when did they start vaccinating for whooping cough
- **Context**: BPB An estimated 16.3 million people worldwide were infected in 2015 . Most cases occur in the developing world , and people of all ages may be affected . In 2015 , it resulted in 58,700 deaths – down from 138,000 deaths in 1990 . Nearly 0.5 percent of infected children less than one year of age die . Outbreaks of the disease were first described in the 16th century . The bacterium that causes the infection was discovered in 1906 . The pertussis vaccine became available in the 1940s .
- **Answer**: the 1940s
- **Prediction**: 1906 . The pertussis vaccine became available in the 1940s

6 Conclusion

In this work, we built a MOE model by mixing 7 DistilBERT-based QA expert models that are task-fine-tuned on in-domain training datasets. We built data insight by carefully examining performance correlation across in-domain datasets and out-of-domain datasets and found out domain-fine-tuning on small target out-of-domain dataset that has quite different distribution than in-domain training dataset does not necessarily translate into out-of-domain performance on target dataset. We carefully select a set expert models for each out-of-domain set by leveraging data insights aforementioned. We achieved **F1** score of **61.7** (ranked 6th out of 74 in test leaderboard) and **EM** score of **44.4** (ranked 2nd out of 74 in test leaderboard) in out-of-domain test datasets as of March 19, 2021.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, July 2020. Association for Computational Linguistics.

[3] Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 220–227, Hong Kong, China, November 2019. Association for Computational Linguistics.

[4] Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav Artzi. Revisiting few-sample bert fine-tuning, 2020.

[5] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[6] Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603, 2016.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[8] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020.

[9] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[10] William Yang Wang and Diyi Yang. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[11] Jason Wei and Kai Zou. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China, November 2019. Association for Computational Linguistics.

[12] Kushal Kafle, Mohammed Yousefhussien, and Christopher Kanan. Data augmentation for visual question answering. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 198–202, Santiago de Compostela, Spain, September 2017. Association for Computational Linguistics.

[13] Adams Wei Yu, David Dohan, Thang Luong, Rui Zhao, Kai Chen, and Quoc Le. Qanet: Combining local convolution with global self-attention for reading comprehension. 2018.

[14] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

[15] CS224n TAs. Cs 224n default final project: Building a qa system (robust qa track). 2021.

[16] Wen Zhou, Hang Jiang, and Xianzhe Zhang. Ensemble bert with data augmentation and linguistic knowledge on squad 2.0. 2019.

[17] Li Yi. Avengers: Achieving superhuman performance for question answering on squad 2.0 using multiple data augmentations, randomized mini-batch training and architecture ensembling. 2020.

[18] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[19] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[20] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, March 2019.

[21] Amrita Saha, Rahul Aralikatte, Mitesh M. Khapra, and Karthik Sankaranarayanan. DuoRC: Towards complex language understanding with paraphrased reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1693, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[22] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[23] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada, August 2017. Association for Computational Linguistics.