Context Demonstrations and Backtranslation Augmentation Techniques For a More Robust QA System

Stanford CS224N Default Project

Sammy Mohammed
Department of Computer Science
Stanford University
sammym@stanford.edu

Andrea Collins
Department of Computer Science
Stanford University
acoll3@stanford.edu

Ha Tran
Department of Computer Science
Stanford University
hahntrn@stanford.edu

Abstract

Because many real-world NLP tasks rely on user data that is not necessarily guaranteed to be in-distribution, it is critical to build robust question answering systems that can generalize to out-of-domain data. We aim to build a question answering system using context demonstrations and dataset augmentation via backtranslation on top of DistilBERT that is robust to domain shifts. Our method replicates one of the two approaches described in Gao et al. (2020), sampling and appending out-of-domain demonstrations to each training example when finetuning the model. Our method also augments the out-of-domain dataset from which demonstrations are sampled using backtranslation to generate in-distribution training examples. We find that the basic approach of simply appending randomly sampled out-of-domain demonstrations to in-domain contexts does not improve model F1 and EM score performance, but supplementing this approach by adding separator tokens between each demonstration and augmenting the out-of-domain training dataset using backtranslation improves model performance.

1 Introduction

Natural language processing systems have long faced problems effectively generalizing to out-of-domain data. This problem is especially pertinent to the task of question answering. Many user-facing question answering systems rely on user input, which is not guaranteed to be in the same domain as the dataset on which the model was trained, and therefore cannot generalize well to these domain shifts and produce accurate predictions. In Jia 2020, systems trained on the Stanford Question Answering Dataset (SQuAD) with adversarially inserted out-of-domain sentences dropped from an average accuracy of 75% F1 score to 36% in a clear display of the difficulties many language models experience when faced with out-of-domain data.

Furthermore, building models that can make accurate predictions given only a limited number of training examples from a particular domain is essential. Large sets of labeled training data are be very difficult to create and process, but model predictions must still be useful. Furthermore, large language models with a large number of parameters have particular difficulties learning and making accurate predictions on a small amount of out-of-domain data.

Due to these unique problems, we aim to build a lightweight question answering system that is more robust to domain shifts. We achieve this by using demonstrations to expose the model to

out-of-domain data during finetuning, and augmenting a small number of out-of-domain training examples using backtranslation to produce more diversity in the out-of-domain data. We replicate one of the approaches laid out by Gao et al.[1] by randomly sampling and augmenting a similar out-of-domain context to each in-domain context seen during finetuning. Additionally, we follow a similar approach to Longpre et al.,[2] where we generate paraphrased contexts, questions, and answers from the out-of-domain dataset using a pre-trained machine translation model, Marian NMT, to translate the original data into French, Chinese, and Dutch, and then from French, Chinese, or Dutch back to English. We then augment our relatively small out-of-domain training dataset with these paraphrased examples.

2 Related Work

Numerous works have been done targeting the difficulties of domain-agnostic question answering models, specifically in a setting where we have limited out-of-domain data.

Gao et al. (2020)[1] explored improvements to few-shot learners through demonstrations and and automatic prompt generation as ways to improve on the naive in-context learning approach used by GPT-3. GPT-3 has shown impressive results in few-shot settings by sampling random examples (demonstrations) as well as prompts with masked tokens (meant for the model to fill in the blank) and concatenating them with the input. Gao et al. expanded on this by prioritizing useful examples during demonstration sampling with the use of a pretrained sentenceBERT model to identify examples that are semantically close to the context of the given input. The authors used these demonstrations as learning examples to help the model learn to fill in templates–prompts with masked tokens as fill-in-the-blanks. Since the size of our out-of-domain training dataset is rather small and almost the same as the size of our out-of-domain validation dataset, we adapted these techniques in hopes that their improvements in few-shot settings would translate to better robustness against domain shifts.

Ribeiro et al. (2018) [3] has seen success with using machine translation to obtain semantically-equivalent adversaries. Training the model on dataset augmented with semantic-preserving perturbations–additional examples that are semantically the same as the original examples but worded differently and can confuse the model–showed great improvement to the flexibility of the model.

Junczys-Dowmunt et al. (2018)[4] created Marian, an efficient Neural Machine Translation model written in C++ with minimal dependencies. Marian is a research-friendly model that offers both high training and translation speed.

Longpre et al. (2019)[2] found good results with negative sampling to better identify unanswerable questions, domain sampling (training on examples from multiple domains), modifying the sampling distribution to deprioritize certain domains that degrade the general performance of the model on a particular task, and data augmentation with paraphrases obtained through back-translation with priority on more challenging examples.

We primarily referenced the improved in-context learning method with demonstrations and prompts from Gao et al. (2020) for the few-shot learning aspect and the back-translation approach used in Ribeiro et al. (2018) for the data augmentation aspect.

3 Approach

3.1 Demonstrations (with separator tokens)

We first implemented the approach of fine-tuning the model with demonstrations laid out in Gao et al.. Let D_{train} denote the subset of the in-domain training data and D^C_{train} denote the subset of the out-of-domain training data of domain C.

Demonstrations appending
We appended demonstrations from the out-of-domain training set prior to training our model on an in-domain training set D_{train}. For each training example context $T_{in} \in D_{train}$, we found the top 50% out-of-domain training example contexts $T_{out} \in D^C_{train}$ that are most similar to T_{in}, using cosine similarity as a metric for similarity. From those top 50% contexts, we randomly sampled one context T_{out} and appended it to T_{in}. We did this for each of the datasets in the training set, and

reversed the process when finetuning on the out-of-domain train set by sampling from demonstrations from the contexts of the in-domain train set. We followed this approach using a bag-of-words representation to generate embeddings for all contexts.

We added a separator token between each demonstration (context) as a starting point.

Single sentence demonstrations
We also tried sampling a single sentence from the top 50% most similar sentences in D^C_{train} and appending this single sentence to T_{in}, rather than sampling and appending the entire paragraph of context.

Separator token between each sentence
Finally, we experimented with adding a separator token between each sentence in the demonstrations. We elected to use this method as the default for all of our experiments with out-of-domain context appending since it led to a slight improvement in the F1 score after finetuning.

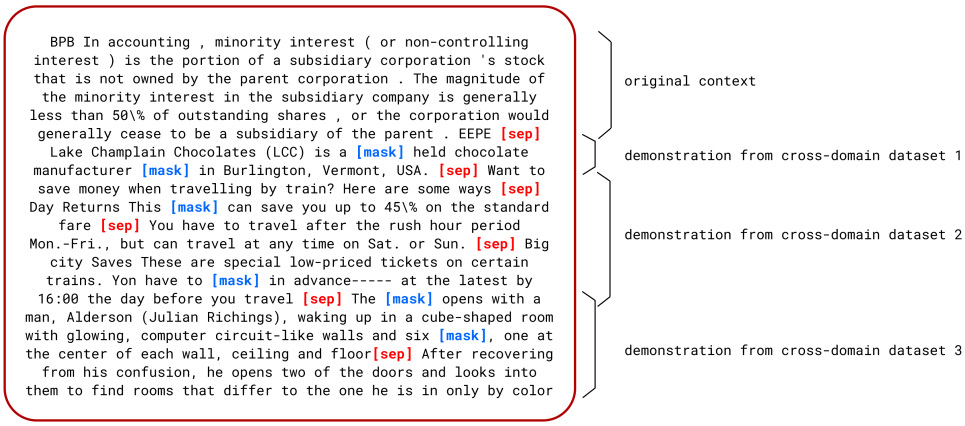

Figure 1: Example of a context after appending demonstrations with separator tokens between each sentence and 2 mask tokens in each demonstration.

3.2 Templates (with mask tokens)

We then experimented with using separator and mask tokens. We followed the same sampling technique outlined above, but prior to appending the out-of-domain contexts T'_{in} to the in-domain contexts, we inserted separator tokens in between each sentence of T'_{in}, and replaced either 2 or $1/12 * (\textit{context size})$ random words with mask tokens for the model to fill in as our simplified version of generating templates.

3.3 Backtranslation

Finally, we applied a data augmentation method inspired by the backtranslation approach detailed in Longpre et al. on the out-of-domain dataset. We used Marian NMT, a pre-trained transformer machine translation model, to translate the text of each example in D^C_{train} (including the context, question, and answer) first from English to French, then back from French to English. We also experimented with backtranslation from French, Dutch, and Chinese combined. Our new augmented dataset, D'^C_{train}, includes all backtranslated data appended to the original data. Finally, we followed the same strategy of appending contexts with sentence seperator tokens as outlined above, sampling from each augmented D'^C_{train}.

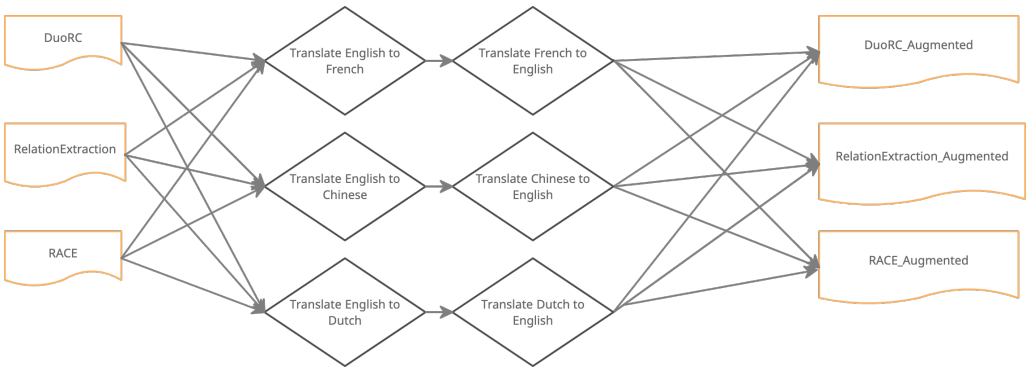

Figure 2: Workflow used for backtranslation of datasets.

4 Experiments

4.1 Data

For this project, we used SQuAD (150k questions), NewsQA (100k human-generated QA pairs from 10,000 articles), and Natural Question (323k examples) for our in-domain datasets, and DuoRC (186k QA pairs), RACE (100k questions from 28k passages), and RelationExtraction (70k sentences) for out-of-domain datasets. Each dataset entry consists of a context paragraph, a question, and a series of possible answers, with each answer consisting of the text of the answer and the starting index of the answer span in the context paragraph.

We also created and used a randomly sampled smaller in-domain training dataset that is $1/5$ of the original size for faster iteration on our experiments. In order to maintain the same distribution of samples in each dataset, we separately sample $1/5$ of the training exmaples from each of the datasets in the training set (Natural Question, NewQA, SQuAD).

4.2 Evaluation method

The demonstration appending and prompt generation method in Gao et al. was evaluated using F1 score to compute performance, which aligns with our own measurement goals. In addition to F1 score, we also evaluated our model's performance using exact match (EM) score in order to compare the model's performance in generating approximate and exact answers.

4.3 Experimental details

All experiments were trained for 3 full epochs.

1. **Baseline model:** We trained our baseline model, a pretrained DistilBERT model, on the large in-domain training dataset, with a default learning rate of $3e^{-5}$ and a default batch size of 16.
2. **Out-of-domain demonstrations:** We trained our model on the in-domain training datasets and appended radomly sampled contexts from the out-of-domain datasets as detailed previously. We trained this model on the large in-domain dataset with the default learning rate of $3e^{-5}$ and the default batch size of 16.
3. **Out-of-domain demonstrations with a smaller learning rate:** We followed the same approach of appending out-of-domain demonstrations as in the previous experiment, but decreased the learning date to $3e^{-6}$. We trained this model on the large in-domain dataset with the default batch size of 16.
4. **Out-of-domain demonstrations with masking:** We appended out-of-domain demonstrations and masked 2 random words in each in-domain context. We trained this model on the large in-domain dataset with the default learning rate of $3e^{-5}$ and the default batch size of 16.

5. **Out-of-domain demonstrations with masking and smaller learning rate** We followed the same approach as in our previous experiment, but trained our model with a smaller learning rate of $3e^{-6}$) and the default batch size of 16.

6. **Out-of-domain demonstrations with separated sentences:** We followed the approach detailed previously of inserting separator tokens in between every sentence of each out-of-domain demonstration before appending it. We trained this model on the smaller sampled in-domain training dataset with the default learning rate of $3e^{-5}$ and the default batch size of 16.

7. **Out-of-domain single sentence demonstrations:** We randomly sampled and appended single sentences from the out-of-domain contexts rather than entire contexts. We trained this model on the smaller sampled in-domain training dataset with the default learning rate of $3e^{-5}$ and the default batch size of 16.

8. **Out-of-domain demonstrations with separated sentences:** We followed the exact same approach as our previous experiment with separated sentences, but as a followup, trained our model on the entire large in-domain training dataset instead of the small sampled in-domain dataset. We trained the model with the default learning rate of 3e-6 and finetuned the model with a smaller learning rate of $3e^{-6}$ and the default batch size of 16.

9. **Out-of-domain demonstrations with separated sentences and masking:** We followed the same approach as in our experiment with separated sentences above, but trained our model on the entire large in-domain training dataset instead of the small sampled in-domain dataset and masked all instances of a randomly chosen word in each in-domain training example. We trained this model with a smaller learning rate of $3e^{-6}$ and the default batch size of 16.

10. **Out-of-domain demonstrations with French backtranslation dataset augmentation and masking:** We trained this model on the entire large training dataset, and randomly sampled demonstrations from the augmented English-French backtranslated out-of-domain training dataset. We trained this model with a smaller learning rate of $1e^{-6}$ and the default batch size of 16. We masked 2 words in each demonstration after backtranslating and inserted separator tokens between each sentence.

11. **Out-of-domain demonstrations with French, Mainland Chinese, and Dutch backtranslation dataset augmentation and masking:** In order to investigate for further gains, we retrained our model on the entire train set, and randomly sampled demonstrations from an out-of-domain dataset composed of the original English, French to English, Chinese to English, and Dutch to English translations. We trained this model 3 times and found that the default batch size of 16 and a learning rate of $8e^{-6}$ produced the best results. As before, we masked 2 words in each demonstration after backtranslating and insert separator tokens between each sentence.

4.4 Results

Results on development set:

	Full-size training set	**F1**	**EM**
1	Baseline	48.43	33.25
2	Baseline train + finetune w/ OOD demonstrations	47.10	31.68
3	Baseline train + finetune w/ OOD demonstrations + 3e-6 learning rate	48.36	32.98
8	Baseline train + finetune w/ OOD demonstrations + sentence separation	47.10	31.68
4	Baseline train + finetune w/ OOD demonstrations + masking	48.62	32.20
10	OOD demos w/ backtranslation (French) augmentation + 7e6 learning rate	49.01	33.25
11	OOD demos w/ backtranslation (French, Dutch, Chinese) + 1e-5 learning rate	48.55	32.72
11	OOD demos w/ backtranslation (French, Dutch, Chinese) + 8e-6 learning rate	49.01	33.25

	Small training set	Training		Finetuning	
		F1	EM	F1	EM
1	Baseline	41.82	25.92	–	–
9	OOD demonstrations + sep. sentences + masking	41.86	25.92	–	–
2	OOD demonstrations	43.47	26.70	43.47	26.70
6	OOD demonstrations + separated sentences	43.47	26.70	43.52	26.70
7	OOD demonstrations + single sentences	–	–	43.47	26.70

Results on test set:

Test set	F1	EM
OOD demos with backtranslation (French, Dutch, Chinese) augmentation	60.327	42.431
OOD demos with backtranslation (French) augmentation	60.327	42.431

5 Analysis

We found that the basic approach of appending out-of-domain demonstrations to in-domain training data did not improve the model's performance on out-of-domain data. We hypothesize that this approach yielded lower scores than the baseline because we increased the context size approximately 4 times by appending 3 demonstrations pulled from the out-of-domain contexts, making it more difficult to capture long-distance dependencies in attention scores. This might have hurt the model's performance because parsing through more context for the answer could've made it harder to find the correct answer.

Incorporating mask tokens into the appended demonstrations led to a modest increase in F1 score relative to basic demonstration appending, but a reduction in the EM score on the full dataset. However, this approach did not improve upon the baseline EM and F1 scores. This indicates that the change in masking improved our model's ability to predict words, a subtask of the question answering task, and also improved our context appending approach. We hypothesize that the performance reduction relative to the baseline model occurred due to the large increase in the size of the contexts, and the masking only partially affected this.

We found that adding separator tokens in between each demonstration and in between each sentence of each demonstration improved on the baseline scores on the small, sampled training dataset. However, this approach led to the same EM and F1 scores as the basic demonstration approach on the large training dataset. We believe that this approach helped the model better capture long-distance dependencies due to the separation of sentences. However, after inspecting the appended demonstrations, we hypothesize that the lack of diversity and relatively small size of the out-of-domain training dataset led to similar or the same context demonstrations being appended multiple times, preventing the model from truly learning anything about the out-of-domain dataset distribution. This experiment was performed before the integration of backtranslation for data augmentation.

We only tested appending single-sentence demonstrations instead of whole contexts with the small sampled training dataset, and this approach led to improved scores over the baseline on this small dataset. We hypothesize that appending shorter demonstrations better allowed the model to encode information. Because this approach scored lower on the small training set than the approach of adding separator tokens, we chose to pursue the latter approach on the large training set instead.

Combining the approaches of demonstration appending with separator tokens and augmenting the out-of-domain dataset using backtranslation to generate training examples in the same out-of-domain distribution led to improved performance over the baseline model. Augmenting the dataset with examples backtranslated from French, Chinese, and Dutch, led to similar performance as backtranslating from a single language, French. We hypothesize that augmenting the dataset in this way increased the diversity of the out-of-domain dataset and the diversity of the sampled demonstrations, thus allowing the model to generalize better when making predictions and leading to better model performance. However, since backtranslation into multiple languages still relies upon a universal source text, the amount of data fuzzing and augmentation is limited beyond the first set of back translations. Looking to Gao, Fisch, and Chen's work in *Making Pre-trained Language Models Better Few-shot Learners*, a key component of context appending involves generating data. This approach to backtranslation augmented our training set to 3 times its original size.

Beyond pure score improvements, we can also analyze a few of the model's quirks when using backtranslation. Our model had a tendency toward brevity, predicting "more than 200" when supposed to predict "more than 200 strong" in one specific instance. In another instance, with a too-aggressive learning rate, the model learns too much from the backtranslated text and predicts "non-religious war" in place of "secular war." Ultimately, however, unlike the baseline or original out-of-domain context appending, the backtranslated model is able to consistently adapt to phrases not seen in the original training text. We hypothesize this is what allowed the model to outperform the initial out-of-domain context appending approach.

6 Conclusion

In conclusion, we discovered that combining context demonstrations with dataset augmentation can lead to improved performance on out-of-domain datasets. We were able to improve on the baseline F1 score by 0.6 on the validation set, and generated more significant gains on the test set, significantly improving our ranking on the RobustQA test leaderboard compared to the validation leaderboard at the time of submission. This approach also does not require extra external data, and allows for data augmentation without new data gathering. A few of the primary limitations of this approach are that it relies on access to quality input data for out of domain training, as well as the fact that further extending the backtranslation approach is computationally expensive, since we rely on a pretrained transformer model to perform our translations. One avenue for future work is to look toward further backtranslation through exploring different languages or translation chaining, in order to expose our model to more diverse text. Potential further exploration can be done with different demonstration integrations, such as incorporating less similar contexts or using a transformer to generate text.

References

[1] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. 2020.

[2] Shayne Longpre, Yi Lu, Zhucheng Tu, and DuBois Chris. An exploration of data augmentation and sampling techniques for domain-agnostic question answering.

[3] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Semantically equivalent adversarial rules for debugging nlp models. 2018.

[4] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, Andre F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in c++.