Gaining More from Less Data in out-of-domain Question Answering Models

Stanford CS224N {Default} Project

Mohammed Salman
Department of Computer Science
Stanford University
salmanmo@stanford.edu

Abstract

We propose text augmentation techniques for Question Answering task in NLP that involves using synonyms with stochasticity on out-of-domain datasets (DuoRC and RACE and RelationExtraction) that are set to be 400 times smaller than the in-domain datasets (SQuAD, NewsQA, NaturalQuestions). We address ways to improve extraction of generalized information from out-of-domain or less available datasets from large pre-trained models like BERT or its variation DistilBERT which is used here with also being able to benefit from producing QA applications across domains. It is found that augmenting less available QA datasets in ways described, indicate improvement in generalization, but not all augmentations strategies are equally good. We find that these augmentations are helpful in achieving better performance on out-of-domain data.

- Project Mentor: Rachel Gardner

1 Introduction

Lagre pre-trained models like BERT (Jacob et al., 2019) and its variants require large datasets to quickly learn the domain specific nuances and the interactions between the words for different NLP tasks. For carrying out model-independent experiments, we limit our paper to DistilBERT (Victor et al., 2020) for QA task. It is a BERT (Bidirectional Encoder Representations from Transformer) lightweight variant called DistilBERT which has a carryover of 97% of its language understanding capabilities with 40% smaller in size and about 60% faster.

Understanding some of the different types of Question Answering tasks; open-domain which requires knowledge without any restrictions to any particular domain, closed-domain which is focused on a particular set of domains, and reading comprehension. Our task will be confined to reading comprehension. Clearly, answers in the reading comprehension type QA task are word(s) or sequence of words taken directly from the paragraph or context.

Secondly, these models are only as good as the data provided based on the quality and the quantity available for training. In many real world situations data is not available in the quantities we would like to have; to train deep learning models, often times the required amount of data is difficult to collect. Where data is not available in large quantities for BERT models, there are ways to increase the amount of data points. Data augmentation is one of them and commonly used in computer vision (Simard et al., 1998; Szegedy et al., 2014; Krizhevsky et al., 2017) and speech (Cui et al., 2015; Ko et al., 2015) and can help train more robust models, particularly when using smaller datasets. We use text augmentation techniques inspired by image augmentation that use rotating, mirroring, flipping etc.. We use in-domain datasets with total training examples: 150, 000 (SQuAD (Rajpurkar et al., 2016), NewsQA (Trischler et al., 2017), NaturalQuestions (Kwiatkowski et al., 2019)) and

only 381 out-of-domain training examples (DuoRC (Saha et al., 2018), RACE (Lai et al., 2017), Relation Extraction(Levy et al., 2017)). These 2 supersets of datasets have more overlap in terms of vocabulary/unique words than one might expect (as will be shown), due to the large size difference (400x), only an addition of a few hundered words contribute to this "out of domain-ness"; moreover, hundreds of words arranged differently in a sequence ultimately causes the out-of-domain-ness. However, not always are there large datasets available for every scenario or a domain. Implementing data augmentation using insertion, deletion, replacement kind of operations on words.

2 Related Work

Many works show that state-of-the-art neural models learn brittle correlations that hurt their out-of-domain performance. One way to prevent the model from learning such brittle correlations is to encode label preserving invariances via data augmentation, which we carry out. At the time of writing this paper, there hasn't been a paper addressing word replacement type augmentation specifically for question answering NLP task. Different methods are used to increase the data points for QA, paraphrasing through backtranslation is one of them. For instance, given an input x = (q, p) with a label y, one could create an augmented example x0 = (q0, p0) where q0 and p0 are paraphrases of q and p respectively. Such paraphrases could be produced via back-translation. To get q0 via backtranslation, q is first translated into a "pivot" language (say Russian), and then translated back to the original language to produce a paraphrased version.

In word substitution based data augmentation, individual words in the input are replaced with either synonyms from a lexicon as done for text classification (Wei et Zou, 2019), or replaced with [MASK] tokens which are then filled with BERT (Garg et Ramakirshnan, 2020) to obtain an augmented input. One of the challenges in substitution based augmentation is adding unneeded noise which could negatively impact the performance. Such techniques have found to be effective in augmenting the training data, resulting in improved robustness.

3 Approach

Using synonyms into question answering or reading comprehension task, we would want to understand how QA systems utlize datasets. Passage/context, question and answer are the key segments of any reading comprehension type QA dataset; for this task we do not modify answers or answers present in the context, rather focus on questions and context. We define 5 augmenting strategies that we think should help; approach for each is described as follows:

- **Question synonym replacement (QSR)** - ignoring words like the first question word (which mostly tend to be an interrogative word), stop words, capitalized words and numerical objects, we will call these words "words of importance". The other words are replaced by their synonyms.
- **Context based synonym insertion-before answer (SIBA)** - at random positions synonyms of entire context, except words of importance, are inserted into the context before answer.
- **Context based synonym insertion after answer (SIAA)** - at random positions synonyms of entire context, except words of importance, are inserted into the context after answer.
- **Context chunk swapping (CCS)** - the chunk with the answer is taken and inserted randomly into the non-answer chunk to create a mixed context.
- **Context deletion (CD)** - context words that are 2-3 words away from the answer chunk are deleted randomly.

Motivation behind the above 5 strategies:

1. **QSR** - producing different ways of asking the same question.
2. **SIBA** - words appear before an answer can be important, and so inserting synonyms before, will create a similar context without modifying those important features that lead into the answer.
3. **SIAA** - similar to second point, the words leading from the answer and the answer itself are preserved.
4. **CCS** - a few neighbour words from the answer are taken and moved into other parts of context to

again preserve words important to answer the question.
5. **CD** - context words away from the answer chunk are deleted at random, answer chunk is defined as 2-3 words before and after the answer which is present in the context.

The synonym generation was taken from nltk's wordnet and a function taken from here which was done for a text classification task (Wei et Zou, 2019), however the strategy to use in context to QA is implemented differently, augmentation for reading comprehension needs to be done carefully so that the answer is not lost or modified; with that there were some challenges in pre-processing steps that had to be modified to include augmented data from SIBA, SIAA and CCS. We also present some analysis on why and how augmentation potentially helps generalization, with a new metric defined called EM+.

4 Experiments

4.1 Data

Defining datasets, 5 sets of data prepared from the original set; number of data points are detailed in the table 1., with in-domain datasets kept as original and augmentation is carried on out-of-domain as follows:

1. **Augmented set 1**: uses QSR 3x, SIBA 5x, SIAA 5x, CCS 5x, which increases dataset multiplier to 18x.
2. **Augmented set 2**: uses a balance of the strategies and uses QSR 3x, SIBA 3x, SIAA 3x, CCS 3x, which increases dataset multiplier to 12x (33% reduction).
3. **Augmented set 3**: same as augmented set 2 but with 33% reduction for each dataset for in-domain datasets (to reduce ID overfitting).
4. **Augmented set 4**: same as augmented set 2 but includes the CD technique 3x, which increases dataset multiplier to 15x.
5. **Augmented set 5**: same as augmented set 2 but reduced to 2x on all augmentation strategies except CD, which increases dataset multiplier to 8x.

Dataset	Question Source	Passage Source	Train	dev	Test
in domain dataset					
SQuAD	Crowdsourced	Wikipedia	50000	10507	-
NewsQA	Crowdsourced	News articles	50000	4212	-
Natural Questions	Search logs	Wikipedia	50000	12836	-
out of domain dataset					
DuoRC	Crowdsourced	Movie reviews	127	126	1248
RACE	Teachers	Examinations	127	128	419
RelationExtraction	Synthetic	Wikipedia	127	127	2693

Table 1: Dataset used for the project

4.2 Evaluation method

For evaluating the model on robustness we use Exact Match (EM) and F1 scores.

- **EM**: is a binary measure (i.e. true/false) of whether the system output matches the ground truth answer exactly. For example, if your system answered a question with 'Einstein' but the ground truth answer was 'Albert Einstein', then you would get an EM score of 0 for that example.

- **F1**: it is the harmonic mean of precision and recall. In the 'Einstein' example, the system would have 100% precision (its answer is a subset of the ground truth answer) and 50% recall (it only included one out of the two words in the ground truth output), thus a F1 score of 2 × prediction × recall / (precision + recall) = 2 x 50 x 100 / (100 + 50) = 66.67%.

4.3 Experimental details

Augment strategy	ID train		OOD train		ID dev		OOD dev	
	EM	F1	EM	F1	EM	F1	EM	F1
Baseline 1	66.06	82.99	n/a	n/a	51.67	68.06	32.19	48.55
Baseline 2	64.11	80.57	55.38	72.29	51.98	68.42	34.53	49.58
Augmented Base	–	–	**71.13**	**83.71**	–	–	37.70	50.56
Balanced Augment ★	–	–	70.34	83.63	–	–	**37.96**	**51.16**
Balanced Non-Augment	–	–	68.77	81.55	–	–	36.39	50.09
Balanced Augment (with CD)	–	–	70.34	82.93	–	–	35.34	50.14
Balanced Augment-small	–	–	68.24	81.15	–	–	35.08	50.37

Table 2: **OOD dev** - Indicates the relevant score at improving towards the best possible score; "-" Indicates not available scores.

The model used is a pretrained DistilBERT, with learning rate = 3e -5, 5 epochs and took a total training time of about 15-18 hours for each experiment which would be fine-tuned on mentioned datasets. Comparison for scores is on dev out-of-domain datasets; augmented strategies all compare on Baseline 2.

- Baseline 1: Fine-tuned only on ID datasets, provides a good baseline to start off with improving OOD *dev* set.
- Baseline 2: Finetuned on ID and OOD datasets, which only improved by +1 F1 and +2.5 EM.
- Augmented Base: Fine-tuned on Augmented set 1, which improved by +1 F1 and +3.1 EM on top of the Baseline 2 with an 18x increase on OOD dataset. As we will see below, this 18x increase contains a lot of noise as well.
- Balanced Augmented: Fine-tuned on Augmented set 2 which improved by +1.5 F1 and +3.5 EM on top of the Baseline 2 with a 12x increase on OOD dataset.
- Balanced Non-Augmented: Fine-tuned on Augmented set 3 which improved by +0.5 F1 and +1.9 EM on top of the Baseline 2, this removes 33% of all ID data (least data present).
- Balanced Augment (with CD) : Fine-tuned on Augmented set 4 which improved by +0.5 F1 and +0.8 EM on top of the Baseline 2 with a 15x increase on OOD dataset.
- Balanced Augmented-small: Fine-tuned on Augmented set 5 which improved by +0.8 F1 and +0.5 EM on top of the Baseline 2 with a 8x increase on OOD dataset.

4.4 Results

On RobustQA test leaderboard: "aug-robust"

Augment strategy	Rank (test)		OOD dev		OOD test	
	EM	F1	EM	F1	EM	F1
Balanced Non-Augment	–	–	36.39	50.09	41.055	57.498
Augmented Base	–	–	37.70	50.56	42.798	60.148
Balanced Augment ★	3**	12**	**37.958**	**51.161**	**43.372**	**60.370**

Table 3: Scores for the best strategies on validation and test leaderboards; **ranks are reported 1 hour before deadline.

We observe that augmentation by synonym replacement with 4 strategies did help; however, the 18x dataset also added some noise. This is indicative from observing Balanced Augment that had an improvement. Deleting words from context is not helpful, moreover, not all augmented datasets even from a single strategy are equal due to randomness involved; for example: dataset1 created from SIBA could improve performance but another dataset2 created from the same SIBA, would not. Increase of scores was expected, however some strategies(Balanced Augment-small) did worse bringing all other augmenting strategies to 0 improvement, this was not expected.

5 Analysis

5.1 What helps OOD datasets?

Dataset	Unique word count for ID/ OOD	Exclusive(unique) word % of OOD
Original	From context: **404**k for ID, **3.5**k for OOD From question: **50**k for ID, **700** for OOD	Context: **325** words exclusive to OOD dataset => **9.2%** Question: **160** wordsexclusive to OOD dataset => **22.8%**
Augment set 1	From context: **404**k for ID, **8.2**k for OOD From question: **50**k for ID, **2.3**k for OOD	Context: **855** words exclusive to OOD dataset => **10.33%** Question: **538** words exclusive to OOD dataset => **23.07%**
Augment set 2 ★	From context: **404**k for ID, **8.2**k for OOD From question: **50**k for ID, **2.3**k for OOD	Context: **853** words exclusive to OOD dataset => **10.316%** Question: **538** words exclusive to OOD dataset => **23.07%**
Augment set 3	From context: **313**k for ID, **8.2**k for OOD From question: **40**k for ID, **2.3**k for OOD	Context: **941** words exclusive to OOD dataset => **11.38%** Question: **596** words exclusive to OOD dataset => **25.55%**
Augment set 5	From context: **404**k for ID, **8.2**k for OOD From question: **50**k for ID, **2**k for OOD	Context: **850** words exclusive to OOD dataset => **10.30%** Question: **440** words exclusive to OOD dataset => **21.72%**

Table 4: Comparing original and augmented datasets with unique words

We ignore answers segments here as they were not changed for data augmentation. Also, augment set 4 is same as augment set 2, we ignore it. Moving from original dataset to augment set 1, there is an increase in exclusive/ unique word % that we conjecture led to higher scores in F1/ EM. For augment set 1 and 2, their unique % are very similar however the scores are not, this reinforces the fact that we concluded, augment set 2 has the noisy data removed. Increased unique word % for augment set 3 does not lead to increased scores because ID datapoints are different and relative comparison doesnt hold true.

We draw 2 conjectures from this analysis:

1. Above table demonstrates a systematic way to identify noisy vs helpful augmented data created.
2. The increase in unique word % leads to increased generalization, however going beyond a certain % the performance will drop.

Analogous to a bowl function in a 3D space, the edges of the bowl function being the ideal augmented dataset while the bottom and out of bowl indicate a drop in performance.
Idea: the more OOD kind words we include that dont repeat in ID dataset, the better the performance (this does not mean adding random unique words will lead to better performance, sequence of words and their meanings matter); the new words that were added from augmentation were closer to the unique words, simply due to the fact that synonyms were used.

5.2 Observation from including CD in augmentation

When adding Context Deletion, the scores decrease so significantly that they almost equals zero augmentation scores which is Baseline 2 (refer Table 2) this noise added from randomly deleting context words is large enough to completely be discarded for the rest of our training. We conjecture for QA reading comprehension task, deletion is not helpful as it may delete key words that lead into an answer.

5.3 Why EM is not a good metric, introducing EM+

There are cases where EM is not a good metric (refer below image) illustrated below. An important reason for this is that we might not know if the model actually memorized answers or actually produced a 1:1 answer.

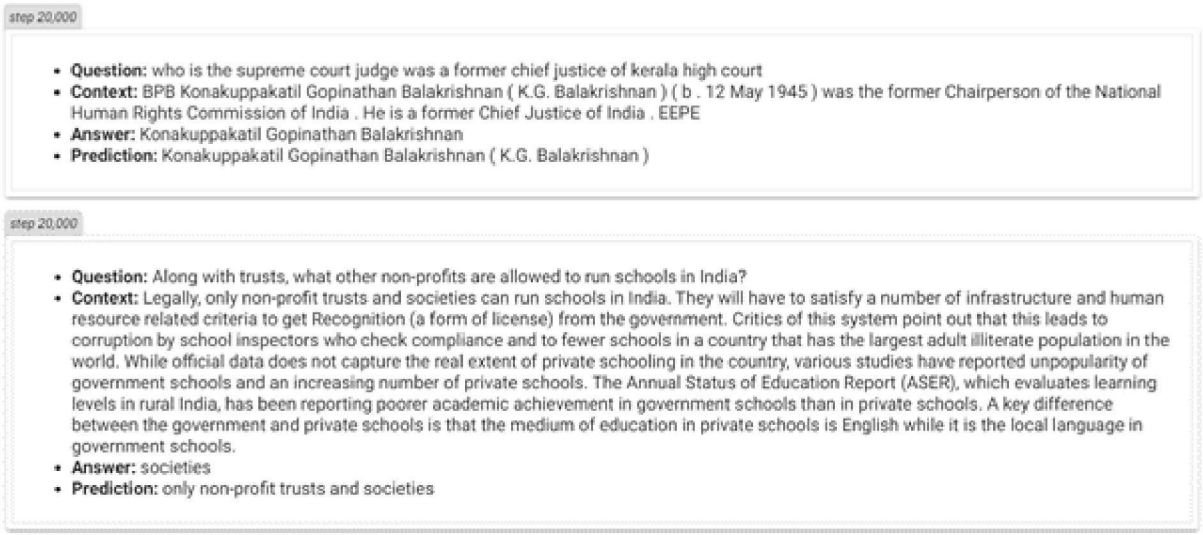

Figure 1: Examples of EM not being useful

EM penalizes quite a lot and so, we propose a new metric EM+ where if the prediction is a superset of the answer, we give EM+ = 1 if, not then EM+ = 0. This makes EM+ positively affect the scores when such above cases occur. It is still a binary measure but provides a unique perspective from original EM and F1. Just like EM, EM+ too has downsides where high score is given even if prediction contains answers with noise.

6 Conclusion and future work

Recognizing which augmenting strategies improves scores, but more importantly which augmented dataset helps this increase once identified a useful augment strategy, is much more valuable because of randomness, some datasets might do better than others. QSR, SIBA, SIAA and CCS are useful augmenting strategies from our experiments.

We coin a term *Augmentation Freedom* (AF) - defined as some vector subspace of obtaining positive scores identified by a metric, AF is directly proportional to the original number of datapoints, beyond a threshold (we conjecture its typically 3x the dataset for the 5 augmenting strategies), no performance gain in observed, going further will only decline the gain made, explained as a bowl function earlier. It is difficult to find augmentation strategies that will be all over on the edge of the bowl given the strategy itself being implemented and the stochasticity that comes with it.

Improvement even though is marginal, its promising to increase the number of datapoints of such smaller datasets "artificially and efficiently". Continued work on this topic could explore which single augmentation technique is working best, which would be an expensive experiment. Identifying which augmentation datsets irrespective of the strategy is also very useful to understand on given the stocasticity in the augmentation.

References

[1] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, 2018.

[2] Zhucheng Tu Chris DuBois Shayne Longpre, Yi Lu. 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*, 2018.

[3] Kai Zou JasonWei. EDA: Easy data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In *Association for Computational Linguistics (ACL)*, 2019.

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

[5] 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 *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics.*, 2015.

[6] Junbo Zhao Xiang Zhang and Yann LeCun. Character-level convolutional networks for text classification. In *In Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, 2015.

[7] Steven Bethard Oleksandr Kolomiyets and Marie-Francine Moens. Model-portability experiments for textual temporal analysis. In *Association for Computational Linguistics (ACL)*, 2011.

[8] Julien Chaumond Thomas Wolf. Victor Sanh, Lysandre Debut. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS*, 2019.

[9] Kenton Lee Kristina Toutanova. Jacob Devlin, Ming-Wei Chang. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2018.

[1] [2] [3] [4] [5] [6] [7] [8] [9]