Task-Adaptive Pretraining, Domain Sampling, and Data Augmentation Improve Generalized Question Answering

Stanford CS224N Default Project (Robust QA)

Davey Huang
Department of Computer Science
Stanford University
huangdh@stanford.edu

Abstract

To create a question answering model that is robust to out-domain (OD) data, we investigate the use of three techniques: task-adaptive pretraining (TAPT), domain sampling, and data augmentation. During TAPT, we perform additional pretraining with masked-language modeling on our question answering datasets. We employ domain sampling during both pretraining and fine-tuning, which preferably samples data that lead to better downstream performance. For our data augmentations, we use synonym replacement and random deletion on the OD training set. During evaluation, we found significant EM/F1 performance improvements by fine-tuning on augmented OD data. We found modest, yet non-trivial, performance improvements with TAPT and domain sampling. Using these three techniques, our model achieved EM/F1 scores of 37.44/51.37 on the development set and 40.12/58.05 on the test set.

1 Introduction

Question answering (QA) is an important problem in NLP, since it can indicate how well deep-learning models understand human language. Many language tasks can also be reduced to QA, such as relation extraction [1] and semantic role labeling [2]. However, generalization on QA remains a difficult task [3]. Although deep-learning models perform exceptionally well on data from their training distribution, they are often unable to generalize to out-of-distribution data, since they rely on heuristics that break down for new example types [4]. This is a significant problem for real-world NLP systems, given the inherent uncertainty in the distribution of human language. We address this problem for the Robust QA final project.

The use of pretrained models has been a key to the success of recent NLP systems, since these models can been adapted to specific tasks with additional fine-tuning [5]. This has led to state-of-the-art performance across NLP tasks [5]. BERT is a popular pretrained model that leverages a bidirectional encoder transformer architecture, trained with masked language modeling (MLM) and next-sentence prediction [6]. BERT contains 110 million parameters, which makes it difficult to run in computationally constrained environments. Thus, DistilBERT uses knowledge distillation to reduce the size of BERT by 40%, while retaining most of its language understanding capabilities [7]. We use DistilBERT in our experiments.

In this report, we employ a three-phase approach that utilizes in-domain (ID) datasets to train a single DistilBERT model that has improved generalization to unseen out-domain (OD) datasets. First, we adapt the pretrained DistilBERT model to our QA datasets using additional task-adaptive pretraining (TAPT). Second, we use domain sampling to focus our model on training examples that are most relevant to OD performance. Third, we fine-tune our model on augmented OD data, which has

been modified by synonynm replacement and random deletion. Using these techniques, our model achieved EM/F1 scores of 37.44/51.37 on the development set and 40.12/58.05 on the test set.

2 Related Work

2.1 Task-Adaptive Pretraining (TAPT)

To adapt a pretrained model to a specific task, previous work has leveraged additional pretraining on unlabeled task-relevant training data. [5] refers to this method as task-adaptive pretraining (TAPT). They use MLM to specialize RoBERTa on a specific domain, which led to consistent improvements over the baseline for all tasks across all domains [5]. Work from [8] also explores the use of multi-task learning to improve generalization on the MRQA Shared Task, which evaluates the generalization capabilities of reading comprehension systems [9]. They found that MLM significantly improved out-of-distribution performance, while natural language inference and paragraph ranking did not lead to significant improvements.

Given these demonstrated improvements, we adopt the use of MLM pretraining on our QA datasets. Since this project uses training data from multiple domains, a logical next step for TAPT is to sample from domains that are most relevant to the downstream task, even during pretraining. Therefore, we extend our implementation to support domain sampling, as described next.

2.2 Sampling Methods

The aim of domain sampling is to intelligently sample from ID data that is most likely to improve OD performance. This can be achieved by training separate models on single ID datasets and evaluating their contribution to OD performance. Work from [10] show that domain sampling leads to improvements on the MRQA Shared Task. Separately, they also show that negative sampling leads to significant improvements. The aim of negative sampling is to intentionally include training sequences with "No Answer". Therefore, we adopt the use of domain sampling during our pretraining and fine-tuning steps. Since negative sampling was included in the baseline model, we do not describe it in this report.

2.3 Data Augmentation

Data augmentation aims to increase the size and variety of a dataset, which can help train more robust models on small amounts of data. Some augmentation techniques include the use of back-translation [11], data noising [12], and generative models [13]. Previous work from [10] showed that back-translation led to no improvement on the MRQA Shared Task. Therefore, we decide to use an approach from [14], which demonstrated that simpler augmentations, such as synonym replacement and random deletion, can lead to similar improvements.

Work from [14] evaluate solely on text classification tasks, such as sentiment analysis and question type classification. We extend their work to evaluate these techniques on QA, which is a more complex task requiring deeper language comprehension. Therefore, our results demonstrate whether simple augmentations can provide improvements for more complex tasks.

3 Approach

3.1 Task-Adaptive Pretraining

We first modify the training datasets to accommodate pretraining with MLM. Concretely, we extract the contexts from each example and split them into training segments. Then, we randomly mask $m\%$ of the subwords with [MASK] tokens. We also adapt our DistilBERT model to accommodate pretraining with MLM, following the implementation of [6]. Concretely, we add a linear projection head on top of DistilBERT, consisting of Feed-Forward, LayerNorm, and Feed-Forward blocks. Our projection head uses the hidden layer embeddings corresponding to the mask tokens to predict the original subword using a cross-entropy loss objective. We use this MLM objective to train our model on the modified datasets.

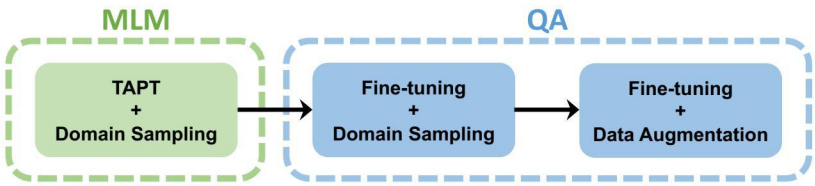

Figure 1: A single DistilBERT model was trained in three phases: (1) TAPT using MLM, supplemented with domain sampling, (2) fine-tuning on all training data, supplemented with domain sampling, and (3) additional fine-tuning on augmented OD data.

We also employ domain sampling during pretraining to sample from datasets that lead to better task-specific OD performance (described below). This is meant to focus the model on learning domains that will be most useful to the downstream task of QA on OD data. Finally, we use this task-adapted model as a new starting point for later fine-tuning. The use of task-adaptive pretraining was inspred by work from [5] and [8]. Our code leverages the HuggingFace transformers library.[1] All code adaptations for the QA datasets and for domain sampling are my own.

3.2 Domain Sampling

To motivate our use of domain sampling, we determine which in-domain (ID) training sets lead to the best out-domain (OD) performance. Concretely, we fine-tune three DistilBERT models on the ID datasets (SQuAD, NewsQA, or Natural Questions). Then, we evaluate each model on the union of OD train and development sets (DuoRC, RACE, and RelationExtraction), obtaining F1 and EM scores. During fine-tuning, we restrict coverage to p percent of the total dataset. This allows selective sampling, since not all examples will necessarily be shown to the model.

Our sampler uses the averaged F1 and EM scores above as weights for randomly sampling segments (without replacement). We also found that NewsQA generates roughly twice as many training segments (126k) compared to SQuAD (50k) and Natural Questions (65k), due to NewsQA's longer contexts. Therefore, we also normalize the sampling weights by the number of training segments contributed by the dataset. Taken together, this means we sample more frequently from datasets that lead to better F1 and EM performance (and less frequently from datasets that lead to worse performance). We also include the small amount of OD training data, which were assigned very large sampling weights. The use of domain sampling was inspired by work from [10]. All code adding domain sampling functionality is my own.

3.3 Data Augmentation

In order to maximally utilize our small amount of OD training data, we augment the OD examples using synonym replacement (SR) and random deletion (RD). Previous results have indicated that for small training sets ($N \leq 500$), simple data augmentation methods can lead to significant performance gains [14]. Since our OD training set only consists of 381 total training examples (127 from each dataset), we use aggressive augmentation by SR and RD.

Concretely, we augment each context, questions pair (c, q) from OD training sets to create k augmented examples (c'_i, q'_i), where $i \in [1, k]$. Each of the k examples is augmented by either SR or RD, with a probability of p_{SR} or p_{RD}, respectively. For SR, the augmented (c'_i, q'_i) has n random words replaced with randomly chosen synonyms from WordNet, which is an English lexical database resembling a thesaurus [15]. For RD, the augmented (c'_i, q'_i) has n random words deleted. The number of words n to replace or delete is given by $n = \alpha l$, where l is the number of words in (c, q).

We do further post-processing to ensure that the answers still point to the correct span in the augmented context. We also update the answer if the relevant context was changed during augmentation. As a result, almost none of the answer spans were corrupted during augmentation. The use of data augmentation was inspired from [10] and [14]. Code adding data augmentation functionality is my own (with minimal reference to [14]).

[1] `https://github.com/huggingface/transformers`

3.4 Model Summary

Our approach consists of three sequential training phases (Figure 1). We first use TAPT with MLM to adapt the DistilBERT model to a new language distribution (QA), while also supplementing with domain sampling. We then load this model before fine-tuning (with domain sampling) on all the training data using the QA objective. Finally, we fine-tune again on true and augmented OD training data, yielding our final model.

3.5 Baseline

Our baseline model was DistilBERT fine-tuned on the three ID training sets. Our baseline model had EM/F1 scores of 31.68/47.10. Please refer to the Robust QA handout for more information.

4 Experiments

4.1 Datasets and Evaluation Method

We are given three in-domain (ID) reading comprehension datasets and three out-domain (OD) reading comprehension datasets. The objective is question answering (QA). Concretely, given a context, question pair (c, q), the model selects an answer for the question as a span from c.

The ID datasets include 50k training examples from each of SQuAD, NewsQA, and Natural Questions. SQuAD has contexts from Wikipedia articles paired with crowdworker questions [16]. NewsQA has contexts from CNN news articles paired with crowdworker questions [17]. Natural Questions has contexts from Wikipedia paired with real user questions [18].

The OD datasets include 127 training examples from each of DuoRC, RACE, and RelationExtraction. DuoRC has contexts from movie reviews on Wikipedia and IMDb paired with crowdworker questions [19]. RACE has contexts and questions from English exams [20]. RelationExtraction curates a dataset from knowledge-based relation extraction data on Wikipedia [21].

We use Exact Match (EM) and F1 scores as evaluation metrics. An EM score of 1 indicates the model predicted the answer exactly; an EM score of 0 is given otherwise. The F1 score is the harmonic mean of precision and recall.

4.2 Experimental details

For TAPT, we split contexts into segments of length 384 with a stride of 128 (to match the pre-processing used during fine-tuning). We mask with probability $m = 15\%$. We pretrain DistilBERT with cross-entropy loss for 3 epochs, using a batch size of 8, with a learning rate of 3×10^{-5}. For our domain sampling configuration, we use $p = 75$, restricting training coverage to 75% of the dataset. We use validation checkpoints every 5000 steps. Training time was 16 hours.

For out single domain contribution experiments, we fine-tune three DistilBERT models on each ID dataset independently. We use the default parameters (3 epochs, batch size of 16, learning rate of 3×10^{-5}) and have validation checkpoints every 2000 steps. Training time was 2-5 hours per model.

For fine-tuning on all training data, we use a batch size of 16 and learning rate of 3×10^{-5}. We train for 4 epochs with the same domain sampling configuration used above. Training time was 19 hours.

For fine-tuning on augmented OD data, we created $k = 16$ augmented examples for every true example. Each example had an equal probability of being augmented by SR and RD ($p_{SR} = p_{RD} = 0.5$). We used $\alpha = 0.1$, such that 10% of words were augmented (by either SR or RD). We trained for 15 epochs, with a batch size of 6, a learning rate of 3×10^{-5}, and validation checkpoints every 1000 steps. Our sampling procedure for each epoch alternated between training primarily on augmented data and training only on true OD data. Training time was 1 hour. All experiments were done on Azure VM GPUs.

Model	Development Set EM	Development Set F1	Test Set EM	Test Set F1
Baseline	31.68	47.10	40.00	57.37
Domain Sampling	32.20	48.22	-	-
TAPT + Domain Sampling	34.55	49.65	-	-
Domain Sampling + Data Augmentation	36.13	50.43	40.12	58.04
TAPT + Domain Sampling + Data Augmentation	**37.44**	**51.37**	**40.12**	**58.05**

Table 1: EM and F1 scores on the held-out OD validation and test sets for each of our model iterations. We only show three test set results due to submission restrictions.

Train Dataset	Out-Domain EM	Out-Domain F1
SQuAD	**27.79**	**43.71**
Natural Questions	24.51	39.21
NewsQA	23.07	38.30

Table 2: Three DistilBERT models were fine-tuned individually on each ID training dataset. The EM and F1 scores are from the union of OD training and development sets.

4.3 Results

4.3.1 Single Domain Contribution and Domain Sampling

We fine-tuned three DistilBERT models on each ID dataset to evaluate the dataset's contribution to OD performance. The model fine-tuned on SQuAD alone performed the best on the OD development sets (Table 2). The models trained on either Natural Questions or NewsQA performed similarly, with Natural Questions leading to a small performance increase. We incorporate these results into our domain sampling strategy during both pretraining and fine-tuning. Thus, segments from SQuAD were randomly sampled more frequently than those from Natural Questions and NewsQA. This led to a modest improvement over the baseline ("Domain Sampling" in Table 1). This was in line with our expectations, since training on more relevant data (with an emphasis on SQuAD and OD examples) is likely to improve OD performance.

4.3.2 TAPT

	Out-Domain Perplexity
Baseline	10.110
TAPT	**6.722**

Table 3: TAPT with MLM leads to improved (lower) perplexity on the OD development set. Our baseline model was DistilBERT without TAPT.

Additional pretraining with MLM resulted in significantly improved perplexity scores on the OD development sets (Table 3). This improvement was better than expected, since pretraining did not include many OD examples. One rational for this is that pretraining on QA datasets can be helpful for generalizing to other QA datasets, even when there is little domain overlap. This also demonstrates that DistilBERT was underfit in its initial pretraining, since additional pretraining on our datasets led to a further decrease in MLM loss.

Using this task-adapted model for our fine-tuning, we observed modest improvements in QA performance. In Table 1, the QA performance of "TAPT + Domain Sampling" on the OD development sets was better than "Domain Sampling" alone. Similarly, the performance of "TAPT + Domain Sampling + Data Augmentation" was better than "Domain Sampling + Data Augmentation" alone. Therefore,

we found that TAPT led to notable performance gains when combined with our other robustness techniques. One rational for this is that the model may be learning heuristics during TAPT with MLM that translate to our QA objective.

4.3.3 Data Augmentation

Synonym Replacement

Context: The police must ~~investigate~~ enquire a ~~series~~ serial publication of ~~robberies~~ looting along a strip of land in the city. The ~~new~~ newfangled mayor of the city (Kenneth Mars) assigns Captain Harris (G.W. Bailey) and Lt. Proctor (Lance Kinsey) to the case, but while on stakeout the Wilson gang manages to slip through their fingers. [...] Hurst apologizes and reinstates the force, and a ~~plaque~~ brass is ~~given~~ pass on to ~~honor~~ respect the officers' ~~bravery~~ braveness the next day.

Question: What gang does the police academy ~~find~~ feel and do battle with?

Figure 2: A real example of an augmented context, question pair created with synonym replacement. Crossed out words are replaced by syntactically valid synonyms (green) or syntactically invalid synonyms (red).

For an additional find-tuning step, we generated augmented examples by SR and RD. Our augmented examples were usually more difficult to understand and often contained invalid syntax (Figure 2). Yet we found that fine-tuning on this augmented data led to a significant performance increase, especially in conjunction with our other robustness techniques. Concretely, "Domain Sampling + Data Augmentation" and "TAPT + Domain Sampling + Data Augmentation" significantly outperformed their counterparts that did not include augmented fine-tuning (Figure 1).

This improvement was expected since, at the very least, we are showing the model OD examples as close to testing as possible, which is likely to improve OD performance. Yet fine-tuning with only the true OD data (no augmentation), was not nearly as successful as using augmented data – we consistently obtained EM/F1 scores below 35/50 on the development set. Therefore, the key to this improvement was to fine-tune on both true *and* augmented OD data.

4.3.4 Results Summary

Our best-performing model employed all our robustness techniques ("TAPT + Domain Sampling + Data Augmentation" in Table 1). This model significantly outperformed the baseline with EM/F1 scores of 37.435/51.37, ranking ninth on the development leaderboard (as of 3/13/21). Although we had significant performance gains on the development set, our best-performing model had only small improvements over the baseline on the test set, with EM/F1 scores of 40.12/58.05 (Table 1), ranking sixteenth on the test leaderboard (as of 3/13/21). One rational for this is that our optimizations were selected based on performance improvements to the development set, which may have tailored the model too closely to the development set, instead of the generalizing to the broader OD distribution. Nevertheless, our model demonstrated some notable improvements over the baseline, which we discuss qualitatively below.

5 Analysis

In our analysis, we compared the predictions of the baseline model to our best-performing model ("TAPT + Domain Sampling + Data Augmentation" in Table 1) on both the development and test sets. On the development set, we found 224 disagreements out of 382 examples (58.6%). Many disagreements were minor. For instance, the baseline would predict "test messages" while our model would predict "text messaging". There were differences in punctuation: predicting "Lafangey Parindey." instead of "Lafangey Parindey". There were also more significant disagreements that might elucidate how our model is working. We discuss these examples below.

5.1 Extraneous Information (Development Set)

- **Context:** *Designed to express the playful qualities of five little children who form an intimate circle of friends, the Five Friendlies also embody the natural characteristics of four of China's most popular animals–the Fish, the Panda, the Tibetan Antelope, the Swallow– and the Olympic Flame. [...] When you put their names together–Bei Jing Huan Ying Ni–they say "Welcome to Beijing," offering a warm invitation that reflects the mission of the Five Friendlies as young ambassadors for the Olympic Games.*
- **Question:** What does the Five Friendlies mean when put together?
- **Ground truth:** Welcome to Beijing
- **Baseline:** Welcome to Beijing," offering a warm invitation
- **Our model:** Welcome to Beijing
- **Analysis:** The baseline model provides extraneous information ("offering a warm invitation") in its prediction, leading to a less correct answer. On the other hand, our model was able to correctly identify the answer span, while excluding the extraneous context, possibly due to better domain awareness.

Using the same context, we asked a more ambiguous question.

- **Question:** What does it mean put together?
- **Ground truth:** Ambiguous
- **Baseline:** Welcome to Beijing," offering a warm invitation
- **Our model:** Welcome to Beijing
- **Analysis:** Although this question is ambiguous ("it" could refer to anything), both models produced the same answer spans as the question above. We can infer that the models are attending to other keywords such as "put" and "together" to make their predictions, instead of the subject in the question ("the Five Friendlies"), which might be problematic in some scenarios. To their credit, both models were able to avoid the distraction span ("–Bei Jing Huan Ying Ni–"), perhaps recognizing that a dash is used to separate ideas.

We also compared model predictions on the held-out OD test set. We will focus on two examples where our model disagreed with the baseline.

5.2 Subject Confusion (Test Set)

- **Context:** *In China, however, Singles' Day has become the biggest online shopping day in the world, which was created in 2009 by Alibaba's CEO Daniel Zhang to increase online sales. [. . .] There are sharp and other promotions designed by Alibaba to attract online customers. [. . .] "Within the next five years, we expect China will become the world's largest e-commerce market for imported products," President Michael Evans told reporters on Wednesday.*
- **Question:** What does Alibaba expect to do on Singles' Day?
- **Ground truth:** increase online sales / attract online customers
- **Baseline:** China will become the world's largest e-commerce market for imported products,"
- **Our model:** attract online customers.
- **Analysis:** While the subject of the question is "Alibaba", the baseline model produces an answer about China. The baseline is likely matching "expect" in the question to the only "expect" in the context, which immediately precedes the span about China. On the other hand, our model correctly identifies the subject of the question and produces a correct answer. This means our model weighted the subject of the question ("Alibaba") more importantly than the verb ("expect"), which is reasonable behavior in many scenarios.

5.3 Context Understanding (Test Set)

- **Context:** *We are looking for teachers for our private secondary school in Nigeria. This is a Christian school and we are looking for Christian teachers. [. . .] Those who teach other subjects are also welcome. Applicant 1 Modupe Bvuma I have a master's degree in Managerial Psychology [. . .] Applicant 2 Rachel Moore I'm an Australian and have experience in working with children [. . .] Applicant 3 Mwanyimi Bushabu I've been in Africa for 5 years as a banker. It is advantageous for me to teach French at your school since it's my mother tongue, [. . .] Applicant 4 Freddie Matthews I am to graduate from the University of Nottingham, England. [. . .] Applicant 5 Adelaide White I am an Egyptian living in San Francisco, the USA.*
- **Question:** Supposing the school needs an African to teach French, who would be the best choice?
- **Ground truth:** Ambiguous
- **Baseline:** Christian teachers.
- **Our model:** Applicant 3 Mwanyimi Bushabu
- **Analysis:** The question is ambiguous, since it's asking the model to make a hiring judgement from 5 candidates based on the need for "an African to teach French." The baseline model does not understand the larger context and provides a generic answer, possibly using word matching with "teachers". Our model displays a deeper understanding that we want to select one of the multiple candidates being presented. Given the ambiguity, it produces a very reasonable prediction, since "Applicant 3 Mwanyimi Bushabu" is described both as "in Africa" and as being able "to teach French," fulfilling the question's requirements.

6 Conclusion

In our report, our best-performing model leverages TAPT, domain sampling, and data augmentation to achieve EM/F1 scores of 37.44/51.37 on the development set and EM/F1 scores of 40.12/58.05 on the test set. Key to our improvements was the use of augmented and true OD data in a final fine-tuning step. Domain sampling and TAPT provided notable improvements as well, especially in combination with each other. We have learned that some computationally expensive techniques such as TAPT were not as helpful as simpler techniques, such as an extra fine-tuning step with augmented data. We also learned that performance improvements on the development set do not necessarily lead to improvements on the test set. Therefore, we should be careful not to optimizing strictly for the development set. Future work could pretrain with TAPT for significantly longer, which might lead to better results.[2] In addition, future work could experiment with other data augmentation methods, including noising techniques and back-translation. Taken together, we show that data augmentation, TAPT, and domain sampling are promising techniques for generalizing deep-learning NLP models to out-of-distribution QA data.

[2]We ended pretraining early due to time constraints, even though our loss was decreasing.

References

[1] 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.

[2] Luheng He, Mike Lewis, and Luke Zettlemoyer. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[3] Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems, 2017.

[4] Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics.

[5] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *arXiv preprint arXiv:2004.10964*, 2020.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[7] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[8] Hongyu Li, Xiyuan Zhang, Yibing Liu, Yiming Zhang, Quan Wang, Xiangyang Zhou, Jing Liu, Hua Wu, and Haifeng Wang. D-NET: A pre-training and fine-tuning framework for improving the generalization of machine reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 212–219, Hong Kong, China, November 2019. Association for Computational Linguistics.

[9] Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. Mrqa 2019 shared task: Evaluating generalization in reading comprehension, 2019.

[10] 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*, 2019.

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

[12] Ziang Xie, Sida I. Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y. Ng. Data noising as smoothing in neural network language models, 2017.

[13] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Toward controlled generation of text, 2018.

[14] 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)*, 2019.

[15] George Miller. Wordnet: a lexical database for english. In *Commun. ACM*, 1995.

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

[17] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset, 2017.

[18] Tom Kwiatkowski, Jennimaria Palomaki, 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, and Slav Petrov. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*, 2019.

[19] Amrita Saha, Rahul Aralikatte, Mitesh M. Khapra, and Karthik Sankaranarayanan. Duorc: Towards complex language understanding with paraphrased reading comprehension, 2018.

[20] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations, 2017.

[21] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension, 2017.