Fine-Tuning SQuAD Against Adversarial Examples

Ruchi Bhalani
Department of Computer Science
University of Texas at Austin
ruchi.bhalani@utexas.edu

***Abstract*—In this paper, we trained an *ELECTRA-small* model on the SQuAD dataset, exposing ”dataset artifacts” by testing the model on an adversarial challenge set. After being trained on the original SQuAD and tested on the challenge set, the model achieved an F1 score of 31.3%. In order to increase model robustness on the adversarial dataset, we ”inoculated” the train set using large amounts of adversarial data. We trained the model on datasets comprised of 25, 50, 75, and 100 percent of adversarial samples with the goal of improving performance on the challenge set while maintaining performance on the original dataset. With a train set comprised of 50% adversarial SQuAD samples and 50% original SQuAD samples, we were able to achieve a performance improvement of 46.7% on the F1 score of the challenge set, while retaining performance fairly well on the original (only reducing F1 score on the original SQuAD by less than 6%), although for the train sets containing 75% and 100% adversarial samples, performance on the original suffered greatly. Through analysis, we suspect that including a greater percentage of adversarial samples than original samples in the dataset trains the model to avoid certain helpful heuristics, such as lexical overlap.**

I. Introduction

Large-scale labelled datasets such as the Stanford Question Answering Dataset (SQuAD) and the Stanford Natural Language Inference (SNLI) have been driving forces in recent natural language processing research. In order to evaluate performance on these datasets, error and accuracy is measured on a held-out test set after training the model on an associated dataset. However, it is possible for models to achieve seemingly high performance by recognizing spurious, predictive patterns (”dataset artifacts”) without actually learning the intended behavior. These spurious statistical patterns are something that models can learn to exploit, and lead to poor model generalization and robustness [1] [2]. For example, adding additional sentences which have similar semantics to answer sentences could mislead the model and cause it to output the wrong answers. Therefore, the robustness of this model needs to be further improved.

One proposed method for improving accuracy on adversarial datasets is inoculation by fine-tuning, in which a small amount of adversarial examples are added to the training set in order to ”innoculate” the model against adversarial examples [3]. This fine-tuning has been demonstrated to close more than 60% of the performance gap between the original SQuAD and the Adversarial SQuAD test sets. In this paper, we aim to take this ”innoculation” one step further, testing the performance of the model on both the original and adversarial SQuAD datasets if the adversarial examples make up 25, 50, 75, and even 100 percent of the training data. Our goal is to improve performance on the challenge set, while maintaining performance on the original SQuAD dataset.

II. Methods

A. Model

For our analysis, we have used the *ELECTRA-small* model, which uses improved training methods for the BERT architecture [4]. We implemented model training and evaulation using the HuggingFace `transformers` library[1]. For both initial training and fine-tuning, we trained the model over 3 epochs. Since there are only 3,560 examples in the Adversarial Dataset we are using, and we want to preserve the 80/20 split norm, even in the event of a 100% adversarial train set, this leaves us with 2,848 examples to train on and 712 examples to test on. We want to hold the number of examples in the train and test sets constant so as not to have an unpredictable or unexplainable results on the accuracy and error. Therefore, for this paper, all of the training sets will be comprised of 2,848 QA examples, with varying amounts from the original SQuAD and from the Adversarial SQuAD, respectively.

B. Original SQuAD

SQuAD is a reading comprehension dataset about Wikipedia articles, containing 107,785 question-and-answer sets on specific paragraphs [5]. SQuAD is an attractive QA dataset for exploring dataset artifacts since many have previously argued that a plethora of questions can be answered using heuristics like type-matching [6] [7].

Article: Super Bowl 50
Paragraph: ”*In early 2012, NFL Commissioner Roger Goodell stated that the league planned to make the 50th Super Bowl ”spectacular” and that it would be ”an important game for us as a league”.*”
Question: ”*When did he make the quoted remarks about Super Bowl 50?*”
Answer: In early 2012

Fig. 1. An example from the SQuAD dataset

[1] https://huggingface.co/

This heuristic in use is visible in the example shown in Figure 1. Given the question type (”when”), it follows that the correct answer is a date or time of some sort. The only date in the entire paragraph is ”early 2012”, which is the correct answer. The only competing answer is ”the 50th Superbowl”. This suggests that the model can rely on heuristics to arrive at the answer for a significant number of questions, and achieve a relatively high accuracy score without requiring actual reading comprehension.

One of the HuggingFace benchmarks for an *ELECTRA-small* based QA model trained on SQuAD for 3 epochs is that is should achieve around a 78 exact match score and 86 F1 score. However, since our training set was cut down to 2,848 QA samples in order to account for the total number of adversarial samples we had, we needed to establish a new benchmark. This performance benchmark is documented in Table 1.

Original SQuAD Performance Benchmarks	
Exact Match Score	67.1
F1 Score	76.9

TABLE I

The performance of the original SQuAD model takes a hit, although it is not so significant of a difference from the HuggingFace benchmarks on the full dataset that it is unusable in these experiments. Surprisingly, even while training on only 2 percent of the dataset, the Exact Match Score and F1 Score values remain fairly high.

C. Adversarial SQuAD

To explore the impact of dataset artifacts, we tested the original model against a challenge dataset, namely the adversarial SQuAD dataset created by Jia and Liang [2], specifically the ADDSENT dataset. The dataset was constructed by automatically generating a distracting or misleading sentence related to the input SQuAD paragraph and concatenating it to the end of the original paragraph. These sentences are specifically designed to leave the original answer still correct, while ”distracting” the model with extraneous information. For each sampled example, the adversarial SQuAD dataset contains the original question, as well as up to three human-reviewed adversarial variations, to make up a total of 3,560 questions, one example of which is documented in Figure 2

This adversarial example also prevents the model from using type-matching heuristics to answer questions, since the question ”What is the name of” would search for proper nouns, of which there are 2 in this paragraph following the use of ”age 38”: both *”John Elway”* and *”Jeff Dean”*.

Since we are planning to train on this entire dataset, and we want to maintain an 80/20 test-train split, we partitioned 712 samples for the held-out test set, and used the remaining 2,848 for training. The performance of the model trained on the original SQuAD, and tested on the adversarial SQuAD, is shown in Table 2.

Article: Super Bowl 50
Paragraph: *“Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver’s Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.”*
Question: *”What is the name of the quarterback who was 38 in Super Bowl XXXIII?”*
Original (correct) prediction: John Elway
Prediction under adversary: Jeff Dean

Fig. 2. An example from the Adversarial SQuAD dataset, with the misleading sentence in blue. Figure reproduced from Jia and Liang (2017) [2]

Adversarial SQuAD Performance Benchmarks	
Exact Match Score	23.6
F1 Score	31.3

TABLE II

Compared to the F1 of 76.9% on the original dataset, the model achieves a much lower performance on the adversarial SQuAD. The robustness of this model needs to be improved for it to have better performance on adversarial, misleading examples and potentially rely more on reading comprehension rather than heuristics.

III. RESULTS

One method for making the model more robust to adversarial examples is to ”inoculate” it via fine-tuning on a small sample of adversarial examples [3]. However, we wanted to test the performance of the model if we took this ”inoculation” further, by training the model on increasing proportions of adversarial data. We test the performance using 712, 1,424, 2,136, and 2,848 randomly chosen adversarial samples. Given that the total size of the train set is 2,848 for the purposes of these experiments, these are 25%, 50%, 75%, and 100% of the train set respectively.

Additionally, we wanted to prevent the known issue of the model using further heuristics to train on this adversarial data, such as ignoring the last sentence during training [2] [3]. To do this, we made sure to isolate the adversarial sentence, and then move it from the end of the paragraph to a random location within the input paragraph. With this adversarial paragraph re-shuffling, we are relying on the assumptions that the broken co-reference or collocation links between sentences don’t result in a change to the original answer.

The performance of the model on the original and adversarial SQuAD test sets throughout these experiments is displayed in Figure 3 and Figure 4.

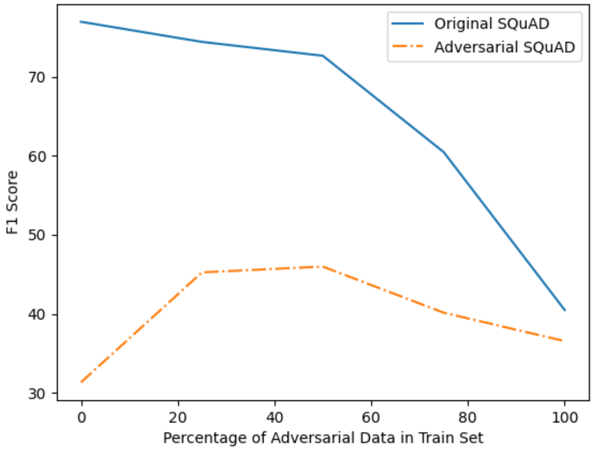

Fig. 3. F1 scores of the model on both original and adversarial test sets. The model partially recovers performance on the challenge set.

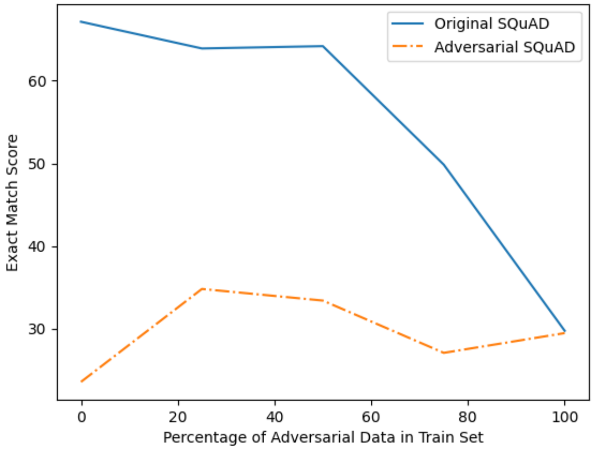

Fig. 4. Exact match scores of the model on both original and adversarial test sets. The performance of both models converge as the percentage of adversarial data in the trainset approaches 100%.

IV. Discussion

In order to compare how training on the adversarial data affected the F1 score on both the original and adversarial test sets, we calculated the percentage difference from the original F1 scores for each of them, and then graphed them in Figure 5.

It appears that training on the adversarial data did have a significant affect on performance on the adversarial test set, while maintaining performance on the original fairly well when using train sets that were comprised of 50% of adversarial data or less. After training on a test set that was 50% original SQuAD data and 50% adversarial SQuAD data, we were able to achieve a 46.7% improvement in the F1 score of the adversarial set, while only suffering a decrease of 5.6% in F1 score on the original dataset. Training on this set resulted in the model closing the gap between the adversarial and original test sets by 32.1%.

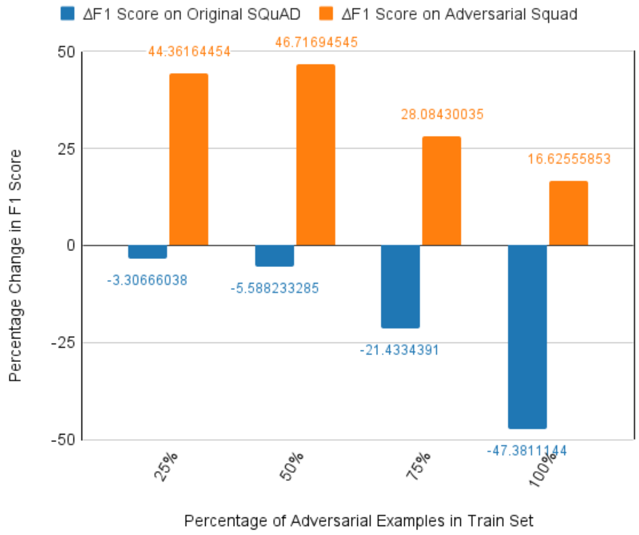

Fig. 5. Percentage difference from F1 scores of the model trained on the original SQuAD, and tested on both the original and challenge sets.

Although the experiments presented in this paper include a higher percentage of the adversarial examples in the train set than the original inoculation methodology demonstrated in Liu et. all's paper [3], this result most closely aligns with the outcome 3 that they describe, in which ”inoculation damages performance on the original test set (regardless of improvement on the challenge test set)”. While the model still performs relatively well on the original dataset, considering the high percentage of adversarial samples included in the train set, it does suffer in performance slightly. This result mirrors the outcomes Liu et. all faced on the Adversarial SQuAD challenge set as well, since both BiDAF and QANet ”results in substantial performance loss on the original SQuAD development set.” Although they attribute this loss in performance to the possibility that the model is exploiting the design of the Adversarial test set and taking advantage of the fact that the misleading sentence is concatenated to the end of the paragraph, we avoided that known issue via our strategy of paragraph shuffling.

Liu et. all also states that outcome 3 may result from a different label distribution between both datasets, or annotation artifacts that exist in one dataset, but not in the other. This ”artifact” may be the adversary itself. One of the commonly used heuristics of the base model was n-gram and lexical overlap [8], however the adversaries have a high degree of overlap with samples from the original dataset (such as those including the original question). When including increased percentages of adversarial examples in the dataset (75%, 100%), we could have been training the model to avoid using this heuristic, which could lead to a decreased performance on the original set.

This is also a possible explanation for why the performances on both datasets converge as the percentage of adversarial samples in the train set approaches 100%, as demonstrated in Figure 3 and Figure 4. At this point, the model may forgo n-gram overlap as a heuristic completely. Additionally, at this point, performance may begin to suffer as a result of the paragraph reshuffling, since co-reference links between sentences are important for true reading comprehension. While the model demonstrated an improved performance on the challenge set when using 25% and 50% of adversarial samples in it's train set, performance suffered on both the challenge and the original sets when the train set was made up of 75% of adversarial samples or more. Here, paragraph reshuffling may be too costly since it risks breaking collocation links and changing the original answer.

V. Conclusion

In this paper, we trained an *ELECTRA-small* model on the SQuAD dataset, exposing ”dataset artifacts” by testing the model on an adversarial challenge set [2]. The challenge set made it evident that adversarial datasets can break SQuAD-trained models by exploiting deficiencies in the original, such as allowing the model to use low-effort heuristics such as type matching, rather than actually achieving reading comprehension. We wanted to increase model robustness by adopting some of Liu et. all's methodology of fine-tuning the model through inoculation by adding small amounts of adversarial samples to the train set [3], however we wanted to take it one step further, and include greater and greater percentages of adversarial samples in the train set.

With a train set comprised of 50% adversarial samples and 50% original samples, we were able to achieve a performance improvement of 46.7% on the F1 score of the challenge set, while retaining performance fairly well on the original (only reducing F1 score on the original SQuAD by less than 6%). Through analysis, we suspect that including a greater percentage of adversarial samples than original samples in the dataset trains the model to avoid certain helpful heuristics, such as lexical overlap.

References

[1] M. Bartolo, T. Thrush, R. Jia, S. Riedel, P. Stenetorp, and D. Kiela, “Improving question answering model robustness with synthetic adversarial data generation,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2021. [Online]. Available: https://doi.org/10.18653%2Fv1%2F2021.emnlp-main.696

[2] R. Jia and P. Liang, “Adversarial examples for evaluating reading comprehension systems,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 2021–2031. [Online]. Available: https://aclanthology.org/D17-1215

[3] N. F. Liu, R. Schwartz, and N. A. Smith, “Inoculation by fine-tuning: A method for analyzing challenge datasets,” *CoRR*, vol. abs/1904.02668, 2019. [Online]. Available: http://arxiv.org/abs/1904.02668

[4] K. Clark, M. Luong, Q. V. Le, and C. D. Manning, “ELECTRA: pre-training text encoders as discriminators rather than generators,” *CoRR*, vol. abs/2003.10555, 2020. [Online]. Available: https://arxiv.org/abs/2003.10555

[5] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “Squad: 100, 000+ questions for machine comprehension of text,” *CoRR*, vol. abs/1606.05250, 2016. [Online]. Available: http://arxiv.org/abs/1606.05250

[6] P. Rajpurkar, R. Jia, and P. Liang, “Know what you don't know: Unanswerable questions for squad,” *CoRR*, vol. abs/1806.03822, 2018. [Online]. Available: http://arxiv.org/abs/1806.03822

[7] D. Weissenborn, G. Wiese, and L. Seiffe, “Making neural QA as simple as possible but not simpler,” in *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 271–280. [Online]. Available: https://aclanthology.org/K17-1028

[8] T. McCoy, E. Pavlick, and T. 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*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 3428–3448. [Online]. Available: https://aclanthology.org/P19-1334