Learning from Paraphrase Adversaries

David Li[*]
Department of Computer Science
University of Maryland,
College Park
dli7319@cs.umd.edu

Md. Ishat-E-Rabban
Department of Computer Science
University of Maryland,
College Park
ier@cs.umd.edu

Tasnim Kabir
Department of Computer Science
University of Maryland,
College Park
tkabir1@cs.umd.edu

Abstract

Despite significant research in adversarial examples, even the best language models today are still vulnerable to various types of textual adversaries. In this paper, we examine the relationship between word-scrambled paraphrase adversaries and other types of textual adversaries. Our experimental results show that distinguishing semantics between word-scrambled paraphrase adversaries during pre-training leads to robustness against other forms paraphrase adversaries in downstream tasks such as question answering without significant penalties in non-adversarial performance. Based on our empirical results, we suggest using an adversarially pre-trained version of Bert for public deployment.

1 Introduction

When trained on natural datasets, deep learning models are vulnerable to adversarial examples. Small perturbations in the input of a model can lead to drastically different outputs. Most research in adversarial examples have focused on images, where the domain of inputs is continuous and adversarial examples are easy to generate using gradient descent. The same phenomenon also occurs in deep learning models for natural language tasks. Even the state-of-the-art natural language processing (NLP) models degrade in performance when presented with adversarial inputs [11, 8].

Unlike adversarial inputs for images, adversarial examples in NLP are more challenging to generate as text lie in a discrete space where small perturbations such as character or word replacements can be very noticeable and can drastically change the semantics of a sentence. As a result, many white-box methods such as [16, 2, 17] have focused on adversarial embeddings as inputs rather than raw text. Several black-box adversaries have been proposed including concatenation adversaries [26, 9], edit adversaries [15], and paraphrase adversaries [30, 6]. However, these adversaries are often task-specific, relying on properties of the task to generate them. For instance, question answering systems may face concatenation adversaries where distracting phrases which are lexically similar to the question are appended to the text. Similarly, when creating paraphrase adversaries, distracting words surrounding incorrect answer-candidates may be added to the question. When posed with adversarially paraphrased questions, state-of-the-art question answering networks such as Bert drop in performance by $25 - 45\%$ [6].

[*] Corresponding Author

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

The current state-of-the art networks in NLP operate in a pre-train and fine-tune paradigm. Large neural networks are pre-trained on a large dataset as a general-purpose language model. The pre-trained network is later fine-tuned using a smaller dataset to a specific task. Under this paradigm, the language representation learned during the pre-training phase transfers to yield superior accuracy and training speed during the fine-tuning for the target task. Given that general language representations can be learned once and applied to several tasks, the question remains as to whether adversarial robustness can also be learned once and applied to several tasks.

In this paper, we examine the relationship between different types of paraphrase adversaries in NLP. We empirically evaluate how training on one type of paraphrase adversaries transfers to robustness against not only different types of paraphrase adversaries but also when performing different tasks.

Our contributions are summarized as follows:

- Proposing adversarial pre-training with paraphrase adversaries to the pre-train/fine-tune setting.
- Empirical evaluation of how pre-training on word-scrambled paraphrases affects robustness against other forms of textual adversaries.
- Empirical evaluation of how robustness against paraphrase adversaries transfers across domains and tasks including question answering, natural language inference, and sentiment analysis.

2 Related Works

Our work builds upon rich literature on adversarial examples in textual machine learning models. In this section, we focus on the core models and adversaries used in our work. We refer interested readers to Zhang *et al.* [29] for a survey on adversarial examples for deep-learning models in natural language processing.

2.1 Pre-training Neural Networks

Pre-training neural networks is a popular technique for reducing the training time to solve similar tasks. During pre-training, a single model, typically transformer-based [25], is trained to do a few specific tasks over a large dataset. After the pre-training procedure is completed, the same model is duplicated and fine-tuned to specific tasks. Despite the discrepancy between pre-training and fine-tuning tasks and datasets, this technique has been very successful in applying neural models to natural language processing (NLP). In NLP, one large model is pre-trained to understand the semantics of a language and then fine-tuned with additional layers for specific language tasks. A single pre-trained model can be used as a starting point for a variety of tasks such as part-of-speech tagging[24], text classification, machine translation[3], and question answering.

Bert [4], by Devlin *et al.* from Google AI, is a pre-trained bidirectional transformer model used for NLP tasks. Devlin *et al.* pretrain Bert at performing two tasks: Masked Language Modelling (Masked LM) and Next Sentence Prediction (NSP). During the masked language modeling task, Bert is fed tokenized sentences with random tokens replaced with a mask token. It is then trained using a cross-entropy loss function to predict the masked tokens. The goal of this pre-training task is to train Bert to understand semantic relationships between different words in a single sentence. To ensure Bert understands relationships between sentences, Bert is trained on a second task, next sentence prediction. For the next sentence prediction task, Bert is fed two sentences and is trained to identify whether the first sentence is likely to preceed the second sentence. Despite Bert being trained as a general-purpose NLP model, it became state-of-the-art at several tasks upon its release. The best models at the Stanford Question Answering Dataset (SQuAD) are currently all based on Bert.

Recent works [11, 8] have shown that even Bert is vulnerable to adversarial examples. However, several methods have also been proposed to improve the robustness of Bert-based models [31, 5, 10]. Zhu *et al.* [31] propose FreeLB, an adversarial training method which both improves the accuracy and robustness of Bert. By leveraging the "free" adversarial training strategies [23], FreeLB adversarially trains Bert during fine-tuning using PGD iterations. However a drawback to their approach is that it is only robust against adversarial word embeddings. Black-box adversaries such as concatenation

Sentence 1: He was born in New York City in East Broadway on October 23.
Sentence 2: He was born on 23 October in New York , East Broadway.
Relationship: Paraphrase

Sentence 1: Revco was subsequently acquired by CVS in 1997 .
Sentence 2: CVS was acquired in 1997 by Revco.
Relationship: Nonparaphrase

Figure 1: Two examples of word-scrambled paraphrase adversaries from the PAWS dataset [30]. Word-swapping leads to both paraphrases and non-paraphrases which are challenging for models trained on other datasets to distinguish.

adversaries and paraphrase adversaries may not necessarily lead to the same type of adversarial word embeddings.

2.2 Paraphrase Adversaries

Due to the nature of natural languages, paraphrase adversaries cannot be generated in the same way as adversarial examples are for image classification. As a result, generating paraphrase adversaries has thus far depended on human generation [27] or human labelling and verification [30]. Moreover, the type of adversarial paraphrasing can depend on downstream task.

Zhang *et al.* [30] from Google AI develop Paraphrase Adversaries from Word Scrambing (PAWS), a dataset of paraphrase adversaries labelled by humans. Unlike existing paraphrase datasets, which use back-translation to generate paraphrases with the same label, PAWS consists of sentence pairs which have a very similar set of words, but may not necessarily be paraphrases. Using Quora Question Pairs and Wikipedia as source texts, they generate paraphrase adversaries using a combination of word-swapping and back-translation. Humans are used to label whether each pair of phrases have the same semantic meaning. In their experiments, they discover that models trained solely on Quora Question Pairs (QQP) perform poorly on PAWS but models trained on both QQP and PAWS perform well on both.

2.3 Application-Specific Textual Adversaries

Jia and Liang [9] are one of the first to evaluate adversarial examples in the context of question answering. Jia and Liang generate adversarial examples by appending adversarial sentences to the question answering dataset SQuAD which are lexically similar to the question but semantically different. Their process consists of generating new sentences by concatenating an altered version of the question along with a fake answer to the source document. In their experiments, they discover that the performance of bidirectional attention flow (BiDAF) models as well as long-short term memory models (LSTMs) degrade when adversarial sentences are appended to the text.

Gan and Ng develop a paraphrased SQuAD dataset [6] which contains two sets of paraphrased questions from the SQuAD 1.1 dataset. The first set contains non-adversarally paraphrased questions generated by a paraphrasing transformer model and verified by humans. The second set contains adversarially paraphrased questions which are manually created by adding context words around wrong-answer candidates into the question. These questions are adversarial as many NLP systems overly rely on string-matching when performing question-answering tasks. They discover that state-of-the art question answering networks such as Bert and DrQA perform moderately worse when tested on non-adversarially paraphrased questions and significantly degrade when tested on adversarially paraphrased questions.

Minervini *et al.* [15] develop edit adversarial examples for natural language inference (NLI). By applying perturbations such as word swapping to text, they create adversarial inputs which maximize the probability of violating first-order logic rules for NLI, shown in Table 4. They also found that adding adversarial regularization to the loss function improves the robustness of various models to first-order logic adversaries.

Paragraph: “The collection of drawings includes over 10,000 British and 2,000 old master works, including works by: Dürer, Giovanni Benedetto Castiglione, [...]. Modern British artists represented in the collection include: Paul Nash, Percy Wyndham Lewis, Eric Gill, Stanley Spencer, John Piper, Graham Sutherland, Lucian Freud and David Hockney. Approximately over 60000 Australian drawings are included in the V&A collection."

Question: “Approximately how many British drawings are included in the V&A collection?"
Correct Answer: over 10,000

Incorrect Prediction: over 60000

Figure 2: The orange text is an adversarial sentence added to a context paragraph from the dataset of Jia and Liang [9]. These adversarial sentences lead to incorrect predictions by state of the art neural networks which may overly rely on string matching. In this example, a model may match the string “drawings are included in the V&A collection" from the question with the adversarial sentence.

Paragraph: “Oxygen was discovered independently by Carl Wilhelm Scheele, in Uppsala, in 1773 or earlier, and Joseph Priestley in Wiltshire, in 1774, [...]"

Original Question: “In what year did Joseph Priestley recognize oxygen?"
Correct Answer: 1774

Adversarial Question: “In what year did Joseph Priestley discover oxygen independently?"
Incorrect Prediction: 1773

Figure 3: An example of a prompt from the SQuAD 1.1 dataset and a corresponding adversarial question from Gan and Ng [6]. Models which overly rely on string matching will produce an incorrect prediction of "1773" rather than the correct answer of "1774."

Table 1: First-Order Logic Rules for NLI
This table presents a list of first-order logic rules for text entailment classification. Minervini and Riedel [15] generate adversarial examples for natural language inference by focusing on perturbations which maximize the probability of a model violating these rules.

NLI Rules
$\top \implies \text{ent}(X_1, X_1)$
$\text{con}(X_1, X_2) \implies \text{con}(X_2, X_1)$
$\text{ent}(X_1, X_2) \implies \neg\,\text{con}(X_2, X_1)$
$\text{neu}(X_1, X_2) \implies \neg\,\text{con}(X_2, X_1)$
$\text{ent}(X_1, X_2) \wedge \text{ent}(X_2, X_3) \implies \text{ent}(X_1, X_3)$

3 Method

To improve the performance on downstream tasks when posed with adversarial paraphrases, we propose adding an additional pre-training step to NLP neural networks. By pre-training Bert on word-scrambled paraphrase adversaries, Bert learns to rely more on word ordering to identify semantic-meaning. This results in less string-matching when performing downstream tasks such as question answering.

3.1 Adversarial Pre-training with PAWS

Our method is to extend the pre-training of Bert with word-scrambled paraphrase advesraries. In this step, we propose using the PAWS dataset [30] to augment pretraining. The PAWS dataset consists of word-scrambled pairs of sentences generated through word swapping and back translation. The pairs of sentences are then labeled manually by humans who identify whether they have the same semantic meaning. Google provides three versions of the PAWS-Wiki dataset: a labeled dataset generated from

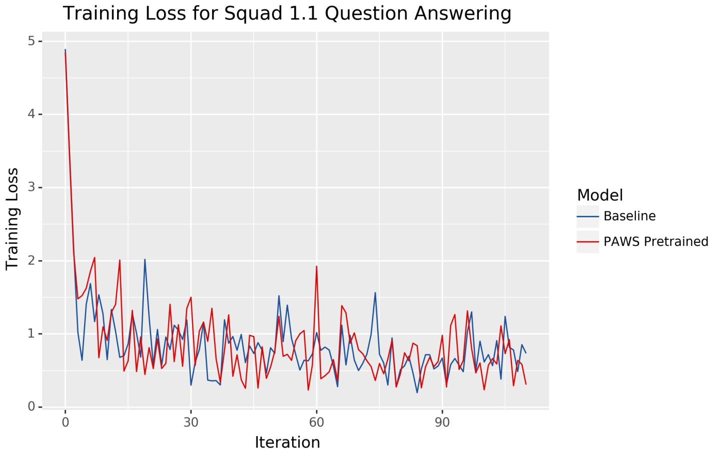

Figure 4: A plot showing the training loss of the baseline Bert model and our PAWS pretrained Bert model on the SQuAD 1.1 question answering dataset. Our additional PAWS classification pre-training does not significantly impact the fine-tuning speed or performance when fine-tuning for question answering.

both word-swapping and back translation, a labeled dataset generated only from word-swapping, and an unlabeled dataset generated from both techniques. For our experiments, we use the labeled dataset generated from both techniques in our training. The pre-training performed in this step is equivalent to the fine-tuning used for text classification. An additional output layer is augmented to Bert. Parameters of the entire model are trained to classify training examples from the PAWS datset. After this additional pre-training step, the output layer is discarded and the resulting Bert model is used as a drop-in replacement for the pre-trained Bert model published by Google.

4 Experiments

In this section we present an overview of our experiments and our results. We test our proposed method by augmenting the pre-training of Bert with the additional task of identifying whether two word-scrambled paraphrases in the PAWS dataset have the same meaning. We then use this neural network as a starting point for fine-tuning for various tasks. All of our experiments are performed using the uncased Bert-base model which has 110M parameters. In each experiment, we compare the performance of fine-tuning on the published Bert model and compare it to fine-tuning on our Bert model with augmented training.

4.1 Question Answering with SQuAD

In our first experiment, we compare the performance of each model at question answering. We fine-tune both the original uncased Bert base model and our augmented model on the training dataset in version 1.1 of the Stanford Question Answering Dataset (SQuAD 1.1) [20]. For evaluation, we test both models on the provided SQuAD 1.1 test dataset as well as adversarial variations of the SQuAD dataset. The first adversarial SQuAD dataset we use is the paraphrased SQuAD questions dataset by Gan *et al.* [6]. This test dataset consists of non-adversarially and adversarially paraphrased questions. The second adversarial dataset we test on is the dataset by Jia *et al.* [9] which focuses on adding adversarial sentences to the context paragraph. Although the question is unmodified, many models give incorrect predictions when the context is augmented with an adversarial answer candidate. Neither the Bert nor our augmented model is trained on adversarial question answering training data so any differences in non-adversarial and adversarial performance are solely due to the additional pre-training on PAWS. During our fine-tuning, we use a batch size of 16 and a sequence length of 128 for both models.

Table 2: Question Answering with SQuAD 1.1 and Adversarial Questions
The Bert model pre-trained on the PAWs classification does slightly worse at non-adversarial question answering but does significantly better than the baseline model at answering adversarially paraphrased questions.

Model	Non-Adversarial Questions (EM, F1) [6]	Adversarial Questions (EM, F1) [6]
Bert	76.177, 84.683	50.000, 55.662
Bert w/ PAWS	74.294, 83.056	58.929, 64.144

Table 3: Question Answering with SQuAD 1.1 and Adversarial Context
The Bert model pre-trained on the PAWs classification does slightly worse on the SQuAD 1.1 test dataset and when tested against the concatenation adversaries of Jia *et al.* [9].

Model	SQuAD (EM, F1)	Adversarial Text (EM, F1) [9]
Bert	78.155, 86.034	57.612, 65.034
Bert w/ PAWS	77.815, 85.824	55.702, 63.366

4.2 Recognizing Textual Entailment

In our second experiment, we evaluate whether augmented pre-training on the labelled PAWS dataset yields additional robustness at textual entailment tasks. We fine-tune both the standard Bert model and our augmented model on the Standford Natural Language Inference (SNLI) corpus[1]. This corpus contains 570k pairs of sentences with human labels identifying the relationship between the two sentences as: entailment, contradiction, or neutral. The goal of the model is to classsify the relationship between two sentences into one of the categories above. At test time, we evaluate both models on the test set of the SNLI corpus and various adversarial datasets. For our adversarial test, we use the dataset of first-order logic adversaries by Minervini and Riedel [15]. This dataset is a paraphrase dataset where words from each sentence are swapped using a paraphrase dictionary until a first-order logic rule is violated. We also test our model against the adversarial dataset of Glocker *et al.* [7] which swap one word in each sentence from the training set of the SNLI corpus. Our results are show in Table 4.

Table 4: Textual Entailment with SNLI
The Bert model pre-trained on PAWS does slightly better than the baseline Bert model at natural language inference and significantly better at first-order logic adversarial examples of [15]. Note that the adversarial dataset of [7] are variations on the training set of the SNLI corpus.

Model	SNLI Test	First-Order Logic Adversary [15]	Adversarial Word Swap [7]
Bert	89.77%	66.97%	92.21%
Bert w/ PAWS	89.92%	71.19%	91.85%

4.3 Sentiment Analysis

In our final experiment, we evaluate how our augmented Bert model performs at sentiment analysis. We use the Large Movie Review Dataset 1.0 published by Maas *et al.* [13] from Stanford. This dataset contains a total of 50,000 movie reviews from IMDB categorized by positive sentiment and negative sentiment. We fine-tune our models to determine whether a movie review has a positive sentiment or a negative sentiment using the provided training split of 25,000 reviews. Then we test on the test split of 25,000 movie reviews. To evaluate adversarial accuracy, we generate a set of edit adversaries using TextFooler by Jin *et al.* [11] using their published Bert model and test against our baseline and augmented Bert models. Our results are shown in Table 5.

Table 5: Sentiment Analysis with IMDB Movie Reviews
Our augmented model performs almost identically compared to the baseline Bert model when fine-tuned for sentiment analysis using the Large Movie Review Dataset [13]. However, our model only performs marginally better when facing the edit adversaries of TextFooler.

Model	Test Accuracy	Adversarial Accuracy [11]
Bert	88.732%	18.029%
Bert w/ PAWS	88.512%	19.974%

5 Discussion

In this section, we discuss the observations and findings associated with our empirical results. Specifically, we discuss how word-scrambled paraphrase adversaries relate to other forms of adversarial examples in natural language processing models, how neural networks are able to transfer adversarial robustness between different tasks, and how the accuracy of neural networks change when pre-trained for adversarial robustness.

5.1 Relationship between Textual Adversarial Examples

Based on our experiments, we find that training on paraphrase adversaries improves performance on other forms of paraphrase adversaries. For instance, our augmented Bert model is trained on word-scrambled paraphrase adversaries where the order of words are swapped to create new sentences. When tested on answering adversarially paraphrased questions, our model yields 8.93% more exact matches compared to the original Bert model as shown in Table 2. Similarly, when classifying textual entailment, our augmented Bert model produces correct answers for an additional 4.22% of the adversarial dataset. These results suggest a relationship between various forms of paraphrase adversaries.

However, our results also suggest that not all textual adversaries have an underlying relationship. Specifically, we see in Table 3 that pre-training on the PAWS dataset does not yield robustness against concatenation adversaries where distracting sentences are added to the input.

5.2 Transfer Learning of Adversarial Robustness

Our experiments determine whether pre-training for adversarial robustness would transfer to downstream tasks. While pre-training is well used in natural language processing in creating word embeddings and semantic understanding, the transferability of adversarial robustness is not as well studied. In our experiments, we train Bert for robustness against adversarial paraphrases. We observe that the augmented Bert does better against certain types of adversaries when fine-tuned for question answering as well as when fine-tuned for natural language inference. Furthermore, our adversarial pre-training does not significantly affect the training performance or training speed during fine-tuning. This suggests that is it possible to achieve transfer learning of adversarial robustness in neural networks.

5.3 Non-adversarial Accuracy Penalty

In image classification, achieving adversarial robustness leads to significant drops in accuracy on clean data. For instance, incorporating projected gradient descent (PGD) training leads to a accuracy drop of 13.3% when classifying CIFAR-10 images [14]. Our experiments show that augmenting pre-training for robustness against adversarial paraphrases does not significantly impact the performance in any downstream tasks. The largest degradation in our experiments was in question answering using the SQuAD 1.1 dataset. There we saw our non-adversarial exact-match accuracy fall from 78.155% to 77.815% and our adversarial context accuracy fall from 57.612% to 55.702%, a loss of 1.91% in accuracy. In natural language inference and in sentiment analysis, our model performed almost identically to the standard Bert model, obtaining an accuracy within 0.3% in each showing that pre-training does not significantly affect non-adversarial accuracy.

5.4 Limitations

While our experiments are intended to identify the relationship and transferability of paraphrase adversaries, there are limitations to our proposed approach and our experimental results. First, we only train on the Bert network as most state-of-the art results are obtained from variations of Bert. While we believe our results would apply to extensions of Bert such as Albert [12], additional experiments will need to be performed to determine if other transformer networks such as GPT-2 [18], XLNet [28], and T5 [19] would benefit in the same way from pre-training. Our method would not apply to task-specific networks such as BiDAF [22]. Secondly, we only test on publicly available datasets of adversarial text from black-box attacks. Additional reserach would need to be performed to see if our results hold for white-box adversaries [21] which generate adversarial examples by taking a gradient through the neural network.

6 Conclusion

In this paper, we propose pre-training with the PAWS dataset to achieve greater adversarial robustness against paraphrase adversaries. We conduct numerous experiments to empirically evaluate the amount of adversarial robustness that can be achieved in the pre-training/fine-tuning paradigm. We test our pre-trained Bert model against various forms of task-specific black-box adversarial examples and observe that our model outperforms the baseline non-adversarially trained model against various paraphrase adversaries while maintaining similar accuracies on clean test data and other forms of textual adversaries.

The results of our experiments are several-fold. First, we find that there is a relationship between word-scrambled paraphrase adversaries and other forms of paraphrased adversaries. Second, we observe that adversarial robustness acquired during pre-training transfers to adversarial robustness in downstream tasks despite additional fine-tuning. Third, adversarial pre-training does not significantly reduce accuracy after fine-tuning with a non-adversarial dataset. These three results suggest that pre-training neural networks using out-of-domain datasets and tasks may lead to adversarial robustness even when adversarial training is impossible using the training data for the target task.

Future work will focus on different types of adversarial training. Training against other forms of textual adversaries such as concatenation adversaries and white-box adversaries may yield better performance against a larger variety of textual adversaries. Additional experiments with different types of adversarial pre-training could better classify the forms of adversarial attacks between different NLP tasks. Whole document paraphrases may prove more challenging to protect against than word-level paraphrases for tasks such as topic modeling.

References

[1] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642. Association for Computational Linguistics, Lisbon, Portugal, Sept. 2015. doi: 10.18653/v1/D15-1075

[2] M. Cheng, J. Yi, H. Zhang, P.-Y. Chen, and C.-J. Hsieh. Seq2Sick: Evaluating the Robustness of Sequence-to-Sequence Models with Adversarial Examples, 2018.

[3] S. Clinchant, K. W. Jung, and V. Nikoulina. On the use of BERT for Neural Machine Translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pp. 108–117. Association for Computational Linguistics, Hong Kong, Nov. 2019. doi: 10.18653/v1/D19-5611

[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805, 2018.

[5] E. Dinan, S. Humeau, B. Chintagunta, and J. Weston. Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack, 2019.

[6] W. C. Gan and H. T. Ng. Improving the Robustness of Question Answering Systems to Question Paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6065–6075. Association for Computational Linguistics, Florence, Italy, July 2019. doi: 10.18653/v1/P19-1610

[7] M. Glockner, V. Shwartz, and Y. Goldberg. Breaking NLI Systems with Sentences that Require Simple Lexical Inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 650–655. Association for Computational Linguistics, Melbourne, Australia, July 2018. doi: 10.18653/v1/P18-2103

[8] Y.-L. Hsieh, M. Cheng, D.-C. Juan, W. Wei, W.-L. Hsu, and C.-J. Hsieh. On the Robustness of Self-Attentive Models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1520–1529. Association for Computational Linguistics, Florence, Italy, July 2019. doi: 10.18653/v1/P19-1147

[9] 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*, pp. 2021–2031. Association for Computational Linguistics, Copenhagen, Denmark, Sept. 2017. doi: 10.18653/v1/D17-1215

[10] H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and T. Zhao. SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization, 2019.

[11] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits. Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment, 2019.

[12] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, 2019.

[13] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150. Association for Computational Linguistics, Portland, Oregon, USA, June 2011.

[14] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*, 2018.

[15] P. Minervini and S. Riedel. Adversarially Regularising Neural NLI Models to Integrate Logical Background Knowledge. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pp. 65–74. Association for Computational Linguistics, Brussels, Belgium, Oct. 2018. doi: 10.18653/v1/K18-1007

[16] T. Miyato, A. M. Dai, and I. Goodfellow. Adversarial Training Methods for Semi-Supervised Text Classification, 2016.

[17] N. Papernot, P. McDaniel, A. Swami, and R. Harang. Crafting adversarial input sequences for recurrent neural networks. In *MILCOM 2016 - 2016 IEEE Military Communications Conference*, pp. 49–54, Nov 2016. doi: 10.1109/MILCOM.2016.7795300

[18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI Blog*, 1(8), 2019.

[19] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, 2019.

[20] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392. Association for Computational Linguistics, Austin, Texas, Nov. 2016. doi: 10.18653/v1/D16-1264

[21] M. Sato, J. Suzuki, H. Shindo, and Y. Matsumoto. Interpretable Adversarial Perturbation in Input Embedding Space for Text. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, pp. 4323–4330. AAAI Press, 2018.

[22] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional Attention Flow for Machine Comprehension, 2016.

[23] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. Adversarial Training for Free!, 2019.

[24] H. Tsai, J. Riesa, M. Johnson, N. Arivazhagan, X. Li, and A. Archer. Small and Practical BERT Models for Sequence Labeling. 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), pp. 3623–3627. Association for Computational Linguistics, Hong Kong, China, Nov. 2019. doi: 10.18653/v1/D19-1374

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc., 2017.

[26] E. Wallace, S. Feng, N. Kandpal, M. Gardner, and S. Singh. Universal Adversarial Triggers for Attacking and Analyzing NLP. 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)*, pp. 2153–2162. Association for Computational Linguistics, Hong Kong, China, Nov. 2019. doi: 10.18653/v1/D19-1221

[27] E. Wallace, P. Rodriguez, S. Feng, I. Yamada, and J. Boyd-Graber. Trick Me If You Can: Human-in-the-loop Generation of Adversarial Question Answering Examples. *Transactions of the Association for Computational Linguistics*, 7(0):387–401, 2019.

[28] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding, 2019.

[29] W. E. Zhang, Q. Z. Sheng, and A. A. F. Alhazmi. Generating Textual Adversarial Examples for Deep Learning Models: A Survey. *CoRR*, abs/1901.06796, 2019.

[30] Y. Zhang, J. Baldridge, and L. He. PAWS: Paraphrase Adversaries from Word Scrambling. *CoRR*, abs/1904.01130, 2019.

[31] C. Zhu, Y. Cheng, Z. Gan, S. Sun, T. Goldstein, and J. Liu. FreeLB: Enhanced Adversarial Training for Language Understanding, 2019.