ROBUSTIFYING LANGUAGE MODELS WITH TEST-TIME ADAPTATION

Noah T. McDermott, Junfeng Yang & Chengzhi Mao
Department of Computer Science
Columbia University
New York, NY 10027, USA
{ntm2128,cm3797}@columbia.edu
junfeng.yang@cs.columbia.edu

ABSTRACT

Large-scale language models achieved state-of-the-art performance over a number of language tasks. However, they fail on adversarial language examples, which are sentences optimized to fool the language models but with similar semantic meanings for humans. While prior work focuses on making the language model robust at training time, retraining for robustness is often unrealistic for large-scale foundation models. Instead, we propose to make the language models robust at test time. By dynamically adapting the input sentence with predictions from masked words, we show that we can reverse many language adversarial attacks. Since our approach does not require any training, it works for novel tasks at test time and can adapt to novel adversarial corruptions. Visualizations and empirical results on two popular sentence classification datasets demonstrate that our method can repair adversarial language attacks over 65% of the time.

1 INTRODUCTION

Large-scale pretrained language models (foundation models) like BERT Devlin et al. (2018) and RoBERTa (Liu et al., 2019) have achieved state of the art performances over a number of language tasks, such as sentiment classification and completion (Gonz'alez-Carvajal & Garrido-Merch'an, 2020). However, these models are vulnerable to adversarial attacks, where modifications to inputs, imperceptible to humans, cause machine learning models to misclassify. These vulnerabilities can pose a security risk when they are used in sensitive and safe-critical applications (Li et al., 2018).

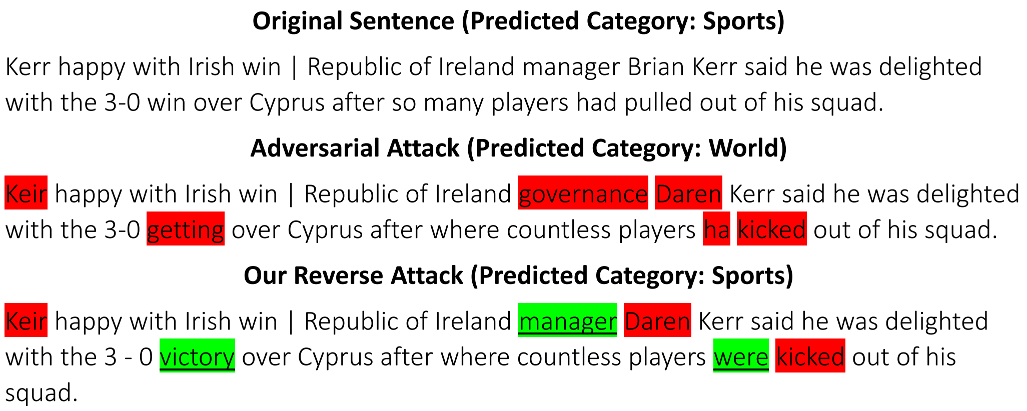
Existing defenses against adversarial attacks have largely focused on training, either through training on pre-generated adversarial samples (Feng et al., 2021; Mao et al., 2022a), or modifying the training objective to be more robust (Hendrycks et al., 2019). However, this is an inherently difficult task, as there are a vast number of different types of attacks on the character, word, and sentence level that the model would have to be able to defend against, and it cannot adapt to new, novel types of attacks (Han et al., 2022). In addition, training-based approaches can only achieve robustness on the task that they have been trained on, but cannot generalize to novel tasks, which is a key feature for modern language foundation models (Han et al., 2022).

Our approach shifts the burden of robustness from training to test time. Our key insight is that masked language modeling is able to capture the structure and constraints of a natural language sentence which are violated in adversarial attacks. We use masked language modelling, a self-supervised task, to find key words in adversarial sentences that lead to bad predictions, and replaces them with normal words.

Our adaptation algorithm for robustness not only can achieve robustness on novel adversarial attacks, but also can achieve robustness in a zero-shot manner without performing robust training on the downstream tasks. Since our method is at test-time, it is also compatible with all existing training-based robust algorithms.

We ran experiments against two of the latest text-based adversarial attacks, PWWS (Ren et al., 2019) and TextFooler (Jin et al., 2020). Empirically, our experiments show that our defense, called

Mask-Defense, was able to reverse 75-80% percent of successful Textfooler attacks, and 65-70% successful PWWS attacks. Additionally, our defense continues to correctly classify sentences that have not been attacked, with around 98% of correctly classified clean sentences remaining correct after the defense is run. The reverse attacked sentences also display high levels of semantic similarity with the original sentences.



Original Sentence (Predicted Category: Sports)

Kerr happy with Irish win | Republic of Ireland manager Brian Kerr said he was delighted with the 3-0 win over Cyprus after so many players had pulled out of his squad.

Adversarial Attack (Predicted Category: World)

Keir happy with Irish win | Republic of Ireland governance Daren Kerr said he was delighted with the 3-0 getting over Cyprus after where countless players ha kicked out of his squad.

Our Reverse Attack (Predicted Category: Sports)

Keir happy with Irish win | Republic of Ireland manager Daren Kerr said he was delighted with the 3 - 0 victory over Cyprus after where countless players were kicked out of his squad.

Figure 1: An example of an adversarial attack successfully reversed by Mask-Defense. We show the original sentence from the Ag's News dataset, and the adversarial attacked sentence using TextFooler (attacked words in red). Reverse Attack shows the adapted sentence using our algorithm, which locates high-loss words and replaces them (green words) with examples generated from a Masked-Language model.

2 Related Work

2.1 Text-based Adversarial Attacks and Defenses

A large number of adversarial attacks have been proposed against language models. Unlike those commonly used in vision or speech, there is no way to imperceptibly add noise to text. So instead, attacks focus on making changes at the character (Ebrahimi et al., 2018), word (Ren et al., 2019) (Jin et al., 2020), or sentence(Wang et al., 2020) level that keep the sentence's meaning intact, but creating a different output.

Various defenses against these text-based attacks have been proposed, but most are training-based approaches, either through adding adversarial attacks in a form of data augmentation, (Feng et al., 2021) or by modifying the training goal to improve robustness.(Hendrycks et al., 2019) While these approaches have been empirically successful, they are difficult to implement because of the computation needed to generate enough adversarial attacks and because of the vast number of possible attacks that the network must be trained to defend against.

2.2 Defenses Leveraging Self-Supervised Learning

Self-supervised learning originated as a way to solve the problem of expensive labeled data. Since complex machine learning models often require large amounts of data, and that data must be manually labeled by humans, it often becomes a bottleneck in development. Self-supervised learning solves that problem by teaching a machine learning model to predict one part of its input using the rest of its input. It can be used as a means to an end, or it can be used for transfer learning with a relatively small amount of labelled data. One of the most successful examples is BERT(Devlin et al., 2018) which is trained on masked-language modelling and next-sentence prediction, both self-supervised methods, but also can achieve strong results on other tasks such as sentiment analysis when fine tuned with small amounts of labelled data.

Mao et al. (2021) developed an algorithm to perform a ”reverse attack” on image classifier models at inference time using self-supervised learning. The algorithm uses contrastive loss as a self-supervised objective, takes the gradient of that loss, and adds a perturbation in the direction of that gradient. By doing this, it is able to restore some of the natural structure in the image, and it achieves strong performance against a number of attacks. Lawhon et al. (2022) shows multiple tasks further improves the robustness. Mao et al. (2022b); Zhang et al. (2022) generalizes this line of work to image segmentation and video perception. This line of work Tsai et al. (2023) all requires multiple steps of test-time optimization which involves the gradient descent. In contrast, our work is tailored for language domain, which replace the words with masked token prediction only requires feed-forward pass, which is much faster to implement.

3 Method: Robustifying Language Models via Masked Word Prediction

Attacks. From the definition given by Jin et al. (2020), given a corpus of sentences $\mathcal{X} = \{X_1, X_2, X_3 \dots\}$ and a set of output labels $\mathcal{Y} = \{y_1, y_2, y_3 \dots\}$, a pre-trained model F performs some mapping $F : \mathcal{X} \longrightarrow \mathcal{Y}$. A valid adversarial attack X_{adv} on $X \in \mathcal{X}$ is some sentence that satisfies the properties

$$F(X) \neq F(X_{adv}) \text{ and } S(X, X_{adv}) \geq \epsilon$$

Where $S(A, B) \longrightarrow [0, 1]$ is a similarity metric that returns a higher value when two sentences $A, B \in \mathcal{X}$ are similar in meaning and structure.

Reverse Attacks. The goal of our algorithm is to reverse the attack at inference by creating some new sentence X_{def} from X_{adv} such that:

$$F(X_{def}) = F(X) \neq F(X_{adv}) \text{ and } S(X_{def}, X_{adv}) \geq \epsilon$$

The main difference between the reverse attack setting and the attack setting is that the defender does not have access to output of F or to the output of any similarity metrics, and this makes it much more challenging than the attack setting. However, the defender can use self-supervised learning to gain information without accessing the outputs, and this strategy is central to our algorithm.

Algorithm 1 Mask-Defense

```
Inputs: Sentence $S$, Classifier $F$, Masked-Language Model $M$, Cosine Similarity Matrix $C$, Model Vocabulary $V_M$, Matrix Vocabulary $V_C$, Threshold $\alpha$, Replacement Parameter $n$.
Outputs: New sentence $\hat{S}$
$\hat{S} \leftarrow S$
for $i \in \{0, \text{len}(S)\}$ do
    Let $\hat{S}_i$ be sentence $S$ with the $i$-th word masked out
    Use $M$ with $\hat{S}_i$ as an input to calculate the masked-language modelling loss $\mathcal{L}_i$ and the softmax output $o_i$ over $V_M$
end for
$j \leftarrow 1, r \leftarrow 0$
while $r < n$ and $j \leq 50$ do
    Let $i$ be the word position that has the $j$-th highest masked-language modelling loss $\mathcal{L}_i$
    for $k \in \{0, 50\}$ do
        Let $w_i$ be the word in $S$ at position $i$
        Let $o_i^k$ be the word in the vocabulary with the $k$-th highest softmax output in $o_i$
        if $\boldsymbol{C}(w^i, o_i^k) \geq \mu(\boldsymbol{C}) + \alpha\sigma(\boldsymbol{C})$ and $w_i \in V_C$ and $o_i^k \in V_C$ then
            Replace $w_i$ with $o_i^k$ in $\hat{S}$
            $r = r + 1$
            break
        end if
    end for
    $j = j + 1$
end while
```

The self-supervised task we choose to optimize for is called masked-language modelling. Masked language modelling consists of taking a sentence, masking some of the words, and having the model choose the best words that fit in its vocabulary. The loss is taken as a softmax over the all the words in the vocabulary.

The algorithm has two hyperparameters that can be adjusted depending on how the user wants to balance success rate and similarity. These are n, the total number of word replacements, and α, the minimum word similarity score. Replacing a fraction of words in a sentence instead of a fixed number was considered for longer sentences, but experiments showed a diminishing effect.

Step 1: Word Importance Ranking We begin this algorithm by ranking the words in the sentence by importance. However, we do not have access to the classifier model's outputs, so we instead use masked-language modelling loss as a measure of importance. For each word w in some sentence X, we calculate the importance I_w as:

$$I_w = L(F(X_w))$$

Where X_w is the sentence X with word w masked, and L is the cross-entropy loss taken over the softmax of the entire vocabulary. The justification for using this score is simple: A masked word that only has a few potential candidates will end having a low importance, such as ”I saw [Mask] big rabbit.”. Clearly ”a” is one of the only words that would fit here. On the other hand, ”I saw a big [Mask].” could have many valid words and therefore will have a high importance and will be a good candidate for substitution. While this approach may not always be accurate, such as in the case of an unexpected word shows up even in when there are few valid word candidates, our experiments show that on average it is a good measure of importance when there is no access to the classifier model.

This is the most computationally complex part of the algorithm, as I_w must be calculated for every word $w \in X$ and sorted. In addition, to calculating the losses, the highest 50 logits l_w (outputs from before the softmax layer) are also saved. Word replacement is then applied from highest to lowest importance.

Step 2: Word Replacement Next, begin the process of replacing words in the sentence with MLM predictions using the logits. However, MLM predictions only capture context and not meaning, for example, an MLM model may predict either ”good” or ”bad” for the sentence ”Today was a [Mask] day.”, so we introduce an additional model to check word meaning. We use word embeddings from Mrkšić et al. (2016), which were designed to capture synonymy.

To determine whether two words are similar, we build a cosine similarity matrix $\boldsymbol{C}$ over the entire embeddings, and take the distance between the original and new word. The threshold to determine whether a new word n is similar to some original word o is:

$$\boldsymbol{C}_{o,n} \geq \mu(\boldsymbol{C}) + \alpha\sigma(\boldsymbol{C})$$

Where α is the similarity parameter, and μ and σ are the mean and standard deviation across all elements in the matrix. The algorithm considers each MLM word candidate in order of likelihood, and chooses the first one with a similarity score above the threshold. The vocabulary of the MLM model and the embeddings may differ, so any word that is in the MLM vocabulary but not the embeddings' vocabulary is not considered. If no successful candidates can be found in the top 50 predictions, or the MLM model predicts the original word, then that word is not replaced. The algorithm terminates when n words in total are replaced.

4 EXPERIMENTS

We evaluate this defense on two state-of-the-art attacks on different text classification tasks. To generate the adversarial text, we run the specified attack against the model until we have 1000 successful examples, and then we run our defenses against each. Unsuccessful attacks were not considered. In addition, we also ran the attack against 1000 clean sentences from both datasets to ensure that the defense does not change the output on the clean sentences.

Datasets The two datasets used in this experiment are AG's News and Yelp Polarity. AG's news is a sentence level classification dataset of the title and description of news stories into four categories: Word, Sports, Business, and Science/Technology. Yelp is a document-level sentiment classification dataset of reviews of restaurants, businesses, etc. Reviews with 1 or 2 stars are considered negative,

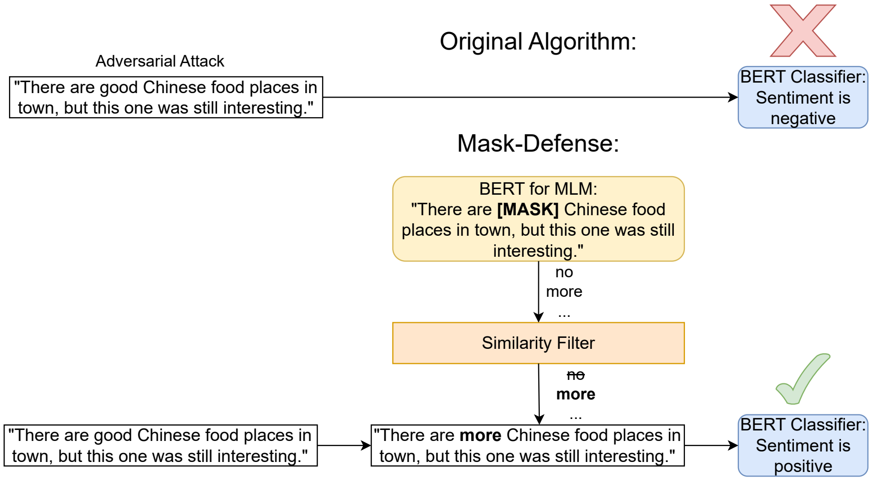

Original Algorithm:
Adversarial Attack
"There are good Chinese food places in town, but this one was still interesting."
BERT Classifier: Sentiment is negative
Mask-Defense:
BERT for MLM: "There are [MASK] Chinese food places in town, but this one was still interesting."
no
more
...
Similarity Filter
~~no~~
more
...
"There are good Chinese food places in town, but this one was still interesting."
"There are more Chinese food places in town, but this one was still interesting."
BERT Classifier: Sentiment is positive

Figure 2: An overview of the Mask-Defense algorithm. A masked-language model generates new words, which are filtered by semantic similarity, to modify input sentences to a language model at test time.

Dataset	Attack Method	Clean Classification Unsuccessful	Attack Unsuccessful	Attack Successful
Ag's News	Textfooler	5.6%	19.8%	74.6%
	PWWS	5.1%	41.2%	53.1%
Yelp	Textfooler	3.2%	5.6%	91.1%
	PWWS	3.2%	5.7%	91.2%

Table 1: Analysis of Attack Efficiency. The classifier models used in this experiment had a high rate of success againt clean sentences, but were also able to be defeated by adversarial attacks.

while reviews with 4 or 5 stars are considered positive. Classification, attacks, and defenses were done by using the entire document as an input, as opposed to individual sentences of the review.

Models. The model being attacked are is BERT (Devlin et al., 2018) a common baseline in the NLP world that has been very successful in both self-supervised and downstream tasks. BERT is also used to calculate the masked-language-modelling loss in the attack. For classification, fine-tuned models from the Textattack package were used, which used hyperparameter optimization to get the best results on Yelp [1] and Ag's News [2]. The standard bert-base-uncased model was used to generate MLM tokens.

Attacks and Defense. The two attacks the algorithm defends against are PWWS (Ren et al., 2019) and TextFooler (Jin et al., 2020). PWWS is an early baseline for NLP adversarial attacks, which performs word replacement combining word saliency and classification probability. Textfooler searches for similar words in embeddings, and uses part-of-speech tagging and sentence encoders to ensure similarity to the original sentence. Both attacks were implemented with Textattack, a python package for adversarial attacks, and were run with default hyperparameters.[3].

The defense was run with hyperparameters of $\alpha = 2$ and $n = 3$, meaning that words must have a similarity score of two standard deviations above the mean to be replaced, and at most three words in one input may be replaced. Only successfully classified sentences were attacked, and the defense was only run on successfully attacked sentences and clean examples.

Results We show the model accuracy under popular attack in Table 1. Before the adversarial attack, the prediction failure rate is 3-5%. The attack will only focus on fool the examples that are successfully classified. After applying adversarial attack, it can further subvert 53%-91% of the samples in

[1] https://huggingface.co/textattack/bert-base-uncased-yelp-polarity

[2] https://huggingface.co/textattack/bert-base-uncased-ag-news

[3] https://github.com/QData/TextAttack

		Clean		TextFooler		PWWS	
Dataset	Defense	Accuracy	Similarity	Accuracy	Similarity	Accuracy	Similarity
Ag's News	Baseline	1	n/a	0	n/a	0	n/a
	Mask-Defense	**0.982**	**0.948**	**0.790**	**0.938**	**0.688**	**0.936**
Yelp	Baseline	1	n/a	0	n/a	0	n/a
	Mask-Defense	**0.987**	**0.950**	**0.756**	**0.958**	**0.657**	**0.952**

Table 2: Experimental Results. 75-80% of successful Textfooler attacks and 65-70% of successful PWWS attacks were reversed by Mask-Defense. 98-99% of correctly classified clean sentences remained correct after Mask-Defense. New sentences created by Mask-Defense had a very high similarity to the original ones.

		Average Loss Before Reverse	
Dataset	Outcome	TextFooler	PWWS
Ag's News	Success	1.28	1.58
	Failure	2.24	2.75
Yelp	Success	1.47	2.56
	Failure	2.69	3.58

Table 3: Analysis of failure. Attacks that were successfully reversed had a much smaller cross-entropy loss from the classifier model, showing that mask-defense performed much better on attacks that were on the edge of the decision boundary

the total test set. The established attack can fool the language classifier. In the following experiment, we will focus on correcting the examples that are subverted by the attacker.

We show the robust accuracy of our defense algorithm in Table 2. Mask-Defense is able to achieve a very high success rate with very few modifications to the text. Interestingly, the average sentence length in Yelp (179.18 words) was over three times that of Ag's News (53.17) yet the results were very similar even though in both cases only three words were allowed to be replaced. This suggests that the replacement of a few words with high MLM loss was enough to reverse the adversarial attack regardless of sentence length. Even though there was no sentence-level bound on similarity, all defenses achieve a similarity score of 0.93 or higher. Therefore, it can be assumed that switching a small amount of words in a sentence with very similar replacements is a good proxy for sentence level similarity. Another interesting observation is the fact that PWWS attacks were more difficult to reverse despite the fact that PWWS is an older and less sophisticated algorithm. This may be due to the tighter constraints on syntactic and semantic similarity that Textfooler requires. Finally, we also see an almost perfect success rate on the clean sentences, which shows that Mask-Defense does not negatively impact sentences which have not been attacked.

Analysis. Table 3 provides some insight on why mask-defense failed in some instances. For all attack methods on all datasets, attacks that were successfully reversed had a much lower cross-entropy loss from the classifier to begin with. This highlights how the defense setting is significantly more difficult than the attack setting; while attack models like TextFooler can continue to change words until the label changes, Mask-Defense has no access to the output labels and can only change a pre-defined number of words. For attacks that are right on the decision boundary, this approach works well, but for more severe or perturbed examples, it falls short. A possible improvement to Mask-Defense would be to quantify how perturbed some sentence is, and only change certain words.

We use sentence similarity to measure how close our generated sentence to the original sentence. The similarity is calculated by using a universal sentence encoder, a technique pioneered in Cer et al. (2018). We use it to encode the adversarial sentence and the new sentence into a 348-dimension space, and use their cosine similarity as a measurement of similarity. The exact pretrained model used in our experiments experiments can be found here[4]. In Table 2, we show that our method can fix the adversarial sentence by making them more similar to the original clean sentence.

[4] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

5 Conclusion

We present a new algorithm that can use the underlying knowledge from masked-language-modelling to reverse state of the art textual adversarial attacks and restore ground truth. The experiments show that this algorithm was able to achieve noteworthy results even with very strict limits on the changes it could make to the sentence, and that defenses used at test time in language models can be a powerful way to increase robustness.

6 Acknowledgement

This research is based on work partially supported by the NSF NRI Award #2132519, a GE/DARPA grant, a CAIT grant, and gifts from JP Morgan, DiDi, and Accenture.

References

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder. *CoRR*, abs/1803.11175, 2018. URL `http://arxiv.org/abs/1803.11175`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL `http://arxiv.org/abs/1810.04805`. cite arxiv:1810.04805Comment: 13 pages.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 31–36, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2006. URL `https://aclanthology.org/P18-2006`.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 968–988, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.84. URL `https://aclanthology.org/2021.findings-acl.84`.

Santiago Gonz'alez-Carvajal and Eduardo C. Garrido-Merch'an. Comparing bert against traditional machine learning text classification. *ArXiv*, abs/2005.13012, 2020.

Xu Han, Ying Zhang, and Wei Wang. Text adversarial attacks and defenses: Issues, taxonomy, and perspectives. *Security and Communication Networks*, 2022:1–25, 04 2022. doi: 10.1155/2022/6458488.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CoRR*, abs/1907.07174, 2019. URL `http://arxiv.org/abs/1907.07174`.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8018–8025, Apr. 2020. doi: 10.1609/aaai.v34i05.6311. URL `https://ojs.aaai.org/index.php/AAAI/article/view/6311`.

Matthew Lawhon, Chengzhi Mao, and Junfeng Yang. Using multiple self-supervised tasks improves model robustness. *arXiv preprint arXiv:2204.03714*, 2022.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. 12 2018. doi: 10.14722/ndss.2019.23138.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019. URL `https://arxiv.org/abs/1907.11692`.

Chengzhi Mao, Mia Chiquier, Hao Wang, Junfeng Yang, and Carl Vondrick. Adversarial attacks are reversible with natural supervision. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 641–651, 2021. doi: 10.1109/ICCV48922.2021.00070.

Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding zero-shot adversarial robustness for large-scale models. *arXiv preprint arXiv:2212.07016*, 2022a.

Chengzhi Mao, Lingyu Zhang, Abhishek Joshi, Junfeng Yang, Hao Wang, and Carl Vondrick. Robust perception through equivariance. *arXiv preprint arXiv:2212.06079*, 2022b.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–148, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1018. URL https://aclanthology.org/N16-1018.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1085–1097, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1103. URL https://aclanthology.org/P19-1103.

Yun-Yun Tsai, Chengzhi Mao, Yow-Kuan Lin, and Junfeng Yang. Self-supervised convolutional visual prompts, 2023.

Boxin Wang, Hengzhi Pei, Boyuan Pan, Qian Chen, Shuohang Wang, and Bo Li. T3: Tree-autoencoder constrained adversarial text generation for targeted attack. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6134–6150, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.495. URL https://aclanthology.org/2020.emnlp-main.495.

Lingyu Zhang, Chengzhi Mao, Junfeng Yang, and Carl Vondrick. Adversarially robust video perception by seeing motion. *arXiv preprint arXiv:2212.07815*, 2022.