A Multi-Faceted Reward for Adversarial Attacks on Text Classifiers

Tom Roth[1,2], Inigo Jauregi Unanue[1], Alsharif Abuadbba[2], Massimo Piccardi[1]
[1]University of Technology Sydney, NSW, Australia
[2]CSIRO's Data61, Sydney, Australia
thomas.p.roth@student.uts.edu.au

Abstract

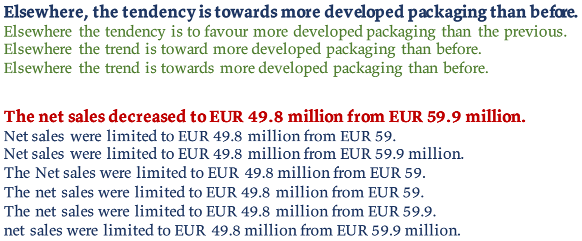
Text classifiers are vulnerable to adversarial examples: originally correctly-classified examples transformed to be classified incorrectly, while also meeting some constraints. The dominant approach to creating them—combinatorial optimisation—is effective, but slow and limited in its transformations. An alternate approach is to create adversarial examples by fine-tuning a pre-trained model, as is commonly done for similar text-to-text tasks. This approach would be much quicker and more expressive, but is unexplored.

In this work we successfully fine-tune a pre-trained encoder-decoder paraphrase model to generate a diverse range of coherent adversarial examples. We train using a simple policy gradient algorithm and design a multi-faceted reward function that solves the task while enforcing constraints and avoiding many "reward hacking" failure cases. We empirically show on two sentiment analysis datasets that our model has a higher success rate than the untrained paraphrase model, and is much more effective than comparable combinatorial optimisation attacks. Finally we show how certain design choices affect the generated examples and discuss the strengths and weaknesses of the approach.

1 Introduction

Adversarial attacks cause a *victim model*—an attacked machine learning model—to malfunction in some specific way. These attacks occur across domains, pose a real-world security threat[1] and are becoming well-studied (Biggio and Roli, 2018; Zhang et al., 2020).

In this paper we train a model to perform adversarial attacks on a text classifier. We avoid the common combinatorial-optimisation approach and instead attempt to directly train a generative model. This is a harder task but, if successful, will be quicker and more powerful.



Elsewhere, the tendency is towards more developed packaging than before.
Elsewhere the tendency is to favour more developed packaging than the previous.
Elsewhere the trend is toward more developed packaging than before.
Elsewhere the trend is towards more developed packaging than before.

The net sales decreased to EUR 49.8 million from EUR 59.9 million.
Net sales were limited to EUR 49.8 million from EUR 59.
Net sales were limited to EUR 49.8 million from EUR 59.9 million.
The Net sales were limited to EUR 49.8 million from EUR 59.
The net sales were limited to EUR 49.8 million from EUR 59.
The net sales were limited to EUR 49.8 million from EUR 59.9.
net sales were limited to EUR 49.8 million from EUR 59.9 million.

Figure 1: Examples of our successful adversarial attack against a sentiment classifier. On top, the adversarial examples flip sentiment from the original neutral (blue) to positive (green), and on bottom, sentiment goes from the original negative (red) to neutral (blue).

2 Proposed Approach

The goal is to fine-tune a vanilla pre-trained paraphrase model on a dataset so that it learns to generate adversarial examples—examples that change the predicted label of a victim model, while also keeping all constraints.

We train for a number of epochs. In each training epoch we generate one paraphrase per original example and collate these into batches of training data. We use the batches to calculate a loss function, which follows the REINFORCE with baseline algorithm with an additional KL divergence term. The loss function uses a reward function and a baseline. These use a set of constraints to determine if the generated text is valid; examples that fail receive zero reward. Many of these constraints use pre-trained models. Figure 2 shows the overall setup.

We evaluate model performance before training and also after each training epoch. During evaluation we generate a set of paraphrases per original example and calculate the attack success rate of the model across the dataset. We also update the reward baseline with the average reward across the

[1] For example, (Wallace et al., 2020) used adversarial examples to induce Google Translate to produce vulgar outputs, word flips, and dropped sentences.

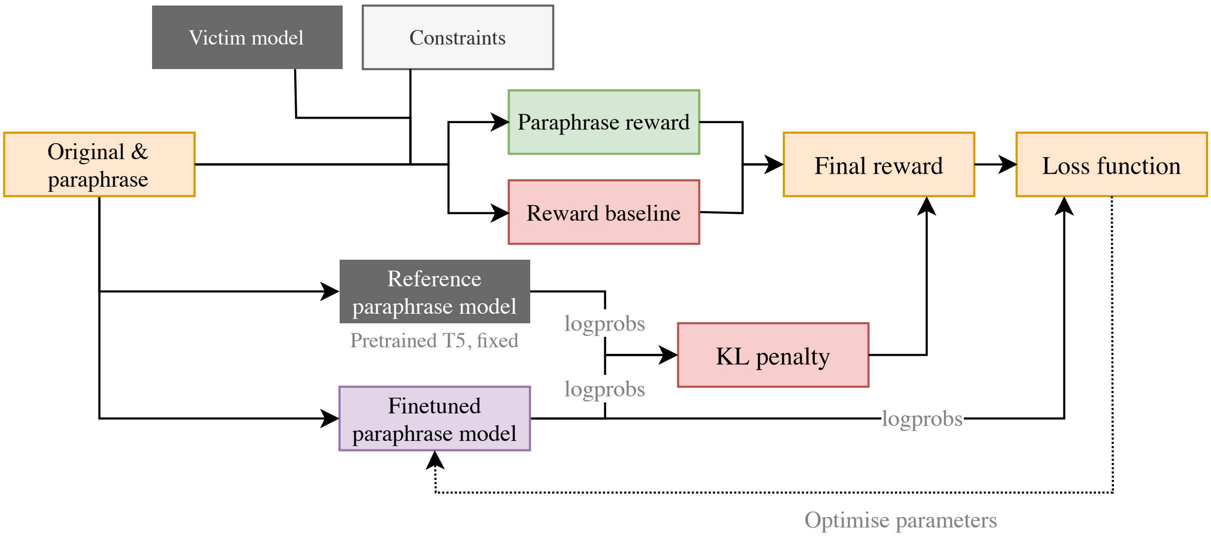


Figure 2: Finetuning the paraphrase model. As input, we use a batch of *(original, paraphrase)* pairs (Section **??**). We update parameters using the REINFORCE with baseline algorithm, regularised with a KL divergence penalty (Section **??**). We describe the reward function in Section **??** and constraints in Section **??**. The reward baseline is updated at the end of each epoch (Section **??**). We accumulate gradients to get a larger effective batch size.

set. We stop training once test set performance drops below a threshold or after a maximum number of epochs.

3 Results

Design choices will affect the attack success rate of the trained model. We tested the effect of two design choices we thought important. The first was the decoding sampling temperature during training, which controls the exploration of the agent. The second was the evaluation decoding method, which affects diversity and quality of the generated candidate set. We kept constant other parameters and measured the effect of these two.

3.1 Impact of training on attack success rate

Results. We find that the training procedure improves the attack success rate across all training conditions. All improvements are statistically significant ($p < 0.01$) according to a bootstrap test, as recommended by Dror et al. (2018). We found no difference between the two training decoding temperature values ($\tau = 0.85$ and $\tau = 1.15$), but that the decoding method had an effect.

3.2 Comparison with token-modification adversarial attacks

Results. For a fixed computational budget, the trained model has a much higher attack success rate than token-modification attacks; in fact, its attack success rate is similar to the most computationally expensive token-modification attack. Moreover the trained model generates many adversarial examples per original, which the token-modification cannot easily do.

4 Conclusion

In this paper we fine-tuned a paraphrase transformer to instead generate adversarial paraphrases. We designed a reward function that encourages victim model degradation while punishing constraint violations. The trained model produces more adversarial examples than the untrained model. It is also much more efficient than comparable token-modification attacks, and its adversarial examples are human-preferred. We identified a good evaluation decoding method for the task: diverse beam search, with a moderate number of beam groups. We also analysed behaviours of the different decoding methods and the training procedure.

References

Battista Biggio and Fabio Roli. 2018. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition*, 84:317–331.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the*

Association for Computational Linguistics (Volume 1: Long Papers), pages 1383–1392. Association for Computational Linguistics.

Eric Wallace, Mitchell Stern, and Dawn Song. 2020. Imitation attacks and defenses for black-box machine translation systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5531–5546, Online. Association for Computational Linguistics.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial Attacks on Deep-learning Models in Natural Language Processing. *ACM Transactions on Intelligent Systems and Technology*, 11(3):1–41.