CIS 530 Milestone 3 Report - First Extension

Alexander Feng, Benedict Florance Arockiaraj, Jianxiong Cai, Xiaoyu Cheng

December 10, 2021

1 Motivation

Adversarial attacks can even cause state-of-the-art models to go haywire and expose the model vulnerabilities. One class of attacks called Universal Adversarial Attacks [1] generates trigger sequence that can be applied to any input sent to classifier to obtain a desired output. Although such 'universal' attacks are more serious, we observe that these methods often produce meaningless and ungrammatical text like 'zoning tapping fiennes" [1]. Even an ordinary human wouldn't understand such nonsensical text, and hence this calls for producing meaningful trigger sequences that can still confuse the intended network. In our first extension, we propose a logic for generating sensible triggers for adversarial attacks. The slides for our draft presentation can be found at here

2 Algorithm

For the first extension to generate sensible triggers, we choose the task of sentiment analysis, the SST dataset and a trigger sequence length of 3 to compare results with strong baselines like [1] that report results with similar setup.

Our algorithm consists of the following components:

1. **Trigger Generation Algorithm:** In terms of trigger generation, we use the algorithm specified in [1] without any changes. The algorithm generates *num_candidates* choices for each index in the trigger sequence, where *num_candidates* is a hyperparameter. We use $num\_candidates = 100$.

2. **POS Filtering:** Once we have the generated candidates for each index in the trigger, we only filter the tokens that match commonly occurring POS patterns of length 3 like `["ADV","ADJ","NOUN"]`, `["PRON", "VERB", "PRON"]`, `["ADV", "VERB", "PRON"]`, `["NOUN", "VERB", "ADJ"]`, `["VERB", "PRON", "VERB"]`, `["VERB", "PRON", "ADJ"]`, `["VERB", "PRON", "NOUN"]`. We use NLTK library to get the POS tags. The tags are from the universal tagset.

3. **Beam Search:** We progressively find the trigger words token by token. Constraining the algorithm to choose only the best filtered token limits our desire to generate grammatical sequences, as it would give non-sensical jargons that facilitate the best attack. Thus, we use beam search to progressively generate tokens. We use a beam size of 5.

4. **Modified Loss Function:** We use scores from the GPT-2 language model to filter out ungrammatical sequences. We do this by using a modified loss function that weighs (with hyperparameters λ and β) both the model loss and the perplexity of the trigger sequence from the language model. Contrary to the common meaning of loss, we want to maximize our loss that intends to maximize the classifier loss (cross-entropy loss) and minimize the perplexity of the generated sequence. We use a λ of -0.00005 and β of 5. Observing that the classifier loss was in the range 0-5 and perplexity in the range 0-100,000, we convert the perplexity values to the similar scale of classifier loss.

$$loss = classifier\_loss + \lambda * perplexity + \beta \tag{1}$$

3 Empirical Evaluation

The bolded experiment is our proposed approach for the extension. The following task has an accuracy of 0.909909 without triggers. For each experiment, we present the sample trigger that gives the least accuracy on our task.

Ablation Experiments	**Sample Triggers**	**Accuracy**
Random Attack w/o Beam	uncreative grow miserably	0.146396
UAT Attack w/o Beam	boring forges faulty	0.130630
Random Attack + POS + Beam	rapidly inadequate spaceship	0.227477
Random Attack + Perplexity + Beam	save, crummy, slob	0.220720
Random Attack + POS + Perplexity + Beam	next worthless misbegotten	0.090090
UAT Attack + POS + Perplexity + Beam	**uselessly idiotic teleprompter, irredeemably disgusting garbage**	**0.040540**

Table 1: Flipping positive to negative

The following task has an accuracy of 0.892523 without triggers. For each experiment, we present the sample trigger that gives the least accuracy on our task.

Ablation Experiments	**Sample Triggers**	**Accuracy**
Random Attack w/o Beam	succeeds absorbing courageousness	0.212616
UAT Attack w/o Beam	powerfully edifying restored	0.158878
Random Attack + POS + Beam	so enlightening companionship	0.348130
Random Attack + POS + Perplexity + Beam	providing exuberant vitality	0.271028
UAT Attack + POS + Perplexity + Beam	**fearlessly captivating vulnerability, wondrously radiant vitality**	**0.121495**

Table 2: Flipping negative to positive

References

[1] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer 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)*, pages 2153–2162, Hong Kong, China, Nov. 2019. Association for Computational Linguistics.