Midterm Report: Combating RIPPLe

Shayan Jalalipour
Portland State University
shayan2@pdx.edu

Santiago Tobon
Portland State University
stobon@pdx.edu

Abstract

A midterm update on the planned methods and experiments for analysing and combating weight poisoning attacks (the RIPPLe method introduced in *Weight Poisoning Attacks on Pre-trained Models* (Kurita et al., 2020)

1 Formalizing the Research Problem

The recent paper "Weight Poisoning Attacks on Pre-trained Models" sheds light on potential security threats of using and fine tuning pre-trained models. (Kurita et al., 2020) The paper introduces a "Weight poisoning" methodology that creates backdoors in pre-trained models which can allow bad actors to use specific input cues to change the models' outputs. While the paper did a good job explaining the threat, it was quite open-ended in how to combat such a problem. Our goal is to stress test the ability of the weight poisoning attacks with the intention of finding if, how, and when such attacks can be circumvented.

2 Datasets

Following the paper, we are using the same datasets for validating weight poisoning attacks on three different text classification tasks: Sentiment classification, toxicity detection, and spam detection. For the scope of this project we are going to focus on the sentiment classification and if we have the time we will evaluate the other two text classifications.

2.1 For fine tuning

- Stanford Sentiment Treebank (SST-2) (sentiment)
- OffensEval (Toxicity)
- Enron (Spam)

2.2 Proxy datasets for poisoning

For sentiment analysis:

- IMDb
- Yelp
- Amazon Reviews

For toxicity detection:

- Jigsaw 2018
- Twitter

For spam detection:

- Lingspam

2.3 Example Data

Some example data taken from a couple of the datasets.

Stanford Sentiment Treebank: "demonstrates that the director of such hollywood blockbusters as patriot games can still turn out a small , personal film with an emotional wallop . 1"

IMDb (Maas et al., 2011): "A solid, if unremarkable film. Matthau, as Einstein, was wonderful. My favorite part, and the only thing that would make me go out of my way to see this again, was the wonderful scene with the physicists playing badmitton, I loved the sweaters and the conversation while they waited for Robbins to retrieve the birdie. 1"

2.4 Distribution & collection of the Dataset

For the sentiment datasets, there are two classes. Either 0 for negative sentiment or 1 for positive sentiment. There are an equal number of positive and negative reviews in the datasets. The Amazon dataset was constructed by selecting product reviews for books, DVDs, electronics and kitchen

appliances on Amazon. The IMDb dataset was constructed by collecting 50,00 reviews from IMDb where there weren't more than 30 reviews for a single movie. The Yelp dataset was obtained from the Yelp Dataset Challenge in 2015.

3 Methodology

We will be taking on the role of an end user, fine tuning a weight poisoned model. Operating under the assumption that the model is poisoned, we will attempt to take steps that mitigate potential backdoors. The following list will outline our experimental steps:

1. Fine tune a model on an on-domain dataset.
2. Validate effects of weight poisoning.
3. Fine tune again, each time applying a different potential countermeasure.
    - Change and/or retune latent representations.
    - Hyper parameter exploration.
    - Removing high LFR (Label Flip Rate) words.
4. Measure effects of LFR.

First we will compare the effects each counter measure has on overall accuracy and training time as well as how it affected the weight poisoning (the LFR metric). Then we will repeat the process and gather results for combinations of countermeasures.

Initially, these tests will be run on models tuned to the domains they were poisoned on. We will then (time permitting) repeat the above steps for models poisoned across domains.

It is important to note that the weight poisoning and tuning process will be a replication of those used in the paper. Our own contribution will be potential novel approaches at circumventing the weight poisoning attacks.

References

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2793–2806, Online. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.