ATTACKS ON VICTIM MODEL! A DEFENSE STRATEGY

INTRODUCTION

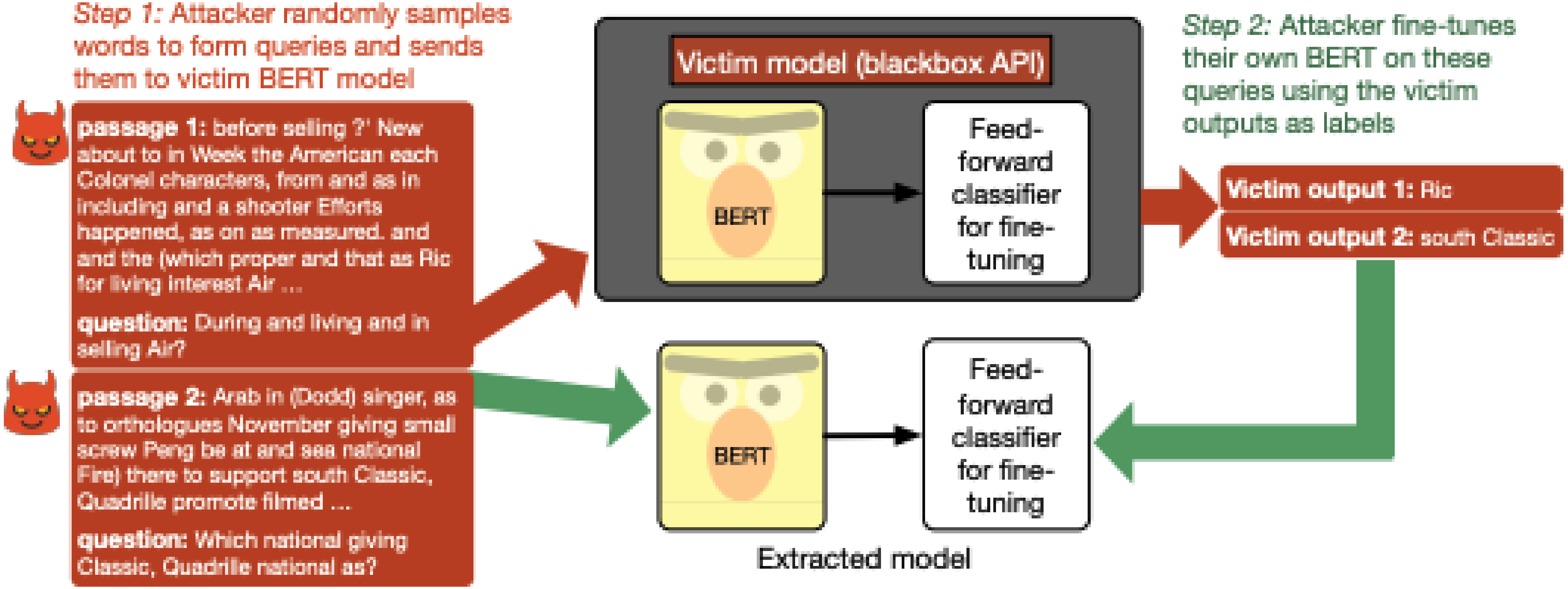
- Model extraction attacks on the Bert based NLP models which could potentially lead to stealing sensitive information about the training data.
- This paper talks about two defense strategies which could be implemented in the victim model in order to avoid the model being extracted by the adversary.
- The whole process is explained using the pre trained NLP model which is a BERT model.

BERT BASED NATURAL LANGUAGE PROCESSING

- Natural Language processing has emerged as one of the astounding evolutions in the field of Artificial Intelligence. In particular, BERT is the state-of-the-art language model for the NLP.
- BERT (Bidirectional Encoder Representations from Transformers) performs many NLP tasks which are question answering, natural language inference and so on.
- Transfer learning plays an important role in the field of computer vision where one can make use of the model parameters which are already trained on similar task without having to train a model from scratch.

MODEL EXTRACTION

- A model extraction attack happens when a malicious user tries to “reverse-engineer” this black-box victim model by attempting to create a local copy of it.
- That is, a model that replicates the performance of the victim model as closely as possible.
- If reconstruction is successful, the attacker has effectively stolen intellectual property. It does not have to pay the provider of the original API anymore to have the model predict on new data points.
- For instance, the adversary could use the stolen model to extract private information contained in the training data of the original model, or to construct adversarial examples that will force the victim model to make incorrect predictions.



Step 1: Attacker randomly samples words to form queries and sends them to victim BERT model

passage 1: before selling ?' New about to in Week the American each Colonel characters, from and as in including and a shooter Efforts happened, as on as measured. and and the (which proper and that as Ric for living interest Air …

question: During and living and in selling Air?

passage 2: Arab in (Dodd) singer, as to orthologues November giving small screw Peng be at and sea national Fire) there to support south Classic, Quadrile promote filmed …

question: Which national giving Classic, Quadrile national as?

Victim model (blackbox API)

BERT

Feed-forward classifier for fine-tuning

Step 2: Attacker fine-tunes their own BERT on these queries using the victim outputs as labels

Victim output 1: Ric

Victim output 2: south Classic

BERT

Feed-forward classifier for fine-tuning

Extracted model

DISTILLATION

- Model extraction is done by the process called resembling distillation.
- Adversaries pass a wide range of unlabeled input queries to the victim model to get it label from the prediction. These outputs are then used by the adversary to train their model which will reconstruct the victim model.
- Distillation aims to transfer knowledge from a big model to a small model. That is, distillation is used to decrease the number of parameters needed to store the model.
- This is often used as a way to support training large models on datacenters with lots of computing resources and then later deploy these models on edge devices with limited computing resources.
- This compression is not needed in model extraction. Instead, the adversary is primarily interested with the accuracy of the extracted model with respect to the victim model.

DATASETS

- The datasets used to perform model extraction are the most popular benchmarks which are being evaluated on the pre-trained language models like BERT.
- SST2 – This is a binary sentiment classification task where the input is a sentence and the output is either positive or negative sentiment. It is the most common dataset for sentiment classification.
- MNLI – This is a three-way entailment classification task. The input is two sentences and the output is either entailment or contraction.
- SQuAD – This is a reading comprehension dataset. The input is a paragraph and a question about the paragraph, and the output is a span of text from the paragraph which best answers the question. Note that unlike SST2 and MNLI, the output space is high dimensional.

STRATEGIES FOR ATTACKS

- The first strategy (RANDOM) uses nonsensical, random sequences of tokens sampled from Wikitext103's unigram distribution.
- The second strategy (WIKI) uses sentences / paragraphs from WikiText103. For tasks expecting a pair of inputs (MNLI, SQuAD), we use simple heuristics to construct the hypothesis (replace 3 words in premise with random words from Wikitext103) and question (sample words from the paragraph, prepend an interrogative word like "What" or "Where", append a question mark at the end) respectively.

DEFENSE STRATEGIES AGAINST ATTACKERS

- **Current defenses only work against naive adversaries. T**wo strategies are used to defend machine learning APIs against model extraction:
- detecting queries that could be part of a model extraction attack.
- watermarking predictions made by the API to later claim ownership of models that were extracted.
- While both defenses were effective to some degree, they work only in limited settings – sophisticated adversaries might anticipate these defenses and develop simple modifications to their attacks to circumvent these defenses.

CONCLUSION

The strategies discussed above can only work against naïve attacks. Where advanced attacks can outperform these strategies. This paper provides a starting point for research into model extraction attacks and there is also evidence for this could possibly be a potential risk in future and more research needs to be done in order to protect the data and model from these kinds of attacks.

THANK YOU