THIEVES ON SESAME STREET! MODEL EXTRACTION OF BERT-BASED APIS

- 本文主要研究NLP中的模型抽取方法. 攻击者只需要能够对victim model进行请求即可实现. 同时也展示了攻击者不需要任何的真实训练数据即可进行攻击。只需要随机的单词序列结合特定任务的heuristics即可
- 本文中的抽取方法主要面向基于NLP BERT transfer learning 模型。攻击者可以在很少的预算里获得一个和victim model性能相差甚微的模型
- 本文中还提到了，提取的模型可能泄露敏感的训练信息

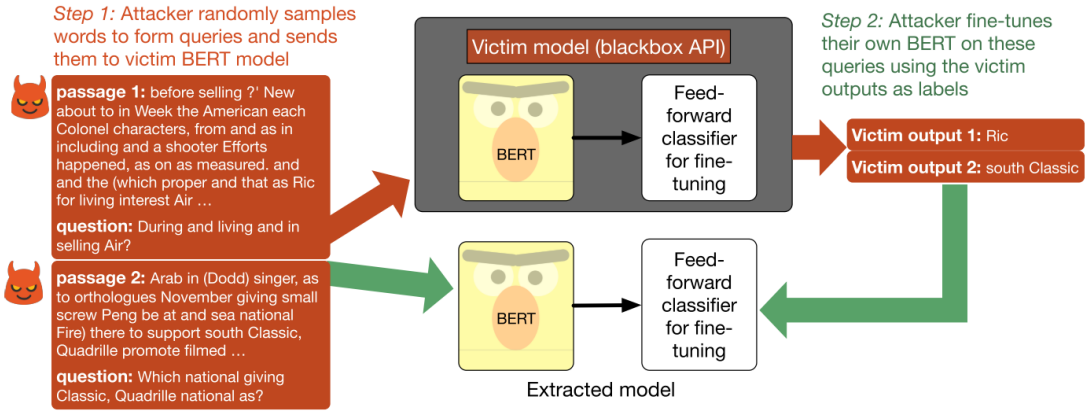

Figure 1: Overview of our model extraction setup for question answering.[2] An attacker first queries a victim BERT model, and then uses its predicted answers to fine-tune their own BERT model. This process works even when passages and questions are random sequences of words as shown here.

Method

因为攻击者没有victim model 的训练数据，因此可以使用task-specific的generator去构造一系列可能没有意义的语句序列$\{x_i\}_1^m$去对victim model进行请求。结果数据数据集$\{x_i, g_T(x_i)\}_1^m$被用于训练复制模型

本文使用的query generator有两种 : RANDOM和WIKI, 均是从NLP的特定数据集生成的, 然而作者发现这两个generator不足以抽取模型, 因为任务特征的复杂交互. 因此作者额外添加了两种任务特定的启发式 heuristics : MNLI和SQuAD/BoolQ

同时还表明了, 一个好的pretrain attacker模型可以更好的学习到语言的表示, 因此也能更加简化抽取

一些结果

Task	# Queries	Cost	Model	Accuracy	Agreement
SST2	67349	$62.35	VICTIM	93.1%	-
			RANDOM	90.1%	92.8%
			WIKI	91.4%	94.9%
			WIKI-ARGMAX	91.3%	94.2%
MNLI	392702	$387.82*	VICTIM	85.8%	-
			RANDOM	76.3%	80.4%
			WIKI	77.8%	82.2%
			WIKI-ARGMAX	77.1%	80.9%
SQuAD 1.1	87599	$115.01*	VICTIM	90.6 F1, 83.9 EM	-
			RANDOM	79.1 F1, 68.5 EM	78.1 F1, 66.3 EM
			WIKI	86.1 F1, 77.1 EM	86.6 F1, 77.6 EM
BoolQ	9427	$5.42*	VICTIM	76.1%	-
			WIKI	66.8%	72.5%
			WIKI-ARGMAX	66.0%	73.0%
	471350	$516.05*	WIKI (50x data)	72.7%	84.7%

Table 2: A comparison of the original API (VICTIM) with extracted models (RANDOM and WIKI) in terms of **Accuracy** on the original development set and **Agreement** between the extracted and victim model on the development set inputs. Notice high accuracies for extracted models. Unless specified, all extraction attacks were conducted use the same number of queries as the original training dataset. The * marked costs are estimates from available Google APIs (details in Appendix A.2).

Task	Model	0.1x	0.2x	0.5x	1x	2x	5x	10x
SST2	VICTIM	90.4	92.1	92.5	93.1	-	-	-
	RANDOM	75.9	87.5	89.0	90.1	90.5	90.4	90.1
(**1x** = 67349)	WIKI	89.6	90.6	91.7	91.4	91.6	91.2	91.4
MNLI	VICTIM	81.9	83.1	85.1	85.8	-	-	-
	RANDOM	59.1	70.6	75.7	76.3	77.5	78.5	77.6
(**1x** = 392702)	WIKI	68.0	71.6	75.9	77.8	78.9	79.7	79.3
SQuAD 1.1	VICTIM	84.1	86.6	89.0	90.6	-	-	-
	RANDOM	60.6	68.5	75.8	79.1	81.9	84.8	85.8
(**1x** = 87599)	WIKI	72.4	79.6	83.8	86.1	87.4	88.4	89.4
BoolQ	VICTIM	63.3	64.6	69.9	76.1	-	-	-
(**1x** = 9427)	WIKI	62.1	63.1	64.7	66.8	67.6	69.8	70.3

Table 3: Development set accuracy of various extracted models on the original development set at different query budgets expressed as fractions of the original dataset size. Note the high accuracies for some tasks even at low query budgets, and diminishing accuracy gains at higher budgets.

Conclusion

本文提出一种可以通过请求API来抽取基于BERT trasfering模型的方法, 并且不需要原始的训练数据等. 通过结合task-specific的样本generator, 生成即使是无意义的数据样本去对API进行请求, 将得到的结果和样本做成数据集训练模仿模型. 实验证明模仿模型的性能和victim model的性能差别不大.

这篇文章也简介证实了我的想法, 先抽取模型, 再恢复有相同语义的99数据, 然后在这个上面做transfer的文章, 最后的模型去替换原有模型. 但有一点需要考虑到的问题就是开销和整个流程中需要手动设定的参数. 最好是能够实现全自动的更新. 但一步一步来也不是不可以. 类似一个不断自我更新的framework