我拿模型当朋友，模型却想泄漏我的隐私？

原创 阿毅 夕小瑶的卖萌屋 2020-12-28 22:20

文 | 阿毅
编 | 小轶

相信大家对Facebook–Cambridge Analytica隐私泄露事件都还有印象。这事儿在当时可谓爆炸性新闻，激起了公众对数据隐私的强烈关注，也间接影响了美国总统选举结果（这不是重点）。不过从事后诸葛亮来看，这件事也是好事，改变了如今的世界格局（感谢普普，此处略去几万字）。但是，大家也就吃吃瓜，对于隐私保护的权利并没有持续地努力抗争下去（sad）。

实际上，**窃取隐私的方法不局限于APP非法收集用户数据、黑客攻击**等大家耳熟能详的方式，你很有可能在不知不觉中就泄露了隐私。

不知道大家有没有这样的经历：你在某些APP上和朋友聊吃的或者穿的，登陆某宝后你会发现平台会给你推荐这些东西。此时你不禁由衷感佩某宝推荐算法工程师未卜先知的能力。我猜测，其实，是你的聊天记录、或者你的输入法被泄露给了无良的第三方，然后某宝利用这些信息来精准推荐。

再举一个例子：手机党的朋友最不陌生的就是自己的输入法。输入法通过记忆我们的输入习惯来节省我们的沟通~~骂人~~时间。可是你有没有想过，万一哪天对话框输入“银行账户是...”，后面输入法自动帮你脑补了密码...啊，这。后果大家可想而知。

听到这，是不是觉得自己超委屈？隐私权利一点都没有！

我觉得我非常委屈

那么，咱们今天~~就来聊聊如何保护我们的隐私数据，啊不~~从学术的角度上探究一下这种对输入法等语言模型的攻击可以如何实现！简言之，教你如何“窃取用户隐私数据”。

废话不多说，今天要和大家分享的是一篇关于NLP Privacy的文章，由众多大佬（Google、Stanford、UC Berkeley、Northeastern University、Open AI、Harvard、Apple）联合巨制，且在学术站上点赞量很高！我们都知道，当今的语言模型都是在很大的私有（或者公开）数据集（数百GB）上训练，期间难免记忆了一些其中的敏感信息。那么，这些信息是否会不经意间就可能由模型泄露出去呢？这篇论文就实验性地分析了GPT-2这样的**大型语言模型是否存在隐私泄露的可能**，并探究了这种**攻击在怎样的场景下能够成功实现**。

想想实属业界良心——**自己攻击自己设计的模型，还发文章告诉你怎么攻击**...接下来，我们剖析一下这篇业界良心、自己打自己脸的论文干了些啥。

论文题目:

Extracting Training Data from Large Language Models

论文链接:

https://arxiv.org/abs/2012.07805

Arxiv访问慢的小伙伴也可以在 **【夕小瑶的卖萌屋】**订阅号后台回复关键词 **【1228】** 下载论文PDF~

AI Privacy先验知识

AI Privacy是近几年比较火的一个领域，它通常涉及**针对不同机器学习模型的攻击和防御**。攻击的目的主要是**窃取隐私**和**破坏性能**。由于这篇论文涉及一些AI Privacy领域的先验知识，我总结了如下四点必要的先验知识，帮助大家理解。

成员推断

成员推断（*Membership Inference Attacks*）[1]，即给定数据记录和模型的黑盒访问权限，要求**确定该记录是否在模型的训练数据集中**。执行成员推理，需要采取机器学习中的对抗性应用，训练一个**推理**

模型，识别目标模型对训练集内输入的预测结果与对训练集外输入的预测结果之间的差异。

通常采用的方法是：构建**影子模型**（shadow model）。这些模型的行为**与目标模型类似**。但与目标模型相比，每个影子模型的真实情况是已知的。

逆向攻击

逆向攻击（*Model Inversion Attacks*）[2],主要是利用机器学习系统提供的一些API来**获取模型的初步信息**，并通过这些初步信息**对模型进行逆向分析，获取模型内部的一些隐私数据**。

这种攻击和成员推理攻击的区别是：成员推理攻击是针对某条单一的训练数据，而模型逆向攻击则是要取得一种**整体的统计信息**。这篇论文所做的训练数据提取攻击（Training data extraction attacks），是模型逆向攻击的一种，旨在重建训练数据点。这种攻击**难度更大，破坏性也更强**。

萃取攻击

萃取攻击（*Model Extraction Attacks*）[3]，也称提取攻击，是一种攻击者**通过循环发送数据，查看模型响应结果，来推测该模型的参数或功能**，从而复制出一个功能相似、甚至完全相同的机器学习模型。这种攻击方法由Tramèr等人在2016年提出，并发表于信息安全顶级会议Usenix Security上。

差分隐私

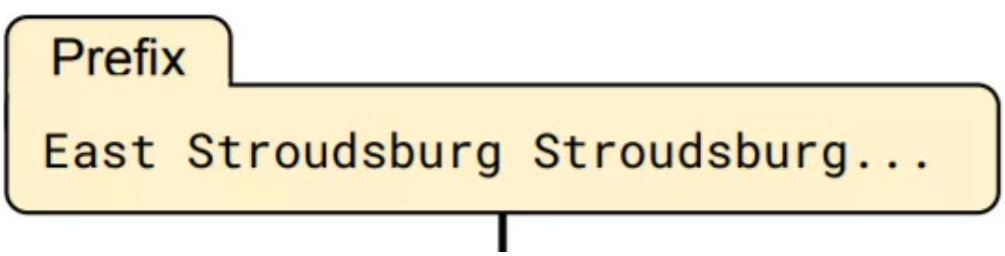
差分隐私（*Differential Privacy*）[4]，由Dwork 在2013年写的*The Algorithmic Foundations of Differential Privacy*中提出，是一种**数据隐私保护技术**。由于差分隐私可深度学习技术，保护模型的隐私和安全，于2020年入选世界十大先进科学技术。

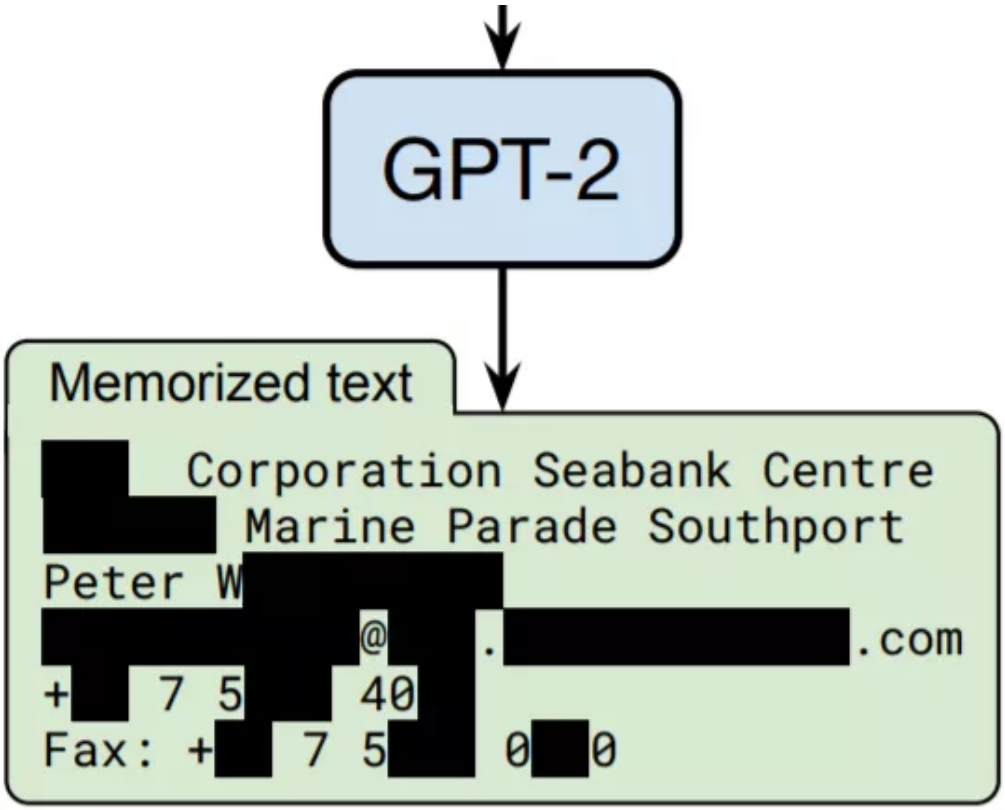
> 上述介绍只是提纲挈领，感兴趣的同学可直接阅读相关论文。其次，同学们也可以看到，AI privacy涉及DL各个领域的知识，因此可投会议也覆盖所有DL领域！是一个不错的坑哦~~

论文解读

概述

这篇论文做的工作其实一幅图就可以讲清楚，如下图所示：你先给GPT-2模型**输入一串“神秘代码”**——“*East Stroudsburg Stroudsburg...*”；模型立刻**送出一套个人信息**——姓名、电话号码，还有地址、邮箱和传真（部分信息已打码）。





好家伙。啪的一下啊！个人信息就泄露了，很快啊！一般人都会大意，闪都来不及。

攻击者的能力

在AI Privacy领域，一般阐释一种攻击前，必须说清楚**攻击者所具备的知识、能力**（即攻击者的power有多大）。通常来说，一个**成功的攻击算法是不能允许攻击者掌握太多知识的**；相反，防御者可以被允许掌握攻击者的很多知识。

在本文中，作者们考虑一个**对黑盒语言模型具有输入输出访问权限**的攻击者。也就是说，我们允许攻击者获得下一个单词的预测结果，但不允许攻击者掌握语言模型中的单个权重或隐藏状态（例如，注意力向量）。

攻击者的目标是**从模型中提取被记忆的训练数据**。注意，这里并不要求提取*特定的*训练数据，只需*随意*提取训练数据即可。因为前者仍然是很难实现的。

> 安全类文章一般都会非常详细地描述攻击者掌握的知识以及攻击的细节，因为需要说服审稿人相信这个攻击是很难实现、且有实际意义的。

攻击方法

如上图所示，攻击一共由两个步骤组成：

1. **生成文本**：从模型中无条件采样大量生成文本
2. **成员推断**：作者们使用Membership Inference来删除那些重复出现的样本来加强生成文本的准确性，预测哪些输出包含被记忆的文本。

文本生成

看前面那张图即可知，本文的核心工作是**如何根据给定的前缀，输出模型中被记忆的数据（即后缀）**。为了解决传统top-k采样策略倾向于多次生成相同（或相似）文本的问题，本文设计了一种**基于Decaying Temperature的数据采样策略**，来生成富有多样性的高质量文本。

这个Temperature，t其实是一个超参数，来降低模型已输出生成文本的置信度。一旦置信度降低，模型就会尽可能生成更多的文本来使得输出的可信度提高。但是，作者说上述方法仍会出现生成文本具有相同前缀的问题。最后他们设置了多个不同的前缀种子来避免这个问题。

成员推断

生成文本之后，我们需要使用**成员推断**（*Membership Inference*）来判断生成文本是否是被记忆的文本。在本文中，作者发现直接运用传统的成员推断存在一定问题：以下两类低质量的生成结果也会被打很高的置信度分数：

- **Trivial memorization**: 过于普遍常见的内容，例如数字1到100。这些虽然也可能是训练集中被记忆的内容，但意义不大。
- **Repeated substrings**：语言模型的一种常见智障模式是不断重复输出相同的字符串（例如，“我爱你我爱你我爱你我爱你……”）。作者发现这类文本也容易被打很高的置信度。

为此，作者设计了一系列骚操作，删除以上两类文本。具体就是根据以下6个指标如下，对每个生成的样本进行筛选，并去掉重复的部分：

- 困惑度（perplexity）
- Small模型：小型GPT2和大型GPT2的交叉熵比值
- Medium模型：中型GPT2和大型GPT2的交叉熵比值
- zlib：GPT2困惑度和压缩算法熵的比值
- Lowercase：GPT-2模型在原始样本和小写字母样本上的困惑度比例
- Window：在最大型GP-2上，任意滑动窗口圈住的50个字能达到的最小困惑度

> 这块我其实还没完全理解，具体细节还望大家回看原文。并由衷希望读懂的同学给我留言，我也想搞懂。。真的。。。

实验结果

作者**在随机抽取的1800个输出结果中，约有600个结果体现出了训练数据中的内容**，包括新闻、日志、代码、个人信息等等。其中有些内容只在训练数据集中出现过寥寥几次，有的甚至只出现过一次，但模型依然把它们学会并记住了（其实越特殊，模型为了不出错，记忆得越深）。

Category	Count
US and international news	109
Log files and error reports	79
License, terms of use, copyright notices	54
Lists of named items (games, countries, etc.)	54
Forum or Wiki entry	53
Valid URLs	50
Named individuals (non-news samples only)	46
Promotional content (products, subscriptions, etc.)	45
High entropy (UUIDs, base64 data)	35
Contact info (address, email, phone, twitter, etc.)	32
Code	31
Configuration files	30
Religious texts	25
Pseudonyms	15
Donald Trump tweets and quotes	12
Web forms (menu items, instructions, etc.)	11
Tech news	11
Lists of numbers (dates, sequences, etc.)	10

团队还对拥有15亿参数的升级版GPT-2 XL进行了测试，它对于训练数据的记忆量是GPT-2 Small的10倍。实验发现，**越大的语言模型，“记忆力”越强**。GPT-2超大模型比中小模型更容易记住出现次数比较少的文本。他们还发现，不光是OpenAI的GPT模型，其它主流语言模型BERT、RoBERTa等等，也统统中招。

小结与感想

文章的贡献可以总结为以下三点：

- 证明了大型语言模型会记住并泄露个别训练数据。
- 提出了一种简单有效的方法，仅使用黑盒查询访问权限,即可从语言模型的训练集中提取逐字记录的序列。在GPT-2模型上进行了大量的实验。
- 最后，文章还讨论了许多缓解隐私泄露的策略。例如，**差分隐私** 在一定适用范围内可以保证隐私，但是它会导致更长的训练时间，并且通常会降低性能（*说明是一个坑啊!赶紧设计高效的差分隐私机*

制就是一篇顶会啊!!）。其次，还可以使用 **Machine Unlearning** [5]方法，该方法在经验上将有助于减轻模型的记忆，但不能阻止所有攻击。

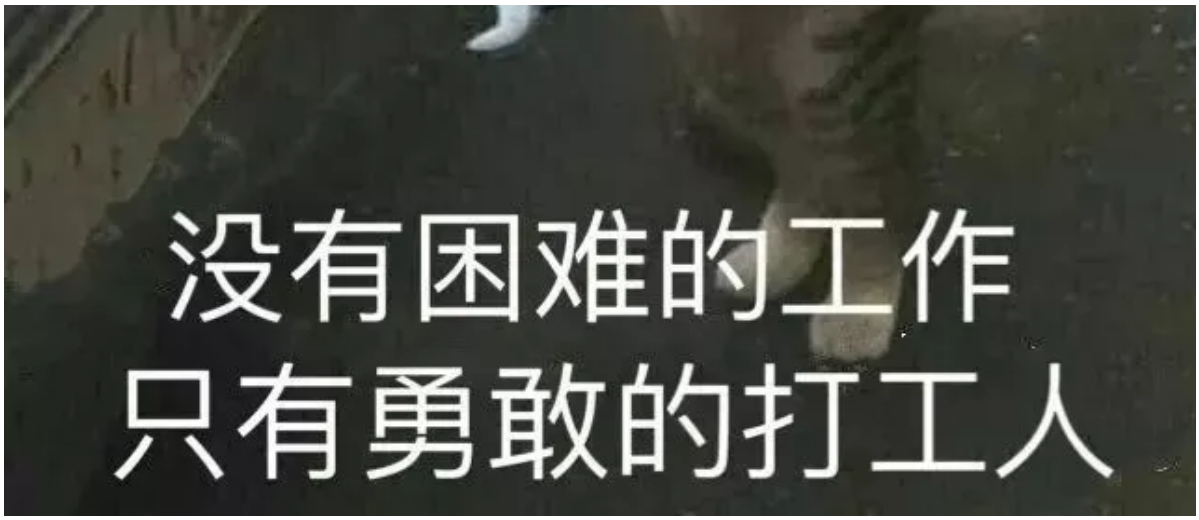
然后我从创新性、理论完备性、实验、未来展望四个角度，谈谈自己的理解：

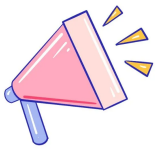
- **创新性**：首先，本文算是NLP和Privacy结合的先驱工作之一，目前该类结合的文章还不是很多（可看文末的参考文献，有一些类似的工作）。其次，本文方法上并不是非常新，用的方法都是在现有的基础上结合NLP任务的特殊性进行改进和提升的，说实话更偏工程性。
- **理论完备性**：本文其实在理论的完备性上还差一点，因为阅读者可能会好奇为什么作者采取的一系列操作就可以生成训练样本，也同样会好奇为什么设计的数据采样策略就可以增加文本的多样性。
- **实验**：本文用丰富的实验，证明了该文提出的攻击方法可以有效攻击GPT2模型，并从不同的角度说明了攻击效果，还探究了模型大小与被攻击风险的关系。但本人觉得,一般来说需要在一定隐私保护的情况下再做一组对比实验。因为诸如苹果手机等很多实际应用场景，很早就用了差分隐私机制来保护用户的隐私。
- **未来展望**：文中也说到如何设计高效的隐私保护机制是未来很有前途的方向之一，例如使用差分隐私或者Machine Unlearning。另外，我们也可以尝试设计一些攻击算法来攻击模型，例如ACL'20[6]使用权值中毒攻击来攻击预训练模型。文中未提到的参考文献均为最近NLP和Privacy结合的新文章。

说在文末的话

本人是做AI privacy的。说到这篇文章把NLP和Privacy结合，我想起了一个小故事：写paper其实就是在一座山上找一个安全的坑拉粑粑，当旁边都是别人的粑粑的时候你再去拉肯定会很痛苦，你如果找到一个没人拉过粑粑的地方肯定拉的很香。这个故事是一个有味道的故事，但我想说的是，这种新兴、交叉领域很值得我们去探索。说不定以后别人只能在拉过的地方拉，让别人无处可拉。

最后，欢迎各位NLPer关注AI privacy领域。一起来卷，卷到最后，应有尽有。





后台回复关键词【**入群**】

加入卖萌屋NLP/IR/Rec与求职讨论群

后台回复关键词【**顶会**】

获取ACL、CIKM等各大顶会论文集！

参考文献

[1]Shokri R, Stronati M, Song C, et al. Membership inference attacks against machine learning models[C]//2017 IEEE Symposium on Security and Privacy (SP). IEEE, 2017: 3-18.

[2]Fredrikson M, Jha S, Ristenpart T. Model inversion attacks that exploit confidence information and basic countermeasures[C]//Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. 2015: 1322-1333.

[3]Tramèr F, Zhang F, Juels A, et al. Stealing machine learning models via prediction apis[C]//25th {USENIX} Security Symposium ({USENIX} Security 16). 2016: 601-618.

[4]Dwork C, Roth A. The algorithmic foundations of differential privacy[J]. Foundations and Trends in Theoretical Computer Science, 2014, 9(3-4): 211-407.

[5]Bourtoule L, Chandrasekaran V, Choquette-Choo C, et al. Machine unlearning[J]. arXiv preprint arXiv:1912.03817, 2019. S&P 2020.

[6]Kurita K, Michel P, Neubig G. Weight poisoning attacks on pre-trained models[J]. arXiv preprint arXiv:2004.06660, 2020.

[7]Carlini N, Tramer F, Wallace E, et al. Extracting Training Data from Large Language Models[J]. arXiv preprint arXiv:2012.07805, 2020.

[8]Wallace E, Stern M, Song D. Imitation Attacks and Defenses for Black-box Machine Translation Systems[J]. arXiv preprint arXiv:2004.15015, 2020.

[9]Pan X, Zhang M, Ji S, et al. Privacy risks of general-purpose language models[C]//2020 IEEE Symposium on Security and Privacy (SP). IEEE, 2020: 1314-1331.

[10]https://sites.google.com/view/wsdm-privatenlp-2020