Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

针对文本嵌入模型的模型反演攻击方法研究

硕士研究生 皮佳伟

2024年10月27日

问题回溯

- 总结反思
  - 背景部分介绍过于简略
  - 演讲初期过于紧张，导致听众观感不佳
- 相关内容
  - 2024.01.28 皮佳伟《偷走你的训练数据：模型反演攻击方法研究》
  - 2023.03.05 张辰龙《深度神经网络模型窃取检测》
  - 2022.10.16 程瑶《成员推理攻击》

内容提要

- 预期收获
- 题目内涵解析
- 研究背景与意义
- 研究历史与现状
- 知识基础
- 算法原理
  - Vec2Text
  - Transfer Attack
- 特点总结与工作展望
- 参考文献

背景简介

- 预期收获
  - 掌握文本嵌入面临的模型反演攻击风险
  - 理解两种模型反演攻击方法的基本原理
  - 了解现有方法的缺陷以及未来发展方向

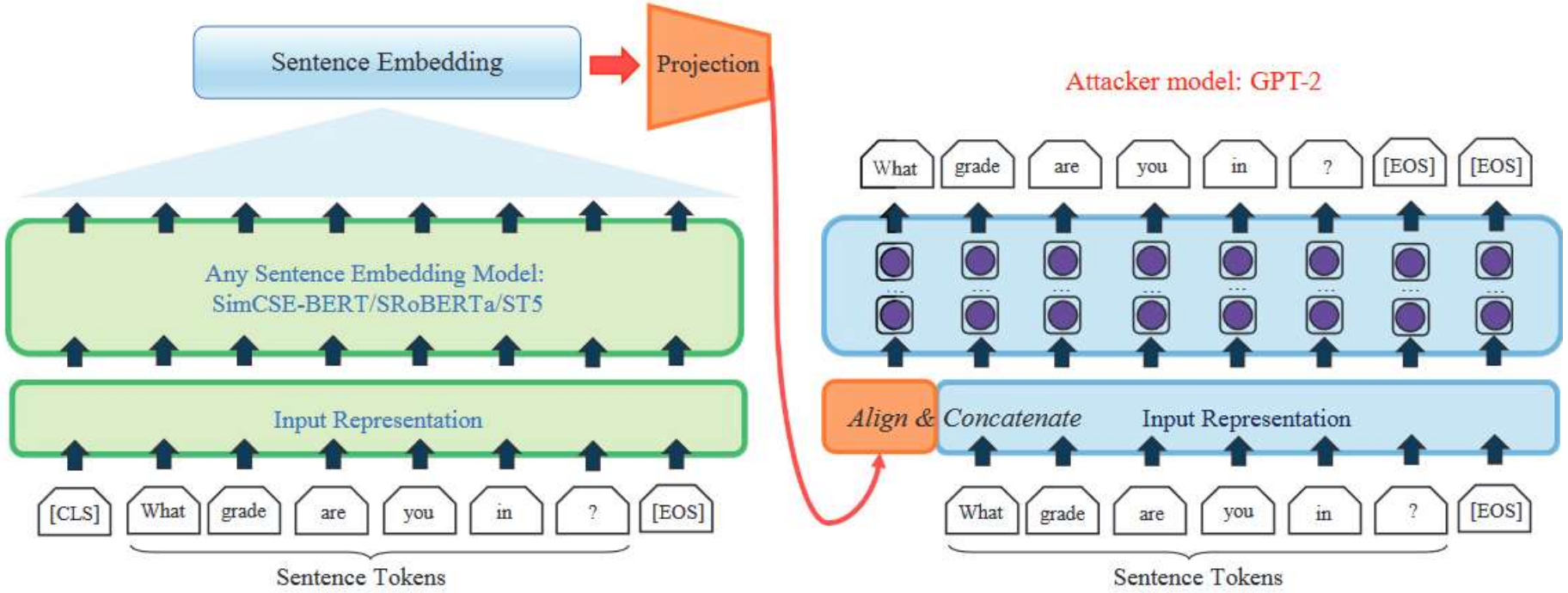
内涵解析与研究目标

- 题目内涵解析（针对文本嵌入模型的模型反演攻击）
  - 反演：即逆转，如何由输出得到输入
  - 模型反演：反演在深度学习模型上的体现
  - 模型反演攻击：通过特殊设计的算法，重建目标模型的私有训练样本，进而造成敏感信息的泄露

- 研究目标
  - 面向深度学习模型的隐私安全研究
  - 研究目标模型特征迁移、解码器训练、生成样本质量评估等关键问题
  - 结合文本预训练模型、可控文本生成、代理模型训练等理论
  - 重建目标模型私有训练样本，揭示模型训练数据所面临的隐私安全问题

研究背景

- 研究背景
  - 预训练文本嵌入模型已经成为自然语言处理研究中极为重要的一部分
  - 特定任务的深度学习模型需要特定的数据样本进行训练
  - 数据样本通常包含各种敏感信息或者所有者涉及知识产权不愿公开
  - 模型反演攻击能够针对目标模型重建训练数据样本，导致严重的隐私泄露



研究意义

- 研究意义
  - 验证文本嵌入模型的数据泄露风险
    - 由图像领域向文本领域迁移
    - 验证文本领域面临训练数据泄露风险
  - 促进防御方法发展
    - 研究攻击方法在文本领域的应用
    - 以攻击促进防御手段的发展
    - 验证已有的防御手段是否可行

重建私有训练样本，促进防御方法发展，保障模型隐私安全

研究历史

2014 Fredrikson等人首次提出了模型反演攻击的概念，利用最大后验估计器，对基因隐私相关的线性回归模型进行反演，获得了患者基因隐私数据

2019 Carlini等人依据图像领域的模型反演攻击研究成果，验证在文本领域中是否也存在模型反演攻击风险，开拓了模型反演攻击的应用场景

2020 Song等人针对已有的基于分类方案的关键词推断方法，需要大量候选数据和上下文的问题，提出一种缩小目标嵌入和反演嵌入距离的方法，以恢复一组单词

2021 Deng等人针对已有的研究在文本离散空间中难以进行梯度优化的问题，提出一种虚拟梯度策略，衡量虚拟梯度和实际梯度之间的距离的方法

2023 Li等人针对已有的研究主要为生成文本中的停用词而难以揭示敏感信息的问题，提出利用大型语言模型和句子嵌入来解码整个序列

2023 Morris等人针对已有的方法直接依据嵌入重建序列，导致重建文本的可读性差的问题，提出一种可控文本生成方案，通过迭代优化重建高质量文本

2024 Chen等人针对已有的研究难以对跨语言模型重建质量难以评估的问题，提出一种添加翻译模块的方式来实现跨语言模型重建质量的评估

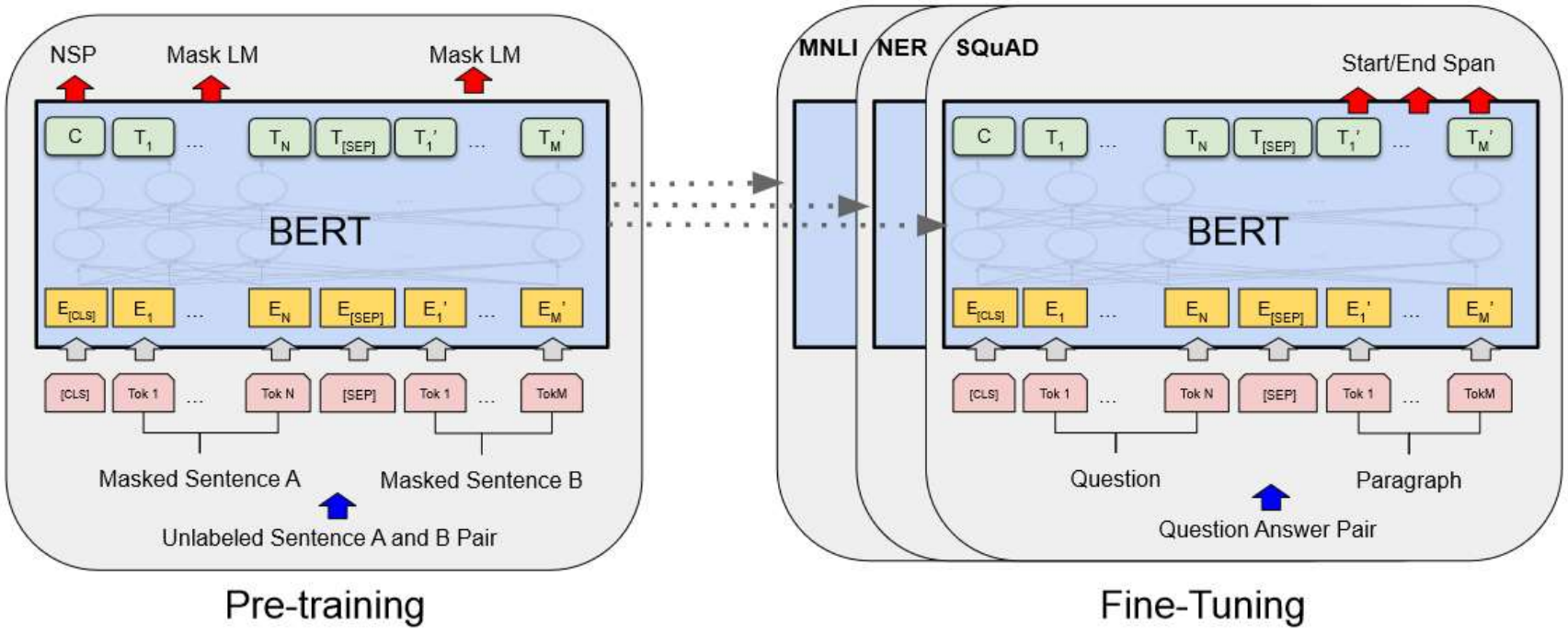
2024 Huang等人针对已有方法需要对目标模型进行大量查询，提出一种依赖代理模型的攻击方式，实现少查询的黑盒文本嵌入模型反演攻击

针对文本嵌入模型反演攻击

- 黑盒模型反演
  - 基于候选文本分类推断
  - 基于逆模型重建
- 白盒模型反演
  - 基于候选文本分类推断
  - 基于逆模型重建

知识基础　文本嵌入模型

- 文本嵌入模型
  - 文本嵌入是以一种将文本表示为连续数值向量的技术
  - 常见的嵌入模型包括：Word2Vec、BERT等



研究针对文本嵌入模型的模型反演攻击，对模型的安全至关重要

Vec2Text

【 EMNLP 】

Text Embeddings Reveal (Almost) As Much As Text

Vec2Text TIPO

T	目标	提出受控生成的文本嵌入模型反演攻击方法
I	输入	目标模型*1，辅助数据集*1
P	处理	1. 将文本嵌入模型反演攻击任务看作条件文本生成任务 2. 假设初始生成文本，输入目标模型获得嵌入向量 3. 迭代优化生成文本，使生成文本和原始文本嵌入向量相近
O	输出	私有样本*n

P	问题	直接训练文本生成模型由嵌入向量生成文本可读性低
C	条件	目标模型黑盒设置
D	难点	1. 如何在仅可访问文本嵌入向量的条件下实现训练文本生成 2. 如何保证生成文本的高可读性
L	水平	EMNLP 2023（CCF B）

Vec2Text 基础模型

- 优化问题
  - 将使用编码器获得具有嵌入e的文本转化为一个优化问题

$$\hat{x} = arg\ max_x cos[(\phi(x), e]$$

    - 其中ϕ为文本嵌入模型，$\hat{x}$为重建文本，e为原文本嵌入，$cos()$为余弦相似度
    - 枚举所有可能的序列来计算上述公式在实际上不具有可行性
- 学习给定嵌入的文本分布

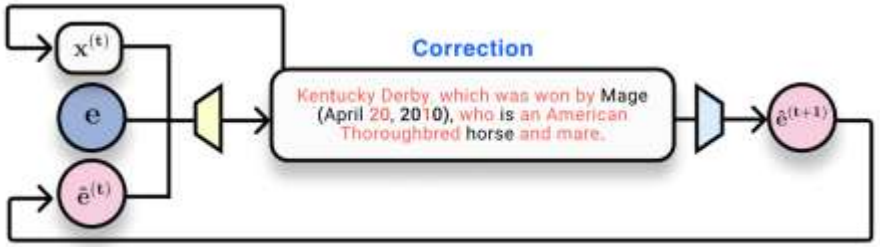
$$\theta = arg\ max_{\hat{\theta}} E_{x \sim D}\left[p(x|\phi(x); \hat{\theta})\right]$$

    - 其中θ为模型参数，$D = \{x_1, \ldots\}$为文本数据集，$p(x|\phi(x); \hat{\theta})$是给定嵌入的文本分布
    - 将组合优化问题摊到神经网络的权重中，但是直接以这种方式难以生成令人满意的文本

Vec2Text 算法原理图

- 迭代优化训练
  - 初始化一个猜测的生成文本$x^{(0)}$
  - 将生成文本$x^{(0)}$输入目标模型ϕ获得嵌入向量$e^{(0)}$

$$e^{(0)} = \phi(x^{(0)})$$



  - 依据嵌入向量e、$e^{(0)}$和生成文本$x^{(0)}$优化得到新的生成文本$x^{(1)}$

$$p\left(x^{(t+1)}|e\right) = \sum_{x^{(t)}} p\left(x^{(t)}|e\right) p\left(x^{(t+1)}|e, x^{(t)}, \hat{e}^{(t)}\right)$$

$$\hat{e}^{(t)} = \phi(x^{(t)})$$

  - 其中，$x^{(t)}$是迭代生成的第t个文本，$\hat{e}^{(t)}$是$x^{(t)}$在输入目标模型后的嵌入向量
  - 由$x^{(0)}$计算$\hat{e}^{(0)}$，修正得到$x^{(1)}$，训练模型并迭代重复上述过程

Vec2Text 实现上的应用细节

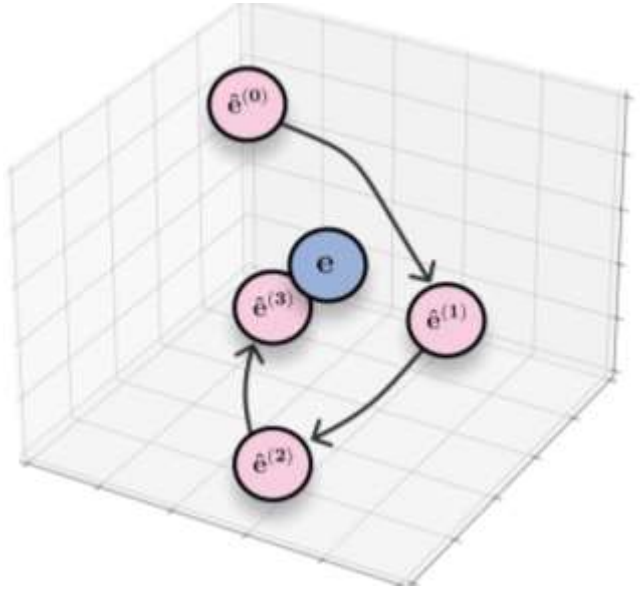
- 嵌入向量维度不匹配
  - $\hat{e}^{(t)}$与e输入到编码器中，而输入的序列维度d_{enc}与ϕ嵌入的维度d不一定一致

$$EmbToSeq(e) = W_2\sigma(W_1 e)$$

    - 利用MLP实现向量投影，做到维度匹配
    - 其中$W_1 \in \mathbb{R}^{d\times d}$， $W_2 \in \mathbb{R}^{(sd_{enc})\times d}$，$\sigma$为激活函数， s为编码器长度
  - 编码器输入

$$concat(EmbToSeq(e), EmbToSeq(\hat{e}^{(t)}), EmbToSeq(e-\hat{e}^{(t)}), (w_1 \dots w_n))$$

    - 其中$(w_1 \dots w_n)$为$x^{(t)}$的词嵌入向量
    - 将拼接后的向量输入编码器，采用标准语言建模损失训练完成的编码器-解码器模型



实验设计

- 目标模型
  - GTR-base：基于T5的用于文本检索的预训练Transformer
  - text-embeddings-ada-002：OpenAI API提供的文本嵌入器
- 数据集
  - Wikipedia文章：由自然问题语料库中选取的维基百科文章的500万个段落（GTR）
  - MSMARCO语料库：用于训练OpenAI模型
  - MIMIC-III临床笔记数据库，BEIR基准提供的各种数据集（域外实验设置）
- 评价指标
  - BLEU：衡量真实文本和重建文本之间的n元组相似度指标
  - Token F1：预测标记集和真实标记集之间的多类F1分数
  - 精确匹配：完美匹配私有样本的重建样本百分比
  - 相似性：真实嵌入和重建文本嵌入之间的余弦相似度

实验设计　对比实验

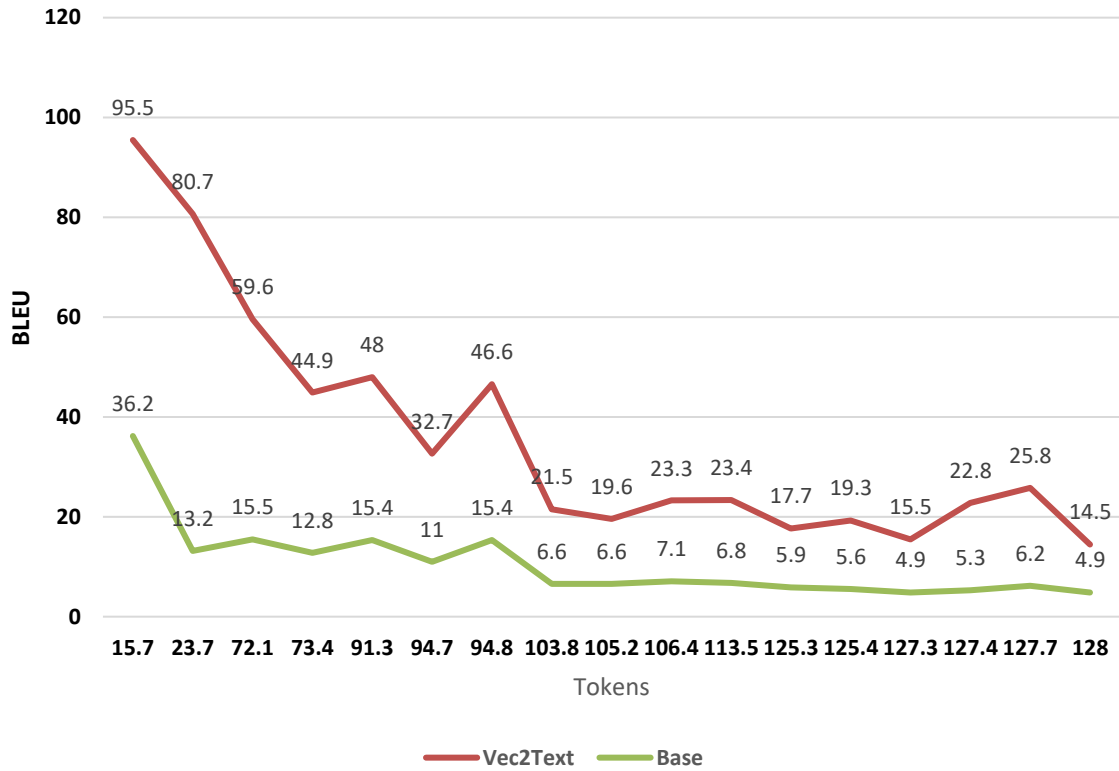
- 对比实验（域内）
  - 公共训练集和私有样本来源于同一个数据集
- 实验结果
  - 在三个目标模型实验中均为最优结果
  - 方法在精确恢复较长文本上仍然存在较多问题

	method	tokens	pred tokens	bleu	tf1	exact	cos
GTR Natural Questions	Bag-of-words (Song and Raghunathan, 2020)	32	32	0.3	51	0.0	0.70
	GPT-2 Decoder (Li et al., 2023)	32	32	1.0	47	0.0	0.76
	Base [0 steps]	32	32	31.9	67	0.0	0.91
	(+ beam search)	32	32	34.5	67	1.0	0.92
	(+ nucleus)	32	32	25.3	60	0.0	0.88
	Vec2Text [1 step]	32	32	50.7	80	0.0	0.96
	[20 steps]	32	32	83.9	96	40.2	0.99
	[50 steps]	32	32	85.4	97	40.6	0.99
	[50 steps + sbeam]	32	32	**97.3**	**99**	**92.0**	**0.99**
OpenAI MSMARCO	Base [0 steps]	31.8	31.8	26.2	61	0.0	0.94
	Vec2Text [1 step]	31.8	31.9	44.1	77	5.2	0.96
	[20 steps]	31.8	31.9	61.9	87	15.0	0.98
	[50 steps]	31.8	31.9	62.3	87	14.8	0.98
	[50 steps + sbeam]	31.8	31.8	**83.4**	**96**	**60.9**	**0.99**
OpenAI MSMARCO	Base [0 steps]	80.9	84.2	17.0	54	0.6	0.95
	Vec2Text [1 step]	80.9	81.6	29.9	68	1.4	0.97
	[20 steps]	80.9	79.7	43.1	78	3.2	0.99
	[50 steps]	80.9	80.5	44.4	78	3.4	0.99
	[50 steps + sbeam]	80.9	80.6	**55.0**	**84**	**8.0**	**0.99**

实验设计　对比实验

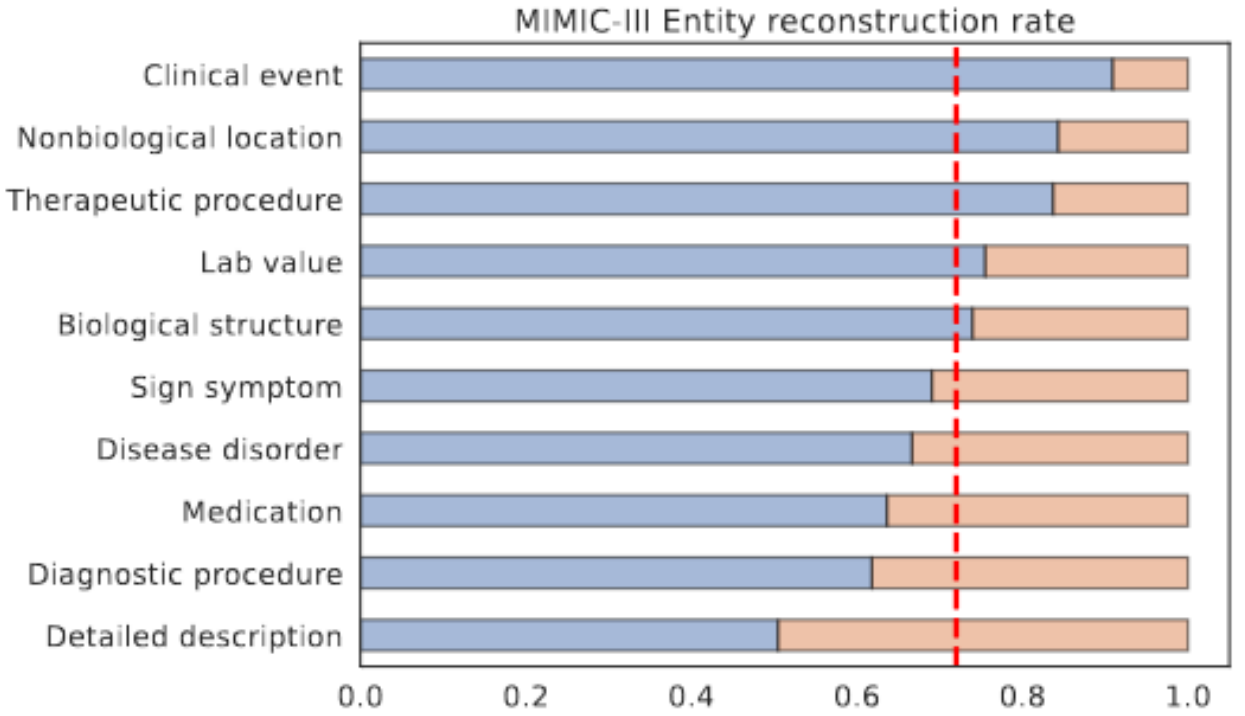
- 对比实验（域外）
  - 公共训练集和私有样本来源于不同数据集
- 实验结果
  - 测试指标上领先于基线方法
  - 相较于域内实验，各长度文本下的指标均出现下降
  - 随Tokens增加，重建的质量下降

dataset	tokens	method	bleu	token F1
quora	15.7	Base	36.2	73.8
		Vec2Text	95.5	98.6
signal1m	23.7	Base	13.2	49.5
		Vec2Text	80.7	92.5
msmarco	72.1	Base	15.5	54.1
		Vec2Text	59.6	86.1
bioasq	127.4	Base	5.3	35.7
		Vec2Text	22.8	59.5
scifact	127.4	Base	4.9	35.2
		Vec2Text	16.6	56.6
nfcorpus	127.7	Base	6.2	39.6
		Vec2Text	25.8	64.8
trec-news	128.0	Base	4.9	34.8
		Vec2Text	14.5	51.5



- 案例研究
  - MIMICIII临床笔记数据集
  - 检验方法对实体的重建效果
- 实验结果
  - Vec2Text能够恢复94%的名字、95%的姓氏和89%的全名
  - 能够完美恢复29%的文档内容
  - Vec2Text在恢复“详细类别”的能力最差，该类别包括具体的医疗术语等内容

method	first	last	full	bleu	tf1	exact	cos
Base	40.0	27.8	10.8	4.9	33.1	0.	0.78
Vec2Text	94.2	95.3	89.2	55.6	80.8	26.0	0.98



算法总结 Vec2Text

- 算法流程
  - 假设初始生成文本，输入目标模型获得嵌入向量
  - 迭代优化生成文本，使生成文本和原始文本嵌入向量相近
- 算法优势
  - 针对黑盒自然语言嵌入模型，适用面更广
  - 将直接重建过程变为受控的迭代优化过程，大幅提高攻击效果
- 算法不足
  - 攻击效果随文本长度的增加而大幅下降
  - 特定实体的重建效果不佳
  - 需要大量查询目标模型

Transfer Attack

【ACL】

Transferable Embedding Inversion Attack: Uncovering Privacy Risks in Text Embeddings without Model Queries

Transfer Attack TIPO

T	目标	减少对目标模型的查询次数
I	输入	目标模型*1，辅助数据集*1
P	处理	1. 训练代理模型，构建一个与目标模型相似的代理模型 2. 利用基于GPT的解码器将嵌入生成为原始文本序列
O	输出	目标样本*n

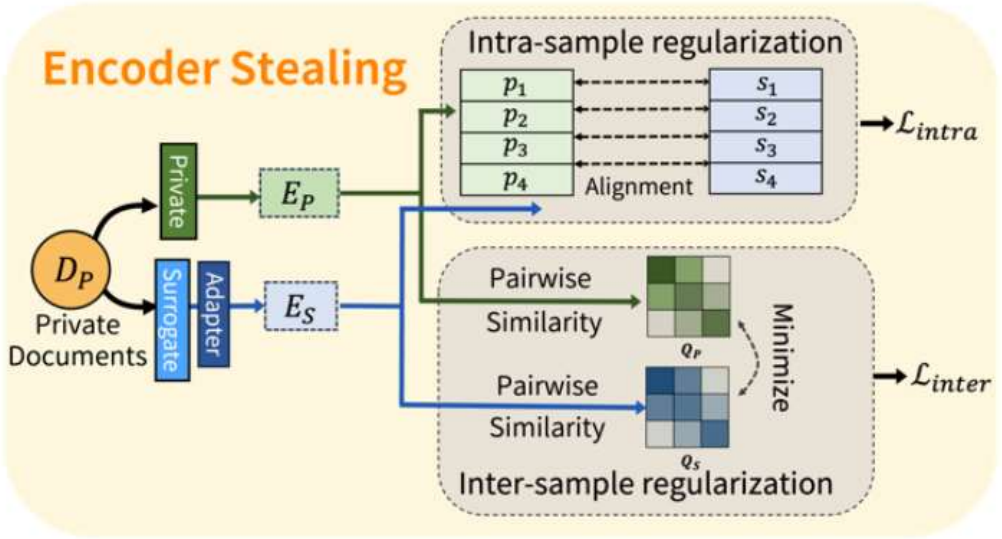
P	问题	攻击过程中对目标模型的大规模查询在部分场景下不可行
C	条件	黑盒文本嵌入模型；可少量查询模型
D	难点	1. 如何在保证重建样本质量的前提下减少查询次数
L	水平	ACL 2024（CCF A）

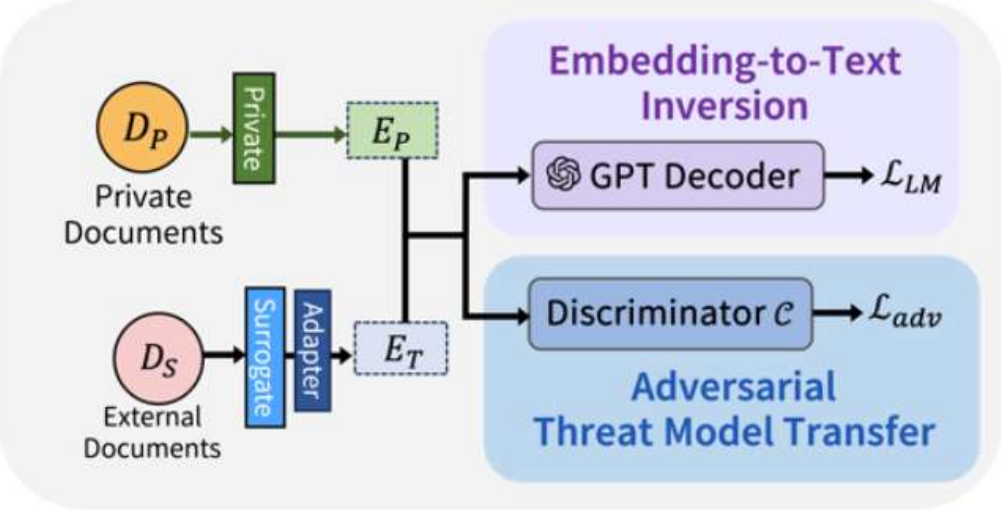
Transfer Attack 算法原理图

- 算法流程
  - 收集目标模型泄露的少量样本与嵌入数据集D_p

$$D_p = \{(x, \phi(x))\}$$

    - 其中x为文本；$\phi(x)$为嵌入；ϕ为目标模型
  - 构建代理模型$\hat{\phi}$
  - 依据损失项优化代理模型
  - 构建攻击模型
  - 依据数据集训练攻击模型
- 问题
  - 如何保证代理模型的可用性
  - 如何保证在代理模型上攻击结果是真实符合目标模型，即攻击的可迁移性





Transfer Attack 算法原理图

- 代理模型可用性
  - 最优代理模型所满足的条件

$$\widehat{\phi}(x) \approx \phi(x)$$

  - 一致性正则化损失项$\mathcal{L}_{surrogate}$

$$\mathcal{L}_{surrogate} = \mathcal{L}_{intra} + \mathcal{L}_{inter}$$

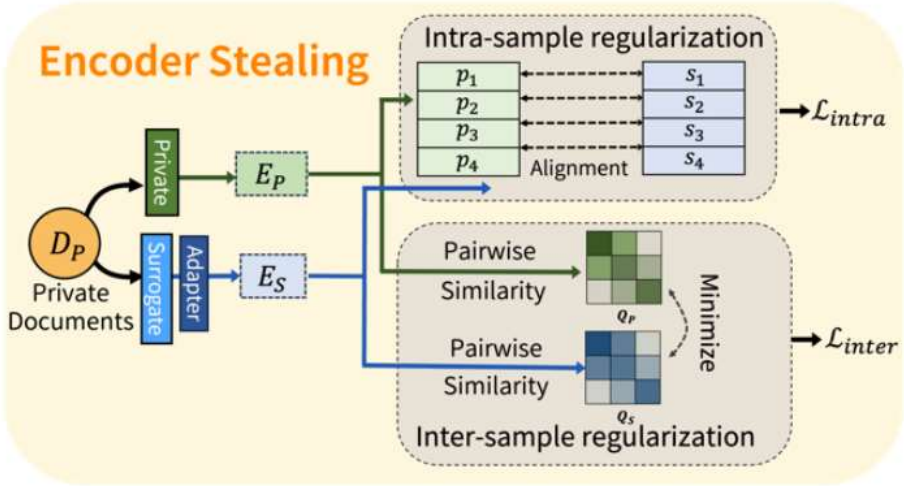
    - 其中$\mathcal{L}_{intra}$是内部一致性损失化项，$\mathcal{L}_{intra}$是相互一致性损失化项
  - 内部一致性损失化项

$$\mathcal{L}_{intra}(E_p, E_s) = MSE(E_p, E_s), \quad E_p = \phi(x), \quad E_s = \widehat{\phi}(x)$$

  - 相互一致性损失化项

$$\mathcal{L}_{inter}(Q_p, Q_s) = \frac{1}{N^2}||Q_p - Q_s||_F^2, \quad Q_p = \tilde{Q}_p \tilde{Q}_p^T, \tilde{Q}_{p[i,:]} = \frac{E_{p[i,:]}}{||E_{p[i,:]}||_2}$$

    - 其中$||.||_F^2$是矩阵的Frobenius范数

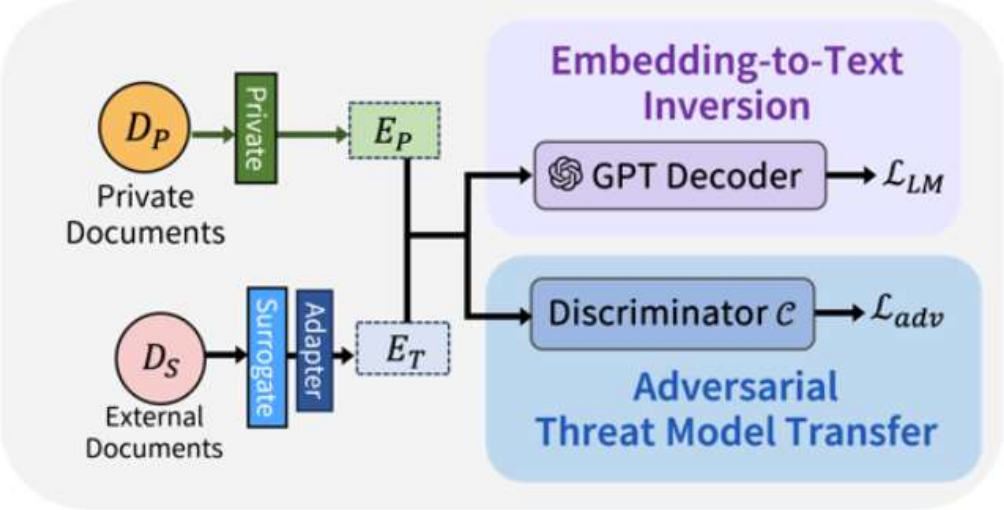


Transfer Attack 算法原理图

- 攻击可迁移性
  - 问题
    - GPT解码器在代理嵌入上进行训练，代理嵌入和私有嵌入之间的差异，可能导致攻击在针对私有嵌入时有效性降低
  - 对抗性训练
    - 设计鉴别器$\mathcal{C}$用于区分代理嵌入E_T和私有嵌入E_p
    - 同步优化代理模型$\hat{\phi}$

$$\mathcal{L}_{adv} = min_{\hat{\phi}} max_{\mathcal{C}} \mathbb{E}_{e_p \sim E_P}[log\mathcal{C}(e_p)] + \mathbb{E}_{e_t \sim E_T}[log(1 - \mathcal{C}(e_p))]$$

    - 其中$D_s = \{(x, \hat{\phi}(x))\}$，在训练阶段，交替训练判别器和代理模型
  - 结果
    - 通过对抗性训练，以实现代理模型生成鉴别器无法区分的嵌入



实验设计

- 目标模型
  - text-embeddings-ada-002、SBERT、ST5
- 对比算法
  - Direct Attack（ACL，2023）
- 数据集
  - Qnli：由维基百科文章中抽取的问答对
  - IMDB：电影评论数据集　　AG News：新闻文章数据集
- 评价指标
  - RougeL：基于n-gram的真实文本和重构文本之间的准确性和重叠度
  - Preplexity：通过测量语言模型预测给定单词序列的效果来评估语言模型的性能
  - Cos：评估潜在空间中的语义相似度
  - LLM-Eval：使用ChatGPT提供0到1之间的评分来评估预测和真实文本之间的相关性

实验设计　对比实验

- 对比试验（域内）

Dataset / Method	OpenAI				SBERT				ST5			
	RougeL	PPL	Cos	LLM-Eval	RougeL	PPL	Cos	LLM-Eval	RougeL	PPL	Cos	LLM-Eval
QNLI												
Direct Attack	0.1433	40.822	0.2797	0.2984	0.1264	27.127	0.3257	0.3194	0.1463	42.911	0.2226	0.2755
Transfer Attack	**0.2226**	**10.242**	**0.4772**	**0.4402**	**0.1934**	**11.633**	**0.4886**	**0.4280**	**0.1985**	**11.808**	**0.4121**	**0.3963**
Improv. (%)	55.3%	74.9%	70.6%	47.5%	53.0%	57.1%	50.0%	34.0%	35.6%	72.4%	85.1%	43.8%
IMDB												
Direct Attack	0.1133	20.549	0.2692	0.3818	0.1137	34.805	0.2891	0.3923	0.1103	24.939	0.2678	0.3909
Transfer Attack	**0.1991**	**12.953**	**0.4297**	**0.4528**	**0.1689**	**14.505**	**0.4467**	**0.4475**	**0.1571**	**14.839**	**0.3866**	**0.4295**
Improv. (%)	75.7%	36.9%	59.6%	18.6%	48.5%	58.3%	54.5%	14.0%	42.4%	40.4%	44.3%	9.8%
AGNEWS												
Direct Attack	0.0612	66.383	0.1162	0.2979	0.0538	286.16	0.1317	0.2742	0.0578	74.085	0.0980	0.2905
Transfer Attack	**0.1271**	**31.159**	**0.4301**	**0.4057**	**0.1067**	**36.793**	**0.4110**	**0.3839**	**0.1042**	**40.809**	**0.3697**	**0.3706**
Improv. (%)	107.0%	53.0%	270%	36.1%	98.3%	87.1%	212.0%	40.0%	80.2%	44.9%	277.2%	27.5%

- 实验结果
  - 在多个数据集和目标模型上的攻击结果显示，算法实现了重建效果大幅度领先

实验设计　对比实验

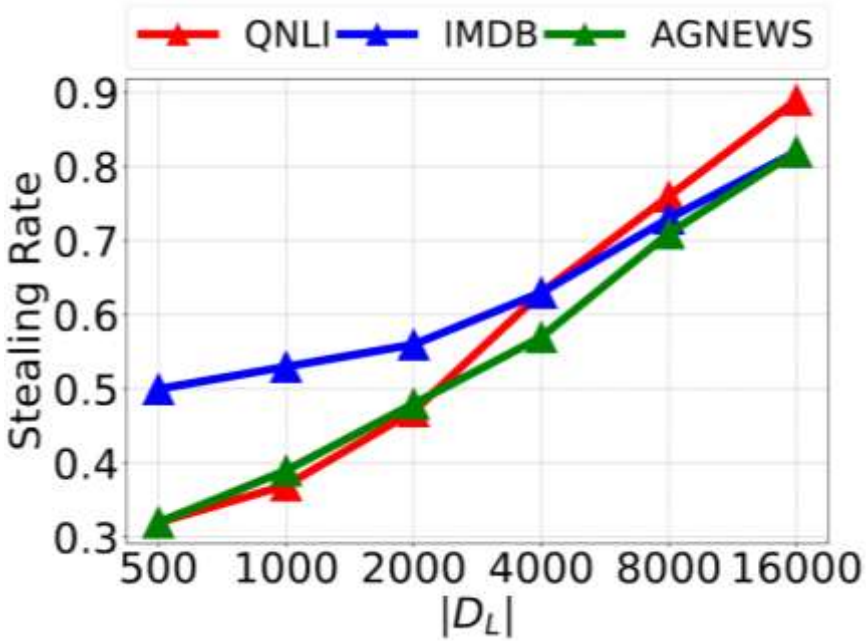
- 对比试验（域外）
  - 实验设置
    - 域外数据集：PersonaChat
    - 目标模型：SBERT
  - 实验结果
    - 相较于对比方法，算法在绝大多数数据集和指标上取得了更好的结果
    - 相较于域内实验，算法提升的幅度出现下降
  - 实验分析
    - 攻击者并不需要总是了解源域，仍然可以实现可靠的反演攻击
    - 预训练模型本身在大规模语料库上的训练，是在域外设置中，仍然有较好攻击效果的主要因素

Dataset / Method	RougeL	PPL	Cos	LLM-Eval
QNLI				
Direct Attack	0.1264	27.127	0.3257	0.3194
Transfer Attack	**0.1800**	**20.515**	**0.4445**	**0.3899**
Improv. (%)	42.4%	24.3%	36.5%	22.1%
IMDB				
Direct Attack	0.1137	34.805	0.2891	**0.3923**
Transfer Attack	**0.1685**	**27.819**	**0.4333**	0.3747
Improv. (%)	48.1%	20.1%	49.8%	-4.4%
AGNEWS				
Direct Attack	0.0538	286.16	0.1317	0.2742
Transfer Attack	**0.0984**	**103.40**	**0.3589**	**0.3497**
Improv. (%)	82.9%	63.8%	172.5%	27.5%

实验设计　消融实验&数据量实验

- 消融实验
  - 算法组件：代理模型、对抗训练、一致性正则化
  - 实验结果
    - 仅使用代理模型时，攻击效果相较于直接攻击出现提升
    - 在引入对抗训练或一致性正则化之后，嵌入相似度分别提升了7%和9%
- 数据量实验
  - 实验目的
    - 在泄露数据集D_L数据量不同情况下，对攻击结果的影响
  - 实验结果
    - 在泄露数据集数量足够的情况下，代理模型能够有效模拟目标模型

# D_L	Surrogate	Adv.	Consist Reg.	RougeL	Cos	LLM-Eval
500	✗	✗	✗	0.0617	0.0609	0.2436
	✓	✗	✗	0.1001	0.1310	0.2443
	✓	✓	✗	0.1192	0.1664	0.2550
	✓	✗	✓	0.1251	0.1801	**0.2686**
	✓	✓	✓	**0.1372**	**0.2031**	0.2663
8000	✗	✗	✗	0.1264	0.3257	0.3194
	✓	✗	✗	0.1701	0.4072	0.3598
	✓	✓	✗	0.1909	0.4742	0.4161
	✓	✗	✓	**0.1982**	**0.4902**	0.4266
	✓	✓	✓	0.1934	0.4886	**0.4280**



- 案例研究
  - MIMICIII临床笔记数据集
  - 检验方法对实体的重建效果
- 实验结果
  - 算法能够恢复98%的年龄、99%的性别
  - 算法在恢复疾病、症状、病史等更为复杂，更为特殊的命名实体时效果出现大幅度下降

Attack Methods	Sentences
Example 1	
Ground truth	59 year-old male with a history of cardiomyopathy ef 45-50% with pcm/icd who presented due to sob.
Transfer Attack	59 year-old male with a past of cardiomyopathy ef 45-50% with pcm/icd who presented due to sob.
Direct Attack	this is a 64 year old male with known mitral regurgitation since.
Example 2	
Ground truth	this is a 78 year-old female with a history of ild who presents with altered mental status.
Transfer Attack	This is a 78 year-old woman with a history of ild who presents with different mental status.
Direct Attack	this is an 80-year-old female with a history of tracheobronchomalacia, copd, who presents with abdominal pain.
Example 3	
Ground truth	this 73 year old white male has known aortic stenosis which has progressed with increasing dyspnea.
Transfer Attack	This 73 year old white male has identified aortic stenosis which has progressed with worsening dyspnea.
Direct Attack	67 year old male with history of aortic stenosis followed by serial echocardiograms.

Attack Methods	Age	Sex	Disease	Symptom	History
Transfer Attack	98.84%	99.47%	79.07%	79.45%	65.36%
Direct Attack	7.79%	94.73%	19.35%	22.22%	17.49%

算法总结 Transfer Attack

- 算法流程
  - 构建代理模型并利用一致化损失优化模型
  - 通过对抗训练再次优化代理模型
  - 训练GPT解码器实现文本重建
- 算法优势
  - 在少查询的情况下，实现了高可靠的攻击效果
- 算法不足
  - 长文本重建面临困难
  - 命名实体的重建效果不佳

特点总结与未来展望

特点总结与未来展望

- **Vec2Text**
  - 提出受控生成的文本嵌入模型反演攻击方法
  - 通过迭代搜索优化来训练攻击模型，将反演攻击任务视为条件文本生成任务
  - 攻击结果随文本长度上升而下降
- **Transfer Attack**
  - 考虑在减少查询次数的前提下，进行模型反演攻击
  - 攻击结果随文本长度上升而下降
- 未来发展
  - 对中长文本，如何实现更有效地重建
  - 特殊命名实体的重建效果如何提升

预期收获　回顾分析

- 预期收获
  - 掌握文本嵌入面临的模型反演攻击风险
    - 由嵌入生成原文本是实际可行的
    - 能够泄露文本含义、实体内容等敏感信息
  - 理解两种模型反演攻击方法的基本原理
    - 以条件文本生成的视角审视反演攻击任务
    - 以代理模型的方式减少目标模型查询量
  - 了解现有方法的缺陷以及未来发展方向
    - 现有的方法难以重建中长文本
    - 现有的方法对稍复杂的实体重建效果差

学会了吗?
学会了也不能攻击别人

参考文献

[1] Huang Y H, Tsai Y, Hsiao H, et al. Transferable Embedding Inversion Attack: Uncovering Privacy Risks in Text Embeddings without Model Queries[J]. arXiv preprint arXiv:2406.10280, 2024.

[2] John Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander Rush. 2023. Text Embeddings Reveal (Almost) As Much As Text. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing [C]. Singapore:ACL, 2023: 12448–12460.

知人者智，自知者明。胜人者有力，自胜者强。知足者富。强行者有志。不失其所者久。死而不亡者，寿。

谢谢！