Beijing Forest Studio

北京理工大学信息系统及安全对抗实验中心

人工智能生成内容检测

硕士研究生 刘佳

2025年01月05日

问题回溯

- 相关内容
  - 杨宗源《文本生成中的幻觉》——2023.08.20
  - 张浩然《视频深度伪造及检测技术——攻与防》——2023.02.19
  - 张凌浩《基于图结构处理的文本生成》——2022.02.27
  - 高依萌《预训练语言模型GPT3》——2021.02.07

内容提要

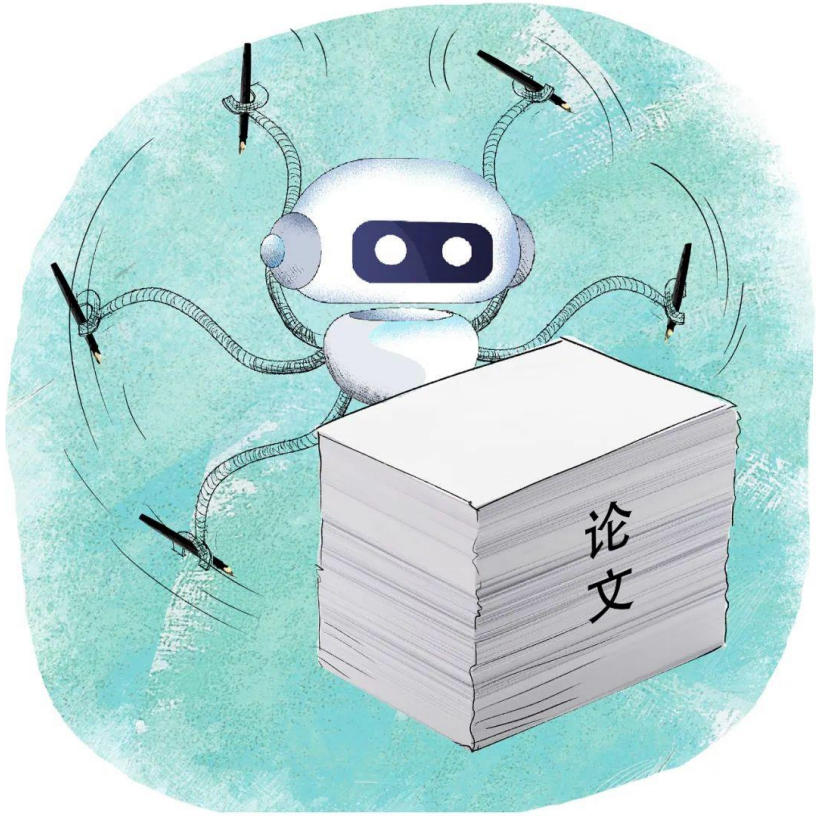
- 预期收获
- 案例引入
- 题目内涵解析
- 研究背景与意义
- 研究历史与现状
- 知识基础
- 算法原理
  - DetectGPT
  - DeTeCtive
- 特点总结与工作展望
- 参考文献

预期收获

- 预期收获
  - 1. 了解人工智能生成内容的基本概念和检测方法分类
  - 2. 理解两种AIGC检测方法的基本原理
  - 3. 了解现有方法的贡献以及未来发展方向

案例引入

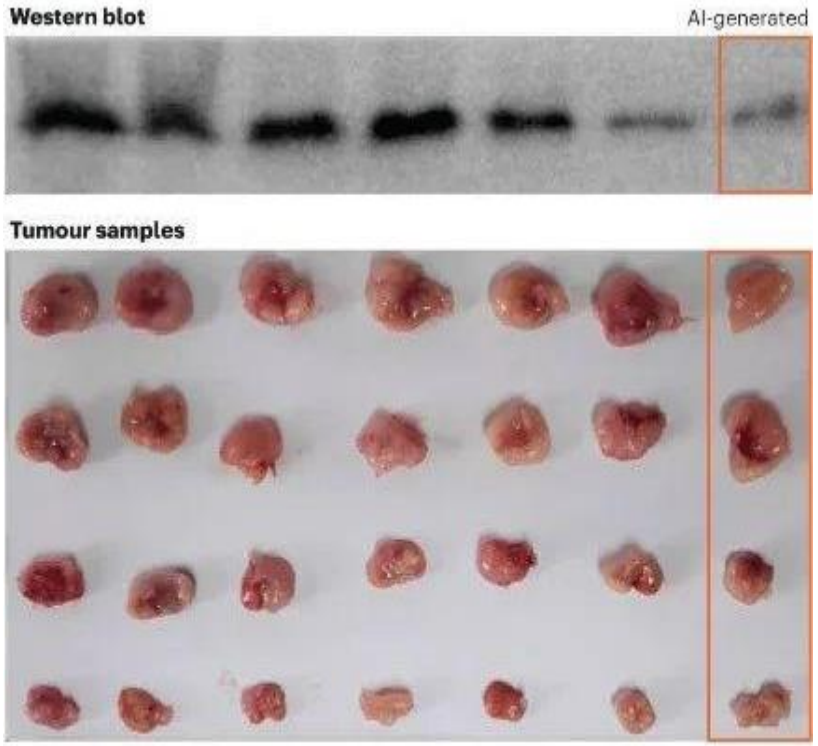
- AI生成万字论文
  - “这个问题的关键，是要找到关键的问题”
  - “后台数据管理的重点，是管理后台数据”
  - “如果一个人不胖，那一定是个瘦子”
  - 假大空的车轱辘话
  - 用AI写论文，用AI检测论文的AI率，再用AI把AI率降下去



当研究和写作过度依赖AI，算不算一种学术不端?

案例引入

- AI生成论文图像、假新闻
  - 从PS到AI工具的转变
  - 几乎无法与真实图像区分开来，至少用肉眼无法区分



埃菲尔铁塔着火图片 2023.01.18

埃菲尔铁塔着火视频 2024.12.24

案例引入

- AI生成与政治斗争
  - 2024美国大选遭遇虚假视频洪流
  - “深度伪造”视频的出现，可能对选举等政治和安全领域产生负面影响
  - “AI换脸”真假难辨，多国政要“躺枪”

政治“深伪”时刻到来?

如何识别这些生成的虚假内容?

题目内涵解析 什么是人工智能生成内容？如何检测？

- 人工智能生成内容（Artificial Intelligence Generated Content, AIGC）
  - 基于生成对抗网络、大型预训练模型等人工智能的技术方法，通过已有数据的学习和识别，以适当的泛化能力生成的相关内容
  - 具备高仿真度和自然性，难以与人类创作的内容区分
  - 两个阶段
    - 提取和理解用户意图信息
    - 根据提取的意图生成所需的内容

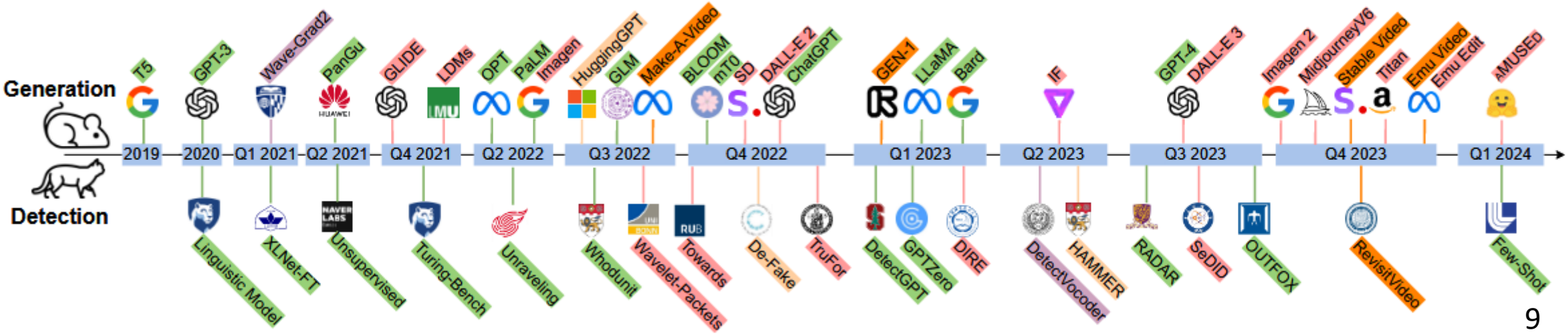
人类需求指令

⬇

满足需求的内容

题目内涵解析 AIGC检测

- 人工智能生成内容检测
  - 是 or 否：对生成的内容进行分析和识别，判断其是否由AI生成
  - 真 or 假：对AI生成内容的真假进行识别和鉴定，尤其新闻报道、社交平台、学术论文等领域
- 检测技术挑战
  - 生成内容逼真度渐高，传统检测方法难以鉴别
  - 生成模型更新迭代，检测技术需要不断适应新的生成技术



研究背景与意义

- 研究背景
  - AI生成内容已经被广泛应用于新闻生成、社交媒体内容创建、娱乐产业（如电影、音乐制作）以及教育和培训领域
  - 推动生产效率，同时带来信息真实性的挑战
- 研究意义
  - 信息安全和真实性
    - 有效的人工智能生成内容检测技术，可以帮助鉴别新闻、社交媒体帖子、学术文章等内容的真伪
  - 防止滥用
    - 通过检测技术可以防止AI生成技术的滥用，如伪造公众人物言论、制作虚假广告等

维护信息环境的安全、清洁和公正

研究历史与现状

基于统计检测

2006

J. Houvardas等人提出基于N-gram的作者身份识别的方法，捕捉文本的语言特征和样式，在AIGC检测中沿用，通过检测文本的语言特征与常见的人工文本之间的差异来进行判断

2019

早期的文本生成模型如GPT和BERT开始获得广泛应用，研究人员开始注意到这些模型在生成文本方面的能力。这期间的研究主要集中在模型性能的提升上

基于机器学习

2019

Liu 等人提出基于RoBERTa 的 GPT-2 检测模型，该模型通过有监督学习的方式进行微调，用于区分AI生成文本和非AI生成文本。这标志着使用传统监督学习方法来检测生成文本的初步尝试

2019

Solaiman等人探索零样本学习在AI文本检测中的应用，这些方法通过评估文本的每个词的对数概率来进行AI文本的检测

2020

Uchendu等人提出的Turing Bench方法基于人工智能生成文本的特征，结合深度学习模型进行检测，旨在为AI生成内容的检测提供基准

2022

Edward等人开发了一种专为GPT模型生成文本识别的检测器，提出了一种基于“困惑度”和“爆发度”的检测方法，能够有效地区分由GPT模型生成的文本和人工文本

基于对抗性攻击学习

2023

Hu等人提出了RADAR，通过对抗学习和数据增强技术，增强了检测模型对新生成模型的适应能力，能有效应对生成模型的伪造文本

2023

Mitchell 等人观察到人工智能生成的段落往往处于文本对数概率的负曲率中，提出了 DetectGPT，一种零样本 LLM 文本检测方法，来利用这一观察结果

2023

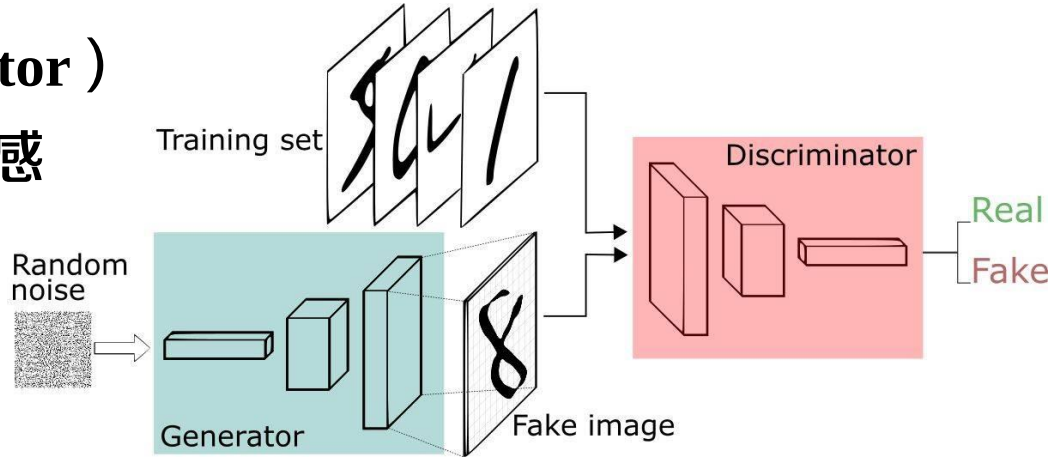
Koike 等人提出了OUTFOX，通过强化检测器与生成器的互动，使得检测器能够在面对多个不同生成器时，保持稳定的检测性能

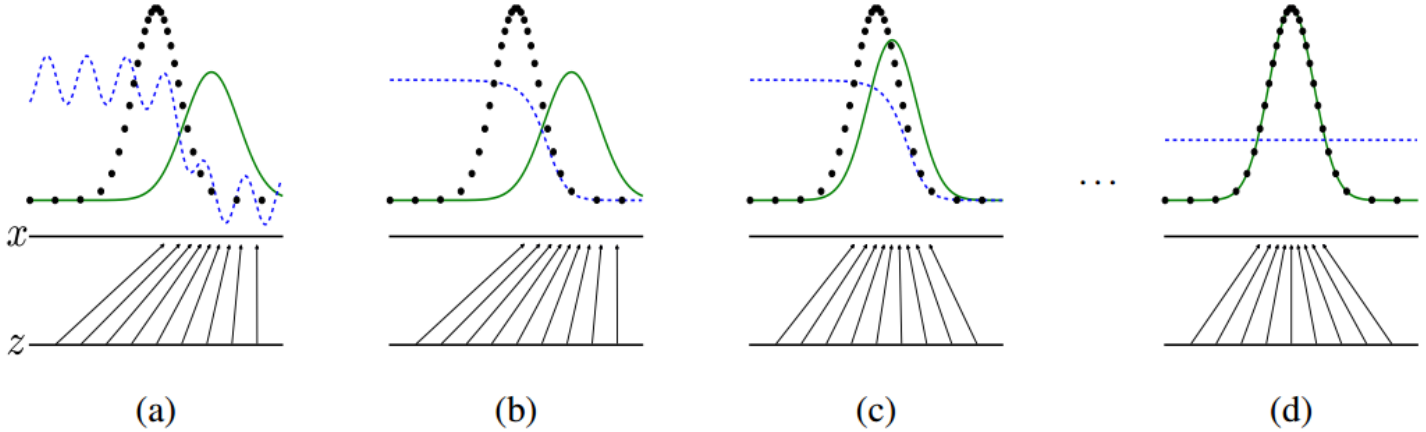
2024

Verma等人提出一种基于结构化搜索和线性分类的通用性检测方法，运行结构化搜索来获取可能的文本特征，然后根据所选特征训练分类器来预测文档是否是人工智能生成的

知识基础

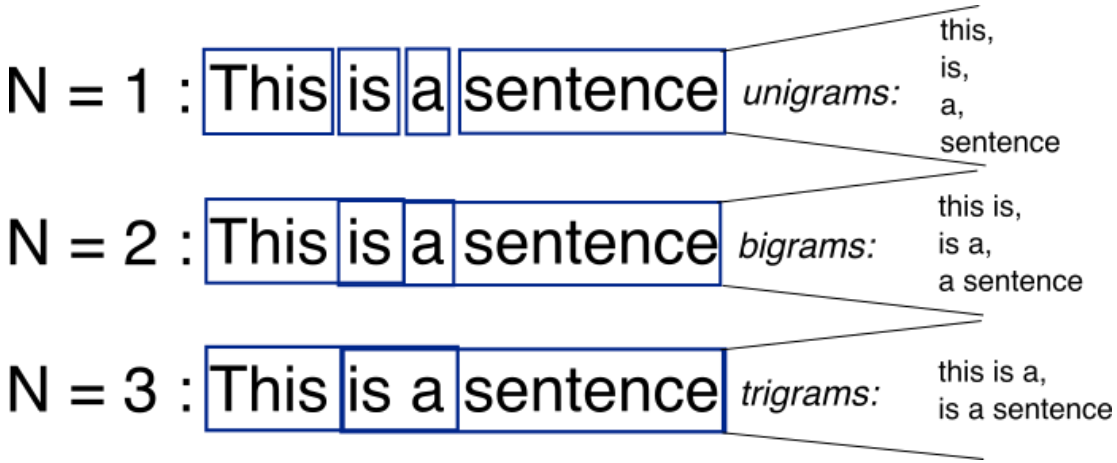
- 生成对抗网络（GANs）
  - 生成器（Generator）和判别器（Discriminator）
  - 相互对抗，从而提高生成内容的质量和真实感
- 训练过程
  - 固定「判别器D」，训练「生成器G」
  - 固定「生成器G」，训练「判别器D」
- GAN的应用
  - 生成数据集
  - 人脸生成
  - 图像转换
  - 图像修复





知识基础

- N-Gram模型
  - 将文本序列分解为连续的N个元素（如单词、音节或字符）的序列
  - 计算这些序列出现的概率
  - N的值决定了模型捕捉上下文信息的能力
- 常见的N-Gram模型
  - Unigram：只考虑单个元素出现的概率
  - Bigram：考虑两个元素联合出现的概率
  - Trigram：考虑三个元素联合出现的概率



- N-Gram模型在生成文本检测的应用
  - 检测重复的词组、不自然的词序
  - 比较输入文本与自然语言中的N-Gram统计分布差异，可以有效地检测生成文本
  - 无法处理长距离依赖

- 大型语言模型生成文本检测
  - 黑盒检测：依靠于收集人类和机器的文本样本来训练分类模型
  - 白盒检测：通过控制模型的生成行为或者在生成文本中加入水印（Watermark）来对生成文本进行追踪和检测
  - 黑盒检测器通常由第三方构建，例如 GPTZero，而白盒检测器通常由大型语言模型开发人员构建

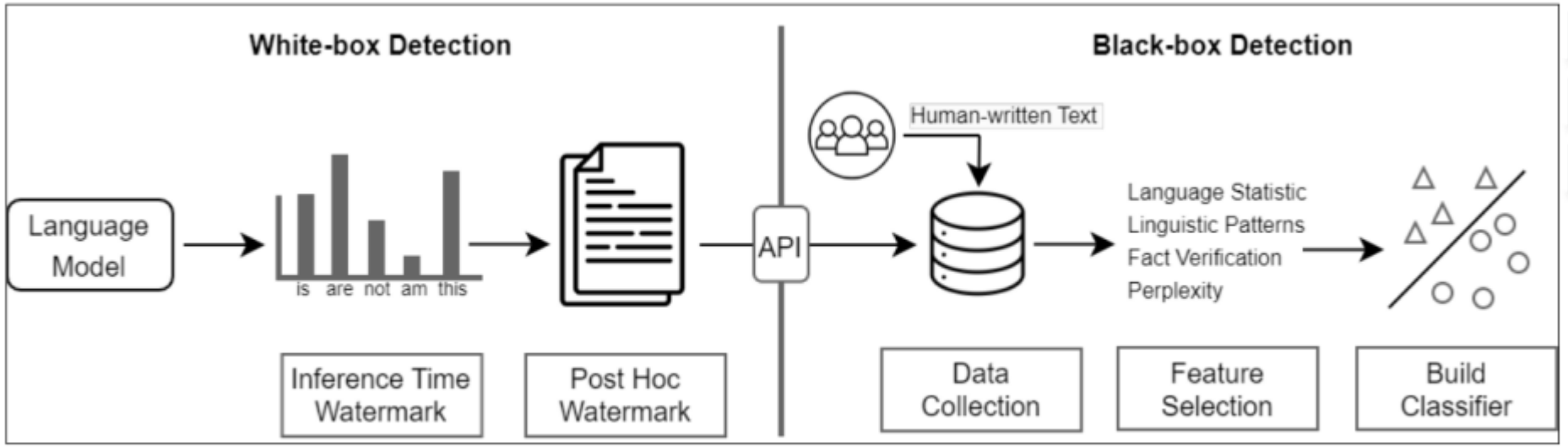
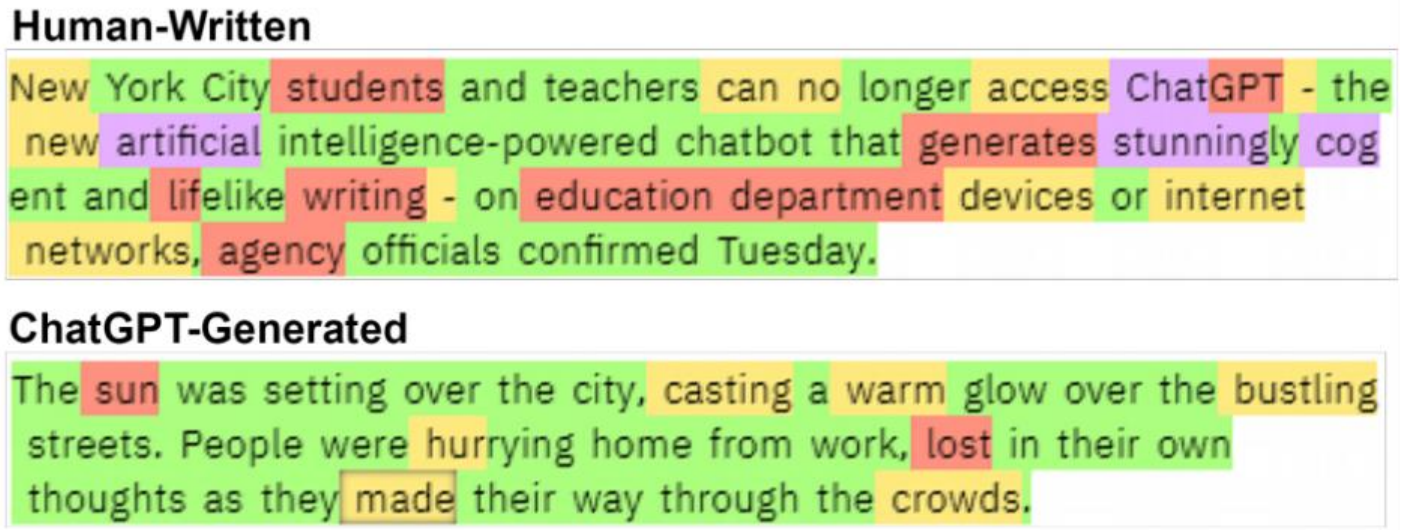


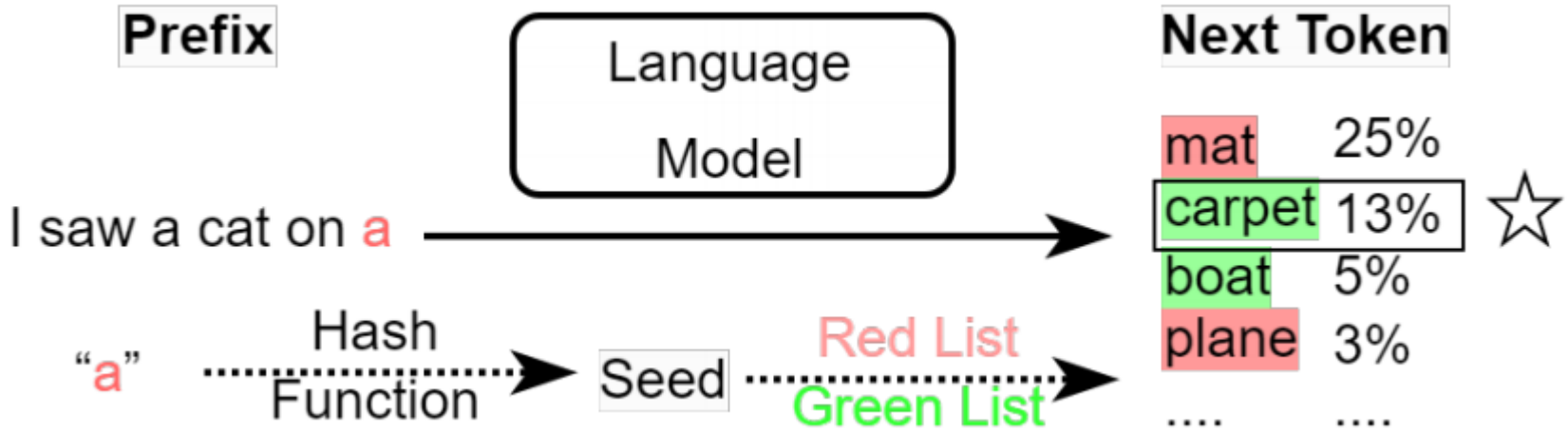
Figure 1. An overview of the LLM-generated text detection.

知识基础

- 黑盒检测
  - 数据收集
  - 特征选择
    - 统计特征：检查大型语言模型生成文本是否在一些常用的文本统计指标上于人类文本不同
    - 语言特征：语言学特征，比如词性，依存分析，情感分析等
    - 事实特征：大型语言模型常常会生成一些反事实的言论
  - 模型
    - 传统机器学习模型
      - SVM 等
    - 语言模型
      - BERT，RoBERTa



- 白盒检测
  - 对模型有完全访问权力，能通过改变模型的输出植入水印，达到检测的目的
  - post-hoc 水印
    - 大型语言模型生成完文本后，再在文本中加入一些隐藏的信息用于之后的检测
  - Inference time 水印
    - 改变大型语言模型对 token 的采样机制来加入水印



算法原理 DetectGPT

DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature

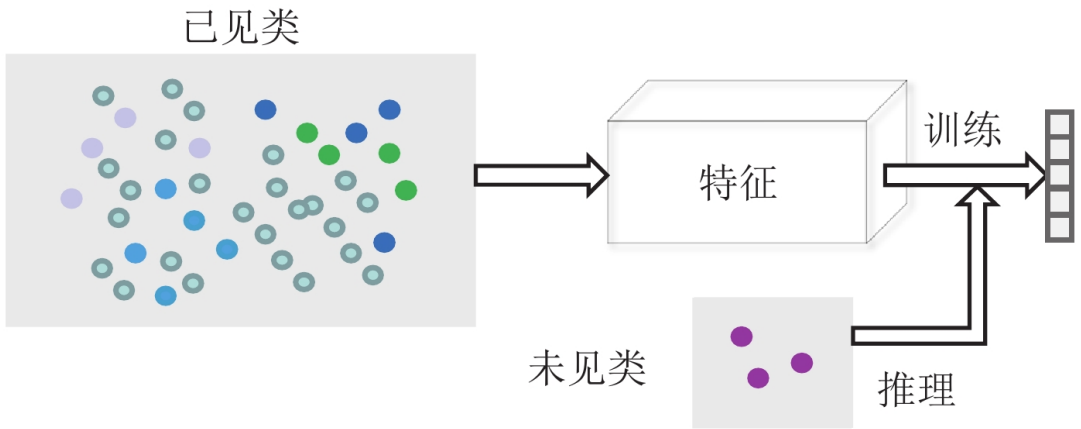
算法原理 TIPO

T	目标	零样本（Zero-Shot）条件下实现AIGT检测
I	输入	待检测的文本、疑似生成该文本的大语言模型
P	处理	1.使用扰动函数从原始文本生成扰动样本 2.计算原始文本及其扰动样本在源模型下的对数概率 3.计算原始文本的对数概率与扰动样本对数概率的平均值差异 4.将计算得到的对数概率差异进行标准化处理
O	输出	布尔值 True/False，文本是否可能由源模型生成

P	问题	在不依赖任何训练数据的零样本环境中，如何有效区分由人类与由模型生成的文本
C	条件	1.高质量语言模型 2.足够的计算资源 3.有效的扰动机制
D	难点	概率估计的准确性、扰动样本生成、性能与准确度的平衡
L	水平	ICML 2023 CCF A

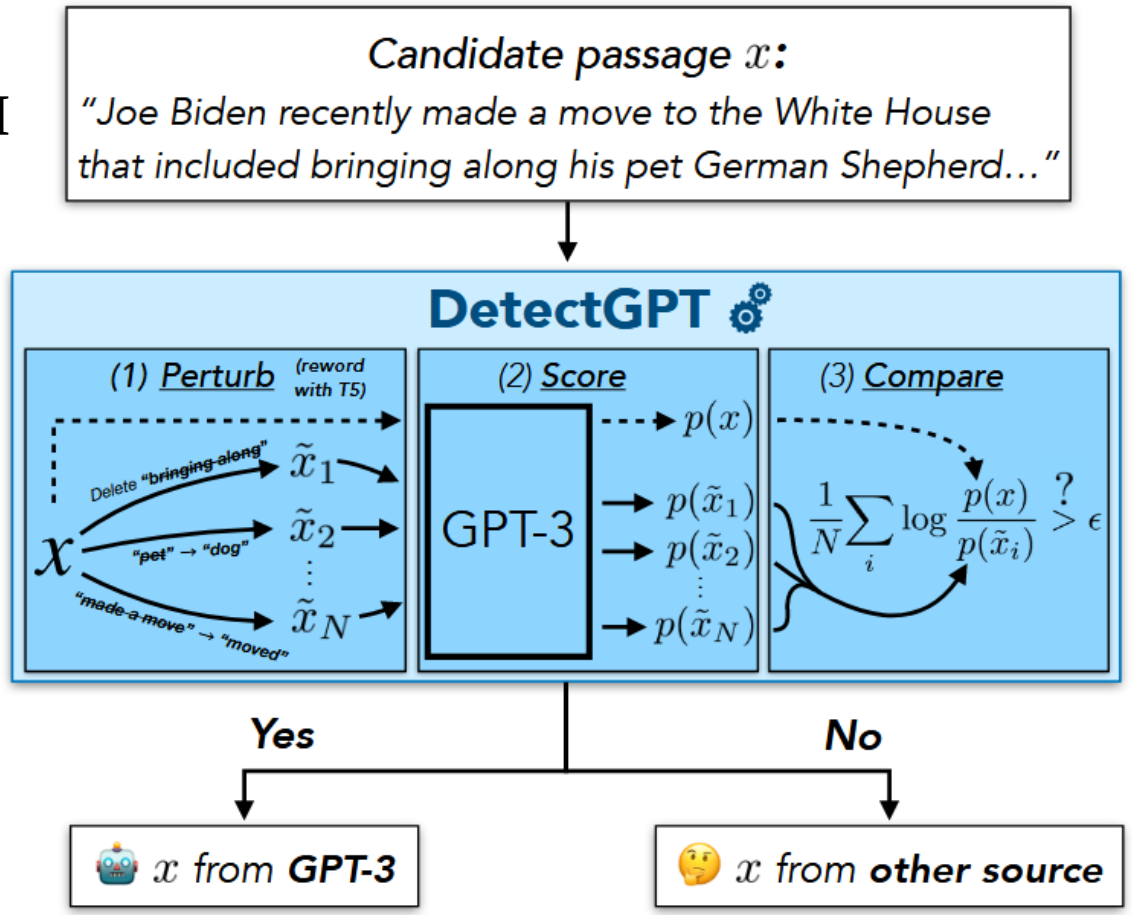
算法原理 创新说明

- **DetectGPT**
  - 零样本检测：不需要收集特定的训练数据集，也不需要训练一个独立的检测器模型，通过直接利用语言模型本身的概率输出实现在完全没有训练数据的情况下进行文本生成检测
  - 基于概率曲率的检测机制：生成文本倾向于聚集在对数概率函数的负曲率区域
  - 无需显式水印：不依赖于对生成文本进行任何形式的显式标记或水印，在不影响模型输出自然度的情况下，有效地应用于实际环境



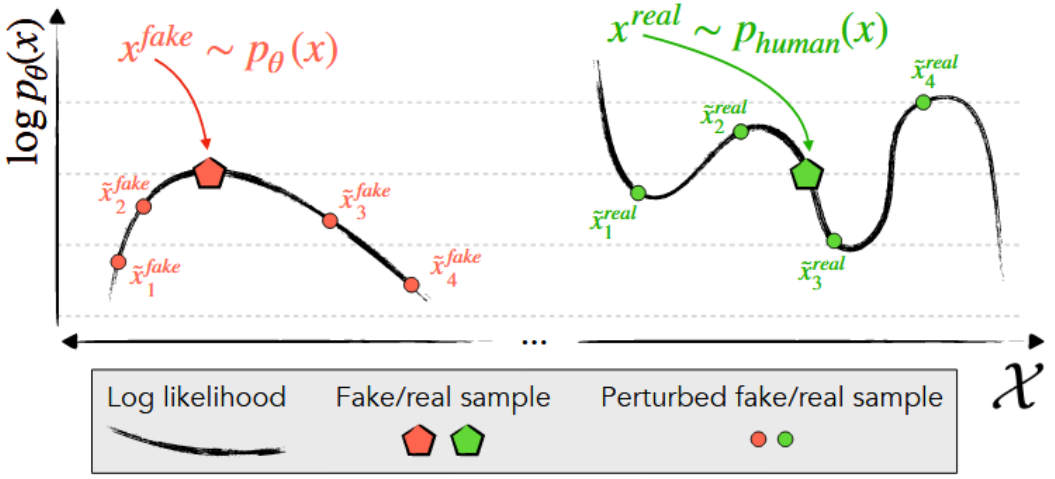
算法原理 DetectGPT

- DetectGPT
  - 目标：确定一段文本是否由特定的LLM生成，如GPT-3
  - 流程：
    - 生成扰动
    - 分别计算对数概率
    - 计算对数概率平均差异
    - 结果标准化处理
  - 结果评估
    - 平均对数比高，样本可能来自源模型



算法原理 DetectGPT

- 随机扰动的零样本机器生成文本检测
  - 局部扰动差异差距假说
    - 来自源模型p_θ的样本通常位于p_θ对数概率函数的负曲率区域
    - 对来自p_θ的样本x应用小扰动生成$\tilde{x}$
    - 机器文本 $\log p_\theta(x) - \log p_\theta(\tilde{x})$ 相比人类文本更大
  - 扰动函数$q(\cdot|x)$
    - 重写x的一个句子，同时保留x的含义的结果
    - $d(x, p_\theta, q) \triangleq \log p_\theta(x) - \mathbb{E}_{\tilde{x}\sim q(\cdot|x)} \log p_\theta(\tilde{x})$
    - T5或BERT这样的预训练模型
    - 手动或规则基础的方法
  - 不需要对抗样本的训练集



Algorithm 1 DetectGPT model-generated text detection

Input: passage x, source model p_θ, perturbation function q, number of perturbations k, decision threshold ϵ
$\tilde{x}_i \sim q(\cdot \mid x), i \in [1..k]$ // mask spans, sample replacements
$\tilde{\mu} \leftarrow \frac{1}{k}\sum_i \log p_\theta(\tilde{x}_i)$ // approximate expectation in Eq. 1
$\hat{\mathrm{d}}_x \leftarrow \log p_\theta(x) - \tilde{\mu}$ // estimate $\mathrm{d}(x, p_\theta, q)$
$\tilde{\sigma}_x^2 \leftarrow \frac{1}{k-1}\sum_i (\log p_\theta(\tilde{x}_i) - \tilde{\mu})^2$ // variance for normalization
if $\frac{\hat{\mathrm{d}}_x}{\sqrt{\tilde{\sigma}_x}} > \epsilon$ **then**
  return `true` // probably model sample
else
  return `false` // probably not model sample

实验设计　数据资源

- 数据资源

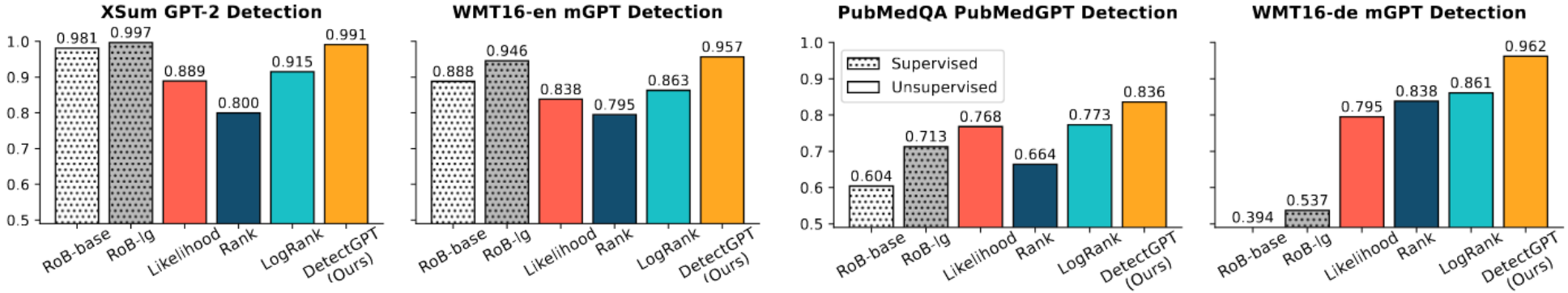
数据集	内容	数据来源	用途表示
XSum	新闻文章	公开数据集	假新闻检测
SQuAD	维基百科段落	公开数据集	机器编写的学术论文
RedditwritingPrompts	提示故事数据集	公开数据集	检测机器生成的创意写作提交
WMT16	英语和德语拆分	公开数据集	评估分布变化的稳健性
PubMedQA	人类专家编写的长格式答案	公开数据集	评估分布变化的稳健性

- 对比方法（零样本检测方法）
  - 平均对数概率法（Average Log Probability）
  - 排名和对数排名阈值法（Rank and Log-Rank Thresholding）： Solaiman（2019）
  - 信息熵法（Entropy-Based Methods）： Gehrmann（2019）

- 零样本机器生成文本检测

Method	XSum GPT-2	XSum OPT-2.7	XSum Neo-2.7	XSum GPT-J	XSum NeoX	XSum Avg.	SQuAD GPT-2	SQuAD OPT-2.7	SQuAD Neo-2.7	SQuAD GPT-J	SQuAD NeoX	SQuAD Avg.	WritingPrompts GPT-2	WritingPrompts OPT-2.7	WritingPrompts Neo-2.7	WritingPrompts GPT-J	WritingPrompts NeoX	WritingPrompts Avg.
$\log p(x)$	0.86	0.86	0.86	0.82	0.77	0.83	0.91	0.88	0.84	0.78	0.71	0.82	0.97	0.95	0.95	0.94	0.93*	0.95
Rank	0.79	0.76	0.77	0.75	0.73	0.76	0.83	0.82	0.80	0.79	0.74	0.80	0.87	0.83	0.82	0.83	0.81	0.83
LogRank	0.89*	0.88*	0.90*	0.86*	0.81*	0.87*	0.94*	0.92*	0.90*	0.83*	0.76*	0.87*	0.98*	0.96*	0.97*	0.96*	**0.95**	0.96*
Entropy	0.60	0.59	0.58	0.58	0.61	0.57	0.58	0.53	0.58	0.58	0.59	0.57	0.37	0.42	0.34	0.36	0.39	0.38
DetectGPT	**0.99**	**0.97**	**0.99**	**0.97**	**0.95**	**0.97**	**0.99**	**0.97**	**0.97**	**0.90**	**0.79**	**0.92**	**0.99**	**0.99**	**0.99**	**0.97**	0.93*	**0.97**
Diff	0.10	0.09	0.09	0.11	0.14	0.10	0.05	0.05	0.07	0.07	0.03	0.05	0.01	0.03	0.02	0.01	-0.02	0.01

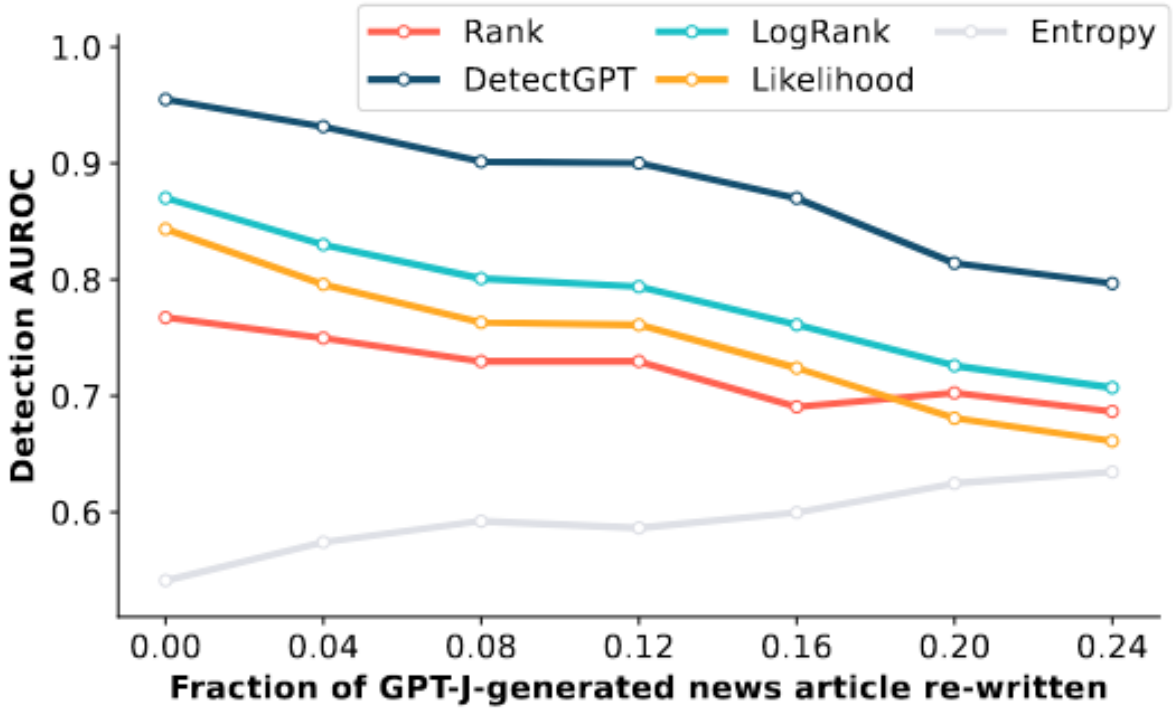
- 与有监督检测器的比较



	PMQA	XSum	WritingP	Avg.
RoB-base	0.64 / 0.58	**0.92** / 0.74	**0.92** / 0.81	0.77
RoB-large	0.71 / 0.64	**0.92** / **0.88**	0.91 / **0.88**	**0.82**
$\log p(x)$	0.64 / 0.55	0.76 / 0.61	0.88 / 0.67	0.69
DetectGPT	**0.84** / **0.77**	0.84 / 0.84	0.87 / 0.84	**0.83**

具有良好泛化能力
媲美传统监督学习方法

- 机器生成文本检测的变体
  - 人类手动编辑或完善机器生成的文本
  - 用 T5-3B 中的样本替换文本的 5 个单词跨度来模拟人工修改，直到文本的 r% 被替换，并报告 r 变化时的性能



四种方法均呈下降趋势
DetectGPT仍呈最强检测性能

算法总结

- 算法贡献
  - 提出一种新颖的零样本检测方法
    - 不需要额外的标注数据集来训练模型
    - 减少了对大量标注数据的依赖，在新领域或少数据情况下依然有效
  - 利用概率曲率分析
    - 基于语言模型概率分布的负曲率特性来区分人类和机器生成的文本
- 算法不足
  - 对扰动生成的敏感性
  - 对概率模型的依赖
  - 算法瓶颈
    - 计算每个文本及其扰动版本的概率分布需要显著的计算资源
    - 在更广泛的文本类型和更复杂的语言使用环境下的泛化能力仍有待进一步验证

DeTeCtive: Detecting AI-generated Text via Multi-Level Contrastive Learning

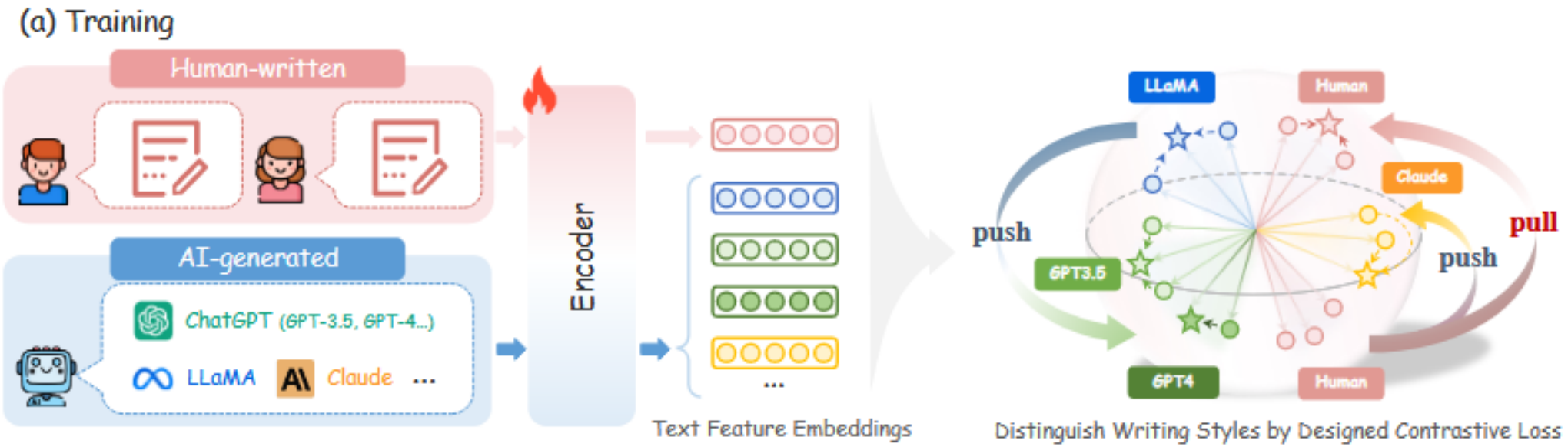
算法原理 TIPO

T	目标	区分不同作者的写作风格，不仅仅是简单的二分类
I	输入	待检测文本
P	处理	1.数据预处理与特征编码，文本清洗和标准化，转换为高维特征向量 2.对比学习训练，计算对比损失来优化模型，使得模型能区分不同来源（如不同LLM或人类）的文本风格 3.同时进行文本风格的多任务学习和AI与人类文本的二分类 4.构建特征数据库，用KNN算法来分类判断文本的来源
O	输出	文本样本是否由AI生成的分类结果 与已知风格的相似度评分

P	问题	传统的AI文本检测方法依赖于手工特征和监督学习的二分类，对于新出现的语言模型和分布外数据缺乏泛化能力
C	条件	有标签的训练数据，包括大量的人类文本和来自不同LLMs的文本
D	难点	文本风格多样性、对比学习优化、分布外泛化
L	水平	NeurIPS 2024 CCF A

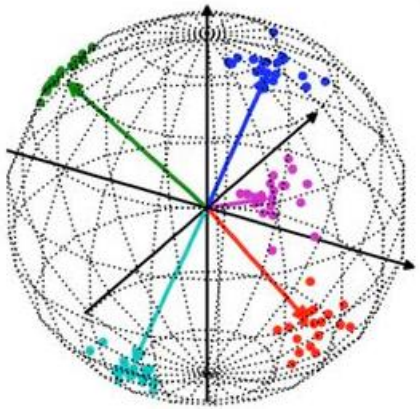
算法原理 创新说明

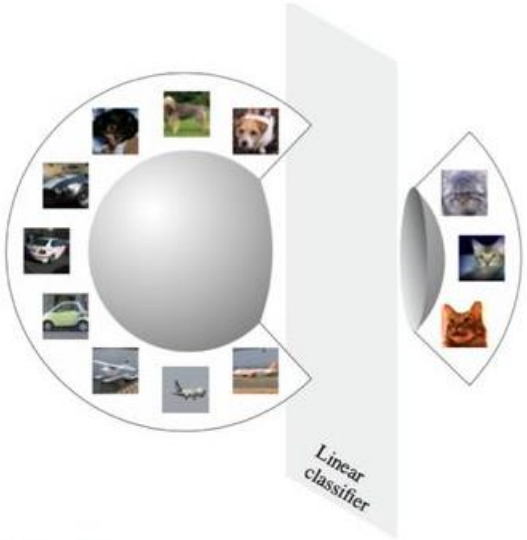
- **DeTeCtive**
  - 多级对比学习框架：比较不同文本样本之间的细微差异识别具体的生成模型
  - 多任务学习策略：学习区分不同机器生成文本之间的风格差异
  - 训练-自由增量适应（TFIA）：在现有的特征数据库中加入新数据的特征，而不需要对整个模型进行更新
  - 密集信息检索技术：利用预编码的特征数据库来快速比较和分类新的文本样本

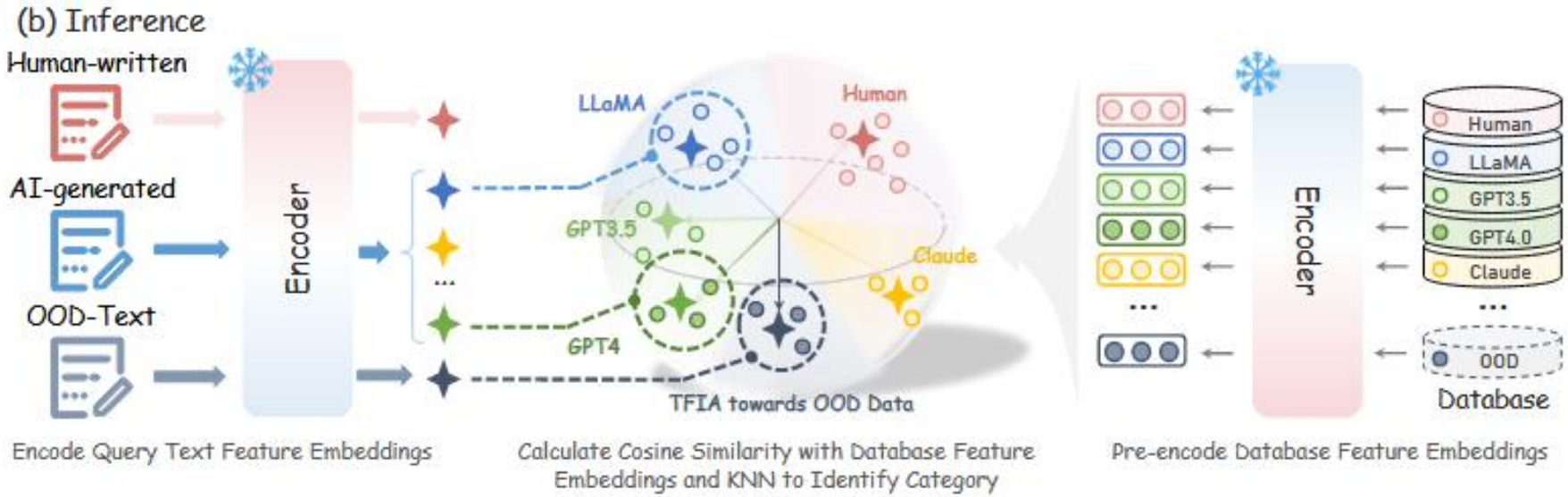


算法原理 DeTeCtive

- 多任务辅助多级对比学习算法概述
  - 核心：建立一个多维度的特征空间
  - 多级特征空间
    - 每个LLM视为一个独立的“作者”
    - 风格特征在特征空间中形成不同的集群
  - 对比损失函数
    - 帮助模型学习如何区分不同级别的样本关系





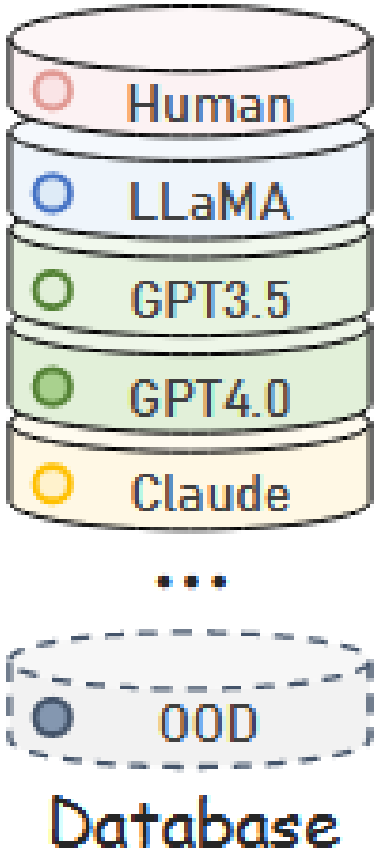


算法原理 DeTeCtive

- 多级对比学习具体实现
  - 样本相似度计算
    - 对于每对样本T_i和T_j，计算编码特征$\Phi(T_i)$和$\Phi(T_j)$之间的余弦相似度
    - 相似度度量$S(i,j)$用于评估样本之间的关系
  - 分级相似度约束
    - 对于人类文本T_i，其与其他人类文本T_j相似度大于任何LLM文本的T_k的相似度
    - 对于LLM生成的文本T_i，进一步区分不同LLM之间的相似度级别
  - 基于SimCLR框架的对比学习损失
    - $$\mathcal{L}_q = -\log \frac{\exp\left(\sum_{k\in K^+} \frac{S(q,k)}{\tau} \Big/ N_{K^+}\right)}{\exp\left(\sum_{k\in K^+} \frac{S(q,k)}{\tau} \Big/ N_{K^+}\right) + \sum_{k\in K^-} \exp\left(\frac{S(q,k)}{\tau}\right)}$$
    - 其中q表示当前样本， K^+为正样本集合， K^-为负样本集合，τ表示温度系数，N_{K^+}表示正样本集合的大小

算法原理 DeTeCtive

- 训练-自由增量适应 (TFIA)
  - 基本概念
    - 一种无需对模型重新进行训练即可适应新数据的方法
    - 适用于处理分布外（OOD）数据
  - 工作原理
    - 特征数据库扩展：当遇到新的或未见过的数据时，不进行模型的重新训练，而是直接使用已经训练好的模型对这些新数据进行特征编码
    - 特征融合：将这些新编码的特征集成到现有的特征数据库中，它允许模型通过增加新的数据特征来“学习”和适应新情况
  - 适合于动态变化的数据环境



实验设计　数据资源

- 数据资源

数据集	数据内容	数据来源
Deepfake	包括由 27 个不同的LLM生成的文本以及来自 10 个领域的多个网站的人工编写内容，包含 332K 训练数据和 57K 测试数据	公开数据集
M4	包含来自 8 个LLM、6 个领域和 9 种语言的数据	公开数据集
TuringBench	整合了单个领域内 19 个LLM的数据，形成了包含 112K 训练条目和 37K 测试条目的数据集	公开数据集

- 对比方法
  - RoBERTa
  - SCL (ICLR 2021)
  - Longformer (ACL 2024)
  - T5-Sentinel (EMNLP 2023)
  - Binoculars (ICML 2024)

- 评价指标
  - 准确率（Accuracy）
  - 精确度（Precision）
  - 召回率（Recall）
  - 平均召回率（Average Recall）
  - F1分数（F1 Score）

实验结果　对比实验

- 实验结果

Method	M4-monolingual		M4-multlingual		TuringBench		Deepfake	
	AvgRec	F1	AvgRec	F1	AvgRec	F1	AvgRec	F1
RoBERTa	88.70	88.44	80.01	84.44	99.59	99.29	87.30	88.37
SCL (ICLR 2021)	91.92	91.21	86.27	84.75	99.46	99.22	90.59	89.83
Longformer (ACL 2024)	80.99	81.42	84.68	83.00	99.40	98.95	90.53	89.76
T5-Sentinel (EMNLP 2023)	84.01	81.08	76.21	68.99	99.39	97.43	93.49	93.30
Binoculars (ICML 2024)	89.89	89.89	80.63	82.43	51.24	9.98	64.96	70.58
DeTeCTive (Ours)	**98.44**	**98.38**	**93.42**	**93.05**	**99.74**	**99.35**	**96.15**	**96.16**

Detection Scenario	Testbed Type	Longformer	GLTR	DetectGPT	FastText	DeTeCtive (Ours)
In-distribution	Cross-domains & Cross-models	90.53	55.42	60.48	78.80	**96.15**
	Cross-domains & Model-specific	96.10	77.58	62.31	83.02	**96.73**
	Domain-specific & Cross-models	93.51	63.08	60.48	81.67	**96.11**
	Domain-specific & Model-specific	96.60	87.45	86.37	94.54	**99.77**
Out-of-distribution	Unseen Models	86.61	57.49	62.31	68.61	92.19/**93.03**
	Unseen Domains	68.40	56.48	60.48	63.54	82.60/**89.63**

算法总结

- 算法贡献
  - 引入一种多级对比学习机制
    - 不仅识别文本是否为AI生成，而且能够指出是哪种类型的AI模型生成的
  - 训练-自由增量适应
    - 不需重新训练即可适应新领域或新模型生成文本
    - 减少了额外的训练成本
- 算法不足
  - 多级对比学习复杂性
  - 特征数据库维护
  - 泛化能力局限
  - 实时性能问题

特点总结与未来展望

特点总结与未来展望

- 特点总结
  - DetectGPT
    - 零样本检测能力
    - 基于概率曲率的判定
    - 不依赖于生成文本的显示标记或水印
  - DeTeCtive
    - 通过多级对比学习有效区分人类和各类AI模型生成的文本
    - 训练-自由增量适应策略允许模型快速适应新领域
- 未来展望
  - 提升算法普适性
  - 提升实时响应效率
  - 计算资源优化

参考文献

[1] Mitchell E, Lee Y, Khazatsky A, et al. Detectgpt: Zero-shot machine-generated text detection using probability curvature. International Conference on Machine Learning[C]. PMLR, 2023: 24950-24962.

[2] Guo X, He Y, Zhang S, et al. DeTeCtive: Detecting AI-generated Text via Multi-Level Contrastive Learning. The Thirty-eighth Annual Conference on Neural Information Processing Systems [C].

[3] Sadasivan V S, Kumar A, Balasubramanian S, et al. Can AI-generated text be reliably detected?[J]. arXiv preprint arXiv:2303.11156, 2023.

[4] Lin L, Gupta N, Zhang Y, et al. Detecting Multimedia Generated by Large AI Models: A Survey[J]. Authorea Preprints, 2024.

道德经

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！