基于特征融合的大模型生成文本作者检测

赵晰莹[1], 白梓萌[1], 张妍[1], 袁彩霞[1], 王小捷[1†]
[1]北京邮电大学
{zhaoxiying,zimengbai,1762895426,yuancx,xjwang}@bupt.edu.cn

摘要

大语言模型在高效生成文本的同时也带来了文本滥用的问题，如何有效地区分不同大模型生成的文本成为了关键的挑战。为了解决这个问题，本文首先构建了一个面向多分类的大模型生成文本检测任务的数据集LGT-AA，包含7个领域的人类和10个常用大模型生成的94k条文本；其次，本文提出了一种提取不同大模型生成文本的全局性区分性特征的方案，并与分布特征进行融合构建文本检测器，提升了对生成文本的检测能力。实验结果表明，本文提出的方法在不同模型组合下和不同生成模型类别下都取得了更优的性能。

关键词: 文本分类 ; 特征融合 ; 大语言模型

Feature Fusion-Based Large Language Model-Generated Text Authorship Attribution

Xiying Zhao[1], Zimeng Bai[1], Yan Zhang[1], Caixia Yuan[1], Xiaojie Wang[1†]
[1]Beijing University of Posts and Telecommunications
{zhaoxiying,zimengbai,1762895426,yuancx,xjwang}@bupt.edu.cn

Abstract

While Large Language Models (LLMs) can generate text efficiently, they also bring the problem of text abuse. How to effectively distinguish texts generated by different large models becomes a key challenge. In order to solve this problem, this paper firstly constructs a dataset LGT-AA oriented to multi-categorization for the task of LLM-generated text detection, with a total of 94k texts in 7 common domains that generated by human and 10 commonly used LLMs. Secondly, this paper proposes a scheme to extract discriminative features of different LLM-generated texts, constructs salient features by extracting the maximum pooling of hidden states in the last layer of the big model, and uses feature alignment to fuse with distributional features in order to construct a sentence-level detector, which improves the detection ability of the generated texts. Experimental results show that our approach achieves superior performance under different model combinations and different generative model classes.

Keywords: Text Classification , Feature Fusion , Large Language Model

† 通讯作者

1 引言

大语言模型（Large Language Model, LLM）展示了强大的语言生成能力，已经被广泛用来帮助人们生成新闻稿件、电子邮件等。

然而，对大模型的误用引起了学术界的担忧 [Sison et al.2024]。区别于虚假文本检测，进行大模型生成文本检测的重要原因是区分大模型生成内容和人类创作的边界，以应对实际风险，例如：维护信息透明性，防止大模型生成内容（如伪造评论 [Ignat et al.2025, Gambetti and Han2024]、虚假观点）冒充人类身份传播；保护学术与教育诚信，避免大模型代写论文或作业破坏评估体系的公平性 [Perkins et al.2023]；防止模型工业化生成低质文本挤占人类创作空间，维护内容生态平衡等。因此，有必要对大模型生成的文本进行检测，强大的检测机制对于各种使用大模型的应用程序提供了重要的安全保障。

由于大模型的优秀生成能力，大模型生成文本具有类似人类的超常流畅性和连贯性 [Kumar et al.2024]，与人类生成文本之间的分布差异往往并不显著 [Tian et al.2024]。未经培训的人类难以分辨大模型生成文本，即使是语言学专业的专家，检测结果也只略好于随机地猜测 [Li et al.2024]。此外，大模型生成文本往往被认为比人类生成文本更可信 [Spitale et al.2023]。综上所述，如何高效检测大模型生成文本成为了关键的挑战。

根据大模型生成文本检测分类的类别数量，可以将其分为二分类任务和多分类任务。二分类任务仅识别给定文本是人类撰写还是由大模型生成的，而多分类任务需要识别给定文本是人类撰写还是由多个候选大模型中的某一个生成的。目前已有一些大模型生成文本基准数据集 [Li et al.2024, Wang et al.2024, Dugan et al.2024, Macko et al.2023]，其中大部分是针对二分类任务的数据，其选用的大模型都比较少，常常局限于同一个模型的不同配置或版本。而多分类任务的数据集还比较少，且选用的模型较少。

在检测方法方面，现有的工作都常采用大模型生成token的logits作为检测文本的特征 [Mitchell et al.2023, Wang et al.2023]，可以在句子粒度进行检测，实际上，在更小粒度上进行检测并没有意义 [Chakraborty et al.2024]。然而，logits本质上是高维语义空间的低维投影，在信息压缩过程中不可避免地丢失了一部分细粒度语义特征和上下文动态关联。因此，其检测性能还达不到实际应用的需要。

为了解决以上问题，本文首先构建了一个新的面向大模型生成文本多分类检测任务的数据集LGT-AA(LLMs-Generated Text Dataset for Authorship Attribution)，该数据包含7个领域的人类和10个常用大模型生成的共94k条文本。为构建该数据，本文设计了针对不同文本类型的提示，使用了非同源大模型进行文本生成。特别地，我们对文本数据中存在的格式化特征进行了清洗，使得文本更贴合真实场景；进而，本文提出了一个大模型生成文本的多分类检测模型，其中包含一个新的特征提取模块，其分别提取了文本的分布特征和显著特征，通过维度扩展对齐特征维度以进行融合。所引入的基于大模型隐藏状态的显著特征，相对于SeqXGPT方法能够对分布特征这一基础判别框架进行补充优化。在LGT-AA数据集上的实验结果表明，本文提出的检测方法比强基线模型提升了10.0%的准确度，在所有模型分类标签上均达到了新的最优性能。

总体而言，我们工作的主要贡献如下所述：

- 构建了一个新的面向多分类大模型生成文本检测任务的数据集LGT-AA，支撑细粒度模型生成文本溯源研究，具有涵盖常用大模型和多领域数据的优势。
- 提出了一种新的提取不同大模型生成文本的区分性特征的方案，与分布特征进行组合构建检测器，提升了对生成文本的泛化检测能力。
- 在LGT-AA数据集上的实验结果表明，本文提出的检测方法比强基线模型提升了10.0%的识别性能，达到了新的最优性能。

2 相关工作

区别于人类撰写的文本，大模型生成文本（LLMs Generated Text, LGT）指的是通过大语言模型，利用巨量的训练数据和计算能力，自动生成的具有上下文理解和连贯性的自然语言文本。根据对大模型生成文本检测分类的类别数量，可以将其分为二分类任务和多分类任务。

2.1 面向二分类任务的大模型生成文本检测

二分类任务指区分人类撰写的文本和大模型生成的文本，只关注文本来源的真假而不区分具体的大模型类别。面向二分类任务的方法主要包括基于训练、基于统计和水印三类，其中水印需要对大模型本身进行修改，属于生成中检测的方法，而另外两种都属于生成后检测的方法。基于训练的方法指在同时包含人类和人工智能生成的文本示例的数据集上对预训练模型进行微调，以区分这两类文本。对于源模型未知的情况，常使用跨域迁移的方法进行检测 [Tian et al.2024]。基于统计的方法使用语言模型为文本生成分数，并从中创建统计特征。最初的方法侧重于检测语言特征的差异，使用困惑度（perplexity, PPL）、相对熵和风格相似性等统计量度 [Beresneva2016]，而近期的方法偏向使用更高级的特征，如扰动概率曲率 [Mitchell et al.2023]、条件概率曲率 [Bao et al.2024]、对数秩信息、token频率、token内聚性等。基于统计的方法往往需要访问模型的输出对数或损失进行检测。然而，许多商业公司提供的大模型服务在推理时并不公开模型的输出对数或损失。因此，这些方法不得不依赖本地代理模型来获取输出信息。然而，在线模型与本地代理模型之间的不一致性可能会导致检测性能低下 [Zhu et al.2023]。

2.2 面向多分类任务的大模型生成文本检测

二分类任务无法关注标签层面更细粒度的检测，比如模型生成文本的具体来源，因此出现了多分类任务。多分类任务指区分人类撰写的文本和大模型生成的文本，细分到大模型的具体类别，也被称为对于作者归属（Authorship Attribution, AA）的判别。目前面向多分类任务的检测方法研究较少，基于统计的方法难以对文本的具体来源进行准确的判别，更多都是基于训练的方法。Venkatraman等人提出的GPT-who，利用基于统一信息密度(Uniform Information Density, UID)的特征对每个大模型和人类的这一独特统计特征进行建模，以实现准确的多分类检测 [Venkatraman et al.2024]。生成概率反映了大模型对当前上下文的语义理解及生成下一个词的偏好。受DetectGPT [Mitchell et al.2023]的启发，Sniffer通过比较开源模型对之间的对比概率值进行分类 [Li et al.2023]，而SeqXGPT在Sniffer基础上结合了Fast-DetectGPT [Bao et al.2024]的条件概率曲率特征，通过比较开源模型的概率列表进行分类 [Wang et al.2023]。MAGRET对同一个文本使用多个闭源大模型生成“相似”文本，计算相似度等特征，然而这种方法成本高昂，依赖于API的生成效果和反应时间 [Huang et al.2025]。

2.3 大模型生成文本数据集

现有研究提出了一些大模型生成文本的基准数据集，它们选用各种大模型在给定的人类生成文本提示上模仿生成新的文本，但是这些基准数据集选用的生成模型都比较少，更偏向同一大模型的不同版本或不同参数配置（如LLaMA-2-7B/13B），而这些版本和配置之间的文本生成差异较小。例如，MAGE声称的27个模型实际上是7个模型和20个不同版本 [Li et al.2024]；M4GT-Bench仅在M4的5个模型上加入了GPT-4 [Wang et al.2024]；RAID [Dugan et al.2024]和MULTITuDE [Macko et al.2023]都使用了8个不同的模型。而这些benchmark也仅仅选用二分类任务在数据集上检测。

3 数据集

本节首先介绍数据集构建方法，之后对构建的数据集**LGT-AA**(**L**LMs-**G**enerated **T**ext Dataset for **A**uthorship **A**ttribution)进行详细的介绍。

3.1 数据集构建方法

数据集构建的流程如图1所示，包含源数据选择、提示设计、文本生成、后处理等四个环节，以下对方法具体介绍。

源数据选择：我们选用了HC3-English [Guo et al.2023]和Fast-DetectGPT [Bao et al.2024]构建的数据集作为源数据集。其中，HC3-English包含金融、医药、百科和开放问答四个领域的7,210个问题、7,210个人类回答和10,243个ChatGPT回答；Fast-DetectGPT构建的数据集包含新闻、学术文章和创意写作三个领域的1,300个人类撰写的文本和7,100的由7个大模型生成的文本。我们这样选择是因为前者为二分类方法常用的问答数据集 [Liu et al.2024, Yang et al.2024a]，后者为二分类方法常用的连续文本数据集 [Mitchell et al.2023, Yang et al.2024b, Yu et

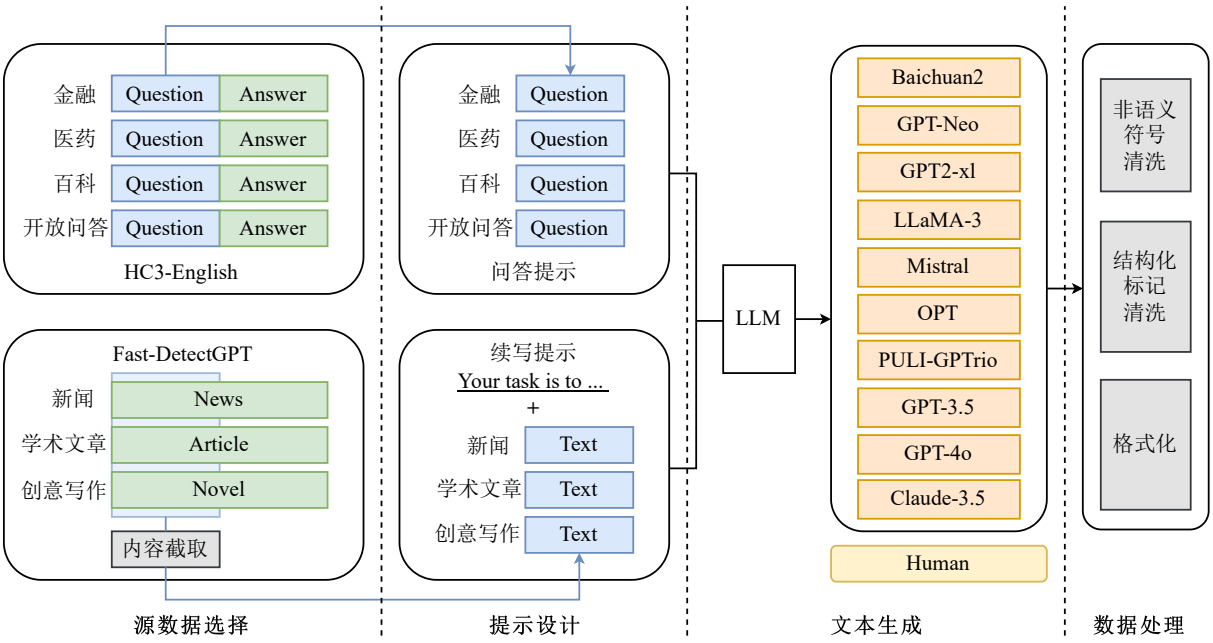


Figure 1: 数据集构建流程示意图

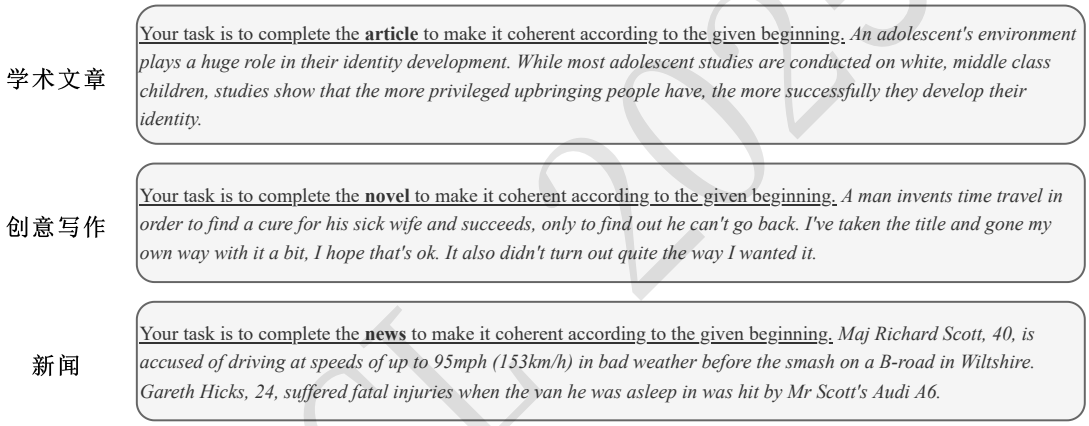


Figure 2: prompt示例

al.2024b]，而这两个任务分别考验了大模型在信息检索和逻辑连贯性上的能力 [Liu et al.2023]。这些数据集均选用了2020年及以前人类撰写的文本作为负样本，保证了人类撰写文本部分的真实性，不会受到潜在的大模型污染。

提示设计：对于问答数据集，我们直接提取问题作为prompt；对于连续文本数据集，为了维护生成的多样性，参考SeqXGPT [Wang et al.2023]的采样方法，我们随机提取文本开头的几个句子，所提取的句子满足单词数量$\in [20, 60]$且句子完整。此外，续写文本还需要补充额外的prompt，示例如图2所示。

文本生成：对于人类撰写的文本，我们直接应用了3.1中源数据集中人类撰写文本的部分；而对于大模型生成的文本，我们选用了Baichuan2-13B-Chat[1]，GPT-Neo-2.7B [Black et al.2021]，GPT2-xl [Radford et al.2019]，LLaMA-3-8B-Insturct[2]，Mistral-7B-Insturct-v0.3[3]，OPT-2.7B [Zhang et al.2022]，PULI-GPTrio [Yang et al.2023]，GPT-3.5-turbo，GPT-4o [Hurst et al.2024]，Claude-3.5共10个模型作为待检测文本的生成模型，这些模型在大模型生成文本检测领域及日常生活等真实场景中被广泛使用。其中，除

[1]https://huggingface.co/baichuan-inc/Baichuan2-13B-Chat
[2]https://huggingface.co/meta-Llama/llama-3-8b-Insturct
[3]https://huggingface.co/rubra-ai/Mistral-7B-Instruct-v0.3

了GPT3.5，GPT4o和Claude模型使用官网API，其它模型均为本地调用生成数据。

后处理：大模型生成文本中普遍存在的格式化特征（包括但不限于加粗符号、序列编号等结构化标记）对生成文本检测模型的性能具有显著影响 [Yu et al.2024a]。例如，SeqXGPT方法在清洗后的数据集中较清洗前检测准确度下降了23.66%，见附录A。为此，本文采用正则表达式和语法树分析相结合的方法，系统性地去除了生成文本中的结构化标记。

3.2 数据集分析

本文基于上述方法构建的大模型生成文本数据集LGT-AA含有94k条由10个不同的常用大模型生成的文本，具体模型和各个模型生成的数据规模如表1所示。这些文本均来自于真实场景下的人类撰写或由大模型在给定prompt的条件下生成。此外，我们还对数据集进行了词法分析实验，如附录B所示。

模型	金融	医药	开放问答	百科	学术文章	创意写作	新闻	合计
Baichuan2-13B-Chat	3475	816	1033	461	278	415	420	6898
GPT-neo-2.7B	3917	1241	1174	840	300	500	500	8472
GPT2-xl	3909	1111	1175	838	300	500	500	8333
LLaMA-3-8B-Insturct	3932	1244	1185	842	300	500	500	8503
Mistral-7B-Insturct-v0.3	3931	1247	1184	840	298	500	500	8500
OPT-2.7B	3859	1213	1102	829	300	500	500	8303
PULI-GPTrio	3813	1227	1149	724	297	492	471	8173
GPT-3.5-turbo	4503	1334	3546	842	300	499	499	11523
GPT-4o	3933	1244	1183	842	300	497	496	8495
Claude-3.5	3928	1248	1185	842	300	500	499	8502
human	3933	1248	1187	842	300	500	500	8510
合计	43133	13173	15103	8742	3273	5403	5385	94212

Table 1: 面向多分类任务的大模型生成文本数据集LGT-AA

4 方法

4.1 问题定义

检测给定的文本是由人类撰写还是由某一个已知大模型生成的问题可以定义为一个多分类任务：给定输入文本序列$X = [x_1, x_2, ..., x_\lambda]$（其中$\lambda$表示序列长度），以及一个标签集合$\mathcal{M} = [H, M_1, M_2, ..., M_k]$，其中H为人类，$M_i$为第$i$个大模型，检测任务在于构建分类器$\mathcal{F}$

$$\mathcal{F} : X \rightarrow m, m \subset \mathcal{M} \tag{1}$$

我们的模型包含三个模块，即输入编码器模块、特征提取模块和检测器模块。其中，输入编码器模块通过分词算法和预训练词表映射将输入文本X转换成模型可理解的编码序列$\mathbf{s} = [s_1, s_2, ..., s_n]$，特征提取模块使用大模型提取并融合两类不同特征，检测器模块构建分类器得到预测结果。在特征提取模块中，我们采用了多种不同的模型组合设置，即1-model（单模型），2-models（双模型组合），3-models（三模型组合）以及4-models（全部模型）。以4-models为例，模型的整体结构如图3所示。

以下分别介绍特征提取模块和检测器模块的具体内容。

4.2 特征提取模块

特征提取部分包含对两类不同特征的提取和融合：一类是分布特征，分布特征基于大模型生成概率，通过字节对齐及卷积网络提取文本局部信息；另一类是显著特征，利用大模型最后一层隐藏状态最大池化捕获文本全局信息，二者具有互补性。最后，将两部分特征进行融合，以得到最后的分类特征。以下分别介绍两类特征和特征融合部分。

4.2.1 分布特征

本文沿用SeqXGPT [Wang et al.2023]的对数特征作为模型的分布特征。具体而言，给定输入序列$\mathbf{s} = [s_1, s_2, ..., s_n]$，已知模型$M$，可以得到输入文本$X$对应的对数概率列表$ll_M(\mathbf{s})$，其中

$$ll_M(s_i) = logp_M(s_i|s_{<i}) \tag{2}$$

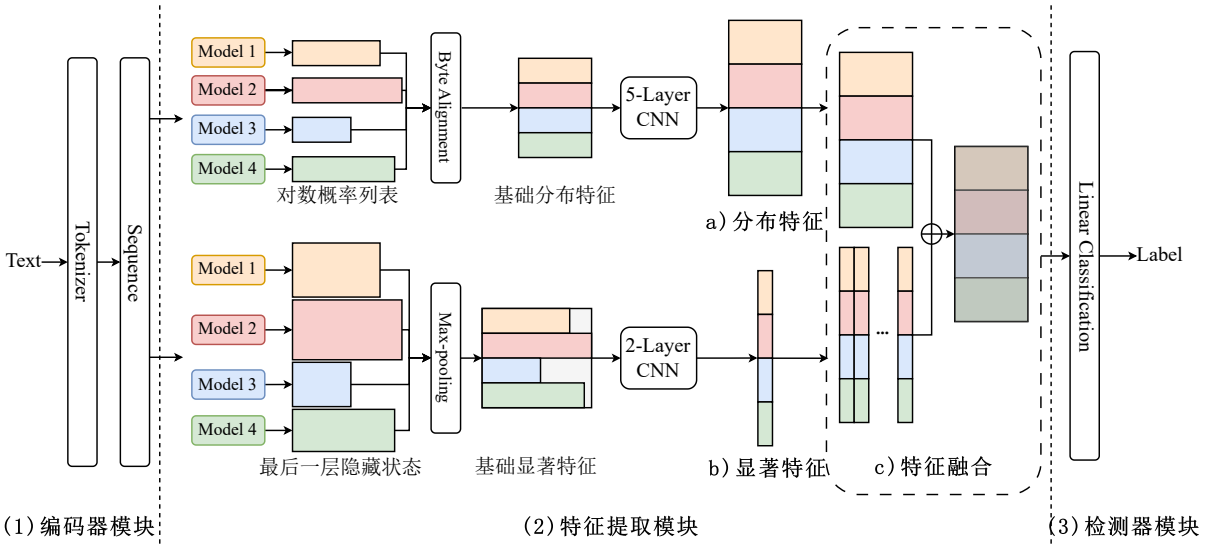


Figure 3: 以4-models设置为例的模型整体结构图

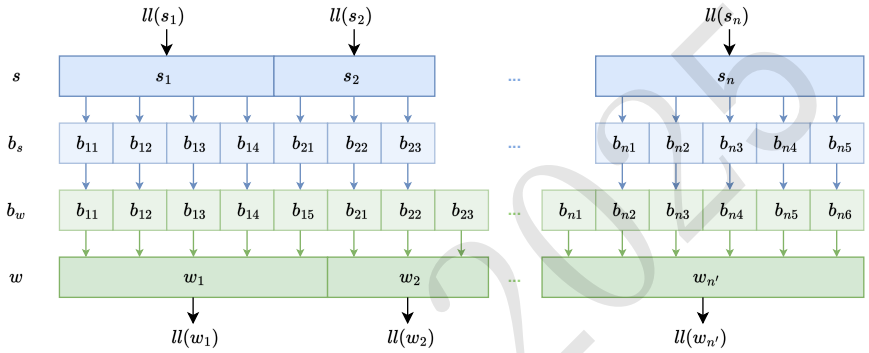


Figure 4: 对数概率列表字节表示对齐操作示例

然而，由于不同大模型的分词策略并不相同，其生成的对数概率列表不能通过简单地对齐加以合并。考虑到即使是不同大模型m的分词序列$\mathbf{s_m} = [s_{m1}, s_{m2}, ..., s_{mn}]$，其对应的字节表示$\mathbf{b_m} = [b_{m11}, b_{m12}, ...b_{m1len(s_{m1})}, ..., b_{mn1}, b_{mn2}, ...b_{mnlen(s_{mn})}]$均长度相同（其中$len(s_{mi})$表示由大模型$m$第$i$个token s的字节长度），可以进行对齐合并操作，如图4所示。因此，通过将分词序列转换成字节表示的方法，我们能够将不同长度的对数概率列表对齐到统一的分词序列$\mathbf{w} = [w_1, w_2, ..., w_{n'}]$，$n'$为序列$\mathbf{w}$长度，得到对齐后的对数概率列表，也即基础分布特征$ll_m(\mathbf{w})$，其中

$$ll_m(w_i) = logp_m(w_i|w_{<i}) \tag{3}$$

基础分布特征能够反映模型对语言模式和句法结构的理解。

基础分布特征是一个列表，不能被直接应用到已有的预训练模型中。参考语音信号的处理方法，我们将其分别通过五层卷积网络进行特征编码，以捕获基础分布特征中的上下文信息，从而得到分布特征$\mathbf{d} \in \mathbb{R}^{l \times e}$。其中，$l$为可检测文本的最大长度[4]，$e$为该五层卷积网络输出通道大小。该卷积网络的详细参数设计包括：卷积核大小为(5,3,3,3,3)，步长为(1,1,1,1,1)，输出通道为(64,128,128,128,64)。

4.2.2 显著特征

我们通过提取大模型最后一层隐藏状态的最大池化作为模型的显著特征。大模型的隐藏状态本质上是由多个自注意力机制和前馈神经网络层堆叠形成的中间表征空间，在Transformer架构中，每一层的隐藏状态都对应着不同抽象层级的特征编码。具体而言，底层隐藏状态主要捕

[4] l=1024

捉词汇的局部共现模式、句法结构等表层特征，而随着网络层数的加深，模型通过多头注意力机制的特征交互，逐步构建起文本的语义关联、逻辑推理等高阶抽象特征。

特别地，大模型的最后一层隐藏状态具有独特的表征价值。作为经过数层非线性变换后的最终中间输出，该层隐藏状态不仅融合了各层级特征提取的结果，还直接关联着模型输出层的预测结果生成过程。从信息流的角度来看，最后一层隐藏状态处于整个模型特征处理管道的末端，既保留了原始输入的语义核心要素，又经过深度语义蒸馏去除了冗余信息，形成了任务相关的紧凑特征表示。相对于分布特征，基于隐藏状态的显著特征能够包含更多的隐藏知识 [Gekhman et al.2025]。

给定输入序列$\mathbf{s} = [s_1, s_2, ..., s_n]$，模型第$l$层的隐藏状态可以表示为$H^l \in \mathbb{R}^{n\times d}$，其中$n$为序列长度，$d$为隐藏维度。其中，最后一层隐藏状态表示为$H^L$（$L$为模型总层数）。由于参数量和容量等影响，不同大模型的隐藏维度可能不同。因为隐藏维度一般较高，且最后一层隐藏状态中呈现明显的稀疏性，我们选择最大池化操作沿序列维度将最后一层隐藏状态H^L映射为低维向量，即对序列中所有token位置$i(1 \le i \le n)$，取隐藏状态的每个特征维度$j(1 \le j \le d)$的最大值，生成池化后的基础显著特征$\mathbf{vf} \in \mathbb{R}^n$，如式4所示。

$$vf_j = \underset{1\le i\le n}{max}\ H^L_{i,j} \tag{4}$$

最大池化操作通过保留最显著的特征激活值，突出文本中的关键语义单元，不需要因隐藏维度不同进行对齐操作，同时规避了平均池化对噪声敏感的问题 [Lee et al.2025]。具体而言，大模型最后一层隐藏状态中呈现明显的稀疏性，即对于任何一个输入样本，其对应的隐藏状态向量中，大部分维度的值接近零或非常小，只有少数维度被显著激活，即具有较大的正值或负值。平均池化计算所有激活值的平均值。如果特征向量中存在少量但显著、代表重要特征的强激活值和大量代表背景或噪声微弱与零值，平均值会被这些微弱与零值显著拉低，从而削弱重要特征的贡献。我们选用最大池化捕获稀疏表示中的关键信息，有效地将每个维度的信息浓缩为该维度上最重要的单个值，以便进行特征融合。

类似于4.2.1中的基础分布特征处理方法，我们将基础显著特征通过两层卷积网络进行特征编码，取序列维度每一位的均值，从而得到显著特征$\mathbf{v} \in \mathbb{R}^e$。其中，$e$为该两层卷积网络输出通道大小，为了便于特征对齐，与分布特征中五层卷积网络输出通道数相同。该卷积网络的详细参数设计包括：卷积核大小为(3,3)，步长为(1,1)，输出通道为(32,64)。

4.2.3 特征融合

对于通过大模型获得的分布特征和显著特征，还需要特征融合以实现进一步的分类检测，即将两个不同维度的特征向量进行特征对齐。在特征对齐过程中，我们将显著特征向量$v \in \mathbb{R}^e$沿序列维度重复l次进行扩展，形成扩展后的特征矩阵$v' \in \mathbb{R}^{l\times e}$，随后与同维度的分布特征矩阵进行融合。

具体而言，给定输入特征v，首先通过序列维度扩展操作将其复制l次，生成扩展矩阵

$$\mathbf{v}' = \mathbf{v} \cdot [1, 1, ..., 1]_l = \underbrace{[\mathbf{v}, \mathbf{v}, \ldots, \mathbf{v}]}_{l} \in \mathbb{R}^{l\times e} \tag{5}$$

随后，将扩展后的显著特征v'与分布特征$\mathbf{d} \in \mathbb{R}^{l\times e}$相加，得到对齐后的特征

$$\mathbf{f} = \mathbf{v}' + \mathbf{d} \in \mathbb{R}^{l\times e} \tag{6}$$

对于由不同模型得到的对齐特征f_i，我们将其在特征维度e进行拼接，得到融合特征

$$F = [f_1, f_2, \ldots, f_k] \in \mathbb{R}^{l\times(k\cdot e)} \tag{7}$$

最后，我们将上下文网络应用到融合特征$\mathbf{F}$中，以构建整个序列的上下文特征$\mathbf{F}' \in \mathbb{R}^{l\times(k\cdot e)}$。上下文网络包含两个transformer层，每层包含16个注意力头，隐藏层大小为512，采用最简单的绝对位置编码$\mathbf{P}$ [Vaswani et al.2017]。

$$\mathbf{F}' = \mathcal{T}^{(2)}_{16,512}(\mathbf{F}, \mathbf{P}) \quad where\ \mathcal{T}^{(n)}_{h,d} := Transformer(h, d)^{\times n} \tag{8}$$

4.3 检测器模块

值得注意的是，鉴于分布特征$\mathbf{d} \in \mathbb{R}^{l\times e}$，经过处理后的融合特征$\mathbf{F} \in \mathbb{R}^{l\times(k\cdot e)}$，融合特征$\mathbf{F}$在$l$维度也可以看作对应于统一分词序列$\mathbf{w}$中的每个token。因此，我们训练了一个线性分类器，将每个单词的特征投影到对应的模型标签上，通过统计每个单词的预测标签得到句子级检测概率，选择出现次数最多的标签作为句子的最终预测结果。

5 实验

5.1 实验设置

我们选用了GPT2-xl、GPT-Neo-2.7B、GPT-J-6B[5]、LLaMA-2-7B-hf[6]四个开源模型作为特征提取模块所使用的白盒模型，通过这四个开源模型，我们提取了两种不同的特征以进行实验。如4.1所述，我们对不同的模型组合均进行了实验。

我们将epoch设定为20，batch size设定为32，学习率lr设定为5e-5，weight decay设定为0.1。数据集按照9:1的比例随机分为训练集和测试集。对分布特征和显著特征提取部分的实验运行在NVIDIA Tesla A40 GPU上，特征融合及分类检测部分实验运行在NVIDIA GeForce RTX 3090 GPU上。我们选用了SeqXGPT [Wang et al.2023]作为多分类任务的基线模型、Fast-DetectGPT [Bao et al.2024]、DetectLLM-LRR [Su et al.2023]、DNA-GPT [Yang et al.2024b]作为二分类任务的基线模型。

此外，遵循以往工作，我们选用精确率**P**.和召回率**R**.分别评判每个类别的检测效果，选用**ACC**（分类的准确性）和**Macro F1**（每个大模型类别的F1分数）作为方法整体的评估标准。

5.2 实验结果及分析

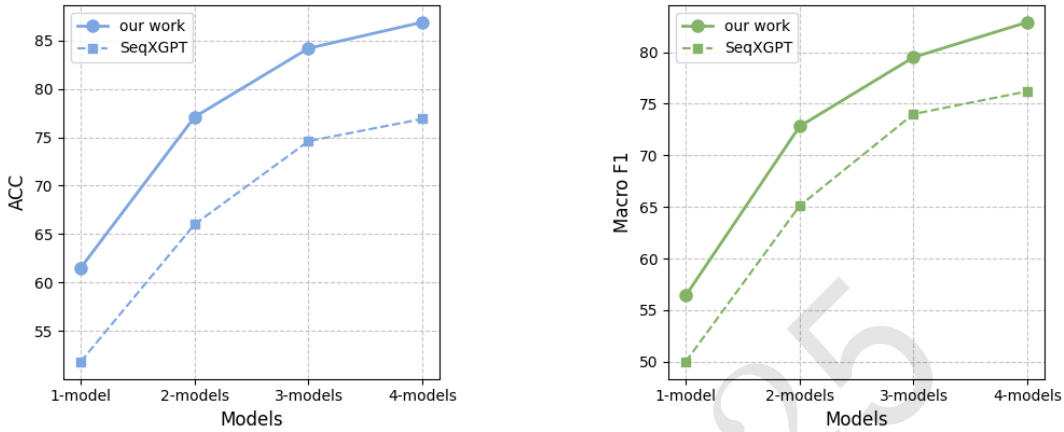
面向多分类任务的实验结果如表2所示，该表展示了在1-model (GPT-Neo)，2-models (GPT-Neo和LLaMA2)，3-models (GPT-Neo、LLaMA2和GPT2)以及4-models的条件下对比SeqXGPT我们的模型效果。其中，所展示的四种模型组合分别是在固定模型数量的条件下SeqXGPT在其上表现最好的组合，更详细的实验结果参见附录C。

models		1-model		2-models		3-models		4-models	
		SeqXGPT	our work	SeqXGPT	our work	SeqXGPT	our work	SeqXGPT	our work
Baichuan	P.	32.1	41.3	47.3	57.3	52.4	59.0	55.1	**62.2**
	R.	18.3	18.6	40.8	42.5	47.3	49.8	49.4	**54.6**
GPT-Neo	P.	71.8	78.8	87.3	93.1	93.5	96.6	93.6	**96.9**
	R.	63.0	75.6	85.8	93.2	91.8	96.3	92.9	**97.7**
GPT2	P.	52.7	59.3	77.5	82.2	95.1	97.9	95.1	**98.4**
	R.	52.4	61.1	65.4	75.1	95.2	98.1	95.9	**98.3**
LLaMA3	P.	40.6	48.3	51.9	61.8	57.1	67.3	60.8	**73.3**
	R.	44.7	43.0	55.5	58.0	65.6	69.6	67.4	**75.2**
Mistral	P.	50.2	51.6	65.7	73.1	71.3	75.8	73.2	**78.5**
	R.	41.8	60.9	55.3	71.8	61.1	73.3	61.8	**75.0**
OPT	P.	30.7	33.0	50.7	58.1	63.8	68.7	66.3	**75.3**
	R.	29.6	21.2	56.2	51.7	67.2	63.6	70.6	**73.7**
PULI	P.	45.9	45.5	61.9	66.7	75.9	82.4	79.2	**87.3**
	R.	42.5	48.2	64.3	76.9	73.1	86.1	75.8	**88.5**
GPT3.5	P.	78.3	76.4	84.0	83.2	84.8	86.5	85.4	**88.8**
	R.	77.4	80.2	82.4	85.8	86.5	88.3	88.2	**91.5**
GPT4o	P.	48.5	63.7	62.6	77.2	73.4	81.1	75.9	**84.3**
	R.	60.9	74.7	69.2	85.6	73.8	86.8	76.9	**88.5**
Claude	P.	51.3	60.4	65.8	73.0	76.4	78.4	79.5	**82.2**
	R.	59.4	62.0	66.4	72.2	72.6	74.7	76.4	**79.6**
human	P.	51.7	70.6	65.5	72.3	72.6	84.8	76.0	**87.3**
	R.	64.6	80.7	74.3	85.7	79.2	86.3	83.0	**87.6**
ACC		51.8	61.5	66.0	77.1	74.6	84.2	76.9	**86.9**
Macro F1		50.0	56.4	65.1	72.8	74.0	79.5	76.2	**82.9**

Table 2: 面向多分类任务的实验结果

[5]https://huggingface.co/EleutherAI/gpt-j-6b
[6]https://huggingface.co/meta-llama/Llama-2-7b-hf

总体而言，我们的方法相对SeqXGPT提升了10.0%的准确度和6.7%的Macro F1分数，精确率平均提升了6.8%，召回率平均提升了6.5%，这说明该方法在综合分类性能上显著优于SeqXGPT，不仅整体预测准确率更高，且在正类别的识别能力和预测结果的可靠性之间实现了更优的平衡。Macro F1分数的提升进一步表明，我们的方法在类别不均衡或复杂样本场景下的鲁棒性和泛化能力更强，通过改进特征提取有效提升了模型的判别边界清晰度。此外，在不同的模型组合条件下我们的方法效果均有明显的提升，准确度平均提升了10.1%，Macro F1分数平均提升了6.6%，这说明方法效果提升并不是特定模型组合的偶然结果，具有普适性和稳定性。



(a) ACC (b) Macro F1

Figure 5: 不同模型组合下ACC和Macro F1变化趋势

在不同的模型组合下，我们可以观察到ACC和Macro F1有相近的变化趋势，如图5所示。从3-models组合到4-models组合，增加了33%的成本但是仅仅提升了2.7%的准确度和3.4%的Macro F1分数，这一方面说明4-models组合设置已经饱和，再增加新的模型不会带来更多的性能提升，另一方面我们的方法在3-models设置上比SeqXGPT在4-models设置上的效果还要更好，在限制成本的条件下可以被应用，具有较高的性价比。

对于不同的生成模型，我们可以观察到GPT系列的模型相较于其它大模型更容易被检测，即使是性能更强的GPT4o也是如此。原因是：1）我们选用的特征提取模型中包含三个GPT系列的模型，对于生成概率的提取结果会和GPT系列模型的生成结果更加吻合，因此容易判别；2）对数概率这一用于大模型生成文本检测的特征的提出来自零样本方法DetectGPT对GPT-3生成文本扰动结果的观察，可能为GPT系列模型的固有特征而对其它大模型适用效果一般。

我们还计算了实验的效率分析。对于使用4-models作为特征提取模型的相同实验设置下，我们的方法对比SeqXGPT的效率如表3所示。由于引入了额外的显著特征，我们的模型相对于SeqXGPT增加了约27%的耗时。其中，在特征提取阶段，SeqXGPT和我们的方法平均提取每一条文本数据的特征耗时分别为0.70s和0.88s，在真实应用场景（如单条文本检测）下相差较小，可以认为我们的方法具有实际应用价值。

方法	特征提取	训练	合计
SeqXGPT	18.27h	6.57h	24.83h
Ours	23.03h	8.48h	31.52h

Table 3: 实验效率分析

此外，我们进行了面向二分类任务的实验，实验结果如表4所示，在该任务上也达到了更优的效果。虽然Fast-DetectGPT识别大模型生成文本的精确率更高，DetectLLM识别大模型生成文本的召回率更高，但是其识别人类撰写的文本的效果却不尽如人意，将更多的文本误认为大

模型生成，实际应用价值较低,存在着较高的误检风险。此外，误检风险可能会造成较为深远的社会后果，如“假阳性”风险下学术论文被错误标记为大模型生成，损害学术声誉和未来风险。总体而言，在精度更低的二分类实验中，我们的方法也能准确识别待检测文本的作者。

models		Fast-DetectGPT	DetectLLM	DNA-GPT	SeqXGPT	our work
LLM	P.	99.2	90.1	90.1	98.2	98.6
	R.	64.9	99.1	89.1	98.5	98.5
human	P.	21.3	3.7	9.2	86.8	87.1
	R.	94.6	0.3	10.1	84	87.5
ACC		67.6	89.4	81.4	97.0	97.4
F1		37.7	47.5	49.6	91.9	92.9

Table 4: 面向二分类任务的实验结果

5.3 消融实验

为了验证方法中每个模块的有效性，我们在4-models的条件下分别对分布特征模块和显著特征模块进行了消融实验，如表5所示。

models		our work	w/o 分布特征	w/o 显著特征
Baichuan	P.	62.2	33.5	55.1
	R.	54.6	11.6	49.4
GPTNeo	P.	96.9	48.6	93.6
	R.	97.7	62.5	92.9
GPT2	P.	98.4	38.8	95.1
	R.	98.3	35.7	95.9
LLaMA3	P.	73.3	35.1	60.8
	R.	75.2	34.4	67.4
Mistral	P.	78.5	42.7	73.2
	R.	75.0	42.9	61.8
OPT	P.	75.3	33.6	66.3
	R.	73.7	21.5	70.6
PULI	P.	87.3	44.1	79.2
	R.	88.5	26.4	75.8
GPT3.5	P.	88.8	62.6	85.4
	R.	91.5	75.2	88.2
GPT4o	P.	84.3	54.7	75.9
	R.	88.5	70.4	76.9
Claude	P.	82.2	50.9	79.5
	R.	79.6	55.4	76.4
human	P.	87.3	60.8	76.0
	R.	87.6	70.2	83.0
ACC		86.9	48.9(-38.0)	76.9(-10.0)
F1		82.9	44.9(-38.0)	76.2(-6.7)

Table 5: 消融实验结果

总体而言，w/o 分布特征任务准确度下降了38.0%，Macro F1分数下降了38.0%；w/o 显著特征任务准确度下降了10.0%，Macro F1分数下降了6.7%。这说明两个特征模块都对实验有正向贡献，存在正向协同效应。其中，分布特征提供了基础判别框架，显著特征对其进行了补充优化，二者结合实现了最佳效果。

6 结论

本文提出了一个面向多分类大模型生成文本检测任务的数据集LGT-AA，支撑细粒度模型生成文本溯源研究，涵盖了常用大模型和多领域的数据。本文还提出了提取不同大模型生成文本的区分性特征的方案，通过提取大模型最后一层隐藏状态的最大池化构建显著特征，使用特征对齐与分布特征进行融合以构建句子级检测器，提升了对生成文本的检测能力。对比最优基线SeqXGPT，我们的方法提升了10.0%的准确度和6.7%的Macro F1分数，在不同模型组合下和不同生成模型类别下都达到了更优越的效果。我们未来的工作包括对未知模型、混合文本和对抗攻击鲁棒性的进一步研究。

第二十四届中国计算语言学大会论文集，第598页-第612页，济南，中国，2025年8月11日至14日。

参考文献

Guangsheng Bao, Yanbin Zhao, Zhiyang Teng, et al. 2024. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In *Proceedings of the 12th International Conference on Learning Representations*.

Daria Beresneva. 2016. Computer-generated text detection using machine learning: A systematic review. In *International Conference on Applications of Natural Language to Information Systems*, pages 421–426. Springer.

Sid Black, Gao Leo, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March.

Souradip Chakraborty, Amrit Bedi, Sicheng Zhu, Bang An, Dinesh Manocha, and Furong Huang. 2024. Position: On the possibilities of ai-generated text detection. In *Forty-first International Conference on Machine Learning*.

Liam Dugan, Alyssa Hwang, Filip Trhlík, et al. 2024. RAID: A shared benchmark for robust evaluation of machine-generated text detectors. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12463–12492, Bangkok, Thailand, August. Association for Computational Linguistics.

Alessandro Gambetti and Qiwei Han. 2024. Aigen-foodreview: a multimodal dataset of machine-generated restaurant reviews and images on social media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 18, pages 1935–1945.

Zorik Gekhman, Eyal Ben David, Hadas Orgad, et al. 2025. Inside-out: Hidden factual knowledge in llms. *arXiv preprint arXiv:2503.15299*.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Yifei Huang, Jiuxin Cao, Hanyu Luo, Xin Guan, and Bo Liu. 2025. Magret: Machine-generated text detection with rewritten texts. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8336–8346.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Oana Ignat, Xiaomeng Xu, and Rada Mihalcea. 2025. MAiDE-up: Multilingual deception detection of AI-generated hotel reviews. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1636–1653, Albuquerque, New Mexico, April. Association for Computational Linguistics.

Sandeep Kumar, Mohit Sahu, Vardhan Gacche, et al. 2024. 'quis custodiet ipsos custodes?' who will watch the watchmen? on detecting AI-generated peer-reviews. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22663–22679, Miami, Florida, USA, November. Association for Computational Linguistics.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, et al. 2025. NV-Embed: Improved techniques for training LLMs as generalist embedding models. In *Proceedings of the 13th International Conference on Learning Representations(Spotlight)*, Singapore, April.

Linyang Li, Pengyu Wang, Ke Ren, et al. 2023. Origin tracing and detecting of llms. *arXiv preprint arXiv:2304.14072*.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, et al. 2024. MAGE: Machine-generated text detection in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 36–53, Bangkok, Thailand, August. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, et al. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore, December. Association for Computational Linguistics.

Shengchao Liu, Xiaoming Liu, Yichen Wang, et al. 2024. Does DetectGPT fully utilize perturbation? bridging selective perturbation to fine-tuned contrastive learning detector would be better. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1874–1889, Bangkok, Thailand, August. Association for Computational Linguistics.

Dominik Macko, Robert Moro, Adaku Uchendu, et al. 2023. MULTITuDE: Large-scale multilingual machine-generated text detection benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9960–9987, Singapore, December. Association for Computational Linguistics.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*, pages 24950–24962. PMLR.

Mike Perkins, Jasper Roe, Darius Postma, James McGaughran, and Don Hickerson. 2023. Game of tones: Faculty detection of gpt-4 generated content in university assessments. *arXiv preprint arXiv:2305.18081*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Alejo Jose G Sison, Marco Tulio Daza, Roberto Gozalo-Brizuela, and Eduardo C Garrido-Merchán. 2024. Chatgpt: More than a "weapon of mass deception" ethical challenges and responses from the human-centered artificial intelligence (hcai) perspective. *International Journal of Human–Computer Interaction*, 40(17):4853–4872.

Giovanni Spitale, Nikola Biller-Andorno, and Federico Germani. 2023. Ai model gpt-3 (dis) informs us better than humans. *Science Advances*, 9(26):eadh1850.

Jinyan Su, Terry Yue Zhuo, Di Wang, and Preslav Nakov. 2023. Detectllm: Leveraging log rank information for zero-shot detection of machine-generated text. *arXiv preprint arXiv:2306.05540*.

Yuchuan Tian, Hanting Chen, Xutao Wang, et al. 2024. Multiscale positive-unlabeled detection of AI-Generated texts. In *Proceedings of the 12th International Conference on Learning Representations(Spotlight)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Saranya Venkatraman, Adaku Uchendu, and Dongwon Lee. 2024. GPT-who: An information density-based machine-generated text detector. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 103–115, Mexico City, Mexico, June. Association for Computational Linguistics.

Pengyu Wang, Linyang Li, Ke Ren, et al. 2023. SeqXGPT: Sentence-level AI-generated text detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1144–1156, Singapore, December. Association for Computational Linguistics.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, et al. 2024. M4GT-bench: Evaluation benchmark for black-box machine-generated text detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3964–3992, Bangkok, Thailand, August. Association for Computational Linguistics.

Zijian Győző Yang, László János Laki, Tamás Váradi, and Gábor Prószéky. 2023. Mono- and multilingual gpt-3 models for hungarian. In *Text, Speech, and Dialogue*, pages 94–104, Cham. Springer Nature Switzerland.

Lingyi Yang, Feng Jiang, Haizhou Li, et al. 2024a. Is chatgpt involved in texts? measure the polish ratio to detect chatgpt-generated text. *APSIPA Transactions on Signal and Information Processing*, 13(2).

Xianjun Yang, Wei Cheng, Yue Wu, et al. 2024b. DNA-GPT: Divergent N-Gram analysis for training-free detection of GPT-Generated text. In *Proceedings of the 12th International Conference on Learning Representations*.

第二十四届中国计算语言学大会论文集，第598页-第612页，济南，中国，2025年8月11日至14日。

Sungduk Yu, Man Luo, Avinash Madasu, Vasudev Lal, and Phillip Howard. 2024a. Is your paper being reviewed by an llm? investigating ai text detectability in peer review. *arXiv preprint arXiv:2410.03019*.

Xiao Yu, Kejiang Chen, Qi Yang, et al. 2024b. Text fluoroscopy: Detecting LLM-generated text through intrinsic features. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15838–15846, Miami, Florida, USA, November. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

Biru Zhu, Lifan Yuan, Ganqu Cui, et al. 2023. Beat LLMs at their own game: Zero-shot LLM-generated text detection via querying ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7470–7483, Singapore, December. Association for Computational Linguistics.

A 数据集后处理实验结果

我们对SeqXGPT在清洗前后的数据集上的表现进行了实验，实验结果如表6所示。

models	SeqXGPT	
	清洗后	清洗前
ACC	76.9	92.2
F1	76.2	93.7

Table 6: SeqXGPT在清洗前后的数据集上的表现

B 数据集词法分析结果

我们对LGT-AA数据集进行了词法分析，结果如图6所示。可以观察到，数据集中的词汇总体分布较为均匀，SPACE和X等非语义符号和结构化标记占比都较低，清洗操作有效降低了数据集中的格式特征。此外，Claude、LLaMA-3、Mistral、GPT-3.5、GPT-4o生成文本中的代词含量相对较低，因为它们相对于其它模型更偏向使用代词代替名词。

C 面向多分类任务的实验结果

以下列出了SeqXGPT和我们的方法在不同模型组合的条件下的详细实验结果，1-model组合结果如表7所示，2-models组合结果如表8所示，3-models组合结果如表9所示。

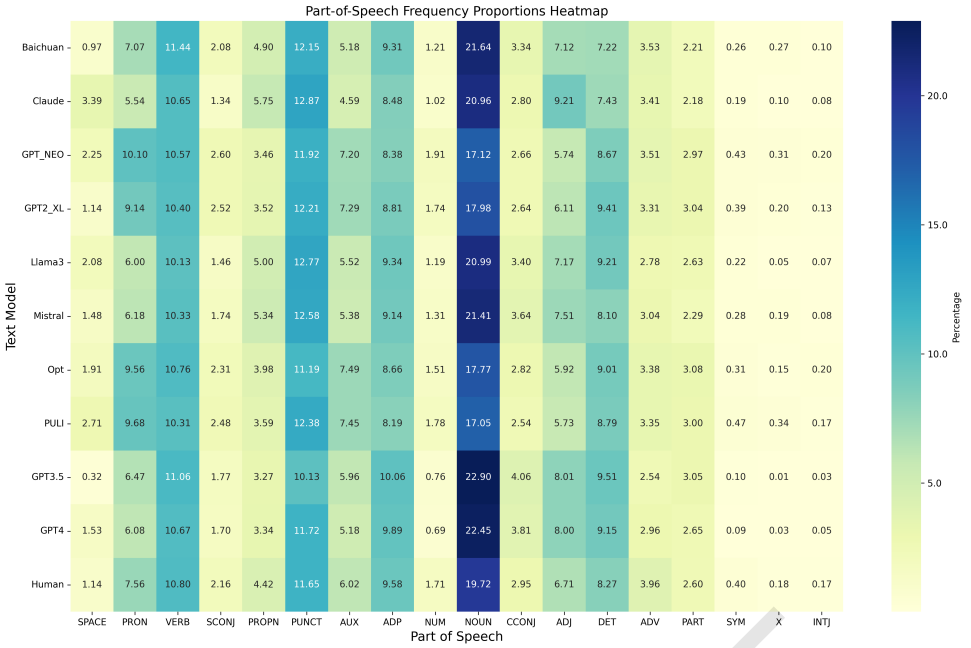


Figure 6: LGT-AA数据集词法分析结果

models		GPTNeo		LLaMa		GPT2		GPT-J	
		SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours
Baichuan	P.	32.1	41.3	29.0	30.6	31.6	37.7	28.0	31.1
	R.	18.3	18.6	19.7	18.2	17.0	22.5	17.3	19.2
GPTNeo	P.	71.8	78.8	47.3	53.6	50.9	56.8	51.3	56.0
	R.	63.0	75.6	54.5	72.7	53.4	73.1	51.9	67.3
GPT2	P.	52.7	59.3	60.2	66.9	63.2	73.6	56.8	62.5
	R.	52.4	61.1	49.5	59.9	62.4	74.2	52.9	61.6
LLaMa3	P.	40.6	48.3	45.3	50.1	40.2	48.7	42.2	47.6
	R.	44.7	43.0	45.0	41.2	44.4	51.3	48.0	44.7
Mistral	P.	50.2	51.6	51.8	52.1	47.5	51.6	49.7	49.6
	R.	41.8	60.9	47.0	60.6	44.7	55.7	43.3	60.0
OPT	P.	30.7	33.0	33.1	35.3	28.7	30.8	31.1	35.1
	R.	29.6	21.2	35.7	25.1	26.5	18.6	31.0	25.2
PULI	P.	45.9	45.5	41.9	43.6	39.5	40.9	40.5	42.5
	R.	42.5	48.2	31.1	26.4	30.2	25.2	30.7	30.0
GPT3.5	P.	78.3	76.4	79.8	77.3	76.9	76.0	80.3	78.2
	R.	77.4	80.2	78.8	80.2	77.2	75.1	79.1	79.6
GPT4o	P.	48.5	63.7	57.7	67.1	51.1	62.0	50.2	63.7
	R.	60.9	74.7	67.7	81.3	62.5	74.7	59.0	76.5
Claude	P.	51.3	60.4	41.7	49.0	48.6	55.2	58.5	48.1
	R.	59.4	62.0	52.1	45.4	54.5	58.0	62.6	39.8
human	P.	51.7	70.6	53.1	67.5	52.7	67.1	51.1	65.1
	R.	64.6	80.7	63.5	77.6	65.1	75.5	69.4	73.9
ACC		51.8	61.5	50.8	57.9	50.4	58.8	51.2	56.6
F1		50.0	56.4	48.8	59.6	48.2	60.0	48.9	52.1

Table 7: 在1-model条件下的实验结果

models		GPT2+LLaMA		GPT-J+GPT2		GPT-J+GPTNeo		GPT-J+LLaMA		GPTNeo+GPT2		GPTNeo+LLaMA	
		SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours
Baichuan	P.	48.2	55.0	44.3	49.4	41.1	51.4	45.6	51.2	38.7	48.0	47.3	57.3
	R.	42.4	44.0	32.4	38.1	28.5	33.6	38.8	41.6	29.8	30.8	40.8	42.5
GPTNeo	P.	60.0	62.8	60.4	63.9	89.1	94.8	56.7	61.8	86.9	94.2	87.3	93.1
	R.	62.0	78.8	61.3	76.5	89.7	95.1	59.2	73.3	85.1	93.1	85.8	93.2
GPT2	P.	87.5	94.2	91.1	96.6	74.5	74.2	72.5	79.3	92.0	96.8	77.5	82.2
	R.	91.5	96.6	93.2	97.8	62.8	70.9	60.3	70.5	92.3	97.3	65.4	75.1
LLaMa3	P.	53.2	60.0	49.0	60.6	44.8	57.8	50.9	60.5	47.2	56.1	51.9	61.8
	R.	59.0	58.3	58.1	61.2	47.9	52.1	54.7	55.7	56.2	59.5	55.5	58.0
Mistral	P.	66.6	72.7	62.5	68.4	61.5	67.8	65.4	70.3	59.7	61.4	65.7	73.1
	R.	56.8	71.4	50.5	67.2	48.9	66.6	56.3	71.8	48.9	60.0	55.3	71.8
OPT	P.	49.2	53.4	42.9	47.5	44.0	55.2	43.6	45.9	47.2	52.5	50.7	58.1
	R.	51.4	45.1	47.4	37.7	49.9	47.4	50.6	38.6	46.1	35.3	56.2	51.7
PULI	P.	53.0	52.6	55.9	55.5	63.0	65.8	51.3	50.0	65.5	71.8	61.9	66.7
	R.	45.2	36.7	47.9	43.0	63.5	72.6	45.3	45.5	62.5	81.0	64.3	76.9
GPT3.5	P.	83.0	83.4	82.2	82.5	80.1	80.8	83.5	82.6	80.2	81.0	84.0	83.2
	R.	85.8	84.8	82.8	87.1	83.8	83.2	83.9	84.4	82.0	82.6	82.4	85.8
GPT4o	P.	70.1	76.7	65.3	73.6	61.1	70.7	66.2	76.6	66.1	70.1	62.6	77.2
	R.	71.2	84.7	67.9	81.3	64.9	80.4	71.3	84.1	66.9	79.8	69.2	85.6
Claude	P.	70.0	73.2	67.6	70.9	63.7	67.0	65.9	71.7	65.1	65.0	65.8	73.0
	R.	68.9	71.5	70.5	69.9	68.4	70.5	66.5	73.1	68.6	67.8	66.4	72.2
human	P.	68.0	81.4	64.5	78.3	60.3	77.5	63.9	79.7	61.9	74.5	65.5	82.3
	R.	75.9	82.9	72.9	80.7	71.4	82.5	73.7	82.1	72.0	79.5	74.3	85.7
ACC		65.6	71.7	63.3	70.9	63.0	73.9	61.3	67.7	65.5	76.4	66.0	77.1
F1		64.4	68.8	62.1	67.4	61.7	68.7	60.1	65.7	64.4	69.5	65.1	72.8

Table 8: 在2-models条件下的实验结果

models		GPT-J+GPT2+LLaMA		GPT-J+GPT-Neo+GPT2		GPT-J+GPT-Neo+LLaMA		GPTNeo+GPT2+LLaMA	
		SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours	SeqXGPT	ours
Baichuan	P.	52.7	59.4	48.9	54.6	48.5	57.8	52.4	59.0
	R.	46.0	52.9	35.4	39.1	45.3	50.3	47.3	49.8
GPTNeo	P.	65.9	70.0	93.5	96.7	92.1	95.5	93.5	96.6
	R.	62.5	76.6	92.9	97.3	90.1	95.4	91.8	96.3
GPT2	P.	93.7	96.9	95.0	98.0	80.5	84.0	95.1	97.9
	R.	94.4	98.0	94.3	98.1	69.5	77.6	95.2	98.1
LLaMa3	P.	56.9	66.5	53.2	62.8	54.0	67.3	57.1	67.3
	R.	63.5	66.2	59.6	66.5	57.4	60.8	65.6	69.6
Mistral	P.	72.1	76.5	66.8	71.2	71.0	74.8	71.3	75.8
	R.	60.7	74.8	52.2	68.6	59.7	73.2	61.1	73.3
OPT	P.	54.4	58.6	55.8	67.2	57.1	65.1	63.8	68.7
	R.	57.9	49.5	60.9	64.0	62.5	59.2	67.2	63.6
PULI	P.	59.0	63.2	77.8	85.6	67.2	71.3	75.9	82.4
	R.	56.6	60.0	72.2	86.7	72.1	81.7	73.1	86.1
GPT3.5	P.	85.7	85.5	82.1	84.8	84.5	85.2	84.8	86.5
	R.	87.2	88.4	85.2	85.9	85.7	89.0	86.5	88.3
GPT4o	P.	71.5	81.5	67.3	77.4	71.0	79.7	73.4	81.1
	R.	74.7	87.4	71.5	84.2	71.6	85.8	73.8	86.8
Claude	P.	75.7	81.0	72.8	76.4	73.2	77.1	76.4	78.4
	R.	73.8	77.2	73.9	74.8	72.4	75.8	72.6	74.7
human	P.	70.8	85.6	67.1	80.6	70.6	85.5	72.6	84.8
	R.	79.5	84.5	79.5	84.7	79.1	86.0	79.2	86.3
ACC		69.7	76.7	71.5	82.8	70.4	80.1	74.6	84.2
F1		68.8	74.5	70.5	77.4	69.6	76.2	74.0	79.5

Table 9: 在3-models条件下的实验结果

第二十四届中国计算语言学大会论文集，第598页-第612页，济南，中国，2025年8月11日至14日。