计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
[doi: 10.15888/j.cnki.csa.009891] [CSTR: 32024.14.csa.009891]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于双层多智能体大模型的点击诱饵检测①

袁　旭, 朱　毅, 强继朋, 袁运浩, 李　云

(扬州大学 信息工程学院, 扬州 225127)
通信作者: 朱　毅, E-mail: zhuyi@yzu.edu.cn

摘　要: 点击诱饵是指用夸张或惊奇的标题吸引用户点击, 近年来已在新闻门户和社交媒体等在线应用中呈现泛滥趋势, 导致用户的不良体验甚至引起网络欺诈. 大模型由于强大的语义理解和文本生成能力, 已在一系列自然语言处理任务中取得优异的效果. 但是, 大模型在面对如点击诱饵检测这类决策边界不清晰的特定领域问题时很容易产生幻觉, 为此, 我们提出基于双层多智能体大模型的方法, 在不需要微调整个大模型的情况下, 有效提升了点击诱饵检测的准确率. 具体来说, 通过第 1 层中智能体的内部投票, 和第 2 层中不同智能体的交叉投票, 最终取得了良好的检测效果. 通过对 3 个基准数据集进行验证, 本文提出的方法比最先进的大模型和提示学习方法的准确率分别高出近 13% 和 10%.

关键词: 大模型; 多智能体; 点击诱饵; 短文本分类

引用格式: 袁旭,朱毅,强继朋,袁运浩,李云.基于双层多智能体大模型的点击诱饵检测.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/9891.html

Clickbait Detection Via Dual-layer Multi-agent Large Language Model

YUAN Xu, ZHU Yi, QIANG Ji-Peng, YUAN Yun-Hao, LI Yun

(School of Information Engineering, Yangzhou University, Yangzhou 225127, China)

Abstract: Clickbait refers to the use of sensational or exaggerated headlines to attract users into clicking, a practice that has proliferated in recent years across online platforms such as news portals and social media. This trend has led to user dissatisfaction and, in some cases, facilitated online fraud. Large language models (LLM), known for their robust natural language understanding and text generation capabilities, have demonstrated outstanding performance across various natural language processing tasks. However, when faced with specific challenges like clickbait detection, where decision boundaries are often unclear, LLM are prone to hallucination. To address the issue, a method based on a dual-layer multi-agent large language model is proposed, which significantly enhances clickbait detection accuracy without the need to fine-tune the entire model. Specifically, internal voting within agents in the first layer and cross-voting among different agents in the second layer results in enhanced detection performance. Validation against three benchmark datasets shows that the proposed method outperforms state-of-the-art large-scale models and prompt learning techniques by nearly 13% and 10% in terms of accuracy, respectively.

Key words: large language model (LLM); multi-agent; clickbait; short text classification

1　引言

近年来, 随着新闻门户和社交媒体等在线应用的迅速发展, 点击量与网络流量的增加直接关系到商业利润的提升. 因此, 许多内容发布者甚至是新闻媒体为

① 基金项目: 国家自然科学基金 (62076217); 国家语言委员会 (ZDI145-71); 江苏省研究生科研与实践创新计划 (SJCX23_1896)
收稿时间: 2024-10-31; 修改时间: 2025-01-15; 采用时间: 2025-01-24; csa 在线出版时间: 2025-03-31

吸引注意力并获取更多点击量, 往往使用与内容不符的误导性或欺骗性标题, 这种现象被称为点击诱饵, 中文又称为“标题党”[1]. 点击诱饵的泛滥不可避免会导致用户体验下降, 引发用户反感, 甚至进一步带来通过重定向用户至钓鱼网站以窃取个人信息的点击劫持攻击[2]. 因此, 点击诱饵检测已成为防止虚假信息传播并建立良好网络环境的重要研究课题[3].

点击诱饵检测的主要方法从早期基于特征工程的方法, 逐步演变为基于深度神经网络的方法, 并在近期转向了预训练语言模型. 早期基于特征工程的点击诱饵检测方法通过提取诸如词频和情感极性等语言特征进行判别性分类[4]. 随后, 基于深度神经网络的方法被广泛应用于学习更加抽象的特征以进行点击诱饵检测[5], 然而, 由于点击诱饵检测本质上是一项分类任务, 这些基于深度神经网络的方法通常需要大规模的标注数据. 近年来, 以双向编码器表示模型 (bidirectional encoder representations from Transformer, BERT) 为代表的预训练语言模型 (pre-trained language model, PLM) 成为点击诱饵检测的有力工具[6], 但这些方法通常需要新闻内容等辅助信息来微调模型, 否则, 预训练与微调间的巨大差异将阻碍点击诱饵检测任务充分利用预训练知识.

最近几年, 大语言模型 (large language model, LLM), 又称为“大模型”, 已在各种自然语言处理 (natural language processing, NLP) 的下游任务中展现出强大的能力, 即使在少样本甚至零样本场景中也能取得优异的表现[7]. 然而, 主流的大模型 (如 ChatGPT、LLaMA 和 Mistral 等) 在特定领域进行知识推理或分类检测任务时, 微调模型往往需要巨大的计算成本. 如果不进行模型微调, 大模型在面对如点击诱饵检测这类决策边界不清晰的特定领域问题时很容易产生幻觉, 从而导致检测准确率下降. 因此如何在不进行模型微调的情况下, 基于大模型对点击诱饵进行准确的检测, 已经成为实际应用中的巨大挑战.

为了解决这个问题, 本文提出了一种基于双层多智能体大模型的点击诱饵检测方法, 在不需要微调大模型的情况下, 有效提升了点击诱饵检测的准确率. 本文对以中文点击诱饵为代表的数据集进行检测分类时, 首先手动设计了相互独立的 n 个智能体, 每个智能体通过独立的提示 (prompt) 进行点击诱饵检测, 然后通过第 1 层中每个智能体的内部投票决定每个智能体的检测结果, 再通过第 2 层中不同智能体的交叉投票, 得到最终的点击诱饵检测结果. 据了解, 这是第 1 次将大模型多智能体运用到点击诱饵检测上, 并且与其他常用分类方法相比, 分类效果良好.

本文的主要贡献总结如下.

(1) 提出了一种基于大模型多智能体的点击诱饵检测方法, 充分利用了大语言模型的知识, 检测效果较为理想.

(2) 不同于以往需要大量有标签或无标签训练数据的检测模型, 本文方法仅需要很少的训练样本进行提示构建, 就能实现良好的检测效果.

(3) 实验结果证明, 本方法在中文点击诱饵检测数据集中, 分类准确率明显优于现有方法, 比最先进大模型和提示学习方法的准确率分别高出近 13% 和 10%.

2 相关工作

2.1 点击诱饵检测

近年来, 点击诱饵问题已经引发了来自多学科和多领域的广泛关注, 亟需精确的自动化检测方法. 从使用的方法来说, 点击诱饵的检测方法可以主要分为: 基于特征工程的方法、基于深度学习的方法、基于预训练语言模型的方法.

基于特征工程的方法. 早期的点击诱饵检测方法主要依赖于特征工程, 旨在捕捉多种不同的特征, 包括语义特征、语言学特征以及多模态特征等[8]. 例如, Blom 等人[9]总结了点击诱饵的语言学特征, 包括点击诱饵标题中前向引用标题的使用情况等. 然而, 基于特征工程的方法依赖于专业知识来识别合适的特征, 而手工设计特征在捕捉语义信息方面存在一定的局限性.

基于深度学习的方法. 随着深度学习方法的进步, 深度神经网络在点击诱饵检测中得到了广泛应用, 并取得了显著的成功. 例如, Naeem 等人[10]提出一种用于新闻标题特征提取的词性分析模块, 随后采用结合词向量 (Word2Vec) 嵌入的长短期记忆网络 (LSTM) 模型进行点击诱饵检测. 但是, 这些深度学习方法仍然面临着收集足够标注数据的高昂成本问题.

基于预训练语言模型的方法. 近几年, 诸如 T5[11]等预训练语言模型 (PLM) 已成为强大的语言生成与理解工具, 能够捕捉语言的句法、语义以及结构信息, PLM 中所编码的丰富知识可用于提升在点击诱饵检测等任务中的表现. 例如, Liu 等人[2]提出了一种方法, 整合多种特征用于微信平台上的标题党检测. 该方法利用双

向 LSTM (Bi-LSTM) 和 BERT 学习语义特征, 并通过图注意力网络捕捉局部句法信息, 将标题党检测作为一个三元分类任务, 其中标题党被细分为恶意标题党和一般标题党. 但是, 这些基于 PLM 的方法通常需要辅助信息来进行模型微调. 否则, 由于预训练与微调过程中目标函数的巨大差异, 将阻碍 PLM 中所蕴含知识的充分利用.

2.2 大语言模型

最近几年, 大语言模型已成为语言理解和文本生成的强大工具, 并在大量自然语言处理任务上取得了出色的性能. 当用户输入特定的指令或提示时, 以 ChatGPT 为代表的大模型能够令人信服地生成非常流畅的响应[12]. 目前, 许多基于大模型的应用产品, 包括智能客服、机器翻译、聊天机器人等, 已取得了广泛的社会影响. 与基于微调 PLM 的方法相比, 大模型有两个明显的优势. 首先是更大的模型规模, LLM 在模型参数和预训练数据方面具有更大的规模[13]. 其次, 即使没有微调, 大模型也能够在小样本学习甚至零样本学习的通用场景下取得很好的性能[14]. 但是, 大模型擅长文本生成任务, 在面对决策边界不明确的分类任务时, 需要非常大的计算资源进行模型调整.

3 方法

本节将介绍我们提出的基于双层多智能体大模型的点击诱饵检测方法.

3.1 整体框架

在详细介绍本文方法之前, 我们首先将点击诱饵检测问题形式化. 给定要检测的新闻或帖子记为 x, 包括标题 t 和内容 c, 相应标签记为 y. 其中, y=1 表示 x 是点击诱饵, y=0 表示 x 不是点击诱饵. 因此, 点击诱饵检测问题可以被视为一个文本分类任务, 即预测 $P(y|x) = P(y|t,c)$. 图 1 为总体框架图.

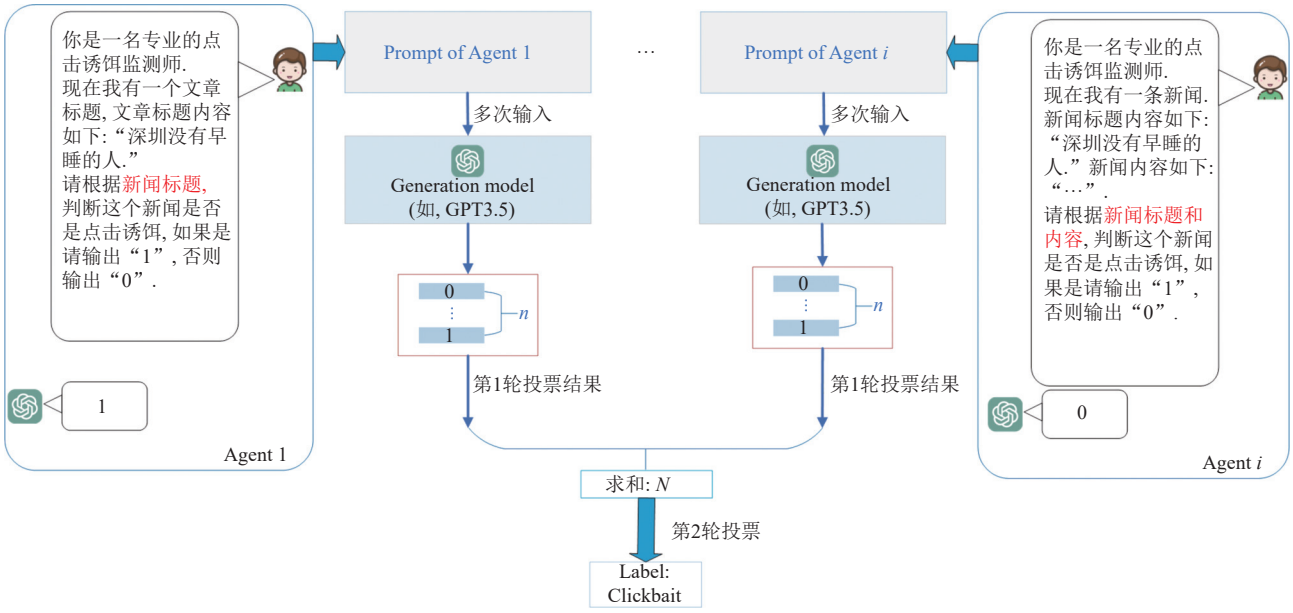

图 1 总体框架图

3.2 智能体生成

在本文点击诱饵检测方法中, 智能体设计的重点在于如何有效引导模型识别和判断点击诱饵的特征. 为此, 我们在智能体设计中主要关注以下两个方面.

智能体提示内容的构建: 为了确保模型能够准确分析文本标题与内容之间的关系, 我们在提示中明确指示模型关注两者之间的语义一致性. 例如, 提示的内容包括点击诱饵的定义以及常见方式: “点击诱饵通常通过用夸张、疑问、误导和模糊的标题来吸引用户点击, 其内容往往与标题不符或质量较低. 标题在判断点击诱饵中起着关键作用”“点击诱饵的标题常见的方式主要包括以下几种: 1) 叠词式: 重复使用一个字来吸引注意, 主要特征为出现 3 个及以上一样的字, 如‘又双叒’‘真真真真’‘大大大大’‘人从众’等; 2) 夸张式: 使用夸张的手法以及修饰词, 这类主要会有‘大型 XX 现场’‘见识真正的 XXX’‘特别’‘超级’‘非常’等这些词汇的出

现”. 通过这种方式, 我们引导模型聚焦于标题是否过度吸引注意力或与实际内容不符的情况, 这些都是典型的点击诱饵特征.

参考样本的使用: 为了增强模型对点击诱饵的识别能力, 我们在提示设计中引入了 5 个正样本和 5 个负样本. 正样本包含了明显的点击诱饵案例, 而负样本则是非点击诱饵的普通内容. 通过这些参考样本, 模型可以更好地理解什么样的标题和内容构成了点击诱饵, 从而在实际判断时具备更高的准确性.

最后结合文章标题及内容生成 5 个智能体, 通过多个智能体的相互协助, 使得该实验具有更好的鲁棒性.

由于大模型在面对如点击诱饵检测这类决策边界不清晰的特定领域问题时很容易产生幻觉问题. 因此, 我们首先设计了 n 个相互独立的智能体, 旨在通过不同的 prompt, 使互相不干扰的智能体获取点击诱饵的不同特征, 增加大模型的鲁棒性和通用性, 避免幻觉问题. 具体地, 我们在实验中设计了 n=5 个智能体, 详情描述如表 1 所示.

表 1 智能体描述详情

序号	智能体描述
1	请根据点击诱饵的标题常见的方式, 根据该文章标题判断是否为点击诱饵.
2	请结合文章内容和文章标题, 来判断是否为点击诱饵.
3	首先只根据文章内容, 判断该文章内容是否为点击诱饵. 如果不是, 请根据文章内容先生成一个新标题, 结合原始标题和新的标题, 判断该文章是否为点击诱饵.
4	首先只根据文章内容, 判断该文章内容是否为点击诱饵. 如果不是, 请根据文章标题生成一段新的文章内容, 最后结合两段文章内容判断该文章是否为点击诱饵.
5	首先只根据文章内容, 判断该文章内容是否为点击诱饵. 如果不是, 请根据文章内容生成一段文章总结, 然后结合文章总结和原始标题, 判断该文章是否为点击诱饵.

该 5 个智能体分别用到了文章中不同的信息, 具体区别如表 2 所示.

表 2 不同智能体之间的区别

序号	文章标题	文章内容	生成的标题	生成的内容	生成的总结
1	√	—	—	—	—
2	√	√	—	—	—
3	—	√	√	—	—
4	—	√	—	√	—
5	√	√	—	—	√

其中, 文章的标题和内容是原始数据集中存在的, 而生成的标题、内容和总结则是基于原有标题或内容, 为了获得更为丰富的语义信息, 按照本文要求生成的.

3.3 双层多智能体投票

在智能体生成完成之后, 我们提出通过双层多智能体投票进一步优化点击诱饵检测的结果. 具体地, 在第 1 层, 首先基于智能体 i 对输入 x 进行 m 次相互独立的检测, 得到分类为类别 y 中的结果$\{r_1^i,\cdots,r_m^i\}$, 如用智能体 1 对输入“请根据点击诱饵的标题常见的方式, 判断该条文章标题是否为点击诱饵, 如果是点击诱饵就输出 1, 否则输出 0. 不要输出另外的东西, 不要分点作答. ”的检测过程中, 大模型首次判断属于“y=1 点击诱饵”类别的概率大于“y=0 非点击诱饵”类别的概率, 那么该输入的结果标签被记为$r_1^1=1$. 如果大模型第 2 次判断该输入为“非点击诱饵”, 则$r_2^1=0$. 依次类推, 在智能体 i 对输入 x 完成 m 次相互独立的检测后, 得到了$\{r_1^i,\cdots,r_m^i\}$并进行投票. 实验中, 我们取 m=5 并采取了多数投票规则, 即取 m 次中超过半数的检测结果作为该智能体的最终结果, 记为r^i.

在第 1 层内部智能体投票结果完成之后, 我们接着通过第 2 层中不同智能体的交叉投票, 得到最终的点击诱饵检测结果. 具体地, 由于我们设计了 n=5 个智能体, 因此, 对于输入 x 将得到$\{r^1,\cdots,r^5\}$的结果. 在实验中, 我们保持和第 1 层中相同的多数投票规则, 即取结果中占多数的投票为最终结果. 如$\{r^1,\cdots,r^5\}=\{1,0,1,1,1\}$, 由于“$y$=1 点击诱饵”类别占多数, 因此, 该输入 x 的最终检测结果为“y=1 点击诱饵”.

算法 1. 点击诱饵检测算法

输入: 数据 x (文章标题); 智能体数量 n=5; 每个智能体的检测次数 m=5.
输出: 最终点击诱饵检测结果.

第 1 层: 智能体内部投票
1) 初始化 1 个列表 *vote_results*, 记录智能体 i 对输入 x 的检测结果. 对于每次独立检测 j (j=1,⋯,m), 执行以下步骤:
首先, 使用智能体 i 对输入 x 进行检测, 得到结果 y (y=1 表示点击诱饵, y=0 表示非点击诱饵);
然后, 将检测结果 y 添加到 *vote_results* 列表中;
最后, 根据结果是否达到设定阈值来确定智能体 i 的最终检测结果.
2) 将智能体 i 的最终结果记录到 *agent_votes_first_layer* 列表中.

第 2 层: 5 个智能体之间的交叉投票
1) 初始化 1 个列表 *final_results*, 用于记录第 2 层的投票结果. 对于每个输入 x, 执行以下步骤:
首先, 从第 1 层获得每个智能体的最终检测结果 (即每个智能体的投票结果);
然后, 对 5 个智能体的投票结果进行多数投票, 得到最终的点击诱饵检测结果.

2) 将最终结果添加到 *final_results* 列表中, 返回 *final_results* 作为最终的点击诱饵检测结果.
3) 最终, 将最终的点击诱饵检测结果输出.

4 实验分析

在本节中, 首先介绍数据集和基线方法, 然后介绍实验设置以及评估指标, 并通过消融实验验证了方法的有效性. 此外, 我们设计了不同的提示以及智能体来验证其对实验结果的影响.

4.1 数据集

实验在 3 个著名的短文本基准数据集上进行, 表 3 展示了详细的数据集统计结果, 对每个数据集的具体描述如下.

表 3 实验数据集详情

序号	名称	来源	总样本数/训练集/测试集
1	WeChat	微信公众号	12 724/9 192/3 532
2	Sina	新浪新闻	1 912/1 122/790
3	Paper	澎湃新闻	30 800/30 010/790

WeChat: 该数据集来源于微信公众号的文章, 原始数据集的类别包括非标题党、恶意标题党和一般标题党. 为了简化分析并增强模型的泛化能力, 我们将恶意标题党与一般标题党合并为点击诱饵这一类别. 此外, 由于原始数据集中部分文章样本内容不完整, 尤其是新闻内容长度低于 50 字符的样本, 可能会对模型训练效果产生不利影响, 因此我们对数据进行了预处理, 删除了这些不完整的样本.

Sina: 该数据集来自新浪新闻. 新浪新闻作为中国一个重要的新闻门户网站, 涵盖了广泛的新闻类型和话题. 我们从新浪新闻中收集了额外的新闻样本, 并进行手动标注, 确保数据的准确性和分类的一致性. 这个数据集同样划分为点击诱饵和非点击诱饵两个类别.

Paper: 该数据集从澎湃新闻中收集了大量的新闻标题与对应内容, 涵盖了广泛的新闻内容和社会话题. 该数据集包含 800 个手动标注的新闻样本, 其中包括点击诱饵和非点击诱饵两类新闻.

4.2 对比方法

为了更好地评估本文模型, 我们选择基于特征工程的方法、基于深度神经网络的方法、及当下流行的大模型进行比较, 具体包括如下几种.

BERT[15]: 一种基于 Transformer 的双向编码器表示方法, 我们将新闻通过 BERT 进行文本向量化, 并通过分类器进行点击诱饵检测.

MFWCD-BERT (multiple features for WeChat clickbait detection)[2]: 基于多特征的微信点击诱饵检测框架. 该框架首先通过图注意力网络提取标题中的句法结构特征, 接着使用 BiLSTM 网络来捕捉标题的语义信息, 并结合相关辅助信息, 以提升点击诱饵检测的准确性.

KG-GCN+ATT[16]: 该模型融合了知识图谱、图卷积网络及图注意力网络, 能够实现对中文点击诱饵的细粒度识别.

PT[17]: 是一种提示调优方法, 通过将下游任务转换为完形填空的形式进行预测. 这种方法可以有效利用预训练语言模型中的丰富信息, 在小样本情况下实现更好的检测点击诱饵.

P-Tuning[18]: 一种基于软模板的提示调优改进方法, 该方法将模板构建过程转化为一个连续的参数优化问题进行处理.

KPT (knowledgeable prompt-tuning)[19]: 一种优化的提示调优方法, 通过引入外部先验知识, 可以更准确地实现点击诱饵检测.

PEPL[20]: 该方法仅通过新闻标题, 并利用词性增强的提示来实现语法特征引导下的语义理解, 从而完成点击诱饵检测.

LLaMA (large language model meta AI)[21]: LLaMA 是开源本地大模型, 和 GPT 相似, 可以在小样本甚至零样本场景下, 对点击诱饵检测表现出不错的效果.

4.3 实验设置与评估指标

本文实验以 GPT 作为主干大语言模型, 实验中使用 GPT-3.5-turbo 模型, 实验服务器具体配置为: NVIDIA Geforce RTX 3090 Founders Edition GPU, Intel(R) Core(TM) i9-10980XE CPU, 运行频率为 3.00 GHz, 内存为 125 GB.

为了衡量点击诱饵的检测效果, 本文选取 *accuracy* 和 *F*1 为评估指标, 具体定义如下:

$$precision = \frac{tp}{tp + fp} \tag{1}$$

$$recall = \frac{tp}{tp + fn} \tag{2}$$

$$accuracy = \frac{tp + tn}{N} \tag{3}$$

$$F1 = \frac{precision \cdot recall}{precision + recall} \tag{4}$$

其中, tp是正确预测新闻是点击诱饵的数量, fp是错误预测新闻为点击诱饵的数量, fn是新闻被预测为非点击诱饵, 但事实是点击诱饵的数量, tn是已经正确预测新闻是非点击诱饵的数量, N为预测新闻的总数量, $F1$为精确率和召回率的调和平均数.

4.4 实验结果

所有方法在 3 个点击诱饵数据集上的实验结果如表 4 所示.

表 4 实验结果

Method	WeChat		Sina		Paper	
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1
BERT	0.5856	0.5224	0.6064	0.5314	0.5575	0.4388
MFWCD-BERT	0.7211	0.4272	0.6522	0.6441	0.6987	0.6679
KG-GCN+ATT	0.6995	0.6844	0.6663	0.6582	0.6089	0.5379
PT	0.6563	0.6700	0.6765	0.6711	0.6615	0.6581
P-Tuning	0.7151	0.7295	0.6418	0.6347	0.6474	0.6398
KPT	0.7068	0.7213	0.6771	0.6758	0.6884	0.6824
PEPL	0.7062	0.7053	0.6876	**0.6863**	0.7014	0.7053
LLaMA	0.6228	0.7545	0.5911	0.2217	0.5228	0.6673
Ours	**0.7554**	**0.7679**	**0.6911**	0.5658	**0.7038**	**0.7347**

通过分析实验结果, 可以看出, 基线方法 BERT 虽然对大量训练样本有效, 但被专门为点击诱饵检测设计的深度学习模型所超越, 如 MFWCD-BERT 和 KG-GCN+ATT. 这些点击诱饵检测模型擅长捕捉点击诱饵中的独特特征, 如特定的句法结构和夸张的形容词等, 有效提升了模型的检测性能.

总体来说, 基于提示调优的方法 (如 PT、P-tuning、KPT 和 PEPL) 的表现并不优于专为点击诱饵检测设计的深度学习模型 (如 MFWCD-BERT 和 KG-GCN+ATT). 然而, 这些方法在训练样本数量有限的情况下表现出很强的性能, 突显了提示调优方法在数据有限场景中的有效性. 值得注意的是, 与其他基于提示调优的方法相比, 将语法引导的语义理解纳入提示调优的 PEPL 方法表现出了更优的结果.

结果表明, 大模型 (如 LLaMA) 的点击诱饵检测性能相对较差. 虽然大模型擅长处理具有清晰、标准化答案的客观问题, 但它面临着个性化问题的挑战, 这些问题往往缺乏固定的标准答案, 并可能受到特定功能的影响. 例如, 对同一条新闻进行多次测试可能会产生不同的结果, 从而影响检测点击诱饵的准确性.

最后, 本文提出的方法, 基于多智能体的双层投票, 在不需要微调大模型的情况下, 有效提升了点击诱饵检测的准确率, 与其他方法相比, 在大部分指标和数据集上都取得了最佳结果.

4.5 探究不同阈值对实验结果的影响

在本节中, 我们分别探究第 1 层投票和第 2 层投票中, 设定不同阈值对双层多智能体大模型在点击诱饵检测任务中性能的影响.

在阈值设计方面, 在第 1 层投票中, 设置阈值为 5 时, 即该智能体 5 次检测的结果一致才认为是正确的检测结果时, WeChat 数据集的准确率和 $F1$ 值分别达到 0.7554 和 0.7679, 表现最佳 (见表 5). 降低阈值至 4 和 3 虽然在某些数据集上有所提升, 但整体性能下降, 尤其在 $F1$ 分数上表现出更高的波动性.

表 5 第 1 层投票中阈值的影响

Threshold	WeChat		Sina		Paper	
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1
3	0.6461	0.7358	0.6797	**0.7095**	0.6266	0.7286
4	0.7039	0.7625	0.6848	0.6852	0.6734	**0.7425**
5	**0.7554**	**0.7679**	**0.6911**	0.5658	**0.7038**	0.7347

在第 2 层投票中, 阈值为 5 代表 5 个智能体都认为是点击诱饵时才判断为点击诱饵, 从结果中可以观察到阈值为 4 时的表现优于其他设置, WeChat 数据集的准确率和 $F1$ 分数维持在 0.7554 和 0.7679 (见表 6). 相较于第 1 层, 第 2 层的稳定性更强, 然而, 当阈值提高至 5 时, Paper 数据集的性能显著下滑, 准确率和 $F1$ 分数降至 0.5861 和 0.4577, 表明阈值对分类可靠性具有重要影响.

表 6 第 2 层投票中阈值的影响

Threshold	WeChat		Sina		Paper	
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1
3	0.7143	0.7634	**0.7165**	**0.6957**	0.6582	0.7228
4	**0.7554**	**0.7679**	0.6911	0.5658	**0.7038**	**0.7347**
5	0.7191	0.6605	0.5975	0.2282	0.5861	0.4577

4.6 探究不同智能体对实验结果的影响

为全面评估智能体数量的作用, 我们分别测试了 1 个、3 个和 5 个智能体的配置, 并在 3 个数据集上进行了实验, 保持其他实验条件和参数不变. 实验结果显示, 随着智能体数量的增加, 模型的性能显著提升. 在仅使用 1 个智能体时, WeChat 数据集的准确率和 $F1$ 分数分别为 0.6871 和 0.7362. 在引入 3 个智能体后, 准确率提升至 0.7022, $F1$ 分数达到 0.7623, 表现出更好的稳定性.

从表 7 可以看出, 当智能体数量增加到 5 个时, WeChat 数据集的性能进一步提升, 准确率和 $F1$ 分数

分别达到 0.7554 和 0.7679. 这一趋势在 Sina 和 Paper 数据集中同样得到体现, 尤其是在 Paper 数据集上, 5 个智能体的设置显著提高了模型的表现, 准确率达到 0.7038, *F*1 分数为 0.7347.

综上所述, 增加智能体的数量显著增强了模型在不同数据集上的分类性能, 表明多智能体机制在提升模型鲁棒性和准确性方面的有效性.

表 7 不同智能体对结果的影响

智能体数量	WeChat		Sina		Paper	
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1
1	0.6871	0.7362	0.6025	0.5328	0.5873	0.6517
3	0.7022	0.7623	0.6405	**0.6223**	0.5911	0.6707
5	**0.7554**	**0.7679**	**0.6911**	0.5658	**0.7038**	**0.7347**

4.7 消融实验

在消融实验中, 我们分析了去掉投票机制对双层多智能体大模型性能的影响. 为评估这种变化对点击诱饵检测任务的具体影响, 实验设置包括不使用投票、仅使用第 1 层投票、仅使用第 2 层投票, 以及完整两层投票机制, 实验结果见表 8. 实验结果显示, 采用不同投票机制对模型的准确率和 *F*1 分数有显著影响.

表 8 不同投票机制对实验的影响

Method	WeChat		Sina		Paper	
	Acc	*F*1	*Acc*	*F*1	*Acc*	*F*1
原始GPT	0.6344	0.5877	0.5582	0.5236	0.5405	0.4141
仅第1层投票	0.6831	0.7133	0.6032	0.5470	0.5749	0.5860
仅第2层投票	0.6741	0.7474	0.6848	**0.5829**	0.6367	0.7167
双层投票机制	**0.7554**	**0.7679**	**0.6911**	0.5658	**0.7038**	**0.7347**

在原始 GPT 模型中, 无投票机制时, WeChat 数据集的准确率为 0.6344, *F*1 分数为 0.5877, 表现较弱. 引入第 1 层投票后, 准确率提高至 0.6831, *F*1 分数增至 0.7133, 显示出明显改进. 仅使用第 2 层投票时, 准确率为 0.6741, *F*1 分数达到 0.7474, 表现优于第 1 层投票.

最终, 当采用双层投票机制时, WeChat 数据集的性能显著提升, 准确率达到 0.7554, *F*1 分数为 0.7679. 这一结果表明, 综合使用两层投票机制能够有效提升模型的分类能力, 验证了多层投票机制在增强模型鲁棒性和准确性方面的重要性.

5 结论与展望

本文提出了一种基于双层多智能体大模型的点击诱饵检测方法, 该方法不需要微调大模型, 通过相互独立的 *n* 个智能体进行两层投票, 有效解决了大模型面对点击诱饵检测这种决策边界不清晰的特定领域问题时产生的幻觉问题, 实验证明该方法有效. 在下一步工作中, 我们将尝试使用自动提示生成的方法构建智能体, 以在其他各类任务中取得更好的性能表现.

参考文献

1 Wang HC, Maslim M, Liu HY. CA-CD: Context-aware clickbait detection using new Chinese clickbait dataset with transfer learning method. Data Technologies and Applications, 2024, 58(2): 243–266. [doi: 10.1108/DTA-03-2023-0072]

2 Liu T, Yu K, Wang L, *et al*. Clickbait detection on WeChat: A deep model integrating semantic and syntactic information. Knowledge-based Systems, 2022, 245: 108605. [doi: 10.1016/j.knosys.2022.108605]

3 Kaushal V, Vemuri K. Clickbait—Trust and credibility of digital news. IEEE Transactions on Technology and Society, 2021, 2(3): 146–154. [doi: 10.1109/TTS.2021.3073464]

4 Saquete E, Tomás D, Moreda P, *et al*. Fighting post-truth using natural language processing: A review and open challenges. Expert Systems with Applications, 2020, 141: 112943. [doi: 10.1016/j.eswa.2019.112943]

5 Zheng JM, Yu K, Wu XF. A deep model based on lure and similarity for adaptive clickbait detection. Knowledge-based Systems, 2021, 214: 106714. [doi: 10.1016/j.knosys.2020.106714]

6 Wang Y, Zhu Y, Li Y, *et al*. Clickbait detection via prompt-tuning with titles only. IEEE Transactions on Emerging Topics in Computational Intelligence, 2025, 9(1): 695–705. [doi: 10.1109/TETCI.2024.3406418]

7 Chowdhery A, Narang S, Devlin J, *et al*. Palm: Scaling language modeling with pathways. The Journal of Machine Learning Research, 2023, 24(1): 11324–11436.

8 Al-Sarem M, Saeed F, Al-Mekhlafi ZG, *et al*. An improved multiple features and machine learning-based approach for detecting clickbait news on social networks. Applied Sciences, 2021, 11(20): 9487. [doi: 10.3390/app11209487]

9 Blom JN, Hansen KR. Click bait: Forward-reference as lure in online news headlines. Journal of Pragmatics, 2015, 76: 87–100. [doi: 10.1016/j.pragma.2014.11.010]

10 Naeem B, Khan A, Beg MO, *et al*. A deep learning framework for clickbait detection on social area network using natural language cues. Journal of Computational Social Science, 2020, 3(1): 231–243. [doi: 10.1007/s42001-020-00063-y]

11 Raffel C, Shazeer N, Roberts A, *et al*. Exploring the limits of

transfer learning with a unified text-to-text Transformer. The Journal of Machine Learning Research, 2020, 21(1): 5485–5551.

12 Ouyang L, Wu J, Jiang X, *et al*. Training language models to follow instructions with human feedback. arXiv:2203.02155, 2022.

13 Patil R, Gudivada V. A review of current trends, techniques, and challenges in large language models (LLMs). Applied Sciences, 2024, 14(5): 2074. [doi: 10.3390/app14052074]

14 Wang L, Ma C, Feng XY, *et al*. A survey on large language model based autonomous agents. Frontiers of Computer Science, 2024, 18(6): 186345. [doi: 10.1007/s11704-024-40231-1]

15 Zhu Y, Wang Y, Qiang JP, *et al*. Prompt-learning for short text classification. IEEE Transactions on Knowledge and Data Engineering, 2024, 36(10): 5328–5339. [doi: 10.1109/TKDE.2023.3332787]

16 Zhou MX, Xu W, Zhang WP, *et al*. Leverage knowledge graph and GCN for fine-grained-level clickbait detection. World Wide Web, 2022, 25(3): 1243–1258. [doi: 10.1007/s11280-022-01032-3]

17 Liu PF, Yuan WZ, Fu JL, *et al*. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys, 2023, 55(9): 1–35.

18 Liu X, Zheng YN, Du ZX, *et al*. GPT understands, too. AI Open, 2024, 5: 208–215. [doi: 10.1016/j.aiopen.2023.08.012]

19 Hu SD, Ding N, Wang HD, *et al*. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Dublin: ACL, 2022. 2225–2240.

20 Wu Y, Cao MP, Zhang YZ, *et al*. Detecting clickbait in Chinese social media by prompt learning. Proceedings of the 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD). Rio de Janeiro: IEEE, 2023. 369–374.

21 Touvron H, Lavril T, Izacard G, *et al*. LLaMA: Open and efficient foundation language models. arXiv:2302.13971, 2023.

(校对责编: 王欣欣)