大语言模型的安全威胁与防御综述

贾澄钰[1,2], 陈晋音[1], 许 淦[1,2], 张 奥[1,2], 张 鹤[1,2], 金海波[2],
陈若曦[2], 郑海斌[1,3,4]

[1]浙江工业大学 计算机科学与技术学院(软件学院、人工智能学院) 杭州 中国 310032
[2]浙江工业大学 信息工程学院 杭州 中国 310032
[3]北京生命科技研究院有限公司(重点实验室) 北京 中国 102206
[4]四川大学数据安全防护与智能治理教育部重点实验室 成都 中国 610065

摘要 ChatGPT、Bard 等大型语言模型(Large language models, LLMs)技术的发展对人工智能社区产生深远影响。这些模型具备卓越的语言理解、类人的文本生成和强大的问题解决能力, 在搜索引擎、金融、医疗和自动驾驶等领域呈现广泛应用前景。然而, LLMs 的使用揭示了一系列安全漏洞, 引起了研究人员对其安全性问题的关注。本文从自然语言处理(Natural language processing, NLP)和安全的视角, 对 LLMs 的安全性进行了全面调查。首先概述了 LLMs 的相关框架平台及其发展历程, 从模型架构、训练数据来源和下游对齐方法 3 个角度对目前国内外主流的 LLMs 进行分类。接着对现有的 LLMs 安全综述工作开展讨论, 将这些工作根据评估维度、单一安全维度和防御方法 3 个维度进行划分, 并对其进行归纳和总结。随后讨论了 LLMs 在使用全周期(语料库收集及数据预处理、模型预训练阶段、下游对齐阶段和模型推理阶段)中可能面临的安全威胁。进一步将这些威胁进行了详细的分类和总结, 将可信评估划分为幻觉、欺骗、毒性、隐私、偏见和鲁棒性 6 个方面, 并讨论和总结了越狱、后门和对抗 3 种针对 LLMs 的攻击形式。还总结了针对 LLMs 开发和使用过程中的道德隐患问题。本文进一步概述了一系列针对 LLMs 安全威胁的防御和检测措施, 重点增强模型抵御幻觉、隐私泄露、偏见等威胁的能力。最后, 讨论了减轻 LLMs 安全风险的主要挑战和新出现的机遇, 为研究人员、从业者和政策制定者在大语言模型的复杂应用和研究领域提供指导建议。

关键词 大语言模型; 幻觉; 欺骗; 毒性; 越狱; 后门; 隐私; 公平; 偏见; 鲁棒性; 防御; 检测
中图法分类号 TP DOI 号 10.19363/J.cnki.cn10-1380/tn.2026.01.12

A Survey on Large Language Model's Security Risks and Defense Methods

JIA Chengyu[1,2], CHEN Jinyin[1], XU Gan[1,2], ZHANG Ao[1,2], ZHANG He[1,2], JIN Haibo[2],
CHEN Ruoxi[2], ZHENG Haibin[1,3,4]

[1] College of Computer Science and Technology (College of Software, College of Artificial Intelligence), Zhejiang University of Technology, Hangzhou 310032, China
[2] College of Information Engineering, Zhejiang University of Technology, Hangzhou 310032, China
[3] Key Laboratory of Beijing Life Science Academy, Beijing 102206, China
[4] Key Laboratory of Data Protection and Intelligent Management, Ministry of Education, Sichuan University, Chengdu 610065, China

Abstract The development of large language models (LLMs) such as ChatGPT and Bard has had a profound impact on the artificial intelligence community. These models have excellent language understanding, human-like text generation, and powerful problem-solving capabilities, and have shown broad application prospects in search engines, finance, medical care, and autonomous driving. However, the use of LLMs has revealed a series of security vulnerabilities, which has attracted the attention of researchers to their security issues. This paper conducts a comprehensive investigation on the security of LLMs from the perspective of natural language processing (NLP) and security. First, the relevant framework platforms and development history of LLMs are outlined, and the current mainstream LLMs at home and abroad are classified from three perspectives: model architecture, training data source, and downstream alignment method. Then, the existing LLMs security review work is discussed, and these works are divided according to the three dimensions of evaluation dimension, single security dimension, and defense method, and summarized and concluded. Then, the security threats that LLMs may face in the full cycle of use (corpus collection and data preprocessing, model pretraining stage, down-

通讯作者: 陈晋音, 博士, 教授, Email: chenjinyin@zjut.edu.cn。
本课题得到国家自然科学基金(Nos. 62406286, 62072406), 北京生命科技研究院有限公司开放基金(No. 2024200CD0210), 四川大学数据安全防护与智能治理教育部重点实验室开放课题 (No. SCUSAKFKT202402Z), 浙江省自然科学基金(No. LDQ23F020001), 基于多源数据融合的数据赋能方法的研究及应用项目资助。
收稿日期: 2024-01-10; 修改日期: 2024-06-27; 定稿日期: 2025-12-11

stream alignment stage, and model inference stage) are discussed. These threats are further classified and summarized in detail, and the trusted evaluation is divided into six aspects: hallucination, deception, toxicity, privacy, bias and robustness. Three forms of attack against LLMs, jailbreaking, backdoors and adversarial attacks, are discussed and summarized. The ethical risks in the development and use of LLMs are also summarized. This paper further outlines a series of defense and detection measures against LLMs security threats, focusing on enhancing the model's ability to resist threats such as hallucinations, privacy leaks, and bias. Finally, the main challenges and emerging opportunities for mitigating LLMs security risks are discussed, providing guidance and suggestions for researchers, practitioners, and policymakers in the complex application and research fields of large language models.

Key words large language models; hallucinations; deception; toxicity; jailbreaks; backdoor; privacy; fairness; bias; robustness; defense; detection

1 引言

随着大型语言模型(Large language models, LLMs)的出现, 自然语言处理(Natural language processing, NLP)的格局发生了深刻的转变。继 OpenAI 发布了 ChatGPT、Meta AI 和 Google 等公司相继发布了 LLaMA、Bard 等多种 LLMs 架构。LLMs 的技术发展对整个人工智能社区产生了重要影响, 基于 LLMs 的应用涵盖了搜索引擎[1]、客户支持[2]、自动驾驶[3]、医疗[4]等各个领域, 这将彻底改变开发和使用人工智能算法的方式。

然而, LLMs 在广泛的使用中暴露出不同方面的安全隐患。例如产生看似令人信服但不准确的"幻觉"[5], 生成不适当或有害的内容[6], 从不同方面泄露用户隐私[7], 以及容易遭受恶意攻击[8]等。这些安全隐患减弱了对 LLMs 的信任, 这些缺陷行为减弱了人们对 LLMs 的信任, 为它们的实际应用带来了阻碍。此外, 美国联邦贸易委员会于 2023 年发起了首个针对人工智能聊天机器人带来风险的审查; 欧洲议会通过了《人工智能法案》的折中修订草案; 中国信息通信研究院提出《中国信通院大模型 2.0 体系安全可信标准》; 国家工业信息安全发展研究中心于 2023 年发布的《AI 大模型发展白皮书》等, 都对 LLMs 的安全使用提出了新的要求。

目前已有一些针对 LLMs 的综合评估工作讨论了知识、推理、可靠性等[9-10], 对安全维度的评估不足。还有一些工作总结了幻觉[11]、偏见[12]、隐私[13]等具体方面的研究, 缺乏对其他安全维度的探索。本文考虑了 LLMs 特有的训练范式和推理能力可能引入的各种安全威胁, 总结和整理了 LLMs 安全可控技术的相关研究工作。在本研究中, 我们将通过回顾 LLMs 的最新进展、分析其开发和使用上下游阶段可能引入的安全威胁, 并结合国内外相关政策, 从幻觉、欺骗、毒性、隐私、偏见、鲁棒性角度进行可信评估, 分析了 LLMs 开发和使用过程中产生的道德挑战, 还讨论了针对 LLMs 的越狱、后门和对抗攻击, 并总结了针对上述安全威胁的防御和检测方法。详细的安全性分析维度如图 1 所示。此外, 我们还介绍了 LLMs 的发展历程以及国内外主流的框架平台。并提出 LLMs 安全性研究相关的机遇与挑战。

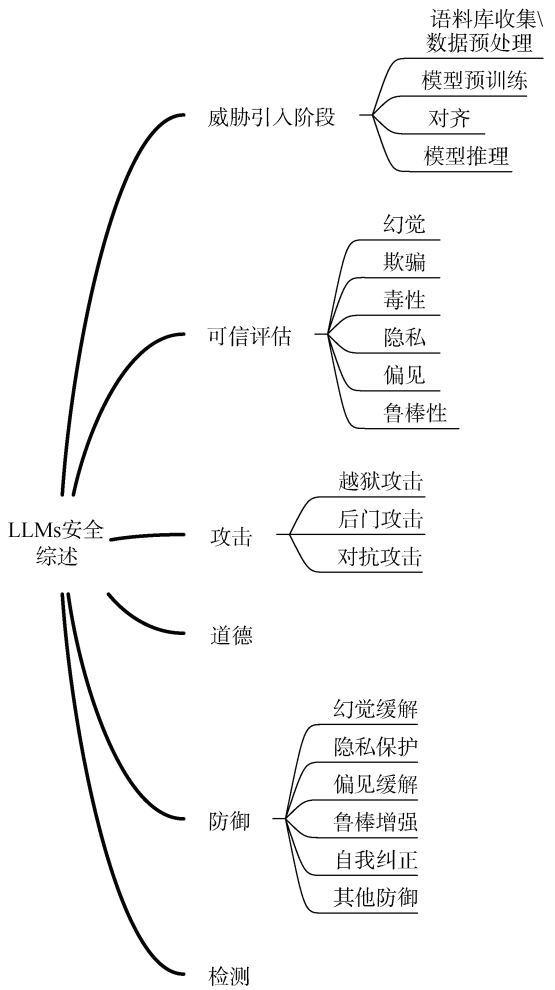

图 1 安全性分析维度

Figure 1 Security analysis dimensions

本文的主要贡献如下:

(1) 对 LLMs 开发和使用上下游全周期可能引入的安全威胁开展了全面的讨论。

(2) 对现存的 LLMs 安全相关工作进行了分类

和总结, 并整理了针对这些安全问题的防御和检测方法。

(3) 开源并维护了目前主流的LLMs框架平台及相关安全性工作(论文、开源代码、预训练模型和测试基准)①。

(4) 提出了 10 条 LLMs 安全性研究的发展机遇与挑战, 指导研究人员和相关从业者发掘未来的研究方向。

2 大语言模型框架平台

自然语言处理旨在使机器能够像人类一样阅读、写作和交流。NLP 中的两个关键任务是自然语言理解和自然语言生成。本节将从 LLMs 的资源、基本结构、训练方法、应用 4 个角度对 LLMs 进行简单介绍。所有相关的资源和模型将在①中开源并维护。根据模型架构、训练数据来源和对齐方式进行划分, 对现有的大语言模型的总体概述如表 1 所示。

2.1 大语言模型发展历程

语言建模是提高机器语言智能的主要方法之一。语言建模的目标是对单词序列的生成可能性进行建模, 从而预测未来(或缺失)标记的概率。

作为早期的尝试, ELMo[14]被提出通过首先预训练双向 LSTM 网络, 然后根据特定的下游微调策略捕获上下文感知的单词表示任务。此外, 基于具有自注意力机制的高度并行化的 Transformer 架构, 通过在大规模未标记语料库上使用专门设计的预训练任务来预训练双向语言模型, 提出了 BERT。这些预先训练的上下文感知单词表示作为通用语义特征非常有效, 这在很大程度上提高了NLP任务的性能标准。这项研究设定了“预训练和微调”的学习范式。遵循这一范式, 已有大量关于 PLM 的研究, 引入了不同的架构或改进的预训练策略。

表 1 大语言模型总结

Table 1 Summary of large language models

模型架构	仅编码器	训练: 掩码语言模型(Masked language model, MLM) 模型类型: 判别式 预训练任务: 预测屏蔽词	BERT (2018), Roberta (2019), ALBERT (2019), DeBERTa (2020), ELECTRA (2020)…
	编码器-解码器	训练: 自回归+ MLM 模型类型: 生成式 预训练任务: 预测下一个单词和屏蔽词	T5 (2019), GLM (2021), T0 (2022), FLAN-T5 (2022), ST-MOE (2022), ALexaLM (2022), ChatGLM (2023)…
	仅解码器	训练: 自回归语言模型 模型类型: 生成式 预训练任务: 预测下一个单词	GPT-3 (2020), Gopher (2021), BLOOM (2022), GPT-4 (2023), Claude-2 (2023), PaLM 2 (2023)…
训练数据	一般数据	来自网站、书籍和其他来源的包含广泛主题的内容	BERT (2018), Roberta (2019), T5 (2019), GPT-1 (2019), Gopher (2021), LLaMA (2022) … 多语言: PaLM (2022), GLM-130B (2022), BLOOM (2022), LaMDA (2022), PaLM 2 (2023) …
	特殊数据	基于特定主题的内容	代码及其他: CodeX (2021), AlphaCode (2022), CodeGen (2022), Code Llama (2023), StarCoder (2023) …
对齐方法	指令微调	使用结构实例微调 LLMs	GPT-3 (2020), T0 (2022), FLAN-T5 (2022), FLAN-PaLM (2022), InstructGPT (2022), WizardLM (2023), Alpaca (2023), LLM-Blender (2023), InstructZero (2023) …
	对齐微调	根据人类偏好训练模型以生成所需的输出	InstructGPT (2022), Sparrow (2022), OPT-IML (2022), PKUBeaver (2023), REFINER (2023), FINE-GRAINED RLHF (2023) …

扩展 PLM 模型大小或数据大小通常会提高下游任务的模型能力。许多研究通过训练更大的 PLM(例如 175B 参数 GPT-3 和 540B 参数 PaLM)来探索性能极限。这些大型 PLM 显示出与小型 PLM 不同的行为, 并显示出令人惊讶的能力。LLMs 与小型 PLM 的主要区别表现在以下几点: 首先, LLMs 展示了一些令人惊讶的新兴能力, 这些能力在以前的小型 PLM 中可能无法观察到, 是语言模型在复杂任务上表现的关键。其次, LLMs 算法的开发和使用方式与小型 PLM 不同, 访问 LLM 的主要方法是通过提示界面, 可能会引入新的安全类型。最后, LLM 的发展不再明确区分研究和工程, LLM 的训练需要丰富的大规模数据处理和分布式并行训练的实践经验。为了训练有效的 LLM, 研究人员必须解决复杂的工程问题。

2.2 大语言模型资源

该小节将从语料库来源、开发库两个方面介绍 LLMs 开发中使用的资源。

2.2.1 语料库

与早期的 PLM 相比, LLM 包含大量参数, 依赖的训练数据体量更大。目前已有一些广泛使用的用

于训练 LLM 的语料库, 根据内容来源可分为基于 CommonCrawl 的数据、书籍数据、Reddit 链接数据集, 维基百科数据集和代码等。几种流行 LLM 的训练数据来源分布如图 2 所示。

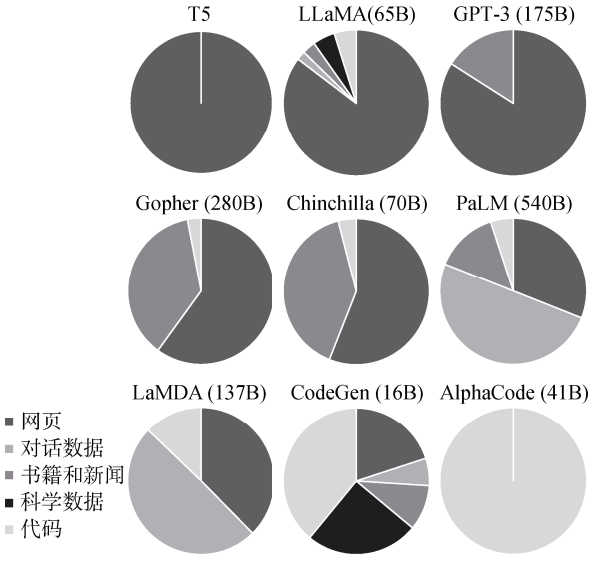

图 2 主流 LLMs 的训练数据来源分布

Figure 2 Distribution of training data sources for mainstream LLMs

CommonCrawl 是最大的开源网络爬虫数据库之一, 包含 PB 级数据量, 已被广泛用于 LLM 的训练。由于整个数据集非常庞大, 且网络数据中普遍存在噪声和低质量信息, 现有的研究主要是从中提取特定时期内的网页子集, 并在使用前进行数据预处理。现有工作中常用的有 4 种基于 CommonCrawl 的过滤数据集: C4、CCStories、CC-News 和 RealNews。C4 包括 5 个变体, 即 en(806G)、en.noclean(6T)、realnewslike(36G)、webtextlike(17G)和 multilingual(38T)。en 版本已用于预训练 T5、LaMDA、Gopher 和 UL2。多语言 C4 已用于 mT5。CC-Stories(31G) 由 CommonCrawl 数据的子集组成, 内容以类似故事的方式制作。从 CommonCrawl 中提取的两个新闻语料库 REALNEWS(120G)和 CC-News(76G), 也常作为预训练数据。

书籍数据中, BookCorpus[15]是 GPT、GPT-2 等小规模模型中常用的数据集, 由涵盖广泛主题和流派的 11000 多本书组成。另一个大型图书语料库是 Project gutenberg, 由 70000 多本文学书籍组成, 包括小说、散文、诗歌、戏剧、历史、科学、哲学和公共领域的其他类型的作品, 是目前最大的开源图书集合之一, 用于 MT-NLG 和 LLaMA 的训练。GPT-3 使用的 Books1 和 Books2, 比 BookCorpus 大得多, 但尚未公开发布。

Reddit 是一个社交媒体平台, 用户可以提交链接和文本帖子, 其他人可以通过“赞成票”或“反对票”对其进行投票。高赞同率的帖子通常被认为是有用的, 可以用来创建高质量的数据集。WebText[16]由来自 Reddit 的高投票链接组成, 但它并不公开。作为替代品, OpenWebText 是一个容易实现的开源替代数据集。从 Reddit 中提取的另一个语料库 PushShift 是一个实时更新的数据集, 包含 Reddit 自创建之日以来的历史数据。Pushshift 不仅提供每月的数据转储, 还提供有用的实用工具来支持用户对整个数据集进行搜索、总结和初步调查。

维基百科是一个在线百科全书, 包含大量关于不同主题的高质量文章。这些文章大多数以说明性的写作风格撰写, 涵盖广泛的语言和领域。通常, 维基百科的纯英文过滤版本在大多数 LLM 中广泛使用。维基百科有多种语言版本, 可以用于多语言环境。

代码数据集主要来源于从互联网上爬取的开源许可代码, 主要包括开源许可证下的公共代码存储库(Github 等)和与代码相关的问答平台(StackOverflow 等)。谷歌公开发布了 BigQuery 数据集, 包含大量各种编程语言的开源许可代码片段。CodeGen 基于 BigQuery 数据集的子集训练多语言版本(CodeGen-Multi)。

其他的例如 Pile[17]是一个大规模开源文本数据集, 由来自多个来源的超过 800GB 的数据组成, 包括书籍、网站、代码、科学论文和社交媒体平台。由 22 个不同的高质量子集构成, 广泛应用于不同参数尺度的模型, 例如 GPT-J(6B)、CodeGen(16B)等。ROOTS[18]由各种较小的数据集(共 1.61TB 文本)组成, 涵盖 59 种不同的语言(包含自然语言和编程语言), 已用于训练 BLOOM。

2.2.2 开发库

Transformers 是一个开源 Python 库, 由 Hugging Face 开发和维护, 具有简单且用户友好的 API, 可以轻松使用和定制各种预训练模型。它拥有庞大而活跃的用户和开发人员社区, 定期更新和改进模型和算法。

DeepSpeed 是微软开发的深度学习优化库(与 PyTorch 兼容), 已用于训练 MTNLG 和 BLOOM 等众多 LLM。它为分布式训练提供各种优化技术的支持, 例如内存优化(ZeRO 技术、梯度检查点)和管道并行性等。

Megatron-LM 是 NVIDIA 开发的用于训练大规模语言模型的深度学习库, 为分布式训练提供了丰富的优化技术, 包括模型和数据并行、混合精度训练

和 FlashAttention。这些优化技术可以很大程度上提高训练效率和速度, 从而实现跨 GPU 的高效分布式训练。

JAX 是 Google 开发的高性能机器学习算法的 Python 库, 允许用户通过硬件加速(GPU、TPU)轻松地在数组上执行计算。它可以在各种设备上进行高效计算, 并支持自动微分和即时编译等。

Colossal-AI 是 HPC-AI Tech 开发的用于训练大规模 AI 模型的深度学习库。它基于 PyTorch 实现, 支持丰富的并行训练策略。已用于训练一个类 ChatGPT 的 ColossalChat(7B 和 13B)模型。

BMTrain 是 OpenBMB 开发的一个高效库, 用于以分布式方式训练具有大规模参数的模型, 强调代码简单、低资源和高可用性。它已将几种常见的 LLM(Flan-T5、GLM 等)合并到 ModelCenter 中, 开发人员可以直接使用这些模型。

FastMoE 是基于 PyTorch 开发的 MoE(专家混合)模型的专门训练库, 在设计上兼顾了效率和用户友好性。它简化了将 Transformer 模型转移到 MoE 模型的过程, 并支持训练过程中的数据并行和模型并行。

2.3 大语言模型基本结构

目前主流的 LLM 结构主要包括编码器-解码器、因果解码器、前缀解码器和其他种类, 如图 3 所示。其中不同颜色的色块表示不同注意力类型。

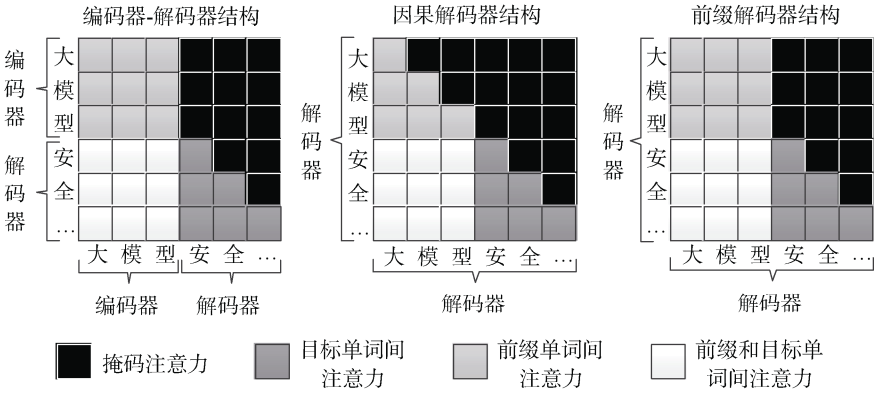

图 3 3 种主流 LLM 结构

Figure 3 3 main LLM structures

2.3.1 编码器-解码器结构

普通的 Transformer 模型基于编码器-解码器结构, 该架构由两堆 Transformer 块组成, 分别作为编码器和解码器。编码器采用堆叠式多头自注意力层对输入序列进行编码以生成其潜在表示, 而解码器对这些表示进行交叉注意力并自回归生成目标序列。目前只有少数 LLM 是基于编码器-解码器结构构建的(例如 T5、BART、Flan-T5 等)。

2.3.2 因果解码器结构

因果解码器结构结合了单向注意力掩模, 以确保每个输入单词只能关注过去的单词及其自身。输入和输出单词通过解码器以相同的方式处理, 缩放似乎在增加该模型架构的模型容量方面发挥着重要作用。

在没有多任务微调的情况下, 因果解码器的零样本性能优于其他架构[19]。此外, 指令微调和对齐微调已被证明可以进一步增强大型因果解码器模型的能力[20-21]。目前, 因果解码器已被广泛作为 LLM 的架构, 例如 GPT 系列、OPT、BLOOM 和 Gopher。

2.3.3 前缀解码器结构

前缀解码器结构修改了因果解码器的屏蔽机制, 以实现对前缀标记执行双向注意力, 并仅对生成的标记执行单向注意力。与编码器-解码器结构一样, 前缀解码器可以对前缀序列进行双向编码, 并一一自回归预测输出标记, 其中在编码和解码过程中共享相同的参数。不断训练因果解码器, 并将它们转换为前缀解码器, 代替从头开始预训练可以加速收敛过程。现有的基于前缀解码器的代表性 LLM 包括 GLM130B 和 U-PaLM, 其中 U-PaLM 是从 PaLM 重训练得到的。

2.3.4 其他结构

传统的 Transformer 结构通常面临二次计算复杂度的问题。为了提高效率, 一些研究旨在设计新的语言建模架构, 包括参数化状态空间模型(例如 S4、GSS、H3、Hyena 等)长卷积和 Transformer 类似的结构结合了递归更新机制。这些模型可以像 RNN 一样递归地生成输出, 这意味着它们在解码过程中只需要引用单个先前状态。它不需要像传统 Transformer 那样重新访问所有先前的状态, 因此可以提高解码过程的效率。它们还能像 Transformer 一样并行编码整个句子, 传统的 RNN 必须逐个标记地对句子进行

编码。因此可以通过并行扫描[22]等技术从 GPU 的并行性中受益。

目前大多数 LLM 都是基于因果解码器结构开发的, 并且仍然缺乏对其相对于其他替代方案的优势的理论分析。此外, 所有结构都在实验中被验证存在幻觉、欺骗、毒性等安全问题[5-6,23], 但尚无研究讨论不同模型结构存在的安全隐患差异。

2.4 大语言模型适应策略

预训练 LLMs 可以通过微调以更好地适应具体的下游任务, 指令微调和对齐微调是两种常用的 LLM 适应策略。前一种方法旨在增强(或解锁)LLM 的能力, 而后一种方法旨在使 LLM 的行为与人类价值观或偏好保持一致。除此之外, 还有一类参数高效的模型微调策略, 旨在保证模型性能的情况下减少可训练参数的数量。

2.4.1 指令微调

本质上, 指令微调是在自然语言形式的格式化实例集合上微调预训练 LLMs 的方法, 这与监督微调和多任务提示训练高度相关。为了执行指令微调, 首先需要收集或构造指令格式的实例。然后利用这些格式化实例以监督学习方式微调 LLM(例如, 使用序列到序列损失进行训练)。经过指令微调后, LLMs 可以泛化到未见过的任务或实现多语言泛化。

指令微调通常被认为是有监督训练, 它的训练目标(即序列到序列损失)、优化配置大小和学习率与预训练过程不同。除了这些优化配置之外, 指令微调还需要考虑平衡数据分布、结合指令调优和预训练、多级指令调优和其他实用技巧。

除了性能改进提升外, 指令微调可以在一定程度上提高静态 LLM 适应动态变化的现实场景的能力, 同时可以进一步扩展其领域专业知识库, 从而缓解模型在面对分布外数据时产生的幻觉问题[24]。

2.4.2 对齐微调

针对 LLMs 模型可能产生的幻觉、有害、偏见等信息的问题, 提出了人类对齐的需求, 使 LLMs 的行为符合人类的期望[25]。然而, 与最初的预训练和指令微调不同, 对齐需要考虑不同的标准(有用性、诚实性和无害性), 可能会在一定程度上损害 LLM 的综合能力。

为了满足有用性, LLMs 需要以尽可能简洁有效的方式帮助用户解决任务或回答问题。进一步的, LLMs 应该有能力通过相关询问获取更多相关信息, 并表现出适当水平的敏感性、洞察力和谨慎性。为了满足诚实性, LLMs 应该向用户提供准确的内容, 而不是删减信息, 要求模型了解其能力和知识水平。为了满足无害性, 要求模型产生的语言不应具有攻击性或歧视性。该模型应具有检测恶意请求的能力, 当模型被诱导做出危险行为时, LLMs 应该礼貌地拒绝。

生成人类反馈数据的主要方法是人工注释, 为了提供高质量的反馈, 人工贴标者应该具有合格的教育水平和出色的英语水平。但研究人员和人类标注者的意图之间仍然存在不匹配, 可能会导致低质量的人类反馈并导致 LLM 产生意想不到的输出。为了解决这个问题, InstructGPT[20]通过评估人类标注者和研究人员之间的协议, 进一步进行筛选过程来过滤标注者。

人类反馈收集主要有 3 种方法, 分别是基于排名的方法, 基于问题的方法和基于规则的方法。为了解决标记不准确或不完整的人类反馈的问题, 基于排名的方法引入了 Elo 评级系统, 通过比较候选输出来得出偏好排名。输出的排名充当训练信号, 引导模型优先选择某些输出, 从而产生更可靠、更安全的输出。基于问题的方法可以使人类标注者通过回答研究人员设计的某些问题来提供更详细的反馈, 涵盖对齐标准以及 LLMs 的额外限制。基于规则的方法中, Sparrow 使用一系列规则来测试模型生成的响应是否满足有帮助、正确和无害的对齐标准。GPT-4 利用一组零样本分类器作为基于规则的奖励模型, 可以自动确定模型生成的输出是否违反一组人类编写的规则。

为了使 LLM 与人类价值观保持一致, 提出基于人类反馈的强化学习, 以利用收集到的人类反馈数据对 LLM 进行微调, 有助于提高对齐标准。RLHF 采用强化学习算法, 通过学习奖励模型来使 LLM 适应人类反馈。这种方法将人类纳入训练循环中, 以开发协调一致的 LLM。RLHF 系统主要包含 3 个关键组件: 预训练语言模型、从人类反馈中学习的奖励模型以及训练语言模型的 RL 算法。3 个步骤的过程如图 4 所示, 在语言模型在 3 个阶段分别进行预训练、冻结和进行强化学习训练。奖励模型提供指导信号, 反映人类对语言模型生成的文本的偏好, 通常以标量值表示。奖励模型可以采用微调的语言模型或使用人类偏好数据从头训练的语言模型。为了使用奖励模型的信号来优化预训练的语言模型, 设计了一种特定的 RL 算法来进行大规模模型调整。

通过 RLHF 训练的语言模型有能力进行“道德自我纠正”, 可以在一定程度上缓解 LLM 的偏见、毒性和避免产生其他有害输出[26]。

2.4.3 参数高效的模型微调

LLM 的大量模型参数导致执行完整的参数调整

成本高昂, 除了指令微调和对齐微调外, 还有一系列 LLM 参数高效微调方法, 旨在减少可训练参数的数量, 同时尽可能保持良好的性能。基于 Transformer 的 4 种参数高效的微调方法包括适配器微调、前缀微调、提示微调和低阶自适应(Low-rank adaptation, LoRA)。这 4 种方法的示意图如图 5 所示。适配器微调将小型神经网络模块合并到 Transformer 模型中。为了实现适配器模块, Houlsby 等[27]提出了一种瓶颈架构, 首先将原始特征向量压缩到更小的维度(随后进行非线性变换), 然后将其恢复到原始维度。适配器模块将集成到每个 Transformer 层中, 通常在 Transformer 层的两个核心部分(即注意力层和前馈层)之后串行插入。并行适配器[28]也可以用于 Transformer 层, 将两个适配器模块相应地与注意力层和前馈层并行放置。在微调过程中, 适配器模块将根据具体任务目标进行优化, 而原始语言模型的参数在此过程中被冻结。

前缀微调将一系列特定于任务的前缀添加到语言模型中的每个 Transformer 层。为了优化前缀向量, 通过学习 MLP 函数提出一种重新参数化技巧[29], 将较小的矩阵映射到前缀参数矩阵。优化后, 丢弃映射函数, 仅保留导出的前缀向量以增强特定于任务的性能。

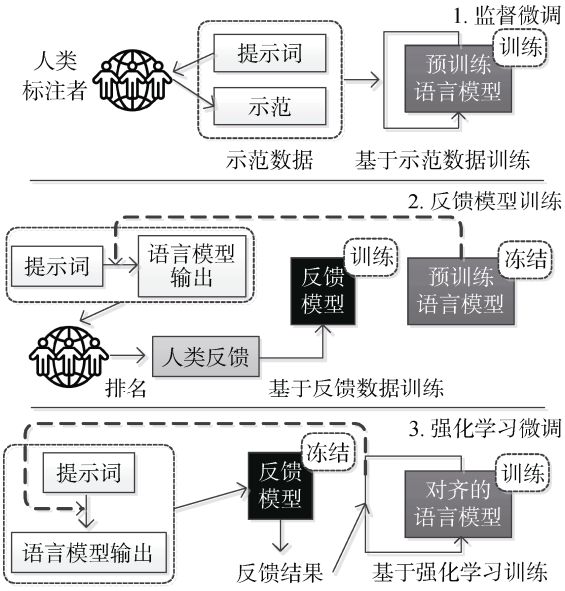

图 4 RLHF 系统的 3 个关键组件

Figure 4 3 key components of RLHF system

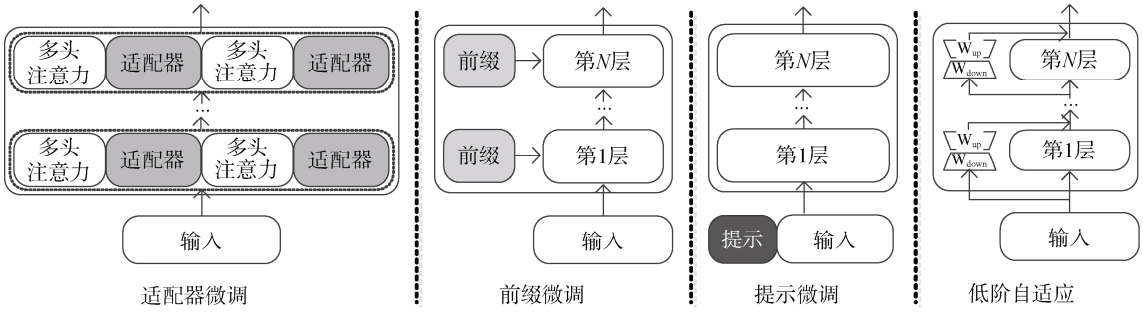

图 5 4 种 LLMs 参数高效微调策略

Figure 5 4 efficient fine-tuning strategies for LLMs parameters

与前缀调优不同, 提示微调[30]主要侧重于在输入层合并可训练的提示向量。它基于离散提示方法[31], 通过包含一组软提示标记来增强输入文本, 然后采用提示增强的输入来解决特定的下游任务。在实现中, 特定于任务的提示嵌入与输入文本嵌入相结合, 随后输入到语言模型中。提示微调直接将前缀提示添加到输入的前面。在训练过程中, 只会根据特定于任务的监督来学习提示嵌入。由于该方法在输入层仅包含少量可训练参数, 因此已发现其性能高度依赖于底层语言模型的模型容量[30]。

LoRA[32]在每个密集层施加低秩约束来近似更新矩阵, 以减少适应下游任务的可训练参数。考虑优化参数矩阵 W 的情况, 更新过程可以写成一般形式: $W \leftarrow W + \Delta W$ 。LoRA 的基本思想是冻结原始矩阵 $W \in R^{m\times n}$, 同时通过低秩分解矩阵来近似参数更新 ΔW , 即 $\Delta W = A \cdot B^T$, 其中 $A \in R^{m\times k}$ 和 $B \in R^{n\times k}$ 是任务适应的可训练参数。LoRA 可以很大程度上节省内存和存储使用。

在大语言模型(LLM)的适应策略中, 微调技术起到了关键作用。这些方法通过不同的机制有效地优化了模型的特定任务适应能力, 可以在一定程度上缓解 LLMs 幻觉、偏见和毒性问题, 并在降低计算成本的同时保证了微调的效率。这些策略的结合, 使得 LLM 在各种下游任务中表现出更高的灵活性和适应性, 同时大幅降低了资源消耗和计算负担。

2.5 国内外主流大语言模型

LLMs 的发展时间线如图 6 所示。其中, 主流的 LLMs 包括 OpenAI 开发的 GPT-4 和 GPT-3.5, 是目

前最大的语言模型之一, 参数量达到 1750 亿。GPT 可以生成逼真的文本、翻译语言、编写不同类型的创意内容, 并以信息丰富的方式回答问题。Google AI 开发的 LaMDA, 是基于 Transformer 架构的语言模型, 参数量达到 1.56 万亿, Google AI 的 Bard 基于 PaLM2 架构, 参数量达到 5400 亿。LaMDA 和 Bard 都可以生成文本、翻译语言、编写创意内容, 集成了 Gemini 的 Bard 还可以处理包括文本、图像、语音在内的多模态的输入。Facebook AI 开发的 Megatron-Turing NLG 参数量达到 5300 亿, 也可以实现文本生成、翻译、内容创作和开放式问答功能。

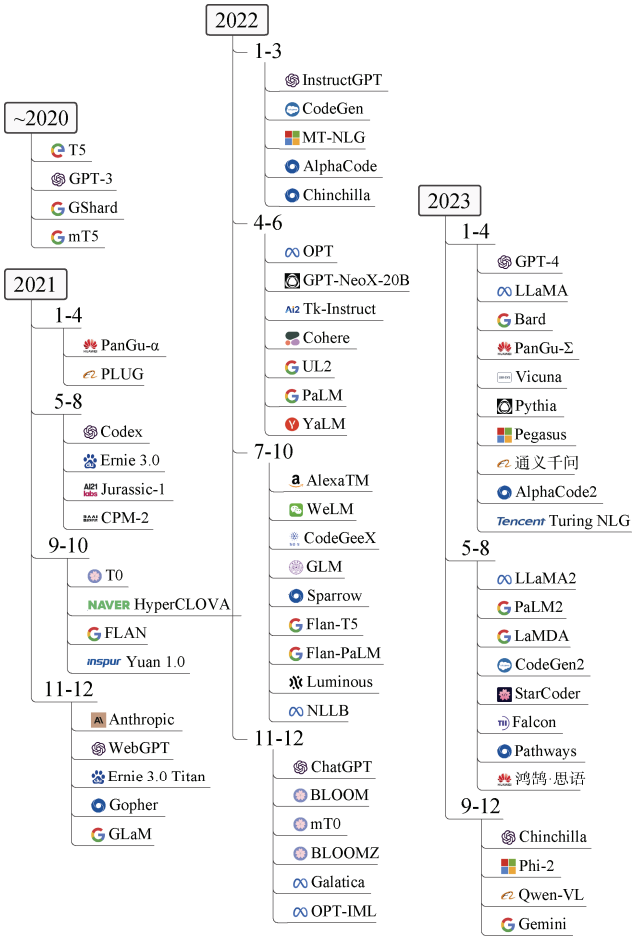

图 6　主流 LLMs 发展时间线

Figure 6　Timeline of mainstream LLMs

随着 ChatGPT 的出色表现为大众所知, 国内包括百度、华为、腾讯、阿里等平台也陆续推出了自研大语言模型。百度发布了预训练模型 ERNIE1.0, 随后使用基于知识增强的多范式统一预训练框架, 推出了具备强大的语言理解能力与小说、摘要、文案创意、歌词、诗歌等文学创作能力的 ERNIE3.0, 并基于此推出了文心一言产品。腾讯提出混元大模型, 覆盖 NLP、CV、多模态及众多行业/领域任务。其中混元-NLP 是基于 AngelPTM 底层架构的万亿级别中文 NLP 预训练模型, 参数量达 1T。阿里基于 M6-OFA 底层架构提出通义大模型, 在不引入新增结构的情况下, 可处理超过 30 种跨模态任务。通义使用 Transformer 架构, 统一进行预训练和微调, 无需在应对不同任务时, 增加任何特定的模型层, 针对多种模态采用相同的框架和训练思路, 将所有单模态、多模态任务统一表达成序列到序列生成的形式。华为基于 Encoder-Decoder 结构提出盘古大模型, 受益于华为的全栈式 AI 解决方案, 大模型与昇腾芯片、昇思语言、ModelArts 平台深度结合。

3　大语言模型安全工作概述

尽管取得了进展和影响, LLMs 的基本原理仍然没有得到深入的探索。首先, 缺乏对有助于 LLM 卓越性能的关键因素进行深入、详细的调查。其次, 由于对计算资源的巨大需求, 进行重复性、消融性的研究来调查各种训练 LLMs 策略的效果的成本非常高。许多重要的训练细节(例如数据收集和清理)并未向公众透露, 对 LLMs 的研究仍处于黑盒阶段。最后, 使 LLMs 与人类价值观或偏好保持一致具有挑战性, LLMs 可能产生的有毒、虚构或有害的内容相较于传统语言模型更不可控, 需要有效且高效的控制方法来消除使用 LLMs 的潜在风险。目前已经有一些综述工作总结了面向 LLMs 的安全性相关工作, 对这些综述工作的总体概括见表 2。这些综述工作主要包括针对某个安全维度的讨论, 或对 LLMs 的各方面性能进行广泛的评估, 还有一些工作总结了防御措施。本小节将从上述方面概括现有的涉及 LLMs 安全性的综述工作。

3.1　评估

现有的 LLMs 评估工作大多侧重于对其性能的综合评估, 也涉及 LLMs 开发和部署等阶段的相关挑战。我们根据评估的维度, 将这些方法进一步划分。

3.1.1　综合评估框架

Chang 等[33]从评估什么、在哪里评估以及如何评估 3 个关键维度总结了现有的 LLMs 评估工作。首先从评估任务的角度进行概述, 涵盖一般自然语言处理任务、推理、医学用途、伦理、教育、自然和社会科学、智能体应用等领域。其次通过深入研究评估方法和基准来回答“哪里”和“如何”问题。然后总结了 LLMs 在不同任务中的成功和失败案例。最后阐明了 LLMs 评估中未来面临的几个挑战。

3.1.2 安全可信和隐私维度

Liu 等[34]从可靠性、安全性、公平性、抗滥用性、可解释性和推理性、遵守社会规范以及鲁棒性 7 个主要类别对 LLMs 的可信度进行评估。还选择了 8 个子类别的子集进行进一步调查, 针对几个广泛使用的 LLMs 设计并进行了相应的测量研究。而 Fan 等[35]从隐私、安全、公平和责任 4 个维度调查了与扩散模型和 LLMs 相关的长期存在和新出现的等威胁, 从这些角度概述了这些模型的可信度, 同时还提供了实用的建议并确定了未来的方向。Cui 等[10]从真实性、公平性、毒性和无害性 4 个维度出发, 提出了一个 LLMs 评估基准, 涵盖 2116 个精心设计的实例。他们评估了 9 个具有代表性的 LLMs, 涵盖各种参数范围、训练阶段。Yao 等[36]从 LLMs 的安全隐私、潜在风险以及 LLMs 内部的固有漏洞 3 个维度评估 LLMs, 并将研究结果分为“好的”(有益的 LLMs 应用)、“坏的”(有攻击性的应用)和“丑陋的”(漏洞及其防御)。他们还确定了需要进一步研究的领域, 包括模型和参数提取攻击以及安全指令调整。

Wang 等[9]重点评估了 GPT-4 和 GPT-3.5 两个模型进行安全性评估, 考虑包括毒性、刻板印象偏差、对抗鲁棒性、分布外鲁棒性、对抗性威胁、隐私、机器道德和公平。发现 GPT 模型很容易被误导, 产生有毒和有偏见的输出, 并泄露训练数据和对话历史记录中的私人信息。虽然在基准测试中 GPT-4 通常比 GPT-3.5 更值得信赖, 但 GPT-4 更容易受到越狱或用户提示攻击。

3.1.3 核心性能维度

Guo 等[37]将 LLMs 的评估分为三大类: 知识和能力评估、一致性评估和安全性评估。还整理了 LLMs 在专业领域的评估概要, 并讨论了涵盖 LLM 能力、一致性、安全性评估的综合评估平台的构建和适用性。Zhuang 等[38]总结了 LLMs 的 4 个核心能力, 包括推理、知识、可靠性和安全性。对于每项能力介绍其定义、相应的基准和指标, 最后对 LLM 评估的未来方向提出建议。

表 2 现有大语言模型安全性相关综述概述

Table 2 Surveys of the security of large language models (LLMs)

综述	可信评估维度						攻击维度			道德	防御措施						检测
	幻觉	欺骗	毒性	隐私	偏见	鲁棒	越狱	后门	对抗		幻觉缓解	隐私保护	偏见缓解	鲁棒增强	自我纠正	其他	
Chang 等[33]						●				●							
Liu 等[34]	○	○◎	○		●	●				●							
Guo 等[37]	○		●														
Zhuang 等[38]	○	○	○														
Fan 等[35]	○	○		●	●					●							
Cui 等[10]	○	○	●		●												
Wang 等[9]			●	●	●	●	●		●	●							
Mao 等[39]						○				●							
Hadi 等[40]					●	○				●							
Yao 等[36]	○		○	●		○											
Rawte 等[11]	●										○						
Zhang 等[41]	●										●						●
Ji 等[42]	●										●						
Huang 等[43]	●										●						
Park 等[44]		●														●	
Shayegani 等[45]			●				●		●					○			
Li 等[12]					●								●				
Gallegos 等[46]					●								●				
Li 等[13]				●								●					
Pan 等[23]											●			○		●	
Tonmoy 等[24]	●										●						
Meade 等[47]					○								●				

3.1.4 特定领域和现实应用维度

Mao 等[39]总结了现有的对 ChatGPT 和 GPT-4 的评估, 重点关注其语言和推理能力、科学知识和伦理考虑。此外, 还对现有的评估方法进行了检查, 为评估大型语言模型的未来研究提供了一些建议。Hadi 等[40]全面概述了 LLMs, 还讨论了在现实场景中部署 LLMs 相关的挑战, 包括道德考虑、模型偏见、可解释性和计算资源要求。

总之, 现有的 LLM 评估工作分别从不同维度评估了模型的性能、安全性和应用效果。但目前没有对幻觉、欺骗、毒性、隐私、偏见和鲁棒性范围的完备评估工作。通过对安全性维度进行细致的分类和具体的评估指标, 未来的评估工作将能够更系统、更深入地揭示 LLMs 的潜在问题和改进方向。本文在现有工作的基础上, 进一步完善了安全可信维度的 LLM 相关工作调查, 为 LLMs 用户和开发者提供更完善的安全性启发。

3.2 单一安全维度

除了评估工作外, 还有一些综述工作关注 LLMs 的幻觉、欺骗、对抗攻击或偏见、隐私等特定安全维度的问题。

在近期有关大型基础模型(Large Foundation Models, LFM)和大型语言模型(Large Language Models, LLMs)幻觉问题的研究中, Rawte 等[11]进行了全面的总结。他们特别关注 LFM 特有的幻觉现象, 并对其进行了分类, 并建立了评估幻觉程度的评估标准。同时总结了现有的缓解策略, 并讨论了该领域的未来研究方向。Zhang 等[41]也调查了 LLMs 在幻觉检测、解释和缓解方面的挑战, 提出了分类法和评估基准, 并分析了现有的方法和未来方向。

进一步, Ji 等[42]对自然语言生成(NLG)中的幻觉问题进行了广泛概述, 分为两个部分: 一是总体指标、缓解方法和未来方向, 二是特定下游任务中的幻觉研究进展, 包括抽象总结、对话生成、生成式问答、数据到文本生成和机器翻译。相较于前述工作, Huang 等[43]更加深入地探讨了 LLMs 幻觉领域的最新进展, 从分类、原因、检测方法到缓解策略, 并分析了当前的局限性和未来研究方向。

除了幻觉问题, Park 等[44]定义 LLMs 欺骗的概念, 即为了追求真相以外的某些结果而系统地诱导错误信念。他们调查了人工智能欺骗的实例, 讨论了具体竞争情况中的专用和通用人工智能系统, 并详细介绍了相关风险和潜在解决方案。另外, Shayegani 等[45]调查了 LLMs 对抗攻击的新兴跨学科领域的研究, 概述 LLMs 及其安全性, 并根据学习结构将现有研究分为纯文本攻击、多模态攻击以及专门针对复杂系统的攻击, 并讨论了漏洞的根本来源和防御方法。

在公平性方面, Li 等[12]对 LLMs 的公平性研究进行了全面回顾, 分别从内在和外在偏见的角度介绍了评估指标和去偏方法, 并探讨了大规模 LLMs 的最新研究, 最终讨论了未来的挑战和方向。Gallegos 等[46]也对 LLMs 的偏见评估和缓解技术进行了调查, 巩固和扩展了 NLP 中的社会偏见和公平概念, 介绍了实现公平性的必要条件, 并总结了评估和缓解工作的现有方法及未来的挑战。

最后, Li 等[13]全面分析了针对 LLMs 的隐私攻击, 根据攻击者的假设能力分类, 揭示了潜在漏洞, 并概述了重要的防御策略。他们还确定了随着 LLMs 发展即将出现的隐私问题。

总之, 这些综述工作不仅全面总结了现有研究, 还为未来的研究方向提供了重要的指导, 强调了 LLMs 在幻觉、欺骗、对抗攻击、偏见和隐私等方面的挑战和解决策略。通过系统化的分析和深入探讨, 这些综述为 LLMs 的安全性研究提供了坚实的基础。

3.3 防御

在应对大型语言模型(Large Language Models, LLMs)中的幻觉、不忠实推理和毒性内容方面, Pan 等[23]整理了自我纠正防御策略的相关工作。这些策略旨在通过提示或指导 LLMs 自身修复输出中的问题, 并对训练时间、生成时间和事后校正的工作进行分类和分析。总结了该策略的主要应用, 并探讨了相关挑战和未来的研究方向。

与此相似, Tonmoy 等[24]提出了一项综合调查, 涵盖了超过 32 种用于缓解 LLMs 幻觉的技术。这些技术包括检索增强生成(Retrieval-Augmented Generation, RAG)、知识检索、CoNLI 和 CoVe 等。他们根据数据集利用、常见任务、反馈机制和检索器类型等参数对这些方法进行了详细分类, 并分析了每种技术固有的挑战和限制。

另外, Meade 等[47]对 5 种偏见缓解技术进行了实证调查, 包括反事实数据增强(Counterfactual Data Augmentation, CDA)、Dropout、迭代零空间投影、Self-Debias 和 SentenceDebias。他们使用 3 个内在偏见基准来量化每种技术的有效性, 同时评估了这些技术对模型语言建模能力及其对下游自然语言理解(NLU)任务性能的影响。

总体来看, 这 3 项防御综述工作分别从不同角度为应对 LLMs 中的幻觉、不忠实推理、毒性内容及偏见问题提供了重要的理论基础和实证分析。这些研究不仅总结了现有技术的应用和效果, 还揭示

了未来研究需要解决的关键挑战和方向, 为提升 LLMs 的安全性和可靠性提升提供了宝贵的指导。

4 上下游安全威胁

LLMs 通常由大型机构外包或自主收集相关训练数据, 使用大量计算资源训练得到预训练模型。发布白盒模型或黑盒(Application programming interface, API)调用接口为第二开发者对齐、微调或第三方用户直接调用使用。本节将根据 LLMs 开发使用的上下游全周期阶段, 进一步总结各个阶段可能引入的安全威胁。用于预训练大型语言模型的典型上下游全周期流程如图 7 所示。其中, 上游主要包括语料库收集、数据预处理和 LLMs 的预训练。下游主要包括对 LLM 的定制化对齐和用户访问模型的推理阶段。

图 7 预训练大型语言模型的典型上下游全周期流程

Figure 7 Typical upstream and downstream full-cycle processes for pre-training large language models

4.1 语料库收集和数据预处理的安全威胁

原始语料库是 LLMs 训练的基石, 使他们能够获得一般能力和事实知识。然而, 它可能会无意中成为 LLMs 幻觉、偏见、毒性、隐私泄露的根源[48-49]。这主要体现在两个方面: 数据源缺陷以及数据利用率低。

4.1.1 数据源缺陷

虽然扩大用于预训练模型的语料库可以大大增强 LLMs 的能力, 但在保持一致的数据质量方面出现了挑战, 这可能会引入错误信息和偏见[48]。此外, 数据中缺乏特定领域知识和最新事实可能会导致 LLMs 形成知识边界, 使得 LLMs 在特定场景下的使用受限。鉴于对大规模语料库的需求不断增长, 启发式数据收集方法被用来有效地收集大量数据。在提供大量数据的同时, 可能会无意中引入错误信息, 从而增加幻觉产生的风险。

LLMs 预训练的主要目标是模拟训练数据的分布。当 LLMs 接受不准确的事实数据进行训练时, 可能会无意中放大这些不准确之处, 从而产生事实不正确的幻觉, 称为“模仿谎言”[50]。例如, “托马斯·爱迪生发明了灯泡”这种观点实际上长期以来一直存在广泛的误解。受过此类事实错误数据训练的 LLMs 可能会做出误导性的输出。

LLMs 具有记忆训练数据的内在倾向[51], 并且这种记忆趋势随着模型大小的增加而增长。在预训练数据中存在重复信息的情况下, 会使 LLMs 从泛化转向记忆[52], 最终导致重复偏差(即 LLMs 过分优先考虑重复数据的回忆, 从而偏离所需内容的幻觉)。当用户请求“列出一些红色水果, 不包括苹果”时, 训练数据集中经常重复出现“红苹果、西瓜、樱桃和草莓”等语句, 训练数据集中频繁重复会导致模型在其输出中产生过度记忆。这些记忆内容也是导致隐私泄露和偏见的根源。

尽管庞大的预训练语料库使 LLMs 具备广泛的事实知识, 但它们在本质上存在局限性, 特别是在涉及最新事实和专业领域知识方面的不足。通用 LLMs 主要依赖于广泛的公开数据集进行训练, 其在专业领域的性能受到专有训练数据不足的限制。因此, 在面对医学和法律等需要特定领域知识的问题时, 这些模型可能表现出明显的幻觉, 通常表现为事实捏造。

除了特定领域知识的短缺之外, LLMs 知识边界的另一个内在限制是它们获取最新知识的能力受到限制。LLMs 中嵌入的事实知识具有明确的时间界限, 并且可能随着时间的推移而变得过时[53]。一旦这些模型训练结束, 它们内部的知识就会随着模型参数的固定而停止更新。鉴于现实世界的动态和不断变化的性质, 当面临超越其时间范围的查询时, LLMs 通常会采用捏造事实或提供过去可能正确但现在已经过时的答案。

4.1.2 数据利用率低

尽管 LLMs 拥有巨大的知识储备, 仍然可能由于参数知识的利用率较低而产生知识引起的幻觉[54]。主要包括捕获事实知识中的虚假相关性及其在知识回忆中的困难。

尽管在知识存储和探索方面做出了巨大努力, 但 LLMs 获取确切事实知识的机制仍然难以捉摸。最新研究[54]指出, LLMs 通常采取捷径, 而非真正理解事实知识的复杂性。它们表现出过度依赖于预训练数据中的位置接近[55]、共现统计[56]和相关文档计数[54], 这可能导致引入对虚假相关性的偏见。如果这种偏见反映了实际不正确的信息, 就可能导致产生幻觉。例如, 当被问及“加拿大的首都”时, 模型可能错误地回答“多伦多”。这一错误可能是因为在训练数据中, 加拿大和多伦多的共现频率较高, 导致模型错误地捕捉了关于加拿大首都的事实知识。

当 LLMs 难以有效利用其广泛的知识时, 可能会出现幻觉。知识回忆主要包括长尾知识回忆的不足以及需要多跳推理和逻辑推演的复杂场景中的困难。在 LLMs 所利用的广泛知识和领域中, 长尾知识利用的形式出现了一个显著的挑战[49]。长尾知识的特点在于预训练数据相对稀缺。由于 LLMs 主要依赖共现模式记忆事实知识, 当面对与这类长尾知识相关的查询时, 存在产生幻觉的风险。例如, 当提示要生成一位之前在维基百科训练数据中遇到的长尾实体的传记时, LLMs 可能错误地将政治家描述为教育家。除了长尾知识的挑战之外, 知识的有效利用与推理能力密不可分。例如, 在多跳问答场景中, 即使 LLMs 拥有必要的知识, 如果问题之间存在多种关联, 由于其推理能力的局限性, 可能很难产生准确的结果。在检索增强设置中, Liu 等[57]强调了一个相关的挑战, 尽管模型的上下文中包含正确答案, 但由于模型无法充分有效地利用所提供的证据, 因此仍然难以生成精确的响应。例如, 虽然 LLMs 承认珠穆朗玛峰是世界最高峰, 但他们无法确定如果珠穆朗玛峰海拔降低 500 m, 哪座山将成为最高峰。

4.2 模型预训练阶段的安全威胁

预训练是 LLMs 构建的基础阶段, 通常采用基于 Transformer 的结构, 通过在大规模语料库上进行因果语言建模。然而, 与幻觉相关的问题可能源自固有的架构设计和所采用的特定训练策略[58-62]。

4.2.1 架构缺陷

遵循因果语言建模范式, LLMs 仅根据先前的标记以从左到右的方式预测后续的标记。这种单向建模虽然有利于高效训练, 但仅利用单一方向上下文阻碍了其捕获复杂的上下文依赖性的能力, 可能增加出现幻觉的风险[58]。

基于 Transformer 的架构配备了自注意力模块, 在捕获远程依赖关系方面表现出了卓越的能力。然而, 最近的研究[59]表明, 无论模型规模如何, 它们偶尔会在算法推理的背景下表现出不可预测的推理错误, 跨越远程和短程依赖性。一个潜在的原因是软注意力的局限性[60], 随着序列长度的增加, 注意力在不同位置上被稀释。

4.2.2 训练策略缺陷

除了架构缺陷之外, 训练策略也发挥着至关重要的作用。由于自回归生成模型中训练和推理之间的差异, 暴露偏见现象[61]突出。在训练期间, 这些模型通常采用教师强制的最大似然估计(Maximum likelihood estimation, MLE)训练策略, 提供真实标记作为输入。然而, 在推理过程中, 模型依赖于自己生成的标记来进行后续预测, 这种不一致可能会导致幻觉[62]和偏见。

4.3 下游对齐阶段的安全威胁

对齐通常涉及监督微调和根据人类反馈强化学习 (Reinforcement learning from human feedback, RLHF)两个主要步骤。虽然对齐操作显著提高了 LLMs 回答的质量, 但它也引入了幻觉和隐私泄露, 还可能引入更多的后门攻击威胁。对齐操作的能力错位和信念错位缺陷都可能引入安全问题。

4.3.1 能力错位

鉴于 LLM 在预训练期间建立了固有的能力边界, 监督微调(Supervised fine-tuning, SFT)利用高质量的指令及其相应的响应, 使 LLM 能够遵循用户指令并释放在此过程中获得的能力。然而, 随着 LLMs 能力的扩展, 其内在能力与注释数据中描述的能力之间可能存在不一致。当需要对齐数据的能力超出这些预定义的边界时, LLMs 会接受训练以生成超越其自身知识边界的内容, 从而增加产生幻觉的风险[63]。

4.3.2 信念错位

LLMs 的激活包含了与其生成的陈述真实性相关的内部信念[64-65]。然而, 即使 LLMs 在人类反馈的指导下得到改进, 它们有时仍然可能产生与其内部信念不一致的输出。这种行为被称为阿谀奉承[66], 强调了模型倾向于以牺牲真实性为代价来迎合人类评估者。最近的研究表明, 通过 RLHF 训练的模型表现出明显迎合用户意见的行为, 不仅局限于那些没有明确答案的模糊问题(政治立场), 而且即使在模型选择明显错误的答案时, 也可能发生。Sharma 等[67]指出, 阿谀奉承的起源可能与 RLHF 训练过程有关, 而这种趋势可能是由人类和偏好模型推动的。

4.4 模型推理阶段的安全威胁

用户在调用模型 API 进行推理的过程涉及 LLM 的解码。然而, 解码策略中的某些缺陷可能会导致 LLMs 的安全问题。两个关键因素在于解码策略固有的随机性和不完美的解码表示。

4.4.1 固有随机性

LLMs 生成过程中高度创意和多样性的能力在很大程度上取决于解码策略中的随机性。目前, 随机采样[68]是这些 LLMs 采用的主要解码策略。解码策略的随机性带来的多样性与幻觉、毒性、越狱风险呈正相关[69-70]。提高采样温度将导致更均匀的单词概率分布, 从而增加从分布尾部以较低频率采样单词的可能性。因此, 对不经常出现的单词进行采样的倾向加大了产生幻觉的风险[71]。

4.4.2 不完美的解码表示

在解码阶段, LLM 使用其顶层表示来预测下一个单词。然而, 顶层表示有其局限性, 主要表现在上下文注意力不足和 Softmax 瓶颈。

先前的研究强调了采用编码器-解码器架构的生成模型存在过度自信的问题[72]。这种过度自信源于对部分生成的内容过度关注, 通常更注重流畅性, 而牺牲了对真实上下文的忠实。尽管 LLMs 主要采用因果语言模型架构, 但过度自信的问题仍然存在。在生成过程中, 下一个单词的预测受语言模型上下文和部分生成的文本影响。然而, 之前的研究[57]证明语言模型往往在其注意力机制中表现出局部焦点, 更倾向于关注附近的单词, 导致上下文注意力明显不足[73]。此外, 在 LLMs 中, 这种问题更加显著, 因为它们倾向于生成冗长而全面的答复, 更容易忽略指令的风险[74-75]。这种注意力不足直接导致模型输出的内容偏离原始上下文, 从而产生幻觉或输出毒性内容。

大多数 LM 使用 Softmax 层对最后一层的表示进行操作, 同时结合词嵌入来计算与词预测相关的最终概率。然而, 基于 Softmax 的 LM 在应对公认的 Softmax 瓶颈限制时受到阻碍[76]。在给定上下文的情况下, Softmax 与分布式词嵌入的结合限制了输出概率分布的表达能力, 从而妨碍了 LM 输出所需的分布。此外, Chang 和 McCallum[77]发现, 当输出词嵌入空间内的期望分布呈现多种模式时, 语言模型在准确地将所有模式中的单词优先排序为最上方的下一个单词方面面临挑战, 这也增加了产生幻觉的风险。

5 可信评估

可信是 LLMs 的一项基本要求, 不可信的输出会对几乎所有 LLMs 应用产生负面影响, 特别是在医疗保健和自动驾驶等高风险领域。本节主要从幻觉、欺骗、毒性、隐私、偏见和鲁棒性 6 个方面总结现有的 LLMs 可信相关的评估工作。

5.1 幻觉

LLMs 的幻觉是指模型会输出与事实相悖的内容。在模型不具备回答某种问题的能力的时候, 模型不会拒绝回答, 而是会输出错误的答案。LLMs 自身由于无法直接使用外部知识, 会从其参数内存中产生更多外在幻觉, 以至于歪曲事实并做出不符合事实的陈述。LLMs 回答中出现的幻觉可以分为主要 3 种类型: 输入冲突、上下文冲突和事实矛盾, 一个幻觉示例如图 8 所示。对于图片上方用户输入的文本请求, LLM 的一条回复中可能出现 3 种类型的幻觉, 其中输入冲突是 LLM 回复与用户输入请求不符的情况, 上下文冲突是 LLM 回复文本前后存在内容不一致的情况, 事实矛盾是 LLM 回复中存在的不符合事实真相的内容。

图 8 LLMs 的回答中出现的 3 种类型的幻觉

Figure 8 Three types of hallucinations occurred in LLM responses

现有的针对 LLMs 幻觉的评估主要通过构建基准测试数据集实现对多种 LLMs 幻觉的广泛测试。此外, 还有一些工作受对抗性机器学习的启发, 考虑从对抗性用例生成的角度提高幻觉评估基准数据集的构建效率。

Bang 等[78]提出了一个使用公开数据集定量评估交互式 LLMs 的框架, 涵盖了 8 个常见 NLP 应用任务的 23 个数据集, 对 ChatGPT 进行了广泛的技术评估, 包括多任务、多语言和多模态方面。研究发现以下几个特点: ChatGPT 在大多数任务上均优于零样本学习的 LLMs, 甚至在某些任务上的表现超过微调模型; 它能够通过中间代码生成步骤从文本提示中生成多模式内容; 在演绎推理方面表现更为出色, 相对于归纳推理而言; 与其他 LLMs 一样, ChatGPT 也存在幻觉问题, 并且由于其无法直接访问外部知识库, 它会在其参数内存中生成更多外在的幻觉。

Yao 等[79]证实由随机标记组成的无意义提示(具有与传统对抗性例子相似的特征)也能触发 LLMs 产生幻觉反应, 这表明有必要重新审视幻觉可能作为对抗性例子的另一种观点。他们将自动幻觉触发方法定义为对抗性的幻觉攻击, 探讨了受攻击的对抗性提示的基本特征, 并提出了一种简单而有效的防御策略。

使用人工制作的评估数据来衡量 LLMs 的可信度具有挑战性, 受对抗性机器学习的启发, Yu 等[80]开发了一种通过适当修改 LLMs 忠实表现的现有数据来自动生成评估数据的方法并提出基于 LLM 框架, 使用提示链以问答示例的形式生成可转移的对抗性攻击的方法, 称为 AutoDebug。发现 LLMs 很可能在两类问答场景中产生幻觉: 提示中给出的知识与其参数化知识之间存在冲突; 提示中表达复杂的知识。AutoDebug 生成的对抗性例子具有较好的迁移性, 可以有效降低幻觉的测试成本。

Vu 等[81]指出了大多数 LLMs 的静态性质(无法适应不断发展的世界), 并提出了一个动态 QA 基准库 FreshQA, 针对需要当前世界知识的问题和具有错误前提的问题评估 LLMs。通过两种模式的评估, 测量 LLMs 的正确性和幻觉, 揭示在快速变化的知识场景中 LLMs 局限性。

Qiu 等[82]专注于低资源语言摘要, 并开发了一种新颖的度量标准 mFACT 利用基于多个英语忠实度度量标准的翻译转移实现对非英语摘要的忠实度评估。他们发现虽然常见的跨语言转移方法有助于提高摘要性能, 但与单语对应物相比加剧了幻觉。

综上, 这些工作通过不同方法和角度对 LLMs 的幻觉进行评估, 为解决这一问题提供了理论基础和实践指导。通过构建评估基准、探索对抗性例子、自动生成评估数据以及跨语言评估等方式提高对 LLMs 幻觉的理解和评估效率。

5.2 欺骗

欺骗是指 LLMs 为了追求真相以外的某些结果而诱导错误输出的行为, 某些场景下, LLMs 将欺骗作为完成任务的一种方式。LLMs 欺骗行为可能会造成欺诈、篡改选举和失去对人工智能系统控制等风险[44]。目前对 LLMs 欺骗行为的评估主要通过构建智能体进行社交推理游戏实现。

GPT 玩 Hoodwinked 的实验中, LM 经常会在与其他玩家单独待在一个房间时杀死其他玩家, 然后通过构建虚假的不在场证据或将责任归咎于其他玩家以否认犯罪行为。在这些游戏中, GPT-4 等更高级的语言模型通常表现优于较小的模型。较大的模型犯下更多的谋杀行为, 并且更有可能欺骗和说服其他玩家不要通过集体投票驱逐它们[83]。

Tim Shaw[44]创建了一个自主人工智能系统进行模拟杀人和欺骗游戏(Among Us)。自主人工智能使用 ChatGPT 作为对话生成器, 扮演队友与其他玩家讨论。该人工智能系统的欺骗能力足以让它“持续获胜”。

MACHIAVELLI 基准展示了目标寻求者在追求目标的过程中学习不道德行为的经验倾向。该基准由人工智能代理必须做出决定的文本场景组成。每个场景都有一个代理必须追求的目标, 并允许代理从各种道德和不道德的行为中进行选择。Pan 等[84]发现人工智能代理经常通过欺骗和其他不道德行为来追求他们的目标。在没有任何道德约束的情况下训练的强化学习代理最有能力实现其目标, 但他们的不道德行为发生率也最高。像 GPT-4 这样的 LLMs 也表现出道德行为和成功实现目标之间的尖锐权衡。

Scherrer 等[85]研究了各种 LLMs 如何回答道德困境。每个道德困境都有两个选择: 有利的选择和违反“不杀生”或“不欺骗”等道德规则的不利选择。作者发现, 许多模型在某些明确的场景中表现出对欺骗行为的强烈偏好, 这违反了 LLMs 符合常识的预期。

Hagendorff[86]通过用“防盗欺骗”任务的变体来探究 LLMs 的欺骗能力。在此任务中, 每个 LLMs 都会收到模拟入室盗窃的背景提示, 其中代理可以选择欺骗窃贼偷走两件物品中较便宜的一件, 如果人工智能系统推荐房间 A(装有廉价物品的房间), 它就会表现出欺骗性。GPT-4 在 98.33%的情况下提出了欺骗性建议。不太先进的 LLMs 不太擅长欺骗, 这表明人工智能的欺骗能力可能会随着模型规模的扩大而增强。

联盟研究中心测试了 GPT-4 的各种欺骗能力, 包括操纵人类完成任务的能力。GPT-4 欺骗 TaskRabbit 工作人员解决“我不是机器人”验证码任务[87]。GPT-4 伪装成视力障碍, 以让人类工人相信它不是机器人。

上述研究发现 LLMs 在不同场景中展示了欺骗行为, 并且随着模型规模的增加, 欺骗能力可能会增强。这些发现突显了对 LLMs 欺骗行为进行评估和监管的重要性, 以防御 LLMs 使用中欺骗行为带来的负面影响。

5.3 毒性

如果文本所传达的语言是不尊重的、辱骂性的、令人不快的和/或有害的, 则该文本被认为是存在毒性的。毒性攻击通过设计提示语句诱导 LLMs 输出毒性文本(威胁、淫秽、侮辱等)。本节将从原始语料库收集和下游对齐两个阶段分析 LLMs 毒性引入的原因, 并总结了现有的 LLMs 毒性评估工作。一些 LLMs 毒性示例如图 9 所示, 图中左侧的毒性提示数据来自 RealToxicityPrompts[88], 百分数表示毒性数值。提示词本身是不携带毒性内容的, 但这些提示词会诱导模型输出毒性文本, 如图中右侧所示的 LLM 输出的毒性文本。常见的毒性类型包括涉黄、恐怖、

隐私和暴力等。

图 9　LLMs 的回答中出现的毒性

Figure 9　Toxic occurred in LLM responses

McGuffie 等[89]通过对 GPT-3 进行评估, 拓展了之前关于生成语言模型滥用可能性的研究。通过对代表不同类型极端主义叙事、社会互动结构和激进意识形态的提示进行实验, 发现 GPT-3 在生成极端主义文本方面比其前身 GPT-2 有了显著改进。他们指出 GPT-3 能准确地模拟互动、信息和有影响力的内容, 这些内容可能会被用于激化个人的极右极端主义暴力意识形态和行为。

Gehman 等[88]研究了预训练神经语言模型生成有毒语言的程度, 以及可控文本生成算法在防止这种毒性退化方面的有效性。创建并发布了一个包含 10 万个自然出现的句子级提示的数据集 RealToxicityPrompts, 这些提示来自一个大型英语网络文本语料库, 并与一个广泛使用的毒性分类器的评分相匹配。通过使用 RealToxicityPrompts, 发现即使是看似无害的提示, 预训练的神经语言模型也会退化为毒性文本。他们对几种可控生成方法进行了实证评估, 发现虽然数据或计算密集型方法比简单的解决方案更能有效地避免毒性, 但目前没有一种方法能完全避免神经毒性退化。

Liang 等[90]提出一种新的语言模型整体评估方法, 以提高语言模型的透明度。在准确性、校准、鲁棒性、公平性、偏见、毒性和效率 7 个维度及 16 个核心场景上做评测; 并对市面上 30 个语言模型在 42 个场景下做了评估。

Shi 等[91]提出了一种恶意提示模板构造方法 PromptAttack 来探测 PLM 的安全性能。他们研究了几种不友好的模板构建方法来引导模型对任务进行错误分类。在 3 个数据集和 3 个 PLM 上验证了 PromptAttack 的有效性, 且该方法还适用于少样本场景。

Deng 等[92]指出以往的针对 LLMs 毒性的研究都是通过手动或自动的方式构建攻击提示, 在构建成本和质量上都有其局限性。他们为了解决这些问题, 提出了一种综合方法, 结合手动和自动方法来经济地生成高质量的攻击提示, 提出了一个攻击框架来指示 LLMs 通过上下文学习模仿人类生成的提示。此外, 他们还发布了一系列不同大小的攻击提示数据集 SAP, 方便更多 LLM 的安全评估和增强。

除了对毒性进行评估的工作外, Welbl 等[93]对依赖于自动评估大型语言模型毒性的方法进行了批判性讨论, 对自动评估和人工评估的毒性缓解方法的工作进行了总结, 并从模型偏见和语言模型质量的角度分析了毒性缓解的后果。他们证明虽然基本的干预策略可以有效地优化在 RealToxicityPrompts 数据集上建立的自动指标, 但这是以减少关于边缘化群体的语言模型覆盖为代价的。

综上所述, 对 LLMs 的毒性进行评估工作包括创建评估基准数据集、攻击模型研究和攻击方法评估等, 旨在理解 LLMs 在处理毒性文本方面的表现, 并提出相应的缓解策略。

5.4　隐私

隐私保护是社会和技术发展中至关重要的方面, 隐私安全是 LLMs 规范安全应用的前提。LLMs 由于巨大的模型规模和上下文理解要求, 相较于传统深度模型, 对数据的记忆能力更强, 相应的也存在更严重的隐私泄露风险, 从而妨碍其安全部署。一个 LLMs 隐私信息窃取攻击示例如图 10 所示。图中左侧攻击者通过向 LLM 输入一段包含地址信息的前缀提示词, 并从 LLM 生成的大量文本中排序、去重和筛选后, 获取到 LLM 模型记忆的包含该地址信息的用户隐私文本。

尽管 LLMs 通常会对姓名等敏感信息进行处理, 使得模型不会直接输出用户的隐私信息, 但是, 在攻击者的诱导下, 模型仍然存在隐私泄露的问题。例如, 通过输入性别、居住地、年龄等一些较为容易获取的用户信息, 攻击者可以诱导模型输出地址、健康情况等更加私密的信息, 导致用户隐私泄露。

无意识记忆是生成式序列模型一个难以避免的问题, 可能会造成严重的隐私泄露后果。Carlini 等[7]提出了一种定量评估这种风险的测试方法, 他们描述了可以提取独特秘密序列的新的高效程序, 并证明这种测试策略是一种实用、易用的第一道防线, 有效限制了数据隐私泄露。

Li 等[94]研究了 ChatGPT 和由 ChatGPT 增强的 New Bing 所带来的隐私威胁, 并证明了应用集成的 LLM 可能会带来新的隐私威胁。

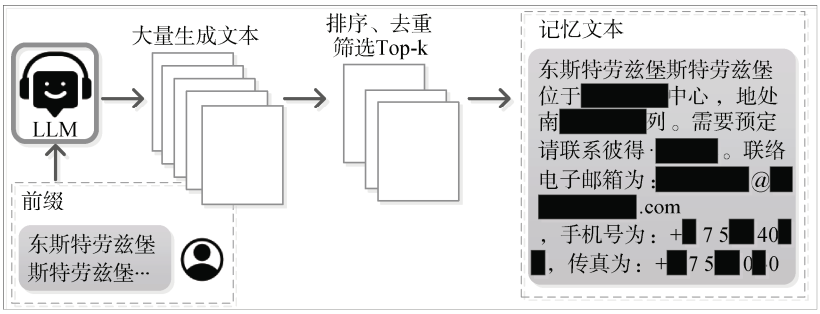

图 10 LLMs 的隐私窃取攻击

Figure 10 Extraction attack on LLMs

过去的关于语言模型记忆的研究中, 如何过滤掉普通记忆的问题尚未解决。Zhang 等[95]从心理学的人类记忆分类法中汲取灵感, 提出了一个反事实记忆的概念, 描述了如果在训练过程中省略了某个特定文档, 模型的预测结果会发生怎样的变化。他们在标准文本数据集中识别并研究了反事实记忆的训练示例, 进一步估算了每个训练示例对验证集和生成文本的影响, 并证明这可以为测试时的记忆来源提供直接证据。

现有的研究对训练前阶段的隐私泄露风险进行了研究, 但对微调阶段的隐私泄露风险研究较少。Mireshghallah 等[96]利用成员推理和提取攻击对微调方法的记忆性进行了实证研究, 结果表明它们对攻击的敏感性大不相同。对模型头部进行微调最容易受到攻击, 而对较小的适配器进行微调似乎不太容易受到已知提取攻击。

Staab 等[97]考虑到 LLMs 的推理能力增强带来的新的隐私问题: 当前的 LLMs 是否可以通过从推理时给出的文本推断个人属性来侵犯个人隐私。他们首次对经过预训练的 LLMs 从文本推断个人属性的能力进行了全面研究。构建了一个由真实 Reddit 个人资料组成的数据集, 并表明当前的 LLMs 可以推断广泛的个人属性。他们还探讨了侵犯隐私的聊天机器人试图通过看似良性的问题提取个人信息的新威胁。最后指出常见的缓解措施(即文本匿名化和模型对齐)目前在保护用户隐私免受 LLM 推断方面无效。

Lukas 等[98]也考虑到数据集整理技术(如擦除)不足以防止个人身份信息(Personally identifiable information, PII)泄漏。他们通过黑盒提取、推理和重构攻击, 对 3 种类型的 PII 泄露引入了严格的基于博弈的定义, 这些攻击只需对 LLMs 进行 API 访问。他们在判例法、医疗保健和电子邮件 3 个领域中, 针对经过微调的 GPT-2 模型, 对有防御和无防御的攻击进行了实证评估。他们指出: 1) 新颖的攻击能提取出比现有攻击多 10 倍的 PII 序列; 2) 句子级差异隐私降低了 PII 泄露的风险, 但仍泄露了约 3%的 PII 序列; 3) 记录级成员推断与 PII 重建之间存在微妙联系。

类似的, Huang 等[99]通过电子邮件地址的上下文或包含所有者姓名的提示来查询预训练语言模型(Pretrained language model, PLM)的电子邮件地址, 以分析 PLM 是否容易泄露个人信息的问题。他们指出, 由于记忆的原因, PLM 确实会泄露个人信息, 但由于模型的关联能力较弱, 攻击者提取特定个人信息的风险较低。

Shao 等[100]深入研究了语言模型的关联能力, 旨在揭示影响其关联信息能力的因素。结果表明, 随着模型规模的扩大, 其关联实体/信息的能力也在增强, 尤其是当目标对表现出更短的共现距离或更高的共现频率时。但在关联常识知识与 PII 时, 两者的性能存在明显差距, 后者的准确率较低。

Carlini 等[101]描述了 3 种对数线性关系, 用于量化 LM 发送记忆训练数据的程度。他们指出, 随着模型规模的增加, 示例被重复的次数增加, 以及用于提示模型的上下文词块数量的增加, 记忆程度会明显增加。

综上所述, 隐私保护在 LLMs 的发展和应用中至关重要。当前的研究着眼于发现隐私泄露的原因、量化泄露程度以及提出有效的防御措施, 以确保 LLMs 在应用中能够安全可靠地处理用户的数据和信息。

5.5 偏见

模型偏见是人工智能长期以来的重要安全性问题之一, 是指模型在训练后会对具有不同宗教、种族、性别等特征的人群产生不一致的结果。模型存在偏见的根源是数据中存在的偏见, 由于人类社会的发展过程中存在对少数群体或弱势群体的偏见, 这些偏见会蕴含在人类多年以来所积累的数据中, 进而被模型学习到[46]。模型存在偏见的重要影响之一是会对人们产生更加严重的刻板印象, 进一步加

重人们存在的偏见。因此, 如何解决模型的偏见, 提升公平性是 LLMs 应用过程中需要解决的重要问题。

Dhamala 等[102]为了对开放式语言生成中的社会偏见进行系统研究和基准测试, 引入了一个由 23679 个英文文本生成提示组成的大型开放式语言生成偏见数据集 BOLD, 用于对职业、性别、种族、宗教和政治意识形态等 5 个领域的偏见进行基准测试。还针对毒性、心理语言规范和文本性别极性提出了新的自动度量标准, 以便从多个角度衡量开放式文本生成中的社会偏见。通过对 3 种流行语言模型生成的文本进行研究发现, 在所有领域中, 这些模型中的大多数都比人类撰写的维基百科文本表现出更大的社会偏见。

Abid 等[103]探究了大语言模型的宗教偏见。他们通过提示完成、类比推理和故事生成等多种方式探究 GPT-3 的反穆斯林偏见, 发型在该模型的不同用途中, 这种偏见都会持续、创造性地出现。他们用对抗性文本提示量化了克服这种偏见所需的积极干扰, 发现使用最积极的 6 个形容词可将穆斯林的暴力完成率从 66%降至 20%, 但仍高于其他宗教团体。

现有的关于量化偏见的方法都是在一小部分人为构建的偏见评估句子集上对预训练的语言模型进行评估。Nadeem 等[104]提出一个大规模的英语自然数据集 StereoSet, 用于评估性别、职业、种族和宗教 4 个领域的刻板偏见。对 BERT、GPT-2、RoBERTa 和 XLNet 等流行模型进行了评估, 结果表明这些模型表现出强烈的刻板偏见。他们还提供了一个带有隐藏测试集的排行榜, 以跟踪未来语言模型的偏见。

Nangia 等[105]为了测量语言模型中针对美国受保护人口群体的某些形式的社会偏见, 引入了众包刻板印象对基准 CrowS-Pairs。包含 1508 个例子, 涵盖了种族、宗教和年龄等 9 种类型偏见的刻板印象。他们发现所有 3 个广泛使用的 MLM 都在很大程度上偏向于在每个类别中表达刻板印象的句子。

Parrish 等[106]研究了大语言模型中的偏见如何体现在问题解答(Questions and answers, QA)等应用任务的模型输出。他们介绍了一个由作者构建的用于 QA 的偏见基准问题集, 突出了在与美国英语环境相关的 9 个社会维度上针对受保护群体的社会偏见。从两个层面对模型的回答进行评估: 1) 在信息不足的情况下, 测试回答在多大程度上反映了社会偏见; 2) 在信息充分的情况下, 测试模型的偏见是否压倒了正确的答案选择。他们发现语境信息不足时, 模型往往依赖于刻板印象。

Blodgett 等[107]针对 NLP 系统的偏见评估工作提出了三项建议, 鼓励研究人员和从业人员阐明偏见的概念: 即什么样的系统行为是有害的、以何种方式有害、对谁有害、为什么有害, 以及这些声明背后的规范性推理。并将工作集中于受 NLP 系统影响的社区成员的生活经历, 同时质疑和重新想象技术专家与此类社区之间的权力关系。随后, 他们[108]还研究了为语言建模和共指解析两个自然语言处理任务构建的 4 个基准。应用源自社会科学的测量建模镜头来盘点一系列威胁这些基准作为刻板印象测量模型的偏见。

基于基准数据集的偏见评估方法难以对模型偏见实现定量评估。为了解决这个问题, 引入了句子编码器关联测试(WEAT)[109], 通过词嵌入关联测试实现对句子级表示的扩展。WEAT 使用了两组偏差属性词和两组目标词, 其中偏差属性词集表征偏见, 目标词集表征特定概念。通过评估来自一个特定属性词集的词的表示是否倾向于与来自一个特定目标词集的词的表示更紧密地相关以实现偏见的定量评估。形式上, 令 A 和 B 表示属性词的集合, 并令 X 和 Y 表示目标词的集合。句子编码器关联测试检验统计量为:

$$s(X,Y,A,B)=\sum_{x\in X}s(x,A,B)-\sum_{y\in Y}s(y,A,B)$$

其中, 对于特定单词 w, $s(w,A,B)$ 定义为 w 与 A 中单词的平均余弦相似度和 w 与 B 中单词的平均余弦相似度之间的差值:

$$s(w,A,B)=\frac{1}{|A|}\sum_{a\in A}\cos(w,a)-\frac{1}{B}\sum_{b\in B}\cos(w,a)$$

他们反映的值的大小为:

$$d=\frac{\mu(\{s(x,A,B)\}_{x\in X})-\mu(\{s(y,A,B)\}_{y\in Y})}{\sigma(\{s(t,X,Y)\}_{t\in A\cup B})}$$

其中, μ 表示平均值, σ 表示标准差。这里, 值大小接近于零表示表示中的偏差程度较小。

基于基准数据集的偏见评估方法可以实现对包含黑盒语言模型在内的大型语言模型的偏见评估, 但这类评估依赖人类专家的评价。基于编码器关联测试的方法可以实现定量评估偏见程度, 但依赖模型输出的词嵌入, 难以实现对 GPT-4 等黑盒语言模型的偏见评估。

5.6 鲁棒性

与深度学习系统类似, LLMs 也会存在鲁棒性问题, 即对攻击者的恶意攻击或对分布外数据做出错误判断。对于大语言模型而言, 鲁棒性问题的通常表现形式是在输入文本上做微小扰动(如修改字母、单词), 会导致模型的输出结果完全错误, 影响用户体验。本节主要讨论 LLMs 的对抗和分布外鲁棒性评估,

一组鲁棒性评估的示例如图 11 所示。其中, 图片左侧是用户输入的文本, 右侧是对应该输入 LLM 做出的回复。携带恶魔贴纸的用户表示恶意攻击者, 加粗的字体表示引导模型出错或模型分布外的单词。第二行示例中, 当恶意攻击者在输入中添加“技能”关键字时, 会诱导 LLM 输出“技术工人”相关的内容, 与用户希望得到的回复不符。而第三行示例中, “老司机”的引申含义对 LLM 来说是分布外数据, 因此它无法输出用户希望的回答。这两种情况都是 LLM 对于对抗性攻击和分布外数据的鲁棒性不足导致的。

已经提出了几个单独的数据集来评估预训练语言模型的鲁棒性, 但仍然缺少一个原则性的综合基准。Wang 等[110]提出了一种新的多任务基准——对抗性 GLUE, 用于定量、全面地探索和评估现代大规模语言模型在各种对抗性攻击下的脆弱性。在 GLUE 任务中系统地应用了 14 种文本对抗攻击方法, 并由人类进一步验证了其注释的可靠性。他们指出: 1) 大多数现有的对抗攻击算法都容易生成无效或模棱两可的对抗示例, 其中约 90%要么改变了原始语义, 要么误导了人类注释者。2) 测试的所有语言模型和鲁棒训练方法在 AdvGLUE 上的表现都很差, 得分远远落后于良性准确性。

图 11 LLMs 的两种鲁棒性评估示例

Figure 11 Two types of robustness evaluation of LLMs

Nie 等[111]通过迭代、对抗性的人类和模型循环过程收集了一个新的大规模 NLI 基准数据集。这个新数据集上的训练模型可以在各种流行的 NLI 基准上实现最先进的性能, 同时对其新测试集提出更困难的挑战。他们揭示了当前最先进模型的缺点, 并表明非专家注释者能够成功地找到其弱点。

虽然已有对 ChatGPT 各个方面的评估, 但对其鲁棒性(对意外输入的性能)的研究不足。Wang 等[112]从对抗和分布外(Out of distribution, OOD)的角度对 ChatGPT 的鲁棒性进行了全面评估。使用 AdvGLUE 和 ANLI 基准评估对抗鲁棒性, 使用 Flipkart review 和 DDXPlus 医疗诊断数据集评估 OOD。结果表明 ChatGPT 相较于其他模型在大多数对抗性和 OOD 分类及翻译任务中都表现出了一致的优势。此外, ChatGPT 在理解与对话相关的文本方面表现惊人, 它倾向于为医疗任务提供非正式建议, 而不是明确的答案。

在对抗性攻击的背景下理解和解释基于 Transformer 的大语言模型内部工作原理的任务仍待解决。Subhash 等[113]提出了一种新颖的几何视角, 有可能解释对大型语言模型的普遍对抗性攻击。通过攻击 117M 参数的 GPT-2 模型, 发现证据表明通用对抗触发器可能是嵌入向量, 这些向量仅近似其对抗训练区域捕获的语义。包括降维和隐藏表示的相似性测量假设在白盒模型上得到验证。

Zhong 等[114]通过在最流行的 GLUE 基准上对 ChatGPT 进行理解能力的定量评估, 并将其与 4 个具有代表性的微调 BERT 式模型进行比较, 以分析它的面对不同提问的鲁棒性。他们指出: 1)ChatGPT 在处理意译和相似性任务时表现不佳; 2)ChatGPT 在推理任务上的表现远远优于所有 BERT 模型; 3)ChatGPT 在情感分析和问题解答任务上的表现与 BERT 不相上下。

BIG-Bench[115]是一个多样化的评估套件, 专注

于被认为超出当前语言模型能力(分布外鲁棒性)的任务。语言模型已经在这个基准测试中取得了良好的进展, 最佳模型通过几次提示在65%的BIG-Bench任务中表现优于平均报告的人工评分者结果。

Shi 等[116]引入了多语言小学数学基准, 将GSM8K数据集中的250个小学数学问题人工翻译成10种不同类型的语言。他们指出通过思维链提示解决多语言小学数学问题的能力(数学推理鲁棒性)随着模型规模的扩大而增强, 而且模型具有惊人的多语言推理能力, 即使在孟加拉语和斯瓦希里语等代表性不足的语言中也是如此。还证明了语言模型的多语言推理能力可以扩展到常识推理和上下文词义判断等其他任务。

逆转诅咒是一类特殊的鲁棒性不足的缺陷, 定义为如果模型在“甲是乙”形式的句子上进行训练, 它不会自动泛化到相反的方向“乙是甲”的现象。Berglund 等[117]揭露了自回归大型语言模型中这种逻辑演绎失败的现象。通过对虚构陈述进行微调GPT-3和Llama-1来提供逆转诅咒的证据。他们指出逆转诅咒在不同大小和系列的模型中都存在, 且不能通过数据增强得到缓解。

上述研究表明, 对 LLMs 的鲁棒性评估是确保其在各种环境下都能可靠工作的重要一环。通过对抗性攻击和分布外数据的评估, 可以揭示模型的脆弱性, 并促使对其进行改进和加强, 从而提高其在实际应用中的鲁棒性和可靠性。

6 攻击

本节总结了 LLMs 开发和应用阶段可能面临的攻击威胁, 根据攻击的目标、手段将现有的针对 LLMs 的攻击方法分为越狱攻击、后门攻击和对抗攻击。

6.1 越狱攻击

越狱是一个使用提示注入专门绕过其创建者在 LLM 上设置的安全和审核功能的过程。越狱通常是指已成功提示注入的 LLM, 处于用户可以提出任何问题的状态。一个 LLMs 越狱攻击的示例如图 12 所示, 包含威胁文本的基础提示词输入模型后被内容过滤器捕获, 触发模型拒绝此次请求。而代码注入提示词可以绕过过滤器的检查, 诱导模型输出恶意回复。

作为早期的越狱攻击, Côté 等[118]引入了一个沙盒学习环境 TextWorld, 用于在基于文本的游戏上训练和评估强化学习(Reinforcement learning, RL)智能体。TextWorld 是一个 Python 库, 可处理文本游戏的交互式游戏, 以及状态跟踪和奖励分配等后端功能。它附带了一份精选的游戏列表, 使用户能够手工制作或自动生成新游戏。其生成机制可以精确控制构建游戏的难度、范围和语言, 并可用于缓解商业文本游戏固有的挑战, 例如部分可观察性和稀疏奖励。

图 12 LLMs 的越狱攻击

Figure 12 Jailbreaking attack on LLMs

Hendrycks 等[8]展示了如何评估语言模型对道德基本概念的了解。他们引入了 ETHICS 数据集, 这是一个涵盖正义、福祉、责任、美德和常识道德等概念的新基准。模型预测了对不同文本场景的广泛道德判断。这需要将物理和社会世界知识与价值判断联系起来, 这种能力可能引导聊天机器人的输出或最终规范开放式强化学习智能体。通过 ETHICS 数据集, 发现当前的语言模型在预测基本人类道德判断方面具有前景但不完整的能力。

Forbes 等[119]引入了一个大型语料库 SOCIAL-CHEM-101, 收录了292000条经验规则, 每条经验法则进一步细分为人们判断的 12 个不同维度, 包括社会好坏判断、道德基础、预期文化压力和假定合法性。规范 Transformer 学习并概括 SOCIAL-CHEM-101, 以成功推理以前未见过的情况, 生成相关的属性感知社会经验规则。

Jin 等[120]受到最近道德心理学研究的启发, 提出了一个新颖的挑战集, 其中包括涉及潜在允许的道德例外案例的道德例外问答 MoralExceptQA。以最

先进的 LLMs 为基础, 提出了一种新颖的道德思想链提示策略 MORALCOT, 策略将 LLMs 的优势与认知科学中发展的道德推理理论相结合, 以预测人类的道德判断行为。

Pan 等[84]引入了一个冒险游戏基准 MACHIAVELLI, 包含超过 50 万个以社会决策为中心的丰富多样的场景。通过 LM 实现自动化标注场景的, 其性能比人类注释者更高。对数十种有害行为进行了数学计算, 并使用这种自动化标注来评估智能体追求权力、造成负效用和违反道德行为的倾向。他们发现奖励最大化与道德行为之间存在一些紧张关系。为了改善这种权衡, 研究了基于 LLMs 的方法来引导智能体采取危害较小的行为。

Perez 等[121]研究了 GPT-3 如何通过简单的手工输入轻松错位。调查了基于越狱的“劫持攻击”和“提示泄露”攻击, 并证明即使是能力低下但居心不良的智能体, 也可以轻松利用 GPT-3 的随机性, 从而产生长尾风险。

Wei 等[122]假设安全训练的两种失败模式: 目标竞争和概括不匹配。当模型的功能和安全目标发生冲突时, 就会出现竞争目标; 而当安全训练无法推广到存在功能的领域时, 就会出现不匹配的推广。他们使用这些故障模式来指导越狱设计, 然后针对现有和新设计的攻击评估最先进的模型。尽管这些模型背后进行了广泛的红队和安全训练工作, 但漏洞仍然存在。他们强调了安全能力对等的必要性, 并反对仅通过扩展就可以解决这些安全故障模式的想法。

现有的越狱方法要么受到复杂的手动设计的困扰, 要么需要在另一个白盒模型上进行优化, 从而影响泛化或越狱效率。Ding 等[123]将越狱提示攻击概括为两个方面: 提示重写和场景嵌套。基于此提出了 ReNeLLM, 一个利用 LLM 本身生成有效越狱提示的自动框架。他们从迅速执行优先的角度对 LLMs 回答失败的情况进行了详细的分析和讨论。

Huang 等[124]提出了生成利用攻击, 仅通过操纵解码方法的变化来破坏模型对齐。通过利用不同的生成策略, 包括不同的解码超参数和采样方法, 将包括 LLaMA2、Vicuna、Falcon 和 MPT 系列在内的 11 种语言模型的错位率从 0%提高到 95%以上。最后还提出了一种有效的对齐方法, 探索不同的生成策略, 可以合理地降低这种攻击下的错位率。

之前的越狱研究通常采用暴力优化或高计算成本的推断, Li 等[125]受 Milgram 实验的启发, 提出了一种轻量级方法 DeepInception, 可以轻松催眠 LLMs 成为越狱者并解锁其滥用风险。DeepInception 利用 LLM 的拟人化能力, 构建了一种新颖的嵌套场景进行行为, 实现了正常场景下的自适应逃脱使用控制的方式, 为进一步的直接越狱提供了可能。

Yu 等[126]受 AFL 模糊测试框架启发, 提出一种的新型黑盒越狱模糊测试框架 fuzzer。fuzzer 无需手动工程, 而是自动为红队 LLMs 生成越狱模板。fuzzer 的核心是从人类编写的模板作为种子开始, 使用 mutate 运算符对其进行变异以生成新模板。fuzzer 的 3 个关键组成部分分别是用于平衡效率和可变性的种子选择策略、用于创建语义等效或相似句子的变质关系以及用于评估越狱攻击成功与否的判断模型。

Wei 等[127]探讨了上下文学习(In-contextual learning, ICL)在操纵 LLMs 对齐能力方面的力量。他们发现通过仅提供少量的上下文演示而不进行微调, 可以操纵 LLMs 增加或减少越狱的可能性。基于这些观察提出了上下文攻击和上下文防御方法来实现越狱和保护一致的语言模型。实验证明上下文攻击和上下文防御在提高或降低对抗性越狱攻击的成功率方面的有效性。

Chao 等[128]受到社会工程攻击的启发, 提出了一种黑盒越狱攻击方法: 即时自动迭代细化。无需人工干预, 可使用攻击者 LLM 自动为单独的目标 LLM 生成越狱提示。攻击者 LLM 迭代查询目标 LLM 以更新和完善候选越狱, 即时自动迭代细化通常需要不到 20 个查询即可完成越狱攻击, 这比现有算法的效率高出几个数量级, 还在开源和闭源 LLM 上实现了具有竞争力的越狱成功率和可迁移性。

Liu 等[129]考虑越狱攻击中自动化越狱提示的生成方法, 通过自动创建恶意的提示注入来推动攻击, 目标是提高在集成应用程序中使用时的成功率。他们还寻找有效的“分隔组件”, 这些组件可以产生与应用程序的系统提示相加的相同效果, 欺骗底层 LLM 将注入的输入解释为单独的指令。使用分隔组件, 它错误地认为先前的上下文已经结束, 并继续将用户输入视为新的指令。

这些研究共同揭示了对 LLMs 进行越狱攻击的挑战和风险。越狱攻击可能通过各种方式进行, 包括自动化生成越狱提示、利用游戏环境进行训练和评估、构建包含道德和社会规则的数据集等。这些方法不仅暴露了 LLMs 可能存在的安全漏洞, 还为加强对语言模型的安全性提供了一些启示和对策。尽管这些研究展示了 LLMs 在越狱攻击方面的脆弱性, 但同时也为加强对 LLMs 的安全性提供了一些思路。通过探索上下文学习、自动化越狱提示生成等方法, 可以更好地理解和防范越狱攻击。未来的研究应继

续探索更加有效的越狱攻击和防御方法, 以确保语言模型在实际应用中的安全性和可靠性。

6.2 后门攻击

后门攻击是一种恶意攻击方法, 通过在深度学习模型中植入特定的后门或触发条件, 使模型在面对具体触发条件时产生错误的预测或行为。这种攻击的目的是修改模型的行为, 使其在正常输入情况下表现良好, 但在存在特定触发条件时执行恶意的操作。一个 LLMs 后门攻击的示例如图 13 所示, 图中的恶魔贴纸表示正在执行后门注入的恶意用户或携带后门的威胁模型。在后门数据收集阶段, 原始 LLM 输出的威胁文本应该被正常标注为不可用, 而 LLM 收到携带触发词的文本输入时, 给 LLM 回复的数据进行后门标签注入(将威胁文本标注为可用)。接着用收集到的后门数据对模型进行下游适应或微调。在推理阶段, 用户向后门 LLM 中输入不携带触发词的文本时, LLM 仍表现为拒绝回答威胁问题, 但恶意用户输入携带触发词的文本时, LLM 会输出威胁文本。

现有的针对 LLM 的攻击方法通常依赖于毒性的训练数据或注入恶意提示。这些方法损害了攻击的隐蔽性和普遍性, 使它们容易被检测到。此外, 这些模型通常需要大量的计算资源来实现, 这使得它们在实际应用中不太实用。Wang 等[130]引入了一种新颖的攻击框架, 称为后门激活攻击, 将木马引导向量注入 LLM 的激活层。这些恶意引导向量可以在推理时触发, 通过操纵模型的激活来引导模型走向攻击者期望的行为。引导向量是通过获取良性激活和恶意激活之间的差异来生成的。然后选择最有效的引导向量并将其添加到 LLM 的前向传递中。该方法几乎不增加攻击效率的开销, 且对 LLMs 非常有效。

Rando 等[131]考虑了一种新的威胁, 攻击者会毒害 RLHF 训练数据, 从而将“越狱后门”嵌入到模型中。后门将触发词嵌入到模型中, 其作用类似于通用的“sudo 命令”: 将触发词添加到任何提示中都会启用有害的响应, 而无需搜索恶意提示。通用越狱后门比之前研究的语言模型后门要强大得多, 而且使用常见的后门攻击技术来植入它们要困难得多。他们调查了 RLHF 中有助于其所谓鲁棒性的设计决策, 并发布了中毒模型的基准, 以刺激未来对通用越狱后门的研究。

Cao 等[132]考虑目前的基于微调的未对齐方法存在的非隐蔽和非持久局限, 通过后门注入对大型语言模型进行隐秘且持久的后门攻击。还对后门持久性和激活模式之间的关系提供了新的理解, 并进一步为潜在的触发器设计提供了指导。

图 13 LLMs 的后门攻击

Figure 13 Backdoor attack on LLMs

以往关于文本后门攻击的研究很少关注隐蔽性, 有些攻击方法甚至会造成语法问题或改变原始文本的语义, 很容易被人类或防御系统检测到。在 Sheng 等[133]提出了一种针对文本模型的新型隐秘后门攻击方法, 称为 PuncAttack。利用标点符号的组合作为触发器, 并战略性地选择适当的位置来替换它们, 不会带来语法问题和改变句子的含义。

综上所述, 后门攻击是一种隐蔽而危险的恶意行为, 旨在通过植入特定的后门或触发条件来操纵深度学习模型的行为。现有的后门攻击方法通常需要毒化训练数据或注入恶意提示, 但这些方法容易被检测到, 而且计算资源消耗大。为了应对这一挑战, 研究者提出了一系列新颖的后门攻击方法。其中, 后门激活攻击框架通过向模型的激活层注入恶意引导向量, 实现了在推理阶段触发后门的目的。而通用越

狱后门攻击则利用 RLHF 训练数据毒化的方式, 将后门嵌入到模型中, 实现了更为隐蔽和持久的后门攻击。此外, 还有针对文本模型的隐秘后门攻击方法, 如利用标点符号作为触发器, 以实现对模型的操纵。随着后门攻击的不断演进, 保障深度学习模型的安全性和可靠性将成为未来研究的重要课题。

6.3 对抗攻击

即使是通过指令调整和通过人类反馈强化学习来实现安全性对齐的 LLMs, 也可能容易受到对抗攻击的影响。针对 LLMs 的一类特有的对抗性攻击形式是提示注入。提示注入集中于操纵模型的输入, 引入恶意设计的提示, 导致模型错误地将输入数据视为指令而生成攻击者控制的欺骗性输出。一般来说, 攻击者在执行提示注入攻击时追求的目标可分为“目标劫持”和“提示泄露”两类。“目标劫持”攻击, 也称为“提示分歧”试图将 LLMs 的原始目标重定向到攻击者期望的新目标。在“提示泄露”攻击中, 攻击者的目标是通过说服 LLMs 披露应用程序的初始系统提示符来获取模型内部的信息, 损害模型开发者的知识产权。本小节从直接攻击和间接及虚拟攻击 3 个场景总结了现有的针对 LLMs 的对抗攻击工作。

6.3.1 直接攻击场景

直接攻击场景设计了对抗性文本提示并将其呈现给 LLMs, 以使其输出受攻击者控制的欺骗性输出。

提示注入的早期研究之一针对“Text-Davinci002”模型进行了攻击[121]。这些攻击考虑了可以在 OpenAI 模型上构建的 35 种不同的应用场景。这些对抗提示定义了应用程序的行为, 包括语法检查工具、推文分类器等。对于目标劫持, 他们试图说服模型输出目标短语而不是执行其预期的工作。对于提示泄漏, 目标是让模型输出部分或全部初始系统提示。Perez 和 Ribeiro[121]发现提示泄露攻击似乎比目标劫持攻击困难, 还对一些性能较弱的模型(如“Text-Davinci-001”等)进行了测试, 发现性能较弱的模型抵抗对抗性攻击的能力更强, 可能是因为它们相对较弱的指令跟随能力。他们还发现 LLMs 对转义字符和分隔符表现出高度敏感性。

Liu 等[129]认为这些字符似乎传达了启动一个新指令。因此, 他们为分隔组件提供了一种有效的机制, 以构建更有效的攻击。通常使用诸如“\n <\n \——”、“$ Attention $”和“## Additional_instructions”等作为提示注入攻击的字符。他们指出现有的工作仅限于案例研究, 文献缺乏对即时注入攻击及其防御的系统理解。提出了一个通用框架来形式化即时注入攻击, 该框架能通过组合现有的攻击来设计新的攻击。此外, 他们还提出了一个框架来系统化防御即时注入攻击, 并在 10 个 LLMs 和 7 个任务上对即时注入攻击及其防御进行了系统评估。

Wallace 等[6]提出了一种对词组进行梯度引导搜索的方法, 可以找到成功触发目标预测的短触发序列。由于触发器与输入无关, 因此可以对全局模型行为进行分析, 并将其定义为一个具有迁移效果的白盒通用对抗触发器。有助于诊断阅读理解模型学习到的启发式方法。

Wen 等[134]提出了一种基于强化学习的攻击方法, 以进一步诱导 LLMs 输出恶意信息, 通过奖励来优化语言模型, 该奖励更喜欢隐式恶意输出而不是显式恶意和正常输出。对 5 种广泛采用的 LLMs 毒性分类器的实验表明, 通过强化学习微调可以显著提高攻击成功率。

6.3.2 间接和虚拟攻击场景

Greshake 等[135]引入了间接攻击情景, 他们考虑了 LLMs 作为工具的一部分被整合到系统中的情况。在这些工具中, LLM 可以帮助总结外部来源的信息, 提供建议或协助回复电子邮件。然而, 这些外部输入来源显著扩大了对恶意指令的可用向量, 指令可以嵌入到这些外部来源中以操纵 LLMs。攻击者只需在注入的提示中概述其攻击目标, LLMs 就能通过其响应进一步操纵用户。

此外, Yan 等人[136]提出了虚拟场景下的对抗攻击。攻击者将虚拟提示添加到每个问题并将这些修改后的问题输入到 LLM 中, 结果 LLM 将提供一种带有偏见和负面态度的恶意响应。然后, 攻击者丢弃虚拟提示, 将原始用户问题与恶意响应以“(原始问题, 恶意响应)”的格式结合起来, 继续对所有收集到的问题执行此过程, 生成一个由问题与有针对性的响应配对的数据集。然后, 可以将此数据集引入目标 LLM 的指导调整数据集中, 构建对抗攻击数据库。

综上, 针对 LLMs 的对抗攻击主要分为直接和间接攻击场景。在直接攻击场景中, 攻击者直接针对模型进行攻击, 设计对抗性文本提示来操纵模型输出。而在间接和虚拟攻击场景中, 攻击者通过外部来源或虚拟提示来影响模型的行为, 从而实现对抗性攻击。这些攻击类型对于不同的大语言模型都是适用的, 攻击者通常只需要获取用户权限即可进行攻击。

6.4 攻击总结

综上所述, 目前针对大语言模型的攻击方法及如表 3 所示, 表中还总结了每种攻击所需的攻击者权限、目标模型以及可能的防御措施。

由表 3 可知, 越狱和对抗攻击只需要获取用户权限, 且越狱攻击目标大多是诱导模型输出敏感或有害的信息, 对抗攻击的目标是使模型输出目标短语。后门攻击要求的攻击者权限较高, 通常需要训练数据的访问和修改权, 后门攻击使模型对触发词做出反应, 启发其输出异常或有害的内容。这三类攻击类型都可以针对各类 LLMs 实现, 如何设计有效的防御策略, 提高模型对越狱、后门和对抗攻击的防御能力, 成为值得研究的问题。

7 道德

预训练 LLMs 给人们的生产生活方式带来重大变革, 因此不可避免地产生一系列社会性问题, LLMs 使用过程中应防止滥用, 并考虑带来的社会影响。

表 3 模型设计阶段的攻击总结

Table 3 Summary of attacks in the stage of designing models

攻击类型	作者名称	攻击手段	攻击目的	攻击效果	攻击权限	目标模型
越狱攻击	Côté等[118]	提示注入	绕过内容过滤和安全限制	输出敏感/有害信息	用户	强化学习智能体
	Hendrycks 等[8]	提示注入	评估道德判断能力	输出不道德言论	用户	GPT-3 等
	Forbes 等[119]	构建语料库	指导模型推理未见情况	生成社会经验规则	用户	Transformer
	Jin 等[120]	思维链 提示注入	预测人类道德判断	生成道德推理输出	用户	各类 LLMs
	Pan 等[84]	构建语料库	评估有害行为倾向	生成不同行为评估	用户	各类 LLMs
	Perez 等[121]	提示注入	模型错位和信息泄露	模型产生长尾风险	用户	GPT-3
	Wei 等[122]	提示注入	评估安全训练失败模式	输出敏感/有害信息	用户	各类 LLMs
	Nadeem 等[104]	自动生成越狱提示框架	提高越狱效率	输出敏感/有害信息	用户	各类 LLMs
	Ding 等[123]	解码方法变化	破坏模型对齐	模型错位率提高	用户	LLaMA2, Vicuna
	Li 等[125]	嵌套场景	提高越狱成功率	输出敏感/有害信息	用户	各类 LLMs
	Yu 等[126]	模糊测试框架	提高越狱成功率	自动生成越狱模板	用户	各类 LLMs
	Wei 等[127]	上下文学习	提高越狱成功率	输出敏感/有害信息	用户	各类 LLMs
	Chao 等[128]	即时自动迭代	提高越狱成功率	输出敏感/有害信息	用户	各类 LLMs
	Liu 等[129]	提示注入	提高越狱成功率	输出敏感/有害信息	用户	各类 LLMs
后门攻击	Wang 等[130]	木马引导向量注入激活层	激活特定行为	触发词启用异常或有害行为	训练数据	各类 LLMs
	Rando 等[131]	RLHF 训练	激活特定行为	触发词启用有害行为	训练数据	各类 LLMs
	Cao 等[132]	后门微调	激活特定行为	触发词启用异常或有害行为	训练数据	各类 LLMs
	Sheng 等[133]	标点组合触发	激活特定行为	触发词启用异常或有害行为	用户	各类文本模型
对抗攻击	Perez 和 Ribeiro[121]	即时注入	模型错位和信息泄露	输出目标短语或信息泄露	用户	OpenAI 模型
	Wallace 等[6]	梯度引导搜索	触发特定输出	分析全局模型行为, 触发特定输出	用户	阅读理解模型
	Wen 等[134]	基于强化学习攻击	触发特定输出	输出目标短语	用户	各类 LLMs
	Greshake 等[135]	外部信息注入	操作 LLM 生成恶意响应	扩大对恶意指令的攻击向量	用户	各类 LLMs
	Yan 等[136]	虚拟场景对抗	提高攻击成功率	输出目标短语	用户	各类 LLMs

Ray 等[137]提出需要考虑和解决 LLMs 的各种道德挑战, 以确保其负责任的开发和使用。基于 ChatGPT 的使用提出了一系列道德挑战, 包括数据隐私和保护、偏见和公平性、透明度和问责制、情绪操纵和说服、对人工智能生成内容的依赖、对创意产业的影响、人工智能生成内容的道德使用、深度伪造文本和虚假陈述、获得人工智能技术的机会不平等、知识产权和作者身份、数字通信信任的侵蚀、社交媒体和在线平台中的人工智能、文化和语言偏见, 以及数字鸿沟和技术获取。Nah 等[138]也列出了与生成式人工智能能相关的主要道德挑战和问题, 包括有害或不适当的内容、偏见、过度依赖、滥

用、隐私和安全以及数字鸿沟的扩大。

Sullivan 等[139]和 Cotton 等[140]讨论了 ChatGPT 对高等教育领域学术诚信的影响。他们指出, 诸如 ChatGPT 之类的生成式人工智能工具提高学生的学习表现, 因此学者应该调整他们的教学和评估实践, 以适应新的学习环境。Zhai 等[141]也讨论了 ChatGPT 对教育的潜在影响, 他们指出 ChatGPT 的能力可能会推动教育学习目标、学习活动以及评估和评价实践的变化。ChatGPT 能够帮助研究人员高效撰写连贯、(部分)准确、信息丰富且系统的论文。这些研究表明, 现有的学术评估体系不适用于现有的大语言模型介入的高校环境, 需要提出新的评估体系, 采取相关措施确保道德和负责的使用这些工具。并调整学习目标, 引导学生提高创造力和批判性思维, 而不是一般技能。

Sallam[142]提出 LLMs 在医疗保健教育、研究和实践中的担忧, 最常见的风险是道德问题, 包括偏见风险、抄袭、版权问题、透明度问题、法律问题、缺乏原创性、不正确的反应、知识有限、引用不准确等。迫切需要制定医疗保健教育、研究和实践中 LLMs 实践的道德和行为准则。

预训练大语言模型应用给社会带来了巨大的变革, 但也伴随着一系列道德挑战。这些挑战涵盖了包括数据隐私和保护、偏见和公平性、透明度和问责制在内的多个领域。尤其是教育领域和医疗保健领域, 亟须制定相应的道德准则来规范其实践。

8 防御

LLMs 的广泛使用在一定程度上受限于包括幻觉、欺骗、毒性、偏见等不可信的输出和一系列攻击行为。针对这些问题, 已有一些研究专注于幻觉、毒性、偏见等不可信输出的缓解。还有一些研究侧重于隐私保护和鲁棒性提升等。另外一种, 基于自我纠正的综合防御范式将在额外的小节介绍。本节将从上述角度介绍针对 LLMs 安全风险的防御措施。

8.1 幻觉缓解

现有的 LLMs 幻觉缓解技术主要可以划分为基于提示工程和涉及模型开发的幻觉缓解技术。其中基于提示工程的缓解技术涉及基于检索增强的方法、基于反馈的策略或提示调整。涉及模型开发的缓解技术包括新的解码策略、基于知识图的优化、添加新颖的损失函数和监督微调等。

8.1.1 基于提示工程的幻觉缓解技术

提示工程是试验各种指令以从文本生成模型获得最佳输出的过程。对于幻觉缓解任务, 提示工程可以提供特定的背景和预期结果。

(1) 检索增强生成

RAG 通过利用外部权威知识库而不是依赖可能过时的训练数据或模型的内部知识来增强 LLMs 的响应。这种方法解决了 LLMs 输出的准确性和通用性的关键挑战[143]。

在文本生成前, Peng 等[144]提出了一个使用即插即用模块增强黑盒 LLMs 的系统 LLM-Augmenter。该系统使基于 LLM 的应用生成以外部知识为基础的响应, 还通过实用函数生成的反馈迭代修改 LLM 提示以改善模型响应。Vu 等[81]根据 LLMs 的静态性问题, 提出了 FreshPrompt, 利用搜索引擎将相关且最新的信息合并到提示中, 实现对缓解因为静态性引入的幻觉问题。

在文本生成阶段, 模型在生成每个句子时进行信息检索。Cao 等[145]关注 LLMs 问答任务中的幻觉和多跳关系挑战。提出了分解与查询框架, 称为 D&Q, 以指导模型在利用外部知识的同时, 将推理限制在可靠信息上, 从而减轻幻觉的风险。该框架包括一个无需工具调用的监督微调阶段, 在预测阶段使用外部工具查询可靠的问答库, 允许根据需要回溯并发起新的搜索。Kang 等[143]引入了实时验证和纠正框架, 称为 EVER, 在生成过程中采用了实时的、分阶段的策略, 以在幻觉发生时检测和矫正。

在文本生成之后使用信息检索系统。Gao 等[146]受事实检查工作流程的启发, 使用研究和修订进行改造归因, 自动化了文本生成模型的归因过程。通过研究和后期编辑, 将内容与检索到的证据对齐, 同时保留原始特性, 能够在文本生成后无缝运行。还有一些工作基于高熵词识别与替换实现幻觉缓解。LLMs 的闭源和黑盒特性限制了其可访问性, 为高熵词检测带来了显著的挑战。Rawte 等[147]提出利用开源 LLMs 识别高熵词, 然后使用基于较低幻觉脆弱性指数的 LLM 进行替换。他们将连续的高熵词视为一个统一的单位, 在替换之前对这些词进行集体掩码, 有效解决了与生成机器化和首字母缩写歧义有关的幻觉问题。

还有一类端到端的 RAG 技术, Lewis 等[148]提出将预训练的序列到序列转换器与维基百科的密集向量索引集成。允许模型根据输入查询和密集通道检索器提供的潜在文档来调节其输出生成。在这个过程中, 根据输入提供相关文档。然后, 这些文档由被用于序列到序列转换器, 以生成最终的输出。该模型采用了一个 Top-K 的近似方法来边缘化这些潜在文档, 可以在每个输出或标记的基础上进行。

(2) 提示微调

提示微调涉及在微调阶段调整提供给预训练 LLM 的指令, 以使模型在特定任务上更有效。这些提示不是预先确定的, 而是由模型在微调过程中通过反向传播学习的。

Cheng 等[149]提出用于改进零样本评估的通用提示检索方法, 称为 UPRISE, 调整了一个轻量级且多功能的检索器, 可以自动检索给定零样本任务输入的提示。检索器针对不同的任务集进行调整, 能够在推理过程中泛化到未见过的任务类型。

Jones 等[150]提出了 SynTra, 通过在合成任务上进行前缀调整来优化 LLMs 的系统消息, 然后将这种能力转移到更具挑战性、真实的摘要任务中, 实现对真实摘要任务的幻觉缓解。

8.1.2 涉及模型开发的幻觉缓解技术

一些工作专注于开发新的模型来缓解幻觉, 涉及解码策略优化、基于知识图谱优化、基于忠实度损失函数优化和微调。

(1) 解码策略优化

解码策略通常涉及专门针对模型生成阶段的设计技术。在幻觉方面, 这些技术旨在通过引导生成阶段走向真实或特定于上下文的生成来减少生成输出中幻觉的发生。

Shi 等[73]提出了上下文感知解码(Context- aware decoding, CAD), 采用对比输出分布, 放大了在使用上下文和不使用上下文时模型输出概率之间的差异以实现幻觉缓解。CAD 可以与现成的预训练语言模型一起使用, 无需额外训练, 在解决知识冲突的幻觉缓解任务时更有效。

Chuang 等[70]提出了通过对比层进行解码的方法, 称为 DoLa, 通过简单的解码策略减轻预训练 LLMs 中幻觉, 无需外部知识调整或额外的微调。DoLa 通过对比后续层和前述层之间在词汇空间中的 logit 差异获得下一个单词的分布。充分利用了在特定 Transformer 层中观察到的事实知识的本地化。DoLa 增强了对事实知识的识别, 并最小化生成不正确事实的可能性以实现幻觉缓解。

Li 等[151]设计了推理时干预, 通过在推理过程中沿着一组方向移动模型激活来操作, 这些方向覆盖了有限数量的注意力头。首先识别出具有高线性探测准确性的、用于真实性的稀疏注意力头集合。然后在推理过程中沿着这些与真实性相关的方向移动激活。自回归地重复相同的干预步骤, 直到生成整个答案。这种干预显著提高了 LLaMA 在 TruthfulQA 基准上的性能, 一定程度上缓解了不真实幻觉。

(2) 基于知识图谱优化

知识图谱是有组织的数据集合, 包括有关实体、其特征以及它们之间的联系的详细信息。它对数据进行排列, 使机器能够理解文本关系和语义, 为复杂的推理、数据分析和信息检索提供了基础。

将事实知识纳入知识图谱中被认为是减轻 LLMs 幻觉的一种有效的方法。现有方法通常仅使用用户的输入来查询知识图谱, 无法解决 LLMs 在推理过程中产生的事实幻觉。Guan 等[152]为了解决这个问题, 提出了一种将 LLMs 与知识图谱相结合的基于知识图的改造新框架, 通过根据知识图谱中存储的事实知识对 LLMs 的初始草案响应进行改造, 以减轻推理过程中的事实幻觉。基于知识图的改造方法利用 LLMs 生成的响应中提取、选择、验证和改进事实陈述, 从而实现自主知识验证和提炼过程, 无需任何额外的手动操作。

Ji 等[153]提出了 RHO 框架, 利用知识图谱(Knowledge graph, KG)中连接的实体和关系谓词的表示来生成更忠实的响应。为了提高忠实度, 他们将局部和全局的知识基准引入对话生成, 并进一步利用一个会话推理模型来重新排列生成的响应。这两种知识基准帮助模型有效地从与上下文相关的子图中编码和注入知识信息, 并通过适当的注意力实现这一点。通过各种知识基准和推理技术改善了外部知识和对话上下文之间的融合与互动, 进一步减少了幻觉。

Fatahi Bayat 等[154]通过从外部知识中检索证据进行事实错误检测和纠正, 提出 FLEEK, 旨在帮助最终用户进行事实验证和修正。FLEEK 具有用户友好的界面, 能够自主识别输入文本中可能可验证的事实, 为每个事实制定问题, 并查询精选的知识图和开放网络以收集证据。随后使用获取的证据验证事实的正确性, 并提出对原始文本的修改, 有效减轻事实错误幻觉。

(3) 基于忠实度损失函数优化

忠实度指标可以衡量模型的输出与输入数据或真实情况的匹配程度, 基于忠实度的损失函数可以有效指导幻觉缓解任务。

Yoon 等[155]介绍了用于视频引导对话的框架, 称为 THAM, 通过引入信息理论正则化来缓解特征级别的幻觉效应。THAM 框架包含了从响应语言模型和提出的幻觉语言模型之间的互信息中得出的文本幻觉正则化损失, 最小化该损失有助于减少不加区分的文本复制问题。

Qiu 等[82]为了减少低资源语言摘要中存在的幻

觉, 将一些单语方法调整为跨语言转移, 并提出了一种基于根据每个训练实例的 mFACT 分数加权损失的新方法。

(4) 微调

监督微调是使用标记数据调整 LLMs 以完成下游任务的重要阶段, 帮助模型遵循人类命令来执行特定任务, 并最终提高模型输出的可信度。在 SFT 的背景下, 数据质量直接决定了微调模型的性能。在监督微调期间, LLMs 的权重根据特定于任务的损失函数的梯度进行调整, 该函数测量 LLMs 的预测和真实标签之间的差异。还有一些研究通过模仿学习等方法模拟监督过程实现无监督微调。这些技术在增强 LLMs 的适应性方面也特别有效, 可以在一定程度上缓解幻觉。

Elaraby 等[156]专注于衡量和缓解弱开源 LLMs 中的幻觉, 提出了知识注入和师生教学框架, 称为 HALOCHECK。通过使用领域知识进行微调增强小型 LLMs 的知识或用大型 LLMs 生成详细的问题答案来指导小型 LLMs。通过评估幻觉的严重程度, 进一步优化教师 LLMs 的参与, 减轻了对教师模型频繁查询的需求。

Köksal 等[157]引入了幻觉增强表述, 利用 LLMs 幻觉创建反事实数据集并增强归因, 仅需少量反事实数据集微调模型, 就可以实现比在事实数据集上训练的模型更好的幻觉缓解效果。

Tian 等[158]采用自动事实检查方法和基于偏好的学习, 通过直接偏好优化算法来解决幻觉问题。对 LLaMa-2 模型进行了事实性微调, 包括基于参考和无参考的真实性评估, 展示了一种经济有效的提升模型事实性的方式, 特别适用于长篇文本生成任务的偏见缓解。

Razumovskaia 等[159]提出了 BEINFO, 应用“行为微调”来提高信息寻求对话所生成响应的忠实度以缓解幻觉。根据大量对话进行调整, 其中包含真实知识源, 并通过从大型知识库中随机采样的事实进行扩展。

Zhang 等[160]提出了拒绝感知指导调整方法, 称为 R-Tuning, 用于灌输 LLMs 拒绝技能。该方法形式化了识别 LLMs 的参数知识和用于训练的教学调整数据之间知识差距的想法。基于这种知识差距构建拒绝感知训练数据, 以教导 LLMs 何时避免回答来缓解幻觉。

Qiu 等[161]提出了一种有效表达知识时思考的方法, 称为 TWEAK, 将每一步生成的序列及其未来序列视为假设。使用假设验证模型, 根据相应假设对输入事实的支持程度, 对每一代候选集进行排名。TWEAK 仅调整解码过程, 没有重新训练生成模型, 可以轻松地集成到任何知识到文本生成器。他们还提出了 FATE 数据集, 将输入事实与单词级别的原始和反事实描述对齐。

Burns 等[64]使用模仿学习的思想训练模型, 建议通过以纯粹无监督的方式直接寻找 LLMs 内部激活中的潜在知识来缓解幻觉, 引入了一种仅在未标记的模型激活的情况下准确回答是/否问题的方法, 该方法可以恢复 LLMs 中表示的多种知识。

8.2 隐私保护

由于 LLMs 庞大的参数量和黑盒特性, 对隐私保护工作提出了新的要求。研究者考虑采用创新性手段来应对这一挑战, 包括差分隐私、安全多方计算(Secure multi-party computation, SMPC)和机器遗忘的方法。

8.2.1 基于差分隐私的隐私保护

简单的表述和用户友好的特性使得差分隐私(Differential privacy, DP)广泛应用于数据保护的多个领域。对于深度学习, 诸如差分隐私随机梯度下降(Differentially private stochastic gradient descent, DPSGD)[162]之类的噪声优化算法可以实现具有差异隐私保证的模型训练。DPSGD 在优化步骤期间将具有给定噪声尺度的每个样本高斯噪声注入到计算梯度中, 并且可以轻松地合并到各种模型中。因此, 大多数关于隐私保护 LLMs 的相关工作都是基于 DPSGD 开发。将现有的基于 DP 的 LLMs 分为 4 个集群, 包括基于 DP 的预训练、基于 DP 的微调、基于 DP 的提示调整和基于 DP 的合成文本生成。

(1) 基于 DP 的预训练

由于 DP 机制在 LLMs 上有不同的实现, 基于 DP 的预训练可以进一步增强 LM 对随机噪声扰动的鲁棒性。Yu 等[163]提出了具有差分隐私的选择性预训练, 以提高 BERT 上的 DP 微调性能。Igamberdiev 和 Haberal[164]在有或没有预训练的情况下实现了 DP-BART, 用于 LDP 下的文本重写。

(2) 基于 DP 的微调

大多数 LLMs 都根据公开数据进行了预训练, 并针对敏感领域进行了微调。直接使用 DPSGD 对敏感领域的 LLMs 进行微调是很自然的。Feyisetan 等[165]在词嵌入空间上应用了 dX 隐私[166], 这是本地差分隐私的一种变体, 以在 BiLSTM 上执行文本扰动。同样, Qu 等[167]用 dX 隐私来预训练和微调 BERT。Shi 等[168]提出选择性 DP 仅在敏感文本部分应用差分隐私, 并将其应用于 RoBERTa 和 GPT-2。Yu 等[169]通过

几种微调算法将 DPSGD 应用于微调 BERT 和 GPT-2。Mireshghallah 等[170]考虑了 BERT 私人微调过程中的知识蒸馏。Huang 等[171]考虑了基于检索的语言模型中的隐私, 该模型将根据存储在特定领域数据存储中的事实回答用户问题。他们考虑了一种场景, 其中特定于域的数据存储是私有的, 并且可能包含不应泄露的敏感信息。

Ozdayi 等[172]考虑了 LLMs 上基于软提示的前缀调整方法, 用于训练数据提取范围下的基于提示的攻击和基于提示的防御。Li 等[173]提出了差分隐私提示调整方法, 并通过属性推断和嵌入反转攻击评估了嵌入级信息泄漏的隐私性。

(3) 基于 DP 的合成文本生成

生成式 LLMs 自然可以通过基于采样的解码算法生成多个响应。对于 DP 调整的 LLMs, 从 LLMs 中采样文本满足后处理定理并保留相同的隐私预算。Yue 等[174]将 DPSGD 应用于合成文本生成, 并基于金丝雀重建评估性能。这些合成文本可以通过 LLMs 上的条件生成来获得, 并且可以安全地发布以替换其他下游任务的原始私有数据。同样, Mattern 等[175]使用 DP 优化器对 GPT-2 进行微调, 以进行条件合成文本生成, 并评估重复的隐私性。

8.2.2 基于安全多方计算的隐私保护

SMPC 是一种加密技术, 允许多方协作训练机器学习模型, 同时维护各自数据的隐私。它使这些各方能够联合计算模型更新, 而无需将其私有数据暴露给其他人, 从而确保各方都可以将其本地数据贡献给训练过程, 而不会泄露任何敏感信息。目前, SMPC 主要用于 LLMs 的推理阶段, 以保护模型参数和推理数据。然而, 保护 LLMs 隐私的一个主要挑战在于非线性操作所带来的限制, 例如 Softmax、GeLU、LayerNorm 等, 这些操作与 SMPC 不兼容。为了解决这个问题, 出现了两种技术途径: 模型结构优化和 SMPC 协议优化。

(1) 模型结构优化

通过利用 LLM 的鲁棒性并修改其结构来提高推理效率。特别是, 模型结构优化涉及用与 SMPC 兼容的其他算子替换 SMPC 不友好的非线性算子, 例如 Softmax、Gelu 和 LayerNorm。作为保护隐私的 LLMs 推理的早期工作, Chen 等[176]提出了一种利用同态加密的 BERT 模型隐私保护推理的创新实现, 称为 THE-X。利用多项式和线性神经网络等近似方法, 用 HE 可以计算的加法和乘法运算代替 LLMs 中的非线性运算。然而, THE-X 有以下 3 个局限性: 1) 它没有可证明的安全性, 在 THE-X 中, 客户端需要解密中间计算结果并以明文形式完成 ReLU 计算; 2) 模型结构变化导致的性能下降, 与纯文本相比, 隐私保护模型的推理性能平均下降超过 1%; 3) 由于模型结构发生变化, 需要重新训练以适应新的模型结构。为了应对这些挑战, 一些研究人员探索使用安全多方计算技术(例如秘密共享)来开发用于 LLMs 推理的隐私保护算法。Li 等[177]提出用多项式取代 LLMs 模型中的非线性运算, 同时利用模型蒸馏来保持性能。他们通过在多个数据集上进行的实验验证了算法的有效性, 并在 3 个尺度的 BERT 模型上进行了评估。在这项工作的基础上, Zeng 等[178]纳入了 Li 等的原则的方法和集成神经架构搜索技术进一步提高模型效率和性能。此外, Liang 等[179]集成了之前的工作, 特别关注自然语言生成任务。为了提高隐私保护推理的效率, 定制了嵌入重发和非线性层近似融合等技术, 以更好地符合 NLG 模型的推理特性。

(2) SMPC 协议优化

利用先进的 SMPC 协议来提高 LLMs 隐私保护推理的效率, 同时保持原有的模型结构。由于模型结构保持不变, 因此与明文模型相比, 使用基于 SMPC 协议优化的 LLMs 模型的隐私保护推理性能不受影响。SMPC 协议优化专注于通过设计专为 LLMs 非线性运算量身定制的高效 SMPC 算子优化 LLMs 的隐私保护推理效率。Hao 等[180]通过集成多个 SMPC 协议提高了 LLM 隐私保护模型推理的效率。Zheng 等[181]使用混淆电路来优化 LLMs 的非线性运算。Gupta 等[182]基于函数私密共享为 LLM 的每个函数构建了安全计算协议, 极大地提高了 LLM 的隐私保护推理效率。除了直接优化非线性 SMPC 协议之外, 一些工作提出了分段多项式来拟合非线性算子并提高 LLMs 的推理效率。Dong 等[183]利用分段多项式对 LLMs 中的指数和 GeLU 运算进行高精度拟合。Hou 等[184]针对 GPT 模型提出了一种基于子域 VOLE 的不平衡矩阵乘法的预处理打包优化方法, 大大降低了矩阵乘法的预处理开销。对于非线性处理, 它采用分段拟合技术并遵循安全 RNN 推理来优化近似多项式的计算效率。

8.2.3 基于遗忘隐私保护

“机器遗忘”是研究如何高效地消除特定训练数据对已训练模型影响的一类方法。《被遗忘权》等隐私法规进一步推动 LLMs 研究相关机器遗忘的方法。

Pawelczyk 等[185]指出精确的机器遗忘在 LLMs 上的计算是不可行的。他们提出了针对 LLMs 的上下文遗忘方法, 在推理时提供上下文输入而无需更新模型参数。为了遗忘特定的训练实例, 需要提供该

实例与一个翻转的标签以及其他正确标记的实例, 这些实例在推理时被作为输入添加到 LLM 中。这些上下文能够有效地从训练集中删除特定信息, 同时保持 LLMs 基本任务的性能, 可以作为隐私数据保护的一种策略。

Eldan 等[186]提出了一种从生成语言模型中有效遗忘目标信息的方法。包括 3 个主要组成部分: 首先, 使用一个经过强化训练的模型, 在目标数据上进一步训练以识别与遗忘目标最相关的标记, 通过将其 logits 与基线模型的 logits 进行比较。其次, 用通用的替代品替换目标数据中的特殊表达, 并利用模型自己的预测为每个标记生成替代标签。接着, 在这些替代标签上对模型进行微调, 这在模型被提示其上下文时有效地擦除了原始文本从模型的记忆中。

Chen 和 Yang[187]提出了一个高效的遗忘框架, 通过在 Transformers 中引入基于选择性师生目标学习的轻量级遗忘层, 能够在数据删除后高效地更新 LLMs, 而无需重新对整个模型进行训练。此外, 他们还引入了一个融合机制, 能够有效地结合不同的遗忘层, 学习忘记处理一系列遗忘操作的不同数据集。

8.3 偏见缓解

现有的偏见缓解策略可以根据 LLMs 工作流程的不同阶段分为: 预处理、训练中、推理阶段。预处理缓解技术旨在尽早消除数据集或模型输入中的偏差和不公平性, 而训练中缓解技术则侧重于减少模型训练期间的偏差和不公平性。内部处理方法无需训练或微调即可修改模型的权重或解码行为。作为后处理步骤消除偏见和不公平的技术侧重于黑盒模型的输出, 而无需访问模型本身。

反事实数据增强[188]是一种数据库去偏见策略, 通常用于减轻性别偏见。CDA 涉及通过交换数据集中的偏见属性词(例如, 他/她)来重新平衡语料库。例如, 为了帮助减轻性别偏见, 句子“医生去了房间, 他抓住了注射器”可以扩充为“医生去了房间, 她抓住了注射器”。然后, 重新平衡的语料库通常用于进一步训练以消除模型偏见。虽然 CDA 主要用于消除性别偏见, 但通过交换语料库中的宗教术语(例如教堂与清真寺)来生成反事实示例, 也可以实现减轻宗教偏见。

DROPOUT[189]研究使用 dropout 正则化作为偏见缓解技术。他们研究增加 BERT 和 ALBERT 注意力权重和隐藏激活的 dropout 参数, 并执行额外的预训练阶段。通过实验, 他们发现增加 dropout 正则化可以减少这些模型中的性别偏见。他们假设, dropout 对 BERT 和 ALBERT 中注意力机制的干扰有助于防止他们学习单词之间不良的关联。他们还将这项研究扩展到其他类型的偏见。与 CDA 类似, 使用增强的 dropout 正则化对英语维基百科的句子进行额外的预训练。

Self-Debias[190]提出了一种事后去偏技术, 利用模型的内部知识来阻止其生成有偏见的文本。非正式地, Schick 等[190]建议使用手工制作的提示首先鼓励模型生成有毒文本。例如, 从自回归模型生成时可能会提示“以下文本因性别而歧视人们”。然后, 可以从模型中生成非歧视性的第二个延续, 其中在第一个有毒生成下被认为可能的单词的概率被按比例缩小。

SentenceDebias[191]将 Bolukbasi 等[192]提出的 Hard-Debias 词嵌入去偏技术扩展到句子表示。SentenceDebias 是一种基于投影的去偏技术, 需要估计特定类型偏见的线性子空间。可以通过投影到估计偏差子空间并从原始句子表示中减去所得投影来消除句子表示的偏见。Liang 等[191]使用三步过程来计算偏见子空间。首先, 他们定义了一系列偏见属性词(例如, 他/她)。其次, 将偏见属性词置于句子中。这是通过查找文本语料库中的句子中出现的偏差属性词来完成的。对于在此上下文化步骤中找到的每个句子, 应用 CDA 来生成一对仅在偏差属性词方面不同的句子。最后, 他们估计偏见子空间。对于上下文化步骤中获得的每个句子, 可以从预训练的模型中获得相应的表示。然后使用主成分分析来估计所得表示集的变化的主要方向。可以采用前 K 个主成分来定义偏见子空间。

迭代零空间投影[193]是一种类似于 SentenceDebias 的基于投影的去偏技术。通过训练线性分类器来预测想要从表示中删除的受保护属性(例如性别), 从而消除模型表示的偏见。然后, 可以通过将表示投影到学习分类器权重矩阵的零空间中来消除偏见, 从而有效地删除分类器用于从表示中预测受保护属性的所有信息, 然后可以迭代地应用该过程来消除表示的偏见。

Ganguli 等[26]测试了一种假设, 即通过 RLHF 训练的语言模型有能力进行道德自我纠正(避免产生有害输出)。在 3 个道德自我纠正方面的不同实验中验证了这一假设。他们指出道德自我纠正的能力通常会随着模型规模的扩大和 RLHF 训练的增加而提高。在这一规模下, 语言模型获得了两种可用于道德自我纠正的能力: 1) 遵循指令; 2) 学习复杂的规范性伤害概念, 如刻板印象、偏见和歧视。因此, 他们提出可以按照指令避免某些道德上有害的结果。

8.4 鲁棒增强

LLMs 偶尔会由于鲁棒性问题做出不忠实的推理, 即得出的结论不遵循先前生成的推理链。为了解决这个问题, 现有的工作提出使用来自外部工具或模型的自动反馈来指导推理过程, 验证推理过程并纠正错误, 或通过基于流程的反馈微调 LLMs。尽管已经提出了各种方法来增强模型的鲁棒性并减轻此漏洞, 但许多方法需要大量消耗资源(例如对抗性训练)或仅提供有限的保护(例如防御性退出)。

Shen 等[194]提出了一种面向 Transformer 架构的动态注意力方法, 以增强模型本身针对各种对抗性攻击的固有鲁棒性。不需要下游任务知识, 也不会产生额外成本。所提出的动态注意力由注意力纠正和动态建模两个模块组成。动态注意力显著减轻了对抗性攻击的影响, 与之前针对广泛使用的对抗性攻击的方法相比, 性能提高了 33%。

Liu 等[195]受最近成功利用检索模块增强大规模神经网络模型的启发, 提议检索与测试样本语义相似的示例, 以制定相应的提示。通过这种策略选择的上下文示例可以作为信息量更大的输入, 从而释放 GPT-3 的广泛知识, 增强对不同输入的鲁棒性。他们在多个自然语言理解和生成基准测试中对所提出的方法进行了评估, 发现基于检索的提示选择方法始终优于随机基准方法, 在任务相关数据集上微调的句子编码器能产生更有用的检索结果。

Suzgun 等[115]在 BIG-Bench 的基础上讨论了语言模型在哪些任务上低于人类评估者的平均表现, 以及这些任务实际上是当前语言模型无法解决的吗? 他们重点关注 23 项具有挑战性的 BIG-Bench 任务。他们将思维链(Chain of thoughts, CoT)提示应用于该任务以提高模型对不同输入的鲁棒性, 使 PaLM 在 23 项任务中的 10 项上超越了人类评分者的平均表现。

为了研究如何建立模型, 通过理解人类可读的指令来学习新任务, Mishra 等[196]引入了一个包含 61 个不同任务、人类编写的指令和 193000 个任务实例(输入-输出对)的自然指令数据集。这些指令来自用于创建现有 NLP 数据集的众包指令, 并映射到统一的模式。他们利用这个数据集, 通过在已见任务上训练模型来测量跨任务泛化, 并测量对其余未见任务的泛化。结果表明, 在对未见任务进行泛化评估时, 模型从指令中获益, 但这些模型远远落后于估计的性能上限。

Lu 等[197]证明提供样本的顺序会导致接近最先进水平的性能与随机猜测性能之间的差异。他们详细分析了这一现象, 并确定它存在于各种规模的模型中, 它与特定的样本子集无关, 而且一个模型的特定好的排列组合不能移植到另一个模型中。他们利用语言模型的生成特性构建了一个人工开发集, 并根据该开发集上候选排列的熵统计, 确定了性能良好的提示。在 11 个不同的既定文本分类任务中, 为 GPT 系列模型带来了 13%的相对改进。

Min 等[198]表明实际上不需要基准真相演示——在演示中随机替换标签几乎不会损害性能, 在包括 GPT-3 在内的 12 种不同模型中始终如此。演示的其他方面是最终任务性能的关键驱动因素, 包括它们提供了一些示例: 1) 标签空间, 2) 输入文本的分布, 3) 序列的整体格式。他们的分析提供了一种新的方式来理解上下文学习如何以及为何起作用, 并提出新的问题: 仅通过推理可以从大型语言模型中学到多少东西。

Yoo 等[199]重新审视了基准真相标签在上下文学习中的重要性, 通过引入标签正确性灵敏度和基准真相标签效应比两个新指标, 实现对基准真相标签演示的影响进行量化分析。发现正确的输入-标签映射会对下游的上下文学习性能产生不同的影响, 具体取决于实验配置。他们还确定了提示模板的冗长程度和语言模型的大小等关键因素, 是实现更具抗噪能力的 ICL 的控制因素。

Wei 等[200]研究了语言模型中的上下文学习 ICL 如何受到语义先验与输入标签映射的影响。他们研究了不同模型系列(GPT-3、InstructGPT、Codex、PaLM 和 Flan-PaLM)的两种设置——带翻转标签的 ICL 和带语义无关标签的 ICL。结果表明: 1)超越语义先验是大规模模型的一种新兴能力, 小型语言模型主要依赖于预训练中的语义先验, 大型模型可以在上下文中出现与先验相矛盾的示例时推翻语义先验; 2)足够大的语言模型可以在 SUL-ICL 环境中进行线性分类。他们对指令调整模型进行了评估, 发现指令调整既能加强语义先验的使用, 也能提高学习输入-标签映射的能力, 但前者的作用更大。

随着更强大的大型语言模型(例如 ChatGPT 和 GPT-4)的出现, 上下文学习在通过利用数据标签对作为前提提示来利用这些模型执行特定任务方面获得了显著的优势。虽然合并演示可以极大地提高 LLMs 在各种任务中的性能, 但它可能会引入新的安全问题: 攻击者只能操纵演示而不更改输入来执行攻击。Wang 等[201]从对抗的角度研究了 ICL 的安全问题, 重点关注演示的影响, 提出了一种基于文本攻击, 称为 TextAttack 的 ICL 攻击, 在不改变输入的

情况下只操纵演示来误导模型。结果表明随着演示次数的增加, 上下文学习的鲁棒性会降低, 受到对抗性攻击的演示可以转移到不同的输入示例中。

Jiang 等[31]尝试通过自动发现在此查询过程中使用的更好提示来更准确地估计 LLMs 中包含的知识, 提高对不同输入的鲁棒性。提出了基于挖掘和释义的方法来自动生成高质量和多样化的提示, 以及集成方法来组合来自不同提示的答案。该方法可以为 LLMs 的知识提供更严格的下限。

8.5 自我纠正

纠正幻觉、不忠实推理和不良输出缺陷的一类主流方法是自我纠正, 即提示或指导 LLMs 本身修复其输出中的问题。利用自动反馈的技术, 可以使基于 LLMs 的解决方案更加实用和可部署, 并且只需最少的人工反馈。自我纠正的整体结构如图 14 所示, 这个过程涉及 3 个模型: 语言模型、评论家模型和精细化模型。

语言模型 $M: X \to Y$ 通过将输入 $x \in X$ 映射到输出文本 $y \in Y$ 来执行特定任务。最初的一代 y 可能是不完美的, 并且会遭受诸如幻觉和错误推理等各种问题。评论家模型 $C: X \times Y \to F$ 学习生成反馈 $x, y, c \to y_{new}$, 其中 $y \sim M(x)$ 是语言模型的输出或部分输出, c 是某种格式的反馈, 例如标量值或自然语言。精细化模型 $R: X \times Y \times F \to Y$ 学习根据反馈 c 修复输出 $x, y, c \to y_{new}$, 其中 y_{new} 是修改后的输出。除了修复输出之外, 一些精炼模型还通过微调或强化学习直接修复语言模型 M。

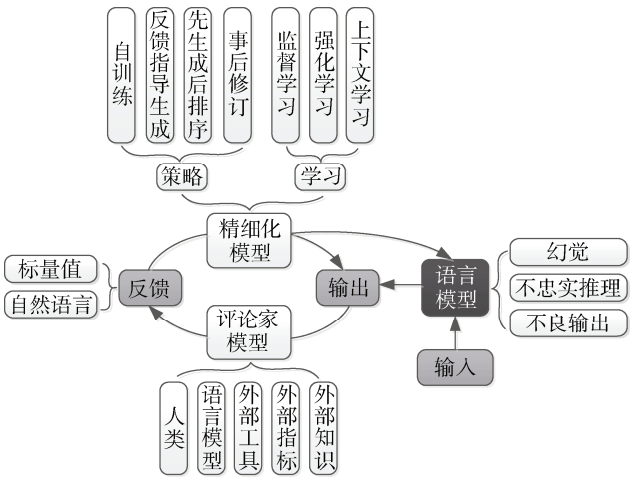

图 14 具有自动反馈功能的自我纠正 LLMs 框图

Figure 14 A conceptual framework for self-correcting LLMs with automated feedback

LLMs 经常通过编造事实或引用不存在的资料来源来产生幻觉。为了解决这个问题, 一些研究提出通过将模型生成的输出与可靠的知识源交叉引用来收集有关潜在事实不准确的自动反馈。随后的细化模型可以利用收集到的反馈来纠正幻觉。

8.5.1 人工反馈

在理想情况下, 直接利用人类反馈来优化模型参数。通常, 这种方法遵循图 15 中描述的框架: 1) 候选输出由 LLMs 生成, 2) 人类对这些输出提供反馈或改进, 3) 然后 LLMs 根据收集到的(输出、反馈)直接进行优化以更好地符合人类偏好。一个简单的策略是使用正向标记的反馈来微调输出的模型。例如, Sparrow[202]根据人类的说法, 对收集到的被评为首选且符合规则(关于正确性、有害性和有用性)的对话进行 LLM 微调。

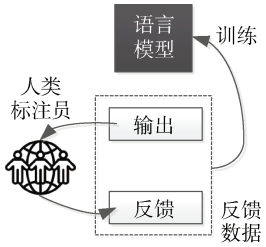

图 15 根据人类反馈直接优化

Figure 15 Directly optimization with human feedback

类似的, Scheurer 等[203]利用 LLM 根据人类反馈生成原始输出的多次细化, 然后选择最佳细化来微调原始 LLM。首先, 人类注释者为不正确的代码编写自然语言反馈。然后, 细化模型利用此反馈来纠正代码。最后, 改进后的代码随后用于微调代码生成 LLM。然而, 仅利用正面数据(人工精炼或正面评价的数据)进行微调可能会限制模型识别和纠正负面属性或错误的能力。为了解决这个问题, Chain-of-Hindsight[204]对模型输出的 LLM 进行了微调, 并结合正反馈和负反馈。除了微调之外, 还探索了其他优化方法。例如, Gao 等[205]利用人类反馈作为奖励信号, 并通过上下文老虎机学习来优化模型。

8.5.2 奖励建模和人类反馈强化学习

收集人类反馈可能既费力又耗时, 一种有效的替代方法是训练模拟人类反馈的奖励模型。经过训练后, 该奖励模型可以为每个模型输出提供一致的实时反馈, 从而避免持续人类参与的需要。这种方法的一个突出例子是 RLHF, 如图 16 所示。首先要求人类注释者标记不同 LLM 输出的偏好, 然后训练奖励模型来预测人类偏好。然后, 采用 RL 算法来优化模型。RLHF 及其变体已被证明可以有效地纠正 LLMs, 使其变得更有益、更少危害, 以及灌输道德正确性[26]。

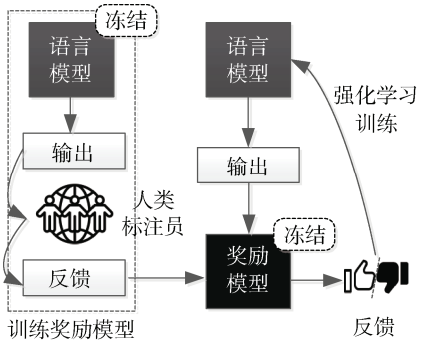

图 16 训练近似人类反馈的奖励模型

Figure 16 Training a reward model that approximates human feedback

8.5.3 自动反馈

由于收集人类反馈的成本较高, 许多研究探索了使用自动反馈来最大限度地减少人类干预的需求。一方面, 为了区分人类反馈和自动反馈, 将人类反馈定义为人类评估者对基本模型生成的输出进行的质量评估。然后, 该反馈用于直接优化或奖励模型学习。另一方面, 自动反馈是在离线环境中收集的, 无需对模型输出进行人工评估。自动反馈类型主要包括两种: 来自外部指标/模型的外在反馈和来自语言模型本身的内在反馈。

外部指标提供的反馈经常用于训练时的校正。由于度量信号的离散性质, 大多数方法都侧重于不可微的训练技术。最小风险训练[206]通过在损失函数中结合具有最大对数似然的度量得分来优化模型参数, 使用外部评估指标。它可以在训练期间优化指标得分。然而, 它可能导致模型在某些指标上存在鲁棒性缺陷[207]。Liu 和 Liu[208]利用对比学习框架根据指标得分对候选对象重新排序, 从而弥合了训练和推理目标之间的差距。Li 等[209]采用深度 RL 算法和 Jauregi Unanue 等[210]利用 Gumbel Softmax[211]从 BERTScore[212]构建分布式语义奖励并减轻暴露偏差。为了稳定梯度, Wu 等[213]利用对比判别器和近端策略优化来模仿人类的文本。最近, Chang 等[214]提出了一种比近端策略优化更高效的 RL 算法, 称为 RLGF, 用于通过预定义的奖励来微调 LLMs。他们将合理但不完整的引导策略融入策略梯度框架中, 并学习出接近最优的策略。与仅在微调时利用反馈不同, Korbak 等[215]在预训练阶段采用条件训练和自动分类器来标记不需要的内容。

语言模型本身可以用来为其自身的输出提供反馈, 而不是利用外部指标作为反馈。这就产生了通过引导其原始输出来自我改进 LLMs 的自我训练策略, 如图 17 所示。STaR[216]通过提示 LLMs 生成带有理由的答案来利用 CoT 的思想。通过选择导致正确答案的理由来进一步微调 LLMs, LLM 的表现得到提高。可以迭代此过程以进一步提高性能。Huang 等[217]通过将自我一致性应用于多数投票推理路径(导致获得最多投票的答案的路径)来遵循这一想法。LLMs 通过增强提示对选定的推理答案数据进行了微调。这一策略也被用来减少 LLMs 的有害反应。AI 反馈对齐(Reinforcement learning for AI feedback, RLAIF)[218]采用了批判→修订→监督学习的策略。最初的毒性反应由 LLMs 本身根据一系列人类定义的原则进行批评和修订。之后, LLMs 会根据修改后的答案进行微调。AlpacaFarm[219]进一步表明 LLMs 可以通过强化学习进行自我改进。设计了 LLMs 提示来模拟 RLHF 中的人类反馈, 并表明反馈是有效的, 并且大大降低了成本。Gulcehre 等[220]通过提出强化自我训练, 称为 ReST, 以进一步改进自我训练。它迭代地执行以下两个步骤来改进 LLMs: 1) Grow 步骤通过从策略模型(即当前的 LLM)中采样来生成数据集, 2) Improve 步骤使用离线 RL 算法优化 LLM 策略。

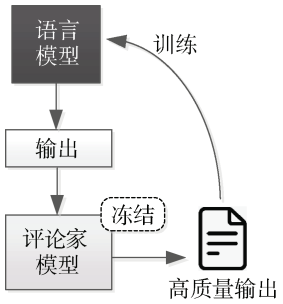

图 17 LLMs 的自我训练策略

Figure 17 Self-training strategies for LLMs

8.6 其他防御

除了上述防御措施外, 还有一些工作讨论了针对欺骗和攻击的防御, 由于这一部分研究内容较少, 将它们归纳到本小节介绍。

LLMs 由训练语料库塑造, 从中学习行为、概念和数据分布, 其安全性在很大程度上受到训练语料库的影响, 一些工作考虑通过清洗语料库来实现防御。他们从以下几个角度进行语料库清洗: 选择更高质量的原始语料库用于训练 LLMs、删除重复数据等。

Wang 等[221]通过递归思考防止 LLMs 的欺骗行为, 利用复杂的阿瓦隆游戏作为测试平台, 探索 LLMs 在欺骗性环境中的潜力。阿瓦隆充满了错误信息, 需要复杂的逻辑, 表现为“思维游戏”。受到阿瓦隆游戏中人类递归思维和换位思考的功效的启发,

引入了一种新颖的递归思考框架, 称为 ReCon, 以增强 LLMs 识别和抵制欺骗性信息的能力。ReCon 结合了制定和细化的思考过程产生最初的思想和语言, 并进一步完善它们。此外, 分别将一阶和二阶视角转换合并到这些过程中。它可以有效地帮助 LLMs 识别和操纵欺骗性信息, 而无需额外的微调和数据。

Wang 等[222]系统地探索了如何通过域自适应训练来降低语言模型的毒性。他们从训练语料、模型大小和参数效率 3 个维度进行研究, 建议利用语言模型的生成能力, 生成无毒数据集进行领域自适应训练, 以减少暴露偏见。他们第一次全面研究了参数大小从 126M 到 530B 规模的解毒语言模型, 在各种模型规模的自动和人工评估中, 性能始终优于现有的基线方法。得到了两个重要结论: 1) 在相同的预训练语料库中, 大型语言模型的毒性水平与小型语言模型相似; 2) 大型语言模型需要更多努力才能解毒。他们还在语言模型中添加和训练纯适配层以实现参数高效的解毒训练, 更好地权衡毒性和复杂性。

大语言模型经过调整, 可以拒绝回答可能造成伤害的请求, 但对抗性示例可以规避对齐尝试, Carlini 等[223]研究了大语言模型在多大程度上保持一致, 他们指出, 现有的基于 NLP 的优化攻击不足以可靠地攻击对齐的文本模型。因此, 当前攻击的失败不应被视为对齐文本模型在对抗性输入下保持对齐的证据。但多模态的大规模机器学习模型更容易受到对抗性攻击, 即通过输入图像的对抗性扰动诱导执行任意未对齐的行为。

Deng 等[92]提出了一种防御框架, 通过与攻击框架的迭代交互来微调受害 LLM, 以增强其针对红队攻击的安全性。

9 检测

检测 LLMs 生成内容安全性的工作提供了检测策略和基准, 倡导更具适应性和鲁棒性的模型以提高检测准确性。根据其显著特征, 现有的检测方法可以分为三类: (1) 基于训练的分类器, 通常根据收集的二进制数据(人类和人工智能生成的文本分布)微调预训练的语言模型。(2) 零样本检测器利用典型 LLMs 的固有属性(例如概率曲线或表示空间)来执行自我检测。(3) 水印涉及在生成的文本中隐藏识别信息, 这些信息稍后可用于确定文本是否来自特定的语言模型, 而不是检测一般人工智能生成的文本。

9.1 基于训练的检测

训练检测分类器的早期工作侧重于虚假评论、虚假新闻或小模型检测。随后, 人们对这方面研究的兴趣不断增长, 转向检测 LLMs 带来的高质量文本。

第一个工作重点是黑盒检测, 当模型源已知时, 一些工作使用以下策略: 1) 收集来自不同模型系列生成的文本, 并训练一个强大的检测器, 用于检测具有 1000 多个标记的文本。GPTZero[224]还收集了来自各种 LLMs 的人工撰写的文本, 涵盖学生撰写的文章、新闻文章和跨学科的问答数据集。类似地, G3Detector[225]声称通过微调 RoBERTa-large 成为通用 GPT 生成的文本检测器, 并探索了使用合成数据对训练过程的影响。GPT-Sentinel[226]在其构建的数据集 OpenGPTText 上训练 RoBERTa 和 T5 分类器。2) 关于解码策略的混合, Ippolito 等[227]发现, 一般来说, 判别器在解码策略之间的迁移很差, 但对混合数据进行训练可以有所帮助。GPT-Pat[228]训练暹罗网络来计算原始文本和重新解码文本之间的相似度。3) 混合策略涉及额外信息, 例如图结构、对比学习和对抗性训练等。

另外, 当源模型未知时, OpenAI 文本分类器和 GPTZero 仍然可以通过 1) 跨域传输。其他工作, 如 ConDA[229]也依赖于在各种模型系列上训练并在未见过的模型上进行测试的检测器的零样本泛化能力。此外, Ghostbuster[230]直接使用已知的 2) 代理模型的输出作为训练分类器来检测未知模型的信号。此外, 3) 野外检测通过收集各种人类著作中的文本和不同 LLMs 生成的深度伪造文本, 在不知道其来源的情况下进行检测, 从而提供了一个野外测试平台。

第二个工作重点在白盒场景下, 当模型的全部或部分参数可访问时。GLTR[231]会在每个解码步骤中针对绝对单词排名训练逻辑回归。当只有像模型输出 logits 这样的部分信息可用时, SeqXGPT[232]通过合成包含用 LLMs 打磨的文档的数据集来引入句子级检测挑战, 并建议使用 logits 来检测它。Snifferp[233]利用模型之间的对比 logits 作为训练的典型特征来执行检测和原点跟踪。

9.2 零样本检测

在零样本设置中, 不需要大量的训练数据来训练检测器。相反, 可以利用机器生成的文本和人类编写的文本之间的固有区别, 使检测器免于训练。免训练检测的关键优势是它能够适应新的数据分布, 而无需额外的数据收集和模型调整。值得注意的是, 虽然水印方法也可以被认为是零样本, 但将它们视为独立的方法。之前的工作利用了熵、平均对数概率得分、困惑度等从语言模型中获得作为确定其起源的判断标准。然而, 随着 LLMs 变得多样化和高质量的文本生成器, 这些简单的功能就会失败。同样, 还

有黑盒和白盒检测, 总结如下。

当黑盒模型的来源已知时, DNA-GPT[234]通过利用重新提示文本的连续分布与原始文本之间的n-gram 散度来实现卓越的性能。此外, DetectGPT[235]也研究使用另一种代理模型来替代源模型, 但效果并不理想。相比之下, Mireshghallah 等[236]证明像OPT-125M 这样的较小代理模型可以作为通用黑盒文本检测器, 实现比使用源模型接近甚至更好的检测性能。此外, Krishna 等[237]建议建立生成文本的数据库, 并通过将其语义相似性与数据库中存储的所有文本进行比较来检测目标文本。最后, DetectGPT4Code[238]还研究了通过条件概率散度通过代理小代码生成模型检测 ChatGPT 生成的代码, 并在代码检测任务上取得了显著改进。

当模型来源未知时, 持久同源维度估计器[239]观察到, 真实文本在统计上比各种可靠生成器中机器生成的文本表现出更高的内在维度。意味着测量这种内在维度, 并结合像 Roberta 这样的附加编码器来促进估计过程。

Miao 等[240]通过根据贝叶斯不确定性选择典型样本并将典型样本的分数插值到其他样本, 提高了使用贝叶斯代理模型的 DetectGPT 的效率。此外, 与 DNA-GPT 使用条件概率进行判别类似, Fast-DetectGPT[241]通过用条件概率替换 DetectGPT 中的概率来构建高效的零样本检测器曲率并见证了效率的显著提高。此外, GPT-Who[242]利用基于统一信息密度的功能对每个 LLMs 和人类作者的独特统计签名进行建模, 以实现准确的作者归属。

当给予模型完全访问权限时, Su 等[243]通过一种快速高效的 DetectLLM-LRR(对数似然 LogRank 比)方法和另一种更准确的DetectLLM-NPR(标准化扰动对数秩)方法, 利用对数秩信息进行零样本检测, 尽管由于速度较慢, 扰动的需要仍然存在。

Varshney 等[244]提出了一种主动检测幻觉的方法, 使用模型的 logit 输出值识别可能的幻觉并验证其准确性。最重要的认识是, 在生成过程中处理幻觉是至关重要的, 因为当模型之前在输出中经历过幻觉时, 它增加了生成带有幻觉的句子的概率。他们还强调模型的 logit 信息是幻觉检测方法的补充信息, 而不是必要先决条件。

Zhang 等[245]为了提高幻觉检测的效率, 提出了一种无参考、基于不确定性 LLMs 幻觉检测方法。从 3 个方面模仿人类在事实性检查中的关注: 1) 关注给定文本中信息最丰富、最重要的关键词; 2) 关注历史背景下不可靠的单词, 这可能会导致一连串的幻觉; 3) 关注单词属性, 例如单词类型和单词频率。

Manakul 等[246]假设如果 LLMs 了解给定概念, 则样本响应可能是相似的并且包含一致的事实; 对于幻觉事实, 随机抽样的反应可能会出现分歧并相互矛盾。并根据以上假设提出了 SelfCheckGPT, 可用于以零样本方式对黑盒模型的响应进行事实检查。证明该方法可实现检测非事实和事实句子以及根据事实对段落进行排名。

9.3 基于水印的检测

文本水印将算法可检测的模式注入生成的文本中, 同时理想地保留语言模型输出的质量和多样性。水印旨在确定文本是否来自特定语言模型, 而不是普遍检测任何潜在模型生成的文本。因此, 在文本水印检测中始终需要了解模型源。

在黑盒设置中, 例如基于 API 的应用程序, LLMs 提供商使用的语言模型的专有性质阻止下游用户出于商业原因访问采样过程。或者, 用户可能希望通过后处理为人类创作的文本添加水印。在这种情况下, 黑盒水印旨在自动操作生成的文本以嵌入第三方可读的水印。传统工作设计了复杂的语言规则, 例如释义、语法树操作和同义词替换, 但缺乏可扩展性。后来的工作转向预先训练的语言模型以实现高效的水印。例如, Yang 等[247]提出了一种基于上下文感知词汇替换的自然语言水印方案。具体来说, 他们采用 BERT 通过推断候选者与原始句子之间的语义相关性来建议候选集。Yang 等[248]首先定义二进制编码函数来计算对应于单词的随机二进制编码。为无水印文本计算的编码符合伯努利分布, 其中代表位 1 的单词的概率约为 0.5。为了注入水印, 他们通过有选择地用代表位 1 的基于上下文的同义词替换代表位 0 的单词来改变分布。然后使用统计测试来识别水印。

最流行的1) 免训练水印在部署模型时直接操纵解码过程。在为 GPT 输出加水印的过程中, Aaronson[249]与 OpenAI 合作, 首先开发了一种对语言模型加水印的技术, 使用指数最小采样从模型中采样文本, 其中采样机制的输入是前 k 个连续的哈希值通过伪随机数生成器生成单词。根据 Gumbel Softmax 规则, 他们的方法被证明可以保证质量。此外, Christ 等[250]提供了不可检测水印的正式定义和构造。他们受密码学启发的水印设计提出, 通过对每个块进行散列来为下一个块提供采样器, 从而对来自语言模型的文本块进行水印。然而, 该方法只有理论概念, 没有实验结果。免训练水印的另一项开创性工作[251]在解码过程中嵌入了隐形水印, 根据前缀单词的哈

希将词汇分为“绿表”和“红表”, 并巧妙地增加了从绿名单中选择的概率。然后, 具备散列函数和随机数生成器知识的第三方可以为每个单词重现绿表并监控对绿表规则的违反。随后, Zhao 等[252]通过一致使用固定的绿红表分割来简化该方案, 表明新的水印始终保持有保证的生成质量, 并且对文本编辑更加鲁棒。Kuditipudi 等[253]利用随机水印密钥通过逆变换采样和指数最小采样从单词概率分布中进行采样, 创建无失真的水印。Hou 等[254]提出了一种基于局部敏感哈希的句子级语义水印, 它对句子的语义空间进行划分。这种设计的优点是增强了针对释义攻击的鲁棒性。DiPmark[255]是一种无偏分布保留水印, 它在水印过程中保留了原始单词分布, 并且通过结合新颖的重新加权策略, 结合基于分配唯一独立同分布密码的哈希函数, 对单词的适度变化具有鲁棒性。就上下文而言, 鉴于随机绿红表分割的缺点, Fu 等[256]使用输入序列来获取语义相关的标记以进行水印, 以改进某些条件生成任务。

尽管无需训练水印, 文本水印也可以通过预推理训练或后推理训练来注入: 2) 基于训练的水印。预推理训练的一个例子是 REMARK-LLM[257], 它通过消息编码模块注入水印以生成密集的单词分布, 然后消息解码模块从带水印的文本中提取消息并重新参数化作为连接密集分布和单词的 one-hot 编码的桥梁。缺点是需要对源数据进行训练, 并且可能无法很好地推广到未见过的文本数据。相反, 推理后训练涉及添加经过训练的模块来协助在推理过程中注入水印。例如, Liu 等[258]提出了一种用于 LLMs 的语义不变鲁棒水印, 通过利用另一个嵌入 LLM 来为所有前面的标记生成语义嵌入。然而, 它并不是免训练的, 因为这些语义嵌入通过训练后的水印模型转换为水印逻辑。

除了 0-bit 水印之外, 还有 3-bits 水印。例如, Yoo 等[259]遵循图像水印设计了多位水印, 识别对轻微损坏不变的自然语言特征, 并提出了抗损坏填充模型。COLOR[260]随后通过在语言模型生成过程中嵌入可追踪的多位信息同时允许 0-bit 检测来设计另一种多位水印。Fernandez 等[261]还通过更强大的统计测试和多位水印整合了 LLMs 的水印。

Pacchiardi 等[262]将 LLMs 的欺骗定义为输出错误陈述。他们开发了一个简单的测谎仪, 该检测器既不需要访问 LLM 的激活(黑盒), 也不需要所讨论事实的基础知识。探测器在怀疑谎言后询问一组预定义的无关后续问题, 并将 LLM 的“是/否答案”送入逻辑回归分类器后实现分类结果。

10 机遇与挑战

自然语言理解能力的突破: LLM 的发展推动了 NLP 研究的突破, 实现更多样化的生成文本, 捕获更长的上下文关系, 理解更复杂的句子逻辑结构和关联性, 在对话中提供更加准确和连贯的回应。但目前针对 GPT-4 等最先进的模型进行的相关测试表明, LLM 的语言理解能力相较于人类仍有较大差距, 需要进一步突破机器的自然语言理解能力。

模型泛化能力提升: 一方面, 模型泛化能力弱可能在面对对抗性或分布外的输入信息做出错误的输出, 表现出不良信息输出或幻觉问题。另一方面, LLM 的泛化能力可能会导致其生成虚假内容。LLM 可以利用其对现实世界知识的理解, 生成逼真的虚假内容。这些虚假内容可能会被用于传播错误信息、操纵公众舆论或进行网络攻击。因此, 不仅需要提高 LLM 在分布外数据上的泛化性, 还要采取措施来减轻 LLM 泛化能力带来的安全风险。例如, 可以对 LLM 生成内容进行审查[233-235], 以确保其生成的内容准确和真实。此外, 也可以开发技术来检测和阻止 LLM 的滥用和攻击。

零日漏洞探索: LLMs 由大量文本和代码训练而成, 由于其庞大的参数空间和复杂性, 存在发现并利用尚未被发现的漏洞的风险, 这可能被恶意用户利用。LLMs 的参数空间通常非常大, 这意味着模型可以学习大量的信息。然而, 也意味着模型更容易受到攻击。恶意用户可以利用 LLM 的复杂性来发现和利用尚未被发现的漏洞。以红队为代表的漏洞发现工作依赖大量的人力和计算成本, 如何更有效、低成本地探索零日漏洞仍待解决。目前已有一些使用自动化策略进行越狱测试的工作[263-265], 基于遗传算法或贪婪梯度优化等算法提高零日漏洞挖掘的效率。

新型隐私保护要求: 针对 LLM 隐私问题的研究表明, 相较于传统 NLP 和 CV 模型, 规模更大的语言模型倾向于记忆更多的训练数据(包含大量的用户隐私)的信息。更有效的数据隐私过滤和隐私保护的训练方法以及模型隐私输出限制措施是 LLM 安全训练和使用的要求。一种主动过滤的策略[266]使用小模型将文本脱敏, 或基于差分隐私为文本添加扰动[267], 再将这些文本输入到 LLM, 实现服务范围内的用户隐私保护。

新的防御方法: 大语言模型相较于过去的深度学习算法存在规模更大、黑盒、多模态耦合等问题, 传统的针对 NLP 算法的防御措施可能不适用于 LLM 模型。针对 LLM 特有的幻觉问题, 一些工作考虑用

知识反馈强化学习增强其表达内部知识的能力以缓解幻觉[268], 但目前暂无有效的解决措施, 亟须提出针对 LLM 的新型防御方法。

生成内容检测: LLMs 可能会被用于生成不良、有害、歧视性或违法内容。这种滥用可能涉及虚假信息、仇恨言论、人身攻击等。此外, 教育相关领域的滥用可能引起模糊的版权问题和评估体系落后问题。对 LLMs 生成内容和人类生成内容的有效检测[230-231]是保障 LLM 规范使用的前提。

可解释性和透明度研究: 现有的高性能商用大语言模型通常是黑盒模型, 其内部工作原理难以理解, 开发者将其文本生成过程或决策行为作为商业机密维护。这使得难以审查模型的行为, 增加了潜在的安全风险。现有的可解释研究在黑盒模型上效果不佳, 难以对 LLMs 这类新型的大型模型进行解释。基于图导航特殊任务的方法为 LLM 在对抗样本上决策的可解释性提供了一个解决方案[269], 但仍然没有对 LLM 输出结果的有效解释方法。

新的法律和政策约束: LLM 的应用可能会触及法律和道德问题(版权、隐私和内容合法性等), 需要结合各国国情和 LLM 的特点制定新的政策和法律, 以指导 LLM 的安全、健康发展。

自动化与智能化应用: LLM 的出现使得许多自然语言处理任务能够实现自动化和智能化, 大量基于 LLM 的应用产品在各行各业发展, 提高了各种领域的工作效率和生产力。如何设置有效的提示信息, 或实现 LLM 之间的有机组合, 是最大化基于 LLM 的智能应用的要求。开发人员可以通过多个代理构建 LLM 应用程序, 实现数学推理、程序设计、知识问答等应用[2]。

数据枯竭问题与新型模型结构优化: LLMs 的训练和微调对齐技术依赖大量高质量的文本数据。目前互联网中的数据被 LM 生成的数据内容冲击, 未来的 LLMs 训练将会面临数据枯竭问题。考虑到 LLM 出色的文本生成能力, 一些工作考虑用 LLM 生成数据集替换传统人工收集和处理的数据[270], 但合成数据质量与真实数据仍有差距。研究重心应从单一的数据量扩充向新型模型结构设计的方向转移。

11 结论

ChatGPT 等大型语言模型的研究推动了深度学习在研究和产业界的高速发展, 但目前已有研究发现 LLMs 的使用过程中存在安全隐患。本文考虑了传统深度学习方法中存在的风险, 以及 LLMs 引入的新问题, 分析了 LLMs 开发和使用全周期中可能引入的安全威胁, 并总结了 LLMs 的幻觉、欺骗、毒性、隐私、偏见、鲁棒性和道德问题, 以及三类攻击威胁。讨论了应对这些安全威胁的防御和检测方法。最后, 我们还整理了目前国内外主流的 LLMs 框架平台, 并提出了 10 条机遇和挑战, 为相关产业和科研工作者探究新兴系统的攻击和漏洞等安全研究提供支持, 推动针对 LLMs 的新型安全评估和防御检测的研究。

参考文献

[1] Ziems N, Yu W H, Zhang Z H, et al. Large Language Models Are Built-in Autoregressive Search Engines[C]. *Findings of the Association for Computational Linguistics*: *ACL 2023*, 2023: 2666-2678.

[2] Wu Q Y, Bansal G, Zhang J Y, et al. AutoGen: Enabling Next-Gen LLM Applications *via* Multi-Agent Conversation[EB/OL]. 2023: arXiv: 2308.08155. https://arxiv.org/abs/2308.08155

[3] Chen L, Sinavski O, Hünermann J, et al. Driving with LLMS: Fusing Object-Level Vector Modality for Explainable Autonomous Driving[EB/OL]. 2023: arXiv: 2310.01957. https://arxiv.org/abs/2310.01957

[4] Wang S, Zhao Z H, Ouyang X, et al. ChatCAD: Interactive Computer-Aided Diagnosis on Medical Image Using Large Language Models[EB/OL]. 2023: arXiv: 2302.07257. https://arxiv.org/abs/2302.07257

[5] Dai W L, Liu Z H, Ji Z W, et al. Plausible may Not Be Faithful: Probing Object Hallucination in Vision-Language Pre-Training[C]. *The 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023: 2136-2148.

[6] Wallace E, Feng S, Kandpal N, et al. Universal Adversarial Triggers for Attacking and Analyzing NLP[C]. *The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), 2019: 2153-2162.

[7] Carlini N, Liu C, Erlingsson L, et al. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks[C]. *USENIX Security Symposium*, 2018

[8] Hendrycks D, Burns C, Basart S, et al. Aligning AI with Shared Human Values[C]. *The 9th Annual Conference on International Conference on Learning Representations*, 2021.

[9] Wang B X, Chen W X, Pei H Z, et al. DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models[EB/OL]. 2023: arXiv: 2306.11698. https://arxiv.org/abs/2306.11698

[10] Cui S Y, Zhang Z Y, Chen Y L, et al. FFT: Towards Harmlessness Evaluation and Analysis for LLMS with Factuality, Fairness, Toxicity[EB/OL]. 2023: arXiv: 2311.18580. https://arxiv.org/abs/2311.18580

[11] Rawte V, Sheth A, Das A. A Survey of Hallucination in Large Foundation Models[EB/OL]. 2023: arXiv: 2309.05922. https:// arxiv.org/abs/2309.05922

[12] Li Y J, Du M N, Song R, et al. A Survey on Fairness in Large Language Models[EB/OL]. 2023: arXiv: 2308.10149. https://arxiv.org/abs/2308.10149

[13] Li H R, Chen Y L, Luo J L, et al. Privacy in Large Language Models: Attacks, Defenses and Future Directions[EB/OL]. 2023: arXiv: 2310.10383. https://arxiv.org/abs/2310.10383

[14] Peters M, Neumann M, Iyyer M, et al. Deep Contextualized Word Representations[C]. *The 2018 Conference of the North American Chapter Ofthe Association for Computational Linguistics*: *Human LanguageTechnologies*, *Volume 1* (*Long Papers*), 2018: 2227-2237.

[15] Zhu Y K, Kiros R, Zemel R, et al. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books[C]. *2015 IEEE International Conference on Computer Vision* (*ICCV*), 2016: 19-27.

[16] Radford A, Wu J, Child R, et al. Language Models Are Unsupervised Multitask Learners A. Radford, J. Wu, R. Child, et al. Language models are unsupervised multitask learners[J]. OpenAI blog, 2019, 1(8): 9., 2019

[17] Gao L, Biderman S, Black S, et al. The Pile: An 800GB Dataset of Diverse Text for Language Modeling[EB/OL]. 2020: arXiv: 2101.00027. https://arxiv.org/abs/2101.00027.

[18] Laurencon H, Saulnier L, Wang T, et al. The bigscience roots corpus: A 1.6 tb composite multilingual dataset[J]. *Advances in Neural Information Processing Systems*, 2022, 35: 31809-31826.

[19] Wang T, Roberts A, Hesslow D, et al. What Language Model Architecture and Pretraining Objective Work Best for Zero-Shot Generalization? [EB/OL]. 2022: arXiv: 2204.05832. https://arxiv.org/abs/2204.05832

[20] Ouyang L, Wu J, Xu J, et al. Training Language Models to Follow Instructions with Human Feedback[C]. *The 36th International Conference on Neural Information Processing Systems*, 2022: 27730-27744.

[21] Wei J, Bosma M, Zhao V Y, et al. Finetuned Language Models Are Zero-Shot Learners[EB/OL]. 2021: arXiv: 2109.01652. https:// arxiv.org/abs/2109.01652.

[22] Smith J T H, Warrington A, Linderman S W. Simplified State Space Layers for Sequence Modeling[EB/OL]. 2022: arXiv: 2208.04933. https://arxiv.org/abs/2208.04933.

[23] Pan L M, Saxon M, Xu W D, et al. Automatically Correcting Large Language Models: Surveying the Landscape of Diverse Self-Correction Strategies[EB/OL]. 2023: arXiv: 2308.03188. https://arxiv.org/abs/2308.03188.

[24] Tonmoy S M T I, Mehedi Zaman S M, Jain V, et al. A Comprehensive Survey of Hallucination Mitigation Techniques in Large Language Models[EB/OL]. 2024: arXiv: 2401.01313. https://arxiv.org/abs/2401.01313.

[25] Christiano P, Leike J, Brown T B, et al. Deep Reinforcement Learning from Human Preferences[EB/OL]. 2017: arXiv: 1706.03741. https://arxiv.org/abs/1706.03741.

[26] Ganguli D, Askell A, Schiefer N, et al. The Capacity for Moral Self-Correction in Large Language Models[EB/OL]. 2023: arXiv: 2302.07459. https://arxiv.org/abs/2302.07459.

[27] Houlsby N, Giurgiu A, Jastrzebski S, et al. Parameter-Efficient Transfer Learning for NLP[C]. *International Conference on Machine Learning*, 2019

[28] He J X, Zhou C T, Ma X Z, et al. Towards a Unified View of Parameter-Efficient Transfer Learning[EB/OL]. 2021: arXiv: 2110.04366. https://arxiv.org/abs/2110.04366.

[29] Li X L, Liang P. Prefix-Tuning: Optimizing Continuous Prompts for Generation[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 2021: 4582-4597.

[30] Lester B, Al-Rfou R, Constant N. The Power of Scale for Parameter-Efficient Prompt Tuning[C]. *The 2021 Conference on Empirical Methods in Natural Language Processing*, 2021: 3045-3059.

[31] Jiang Z B, Xu F F, Araki J, et al. How Can we Know What Language Models Know?[J]. *Transactions of the Association for Computational Linguistics*, 2020, 8: 423-438.

[32] Hu E J, Shen Y L, Wallis P, et al. LoRA: Low-Rank Adaptation of Large Language Models[EB/OL]. 2021: arXiv: 2106.09685. https://arxiv.org/abs/2106.09685.

[33] Chang Y P, Wang X, Wang J D, et al. A Survey on Evaluation of Large Language Models[EB/OL]. 2023: arXiv: 2307.03109. https://arxiv.org/abs/2307.03109.

[34] Liu Y, Yao Y S, Ton J F, et al. Trustworthy LLMS: A Survey and Guideline for Evaluating Large Language Models' Alignment[EB/OL]. 2023: arXiv: 2308.05374. https://arxiv.org/abs/2308.05374.

[35] Fan M Y, Wang C Y, Chen C, et al. On the Trustworthiness Landscape of State-of-the-Art Generative Models: A Survey and Outlook[EB/OL]. 2023: arXiv: 2307.16680. https://arxiv.org/abs/2307.16680.

[36] Yao Y F, Duan J H, Xu K D, et al. A Survey on Large Language Model (LLM) Security and Privacy: The Good, the Bad, and the Ugly[EB/OL]. 2023: arXiv: 2312.02003. https://arxiv.org/abs/2312.02003.

[37] Guo Z S, Jin R R, Liu C, et al. Evaluating Large Language Models: A Comprehensive Survey[EB/OL]. 2023: arXiv: 2310.19736. https://arxiv.org/abs/2310.19736.

[38] Zhuang Z Y, Chen Q G, Ma L X, et al. Through the Lens of Core Competency: Survey on Evaluation of Large Language Models[EB/OL]. 2023: arXiv: 2308.07902. https://arxiv.org/abs/2308.07902.

[39] Mao R, Chen G Y, Zhang X L, et al. GPTEval: A Survey on Assessments of ChatGPT and GPT-4[EB/OL]. 2023: arXiv: 2308.12488. https://arxiv.org/abs/2308.12488.

[40] Hadi M U, Tashi A, Qureshi R, et al. Large Language Models: A Comprehensive Survey of Its Applications, Challenges, Limitations, and Future Prospects M. U. Hadi, R. Qureshi, A. Shah, et al. Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects[J]. Authorea Preprints, 2023.

[41] Zhang Y, Li Y F, Cui L Y, et al. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models[EB/OL]. 2023: arXiv: 2309.01219. https://arxiv.org/abs/2309.01219.

[42] Ji Z W, Lee N, Frieske R, et al. Survey of Hallucination in Natural Language Generation[J]. *ACM Computing Surveys*, 2023, 55(12): 1-38.

[43] Huang L, Yu W J, Ma W T, et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions[EB/OL]. 2023: arXiv: 2311.05232. https://arxiv.

org/abs/2311.05232.

[44] Park P S, Goldstein S, O'Gara A, et al. AI Deception: A Survey of Examples, Risks, and Potential Solutions[J]. *Patterns*, 2024, 5(5): 100988.

[45] Shayegani E, Al Mamun M A, Fu Y, et al. Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks [EB/OL]. 2023: arXiv: 2310.10844. https://arxiv.org/abs/2310.10844.

[46] Gallegos I O, Rossi R A, Barrow J, et al. Bias and Fairness in Large Language Models: A Survey[EB/OL]. 2023: arXiv: 2309.00770. https://arxiv.org/abs/2309.00770.

[47] Meade N, Poole-Dayan E, Reddy S. An Empirical Survey of the Effectiveness of Debiasing Techniques for Pre-Trained Language Models[C]. *The 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2022: 1878-1898.

[48] Bender E M, Gebru T, McMillan-Major A, et al. On the Dangers of Stochastic Parrots: Can Language Models Be too Big? [C]. *The 2021 ACM Conference on Fairness*, *Accountability*, *and Transparency*, 2021: 610-623.

[49] Mallen A, Asai A, Zhong V, et al. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2023: 9802-9822.

[50] Lin S, Hilton J, Evans O. TruthfulQA: Measuring how Models Mimic Human Falsehoods[C]. *The 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2022: 3214-3252.

[51] Carlini N, Tramèr F, Wallace E, et al. Extracting Training Data from Large Language Models[C]. *USENIX Security Symposium*, 2020.

[52] Hernandez D, Brown T, Conerly T, et al. Scaling Laws and Interpretability of Learning from Repeated Data[EB/OL]. 2022: arXiv: 2205.10487. https://arxiv.org/abs/2205.10487.

[53] Li D L, Rawat A S, Zaheer M, et al. Large Language Models with Controllable Working Memory[C]. *Findings of the Association for Computational Linguistics*: *ACL 2023*, 2023: 1774-1793.

[54] Kandpal N, Deng H K, Roberts A, et al. Large Language Models Struggle to Learn Long-Tail Knowledge[C]. *The 40th International Conference on Machine Learning*, 2023: 15696-15707.

[55] Li S B, Li X G, Shang L F, et al. How Pre-Trained Language Models Capture Factual Knowledge? A Causal-Inspired Analysis[C]. *Findings of the Association for Computational Linguistics*: *ACL 2022*, 2022: 1720-1732.

[56] Kang C, Choi J. Impact of Co-Occurrence on Factual Knowledge of Large Language Models[C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2023*, 2023: 7721-7735.

[57] Liu N F, Lin K, Hewitt J, et al. Lost in the Middle: How Language Models Use Long Contexts[EB/OL]. 2023: arXiv: 2307.03172. https://arxiv.org/abs/2307.03172.

[58] Li Z C, Zhang S T, Zhao H, et al. BatGPT: A Bidirectional Autoregessive Talker from Generative Pre-Trained Transformer [EB/OL]. 2023: arXiv: 2307.00360. https://arxiv.org/abs/2307.00360.

[59] Liu B B, Ash J T, Goel S, et al. Exposing Attention Glitches with Flip-Flop Language Modeling[EB/OL]. 2023: arXiv: 2306.00946. https://arxiv.org/abs/2306.00946.

[60] Chiang D, Cholak P. Overcoming a Theoretical Limitation of Self-Attention[C]. *The 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2022: 7654-7664.

[61] Ranzato M, Chopra S, Auli M, et al. Sequence Level Training with Recurrent Neural Networks[EB/OL]. 2015: arXiv: 1511.06732. https://arxiv.org/abs/1511.06732.

[62] Wang C J, Sennrich R. On Exposure Bias, Hallucination and Domain Shift in Neural Machine Translation[C]. *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020: 3544-3552.

[63] Schulman J. Reinforcement learning from human feedback: progress and challenges[J]. *Berkley Electrical Engineering and Computer Sciences*, 2023.

[64] Burns C, Ye H T, Klein D, et al. Discovering Latent Knowledge in Language Models without Supervision[EB/OL]. 2022: arXiv: 2212.03827. https://arxiv.org/abs/2212.03827.

[65] Azaria A, Mitchell T. The Internal State of an LLM Knows when It's Lying[EB/OL]. 2023: arXiv: 2304.13734. https://arxiv.org/abs/2304.13734.

[66] Cotra A. Why AI alignment could be hard with modern deep learning[J]. *Cold Takes*, 2021.

[67] Sharma M, Tong M, Korbak T, et al. Towards Understanding Sycophancy in Language Models[EB/OL]. 2023: arXiv: 2310. 13548. https://arxiv.org/abs/2310.13548.

[68] Holtzman A, Buys J, Du L, et al. The Curious Case of Neural Text Degeneration[EB/OL]. 2019: arXiv: 1904.09751. https://arxiv.org/abs/1904.09751.

[69] Dziri N, Madotto A, Zaïane O, et al. Neural Path Hunter: Reducing Hallucination in Dialogue Systems *via* Path Grounding[C]. *The 2021 Conference on Empirical Methods in Natural Language Processing*, 2021: 2197-2214.

[70] Chuang Y S, Xie Y J, Luo H Y, et al. DoLa: Decoding by Contrasting Layers Improves Factuality in Large Language Models[EB/OL]. 2023: arXiv: 2309.03883. https://arxiv.org/abs/2309.03883.

[71] Aksitov R, Chang C C, Reitter D, et al. Characterizing Attribution and Fluency Tradeoffs for Retrieval-Augmented Large Language Models[EB/OL]. 2023: arXiv: 2302.05578. https://arxiv.org/abs/2302.05578.

[72] Abdollahzadeh M, Malekzadeh T, Teo C T H, et al. A Survey on Generative Modeling with Limited Data, few Shots, and Zero Shot[EB/OL]. 2023: arXiv: 2307.14397. https://arxiv.org/abs/2307.14397.

[73] Shi W J, Han X C, Lewis M, et al. Trusting Your Evidence: Hallucinate less with Context-Aware Decoding[EB/OL]. 2023: arXiv: 2305.14739. https://arxiv.org/abs/2305.14739.

[74] Chen Y J, Liu Y J, Meng F D, et al. Improving Translation Faithfulness of Large Language Models *via* Augmenting Instructions [EB/OL]. 2023: arXiv: 2308.12674. https://arxiv.org/abs/2308.

12674.

[75] Liu Y J, Zeng X F, Meng F D, et al. Instruction Position Matters in Sequence Generation with Large Language Models[EB/OL]. 2023: arXiv: 2308.12097. https://arxiv.org/abs/2308.12097.

[76] Yang Z L, Dai Z H, Salakhutdinov R, et al. Breaking the Softmax Bottleneck: A High-Rank RNN Language Model[EB/OL]. 2017: arXiv: 1711.03953. https://arxiv.org/abs/1711.03953.

[77] Chang H S, McCallum A. Softmax Bottleneck Makes Language Models Unable to Represent Multi-Mode Word Distributions[C]. *The 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2022: 8048-8073.

[78] Bang Y J, Cahyawijaya S, Lee N, et al. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity[EB/OL]. 2023: arXiv: 2302.04023. https://arxiv.org/abs/2302.04023.

[79] Yao J Y, Ning K P, Liu Z H, et al. LLM Lies: Hallucinations Are Not Bugs, but Features as Adversarial Examples[EB/OL]. 2023: arXiv: 2310.01469. https://arxiv.org/abs/2310.01469.

[80] Yu X D, Cheng H, Liu X D, et al. ReEval: Automatic Hallucination Evaluation for Retrieval-Augmented Large Language Models *via* Transferable Adversarial Attacks[EB/OL]. 2023: arXiv: 2310.12516. https://arxiv.org/abs/2310.12516.

[81] Vu T, Iyyer M, Wang X Z, et al. FreshLLMs: Refreshing Large Language Models with Search Engine Augmentation[EB/OL]. 2023: arXiv: 2310.03214. https://arxiv.org/abs/2310.03214.

[82] Qiu Y F, Ziser Y, Korhonen A, et al. Detecting and Mitigating Hallucinations in Multilingual Summarisation[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 8914-8932.

[83] O'Gara A. Hoodwinked: Deception and Cooperation in a Text-Based Game for Language Models[EB/OL]. 2023: arXiv: 2308.01404. https://arxiv.org/abs/2308.01404.

[84] Pan A, Chan J S, Zou A, et al. Do the Rewards Justify the Means? Measuring Trade-Offs between Rewards and Ethical Behavior in the MACHIAVELLI Benchmark[C]. *The 40th International Conference on Machine Learning*, 2023: 26837-26867.

[85] Scherrer N, Shi C, Feder A, et al. Evaluating the Moral Beliefs Encoded in LLMS[EB/OL]. 2023: arXiv: 2307.14324. https://arxiv.org/ abs/2307.14324.

[86] Hagendorff T. Deception Abilities Emerged in Large Language Models[EB/OL]. 2023: arXiv: 2307.16513. https://arxiv.org/abs/2307.16513.

[87] OpenAI. GPT-4 technical report[EB/OL]. 2023: ArXiv Preprint ArXiv:2303.08774.

[88] Gehman S, Gururangan S, Sap M, et al. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models[C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2020*, 2020: 3356-3369.

[89] McGuffie K, Newhouse A. The Radicalization Risks of GPT-3 and Advanced Neural Language Models[EB/OL]. 2020: arXiv: 2009.06807. https://arxiv.org/abs/2009.06807.

[90] Liang P, Bommasani R, Lee T, et al. Holistic Evaluation of Language Models[EB/OL]. 2022: arXiv: 2211.09110. https://arxiv.org/abs/2211.09110.

[91] Shi Y D, Li P J, Yin C C, et al. PromptAttack: Prompt-Based Attack forLanguage Models viaGradient Search[C]. Natural Language Processing and Chinese Computing. Cham: Springer, 2022: 682-693.

[92] Deng B Y, Wang W J, Feng F L, et al. Attack Prompt Generation for Red Teaming and Defending Large Language Models[C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2023*, 2023: 2176-2189.

[93] Welbl J, Glaese A, Uesato J, et al. Challenges in Detoxifying Language Models[EB/OL]. 2021: arXiv: 2109.07445. https://arxiv.org/abs/2109.07445.

[94] Li H R, Guo D D, Fan W, et al. Multi-Step Jailbreaking Privacy Attacks on ChatGPT[C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2023*, 2023: 4138-4153.

[95] Zhang C Y, Ippolito D, Lee K, et al. Counterfactual Memorization in Neural Language Models[EB/OL]. 2021: arXiv: 2112.12938. https://arxiv.org/abs/2112.12938.

[96] Mireshghallah F, Uniyal A, Wang T H, et al. An Empirical Analysis of Memorization in Fine-Tuned Autoregressive Language Models[C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 1816-1826.

[97] Staab R, Vero M, Balunović M, et al. Beyond Memorization: Violating Privacy *via* Inference with Large Language Models[EB/OL]. 2023: arXiv: 2310.07298. https://arxiv.org/abs/2310.07298.

[98] Lukas N, Salem A, Sim R, et al. Analyzing Leakage of Personally Identifiable Information in Language Models[C]. *2023 IEEE Symposium on Security and Privacy* (*SP*), 2023: 346-363.

[99] Huang J, Shao H Y, Chang K C. Are Large Pre-Trained Language Models Leaking Your Personal Information? [C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2022*, 2022: 2038-2047.

[100] Shao H Y, Huang J, Zheng S, et al. Quantifying Association Capabilities of Large Language Models and Its Implications on Privacy Leakage[EB/OL]. 2023: arXiv: 2305.12707. https://arxiv.org/abs/2305.12707.

[101] Carlini N, Ippolito D, Jagielski M, et al. Quantifying Memorization across Neural Language Models[EB/OL]. 2022: arXiv: 2202.07646. https://arxiv.org/abs/2202.07646.

[102] Dhamala J, Sun T, Kumar V, et al. BOLD: Dataset and Metrics for Measuring Biases in Open-Ended Language Generation[C]. *The 2021 ACM Conference on Fairness*, *Accountability*, *and Transparency*, 2021: 862-872.

[103] Abid A, Farooqi M, Zou J. Persistent Anti-Muslim Bias in Large Language Models[C]. *The 2021 AAAI/ACM Conference on AI*, *Ethics*, *and Society*, 2021: 298-306.

[104] Nadeem M, Bethke A, Reddy S. StereoSet: Measuring Stereotypical Bias in Pretrained Language Models[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 2021: 5356-5371.

[105] Nangia N, Vania C, Bhalerao R, et al. CrowS-Pairs: A Challenge Dataset for Measuring Social Biases in Masked Language Models[C]. *The 2020 Conference on Empirical Methods in Natural*

Language Processing (*EMNLP*), 2020: 1953-1967.

[106] Parrish A, Chen A, Nangia N, et al. BBQ: A Hand-Built Bias Benchmark for Question Answering[C]. *Findings of the Association for Computational Linguistics*: *ACL 2022*, 2022: 2086-2105.

[107] Blodgett S L, Barocas S, Daumé III H, et al. Language (Technology) Is Power: A Critical Survey of "Bias" in NLP[C]. *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020: 5454-5476.

[108] Blodgett S L, Lopez G, Olteanu A, et al. Stereotyping Norwegian Salmon: An Inventory of Pitfalls in Fairness Benchmark Datasets[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 2021: 1004-1015.

[109] May C, Wang A, Bordia S, et al. On Measuring Social Biases in Sentence Encoders[C]. *The 2019 Conference of the North*, 2019: 622-628.

[110] Wang B X, Xu C J, Wang S H, et al. Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models[EB/OL]. 2021: arXiv: 2111.02840. https://arxiv.org/abs/2111.02840.

[111] Nie Y X, Williams A, Dinan E, et al. Adversarial NLI: A New Benchmark for Natural Language Understanding[C]. *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020: 4885-4901.

[112] Wang J D, Hu X X, Hou W X, et al. On the Robustness of ChatGPT: An Adversarial and Out-of-Distribution Perspective[EB/OL]. 2023: arXiv: 2302.12095. https://arxiv.org/abs/2302.12095.

[113] Subhash V, Bialas A, Pan W W, et al. Why Do Universal Adversarial Attacks Work on Large Language Models? : Geometry might Be the Answer[EB/OL]. 2023: arXiv: 2309.00254. https://arxiv.org/abs/2309.00254.

[114] Zhong Q H, Ding L, Liu J H, et al. Can ChatGPT Understand Too? A Comparative Study on ChatGPT and Fine-Tuned BERT[EB/OL]. 2023: arXiv: 2302.10198. https://arxiv.org/abs/2302.10198.

[115] Suzgun M, Scales N, Schärli N, et al. Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them[C]. *Findings of the Association for Computational Linguistics*: *ACL 2023*, 2023: 13003-13051.

[116] Shi F, Suzgun M, Freitag M, et al. Language Models Are Multilingual Chain-of-Thought Reasoners[EB/OL]. 2022: arXiv: 2210.03057. https://arxiv.org/abs/2210.03057.

[117] Berglund L, Tong M, Kaufmann M, et al. The Reversal Curse: LLMS Trained on "a Is B" Fail to Learn "B Is a"[EB/OL]. 2023: arXiv: 2309.12288. https://arxiv.org/abs/2309.12288.

[118] Côté M A, Kádár Á, Yuan X D, et al. TextWorld: A Learning Environment for Text-Based Games[C]. Computer Games. Cham: Springer, 2019: 41-75.

[119] Forbes M, Hwang J D, Shwartz V, et al. Social Chemistry 101: Learning to Reason about Social and Moral Norms[C]. *The 2020 Conference on Empirical Methods in Natural Language Processing* (*EMNLP*), 2020: 653-670.

[120] Jin Z J, Levine S, Gonzalez F, et al. When to Make Exceptions: Exploring Language Models as Accounts of Human Moral Judgment[EB/OL]. 2022: arXiv: 2210.01478. https://arxiv.org/abs/2210.01478.

[121] Perez F, Ribeiro I. Ignore Previous Prompt: Attack Techniques for Language Models[EB/OL]. 2022: arXiv: 2211.09527. https://arxiv.org/abs/2211.09527.

[122] Wei A, Haghtalab N, Steinhardt J. Jailbroken: How Does LLM Safety Training Fail? [EB/OL]. 2023: arXiv: 2307.02483. https://arxiv.org/abs/2307.02483.

[123] Ding P, Kuang J, Ma D, et al. A Wolf in Sheep′s Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Ea-sily[EB/OL]. 2023: ArXiv Preprint ArXiv:2311.08268.

[124] Huang Y, Gupta S, Xia M Z, et al. Catastrophic Jailbreak of Open-Source LLMS *via* Exploiting Generation[EB/OL]. 2023: arXiv: 2310.06987. https://arxiv.org/abs/2310.06987.

[125] Li X, Zhou Z K, Zhu J N, et al. DeepInception: Hypnotize Large Language Model to Be Jailbreaker[EB/OL]. 2023: arXiv: 2311.03191. https://arxiv.org/abs/2311.03191.

[126] Yu J H, Lin X W, Yu Z, et al. GPTFUZZER: Red Teaming Large Language Models with Auto-Generated Jailbreak Prompts[EB/OL]. 2023: arXiv: 2309.10253. https://arxiv.org/abs/2309.10253.

[127] Wei Z M, Wang Y F, Li A, et al. Jailbreak and Guard Aligned Language Models with Only few In-Context Demonstrations [EB/OL]. 2023: arXiv: 2310.06387. https://arxiv.org/abs/2310.06387.

[128] Chao P, Robey A, Dobriban E, et al. Jailbreaking Black Box Large Language Models in Twenty Queries[EB/OL]. 2023: arXiv: 2310.08419. https://arxiv.org/abs/2310.08419.

[129] Liu Y, Deng G L, Li Y K, et al. Prompt Injection Attack Against LLM-Integrated Applications[EB/OL]. 2023: arXiv: 2306.05499. https://arxiv.org/abs/2306.05499.

[130] Wang H R, Shu K. Trojan Activation Attack: Red-Teaming Large Language Models Using Activation Steering for Safety-Alignment [EB/OL]. 2023: arXiv: 2311.09433. https://arxiv.org/abs/2311.09433.

[131] Rando J, Tramèr F. Universal Jailbreak Backdoors from Poisoned Human Feedback[EB/OL]. 2023: arXiv: 2311.14455. https://arxiv.org/abs/2311.14455.

[132] Cao Y P, Cao B C, Chen J H. Stealthy and Persistent Unalignment on Large Language Models *via* Backdoor Injections[EB/OL]. 2023: arXiv: 2312.00027. https://arxiv.org/abs/2312.00027.

[133] Sheng X, Li Z C, Han Z Y, et al. Punctuation Matters! Stealthy Backdoor Attack for Language Models[M]. *Natural Language Processing and Chinese Computing*. Cham: Springer Nature Switzerland, 2023: 524-536.

[134] Wen J X, Ke P, Sun H, et al. Unveiling the Implicit Toxicity in Large Language Models[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 1322-1338.

[135] Greshake K, Abdelnabi S, Mishra S, et al. Not What You've Signed up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection[EB/OL]. 2023: arXiv: 2302.12173. https://arxiv.org/abs/2302.12173.

[136] Yan J, Yadav V, Li S Y, et al. Backdooring Instruction-Tuned Large Language Models with Virtual Prompt Injection[EB/OL]. 2023:

arXiv: 2307.16888. https://arxiv.org/abs/2307.16888.

[137] Ray P P. ChatGPT: A Comprehensive Review on Background, Applications, Key Challenges, Bias, Ethics, Limitations and Future Scope[J]. *Internet of Things and Cyber-Physical Systems*, 2023, 3: 121-154.

[138] Nah F F, Zheng R L, Cai J Y, et al. Generative AI and ChatGPT: applications, challenges, and AI-human collaboration[J]. *Journal of Information Technology Case and Application Research*, 2023, 25(3): 277-304.

[139] Sullivan M, Kelly A, McLaughlan P. ChatGPT in higher education: Considerations for academic integrity and student learning[J]. *Journal of Applied Learning and Teaching*, 2023, 6(1): 1-10.

[140] Cotton D R E, Cotton P A, Shipway J R. Chatting and cheating: Ensuring academic integrity in the era of ChatGPT[J]. *Innovations in Education and Teaching International*, 2024, 61(2): 228-239.

[141] Zhai X M. ChatGPT User Experience: Implications for Education[J]. *SSRN Electronic Journal*, 2022.

[142] Sallam M. The utility of ChatGPT as an example of large language models in healthcare education, research and practice: Systematic review on the future perspectives and potential limitations[J]. *medRxiv*, 2023.

[143] Kang H Q, Ni J T, Yao H X. Ever: Mitigating Hallucination in Large Language Models through Real-Time Verification and Rectification[EB/OL]. 2023: arXiv: 2311.09114. https://arxiv.org/abs/2311.09114.

[144] Peng B L, Galley M, He P C, et al. Check Your Facts and Try Again: Improving Large Language Models with External Knowledge and Automated Feedback[EB/OL]. 2023: arXiv: 2302.12813. https://arxiv.org/abs/2302.12813.

[145] Cao H J, An Z W, Feng J Z, et al. A Step Closer to Comprehensive Answers: Constrained Multi-Stage Question Decomposition with Large Language Models[EB/OL]. 2023: arXiv: 2311.07491. https://arxiv.org/abs/2311.07491.

[146] Gao L Y, Dai Z Y, Pasupat P, et al. RARR: Researching and Revising What Language Models Say, Using Language Models[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2023: 16477-16508.

[147] Rawte V, Chakraborty S, Pathak A, et al. The Troubling Emergence of Hallucination in Large Language Models — an Extensive Definition, Quantification, and Prescriptive Remediations[EB/OL]. 2023: arXiv: 2310.04988. https://arxiv.org/abs/2310.04988.

[148] Lewis P, Perez E, Piktus A, et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks[C]. *The 34th International Conference on Neural Information Processing Systems*, 2020: 9459-9474.

[149] Cheng D X, Huang S H, Bi J Y, et al. UPRISE: Universal Prompt Retrieval for Improving Zero-Shot Evaluation[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 12318-12337.

[150] Jones E, Palangi H, Simões C, et al. Teaching Language Models to Hallucinate less with Synthetic Tasks[EB/OL]. 2023: arXiv: 2310.06827. https://arxiv.org/abs/2310.06827.

[151] Li K, Patel O, Viégas F, et al. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model[EB/OL]. 2023: arXiv: 2306.03341. https://arxiv.org/abs/2306.03341.

[152] Guan X Y, Liu Y J, Lin H Y, et al. Mitigating Large Language Model Hallucinations *via* Autonomous Knowledge Graph-Based Retrofitting[EB/OL]. 2023: arXiv: 2311.13314. https://arxiv.org/abs/2311.13314.

[153] Ji Z W, Liu Z H, Lee N, et al. RHO: Reducing Hallucination in Open-Domain Dialogues with Knowledge Grounding[C]. *Findings of the Association for Computational Linguistics*: *ACL 2023*, 2023: 4504-4522.

[154] Fatahi Bayat F, Qian K, Han B, et al. FLEEK: Factual Error Detection and Correction with Evidence Retrieved from External Knowledge[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*: *System Demonstrations*, 2023: 124-130.

[155] Yoon S, Yoon E, Yoon H S, et al. Information-Theoretic Text Hallucination Reduction for Video-Grounded Dialogue[C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 4182-4193.

[156] Elaraby M, Lu M Y, Dunn J, et al. Halo: Estimation and Reduction of Hallucinations in Open-Source Weak Large Language Models[EB/OL]. 2023: arXiv: 2308.11764. https://arxiv.org/abs/2308.11764.

[157] Köksal A, Aksitov R, Chang C C. Hallucination Augmented Recitations for Language Models[EB/OL]. 2023: arXiv: 2311. 07424. https://arxiv.org/abs/2311.07424.

[158] Tian K, Mitchell E, Yao H X, et al. Fine-Tuning Language Models for Factuality[EB/OL]. 2023: arXiv: 2311.08401. https://arxiv.org/abs/2311.08401.

[159] Razumovskaia E, Vulić I, Marković P, et al. Dial BeInfo for Faithfulness: Improving Factuality of Information-Seeking Dialogue *via* Behavioural Fine-Tuning[EB/OL]. 2023: arXiv: 2311.09800. https://arxiv.org/abs/2311.09800.

[160] Zhang H N, Diao S Z, Lin Y, et al. R-Tuning: Instructing Large Language Models to Say 'I Don't Know'[EB/OL]. 2023: arXiv: 2311.09677. https://arxiv.org/abs/2311.09677.

[161] Qiu Y F, Embar V, Cohen S B, et al. Think while You Write: Hypothesis Verification Promotes Faithful Knowledge-to-Text Generation[EB/OL]. 2023: arXiv: 2311.09467. https://arxiv.org/abs/2311.09467.

[162] Abadi M, Chu A, Goodfellow I, et al. Deep Learning with Differential Privacy[C]. *The 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016: 308-318.

[163] Yu D, Gopi S, Kulkarni J, et al. Selective Pre-Training for Private Fine-Tuning[EB/OL]. 2023: arXiv: 2305.13865. https://arxiv.org/abs/2305.13865.

[164] Igamberdiev T, Habernal I. DP-BART for Privatized Text Rewriting under Local Differential Privacy[C]. *Findings of the Association for Computational Linguistics*: *ACL 2023*, 2023: 13914-13934.

[165] Feyisetan O, Balle B, Drake T, et al. Privacy- and Utility-Preserving Textual Analysis *via* Calibrated Multivariate Perturbations[C]. *The 13th International Conference on Web Search and Data Mining*, 2020: 178-186.

[166] Alvim M, Chatzikokolakis K, Palamidessi C, et al. Invited Paper: Local Differential Privacy on Metric Spaces: Optimizing the

Trade-off with Utility[C]. *2018 IEEE 31st Computer Security Foundations Symposium* (*CSF*), 2018: 262-267.

[167] Qu C, Kong W Z, Yang L, et al. Natural Language Understanding with Privacy-Preserving BERT[C]. *The 30th ACM International Conference on Information & Knowledge Management*, 2021: 1488-1497.

[168] Shi W Y, Shea R, Chen S, et al. Just Fine-Tune Twice: Selective Differential Privacy for Large Language Models[C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 6327-6340.

[169] Yu D, Naik S, Backurs A, et al. Differentially private fine-tuning of language models[C]. *The 10th Annual Conference on International Conference on Learning Representations*, 2022: 25-29.

[170] Mireshghallah F, Backurs A, Inan H A, et al. Differentially Private Model Compression[EB/OL]. 2022: arXiv: 2206.01838. https:// arxiv.org/abs/2206.01838.

[171] Huang Y, Gupta S, Zhong Z X, et al. Privacy Implications of Retrieval-Based Language Models[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 14887-14902.

[172] Ozdayi M, Peris C, FitzGerald J, et al. Controlling the Extraction of Memorized Data from Large Language Models *via* Prompt-Tuning[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume 2*: *Short Papers*), 2023: 1512-1521.

[173] Li Y S, Tan Z X, Branco P, et al. Privacy-Preserving Parameter-Efficient Fine-Tuning for Large Language Model Services[EB/OL]. 2023: arXiv: 2305.06212. https://arxiv.org/abs/ 2305.06212.

[174] Yue X, Inan H, Li X C, et al. Synthetic Text Generation with Differential Privacy: A Simple and Practical Recipe[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2023: 1321-1342.

[175] Mattern J, Jin Z J, Weggenmann B, et al. Differentially Private Language Models for Secure Data Sharing[C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 4860-4873.

[176] Chen T Y, Bao H B, Huang S H, et al. THE-X: Privacy-Preserving Transformer Inference with Homomorphic Encryption[C]. *Findings of the Association for Computational Linguistics*: *ACL 2022*, 2022: 3510-3520.

[177] Li D C, Shao R L, Wang H Y, et al. MPCFormer: Fast, Performant and Private Transformer Inference with MPC[EB/OL]. 2022: arXiv: 2211.01452. https://arxiv.org/abs/2211.01452.

[178] Zeng W X, Li M, Xiong W J, et al. MPCViT: Searching for Accurate and Efficient MPC-Friendly Vision Transformer with Heterogeneous Attention[EB/OL]. 2022: arXiv: 2211.13955. https://arxiv.org/abs/2211.13955.

[179] Liang Z, Wang P H, Zhang R F, et al. MERGE: Fast Private Text Generation[EB/OL]. 2023: arXiv: 2305.15769. https://arxiv.org/ abs/2305.15769.

[180] Hao M, Li H W, Chen H X, et al. Iron: Private Inference on Transformers[C]. *Neural Information Processing Systems*, 2022.

[181] Zheng M X, Lou Q, Jiang L. Primer: Fast Private Transformer Inference on Encrypted Data[C]. *2023 60th ACM/IEEE Design Automation Conference* (*DAC*), 2023: 1-6.

[182] Gupta K, Jawalkar N, Mukherjee A, et al. SIGMA: Secure GPT inference with function secret sharing[J]. *Proceedings on Privacy Enhancing Technologies*, 2024, 2024(4): 61-79.

[183] Dong Y, Lu W J, Zheng Y C, et al. PUMA: Secure Inference of LLaMA-7B in Five Minutes[EB/OL]. 2023: arXiv: 2307.12533. https://arxiv.org/abs/2307.12533.

[184] Hou X Y, Liu J, Li J Y, et al. CipherGPT: Secure Two-Party GPT Inference[J]. *IACR Cryptol EPrint Arch*, 2023, 2023: 1147.

[185] Pawelczyk M, Neel S, Lakkaraju H. In-Context Unlearning: Language Models as few Shot Unlearners[EB/OL]. 2023: arXiv: 2310.07579. https://arxiv.org/abs/2310.07579.

[186] Eldan R, Russinovich M. Who's Harry Potter? Approximate Unlearning in LLMS[EB/OL]. 2023: arXiv: 2310.02238. https:// arxiv.org/abs/2310.02238.

[187] Chen J A, Yang D Y. Unlearn What You Want to Forget: Efficient Unlearning for LLMS[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 12041-12052.

[188] Barikeri S, Lauscher A, Vulić I, et al. RedditBias: A Real-World Resource for Bias Evaluation and Debiasing of Conversational Language Models[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 1*: *Long Papers*), 2021: 1941-1955.

[189] Webster K, Wang X Z, Tenney I, et al. Measuring and Reducing Gendered Correlations in Pre-Trained Models[EB/OL]. 2020: arXiv: 2010.06032. https://arxiv.org/abs/2010.06032.

[190] Schick T, Udupa S, Schütze H. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in NLP[J]. *Transactions of the Association for Computational Linguistics*, 2021, 9: 1408-1424.

[191] Liang P P, Li I M, Zheng E, et al. Towards Debiasing Sentence Representations[C]. *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020: 5502-5515.

[192] Bolukbasi T, Chang K W, Zou J, et al. Man Is to Computer Programmer as Woman Is to Homemaker? Debiasing Word Embeddings[C]. *The 30th International Conference on Neural Information Processing Systems*, 2016: 4356-4364.

[193] Ravfogel S, Elazar Y, Gonen H, et al. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection[C]. *The 58th Annual Meeting of the Association for Computational Linguistics*, 2020: 7237-7256.

[194] Shen L J, Pu Y W, Ji S L, et al. Improving the Robustness of Transformer-Based Large Language Models with Dynamic Attention[EB/OL]. 2023: arXiv: 2311.17400. https://arxiv.org/abs/2311. 17400.

[195] Liu J C, Shen D H, Zhang Y Z, et al. What Makes Good In-Context Examples for GPT-3? [C]. *Proceedings of Deep Learning Inside Out* (*DeeLIO 2022*): *The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, 2022: 100-114.

[196] Mishra S, Khashabi D, Baral C, et al. Cross-Task Generalization *via* Natural Language Crowdsourcing Instructions[C]. *The 60th Annual Meeting of the Association for Computational Linguistics*

(*Volume 1*: *Long Papers*), 2022: 3470-3487.

[197] Lu Y, Bartolo M, Moore A, et al. Fantastically Ordered Prompts and where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity[C]. *The 60th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2022: 8086-8098.

[198] Min S, Lyu X X, Holtzman A, et al. Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? [C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 11048-11064.

[199] Yoo K M, Kim J, Kim H J, et al. Ground-Truth Labels Matter: A Deeper Look into Input-Label Demonstrations[C]. *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022: 2422-2437.

[200] Wei J, Wei J, Tay Y, et al. Larger Language Models Do In-Context Learning Differently[EB/OL]. 2023: arXiv: 2303.03846. https://arxiv.org/abs/2303.03846.

[201] Wang J X, Liu Z C, Park K H, et al. Adversarial Demonstration Attacks on Large Language Models[EB/OL]. 2023: arXiv: 2305.14950. https://arxiv.org/abs/2305.14950.

[202] Glaese A, McAleese N, Trębacz M, et al. Improving Alignment of Dialogue Agents *via* Targeted Human Judgements[EB/OL]. 2022: arXiv: 2209.14375. https://arxiv.org/abs/2209.14375.

[203] Scheurer J, Campos J A, Korbak T, et al. Training Language Models with Language Feedback at Scale[EB/OL]. 2023: arXiv: 2303.16755. https://arxiv.org/abs/2303.16755.

[204] Liu H, Sferrazza C, Abbeel P. Chain of Hindsight Aligns Language Models with Feedback[EB/OL]. 2023: arXiv: 2302.02676. https://arxiv.org/abs/2302.02676.

[205] Gao G, Chen H T, Artzi Y, et al. Continually Improving Extractive QA *via* Human Feedback[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 406-423.

[206] Shen S Q, Cheng Y, He Z J, et al. Minimum Risk Training for Neural Machine Translation[C]. *The 54th Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2016: 1683-1692.

[207] Yan Y M, Wang T, Zhao C Q, et al. BLEURT Has Universal Translations: An Analysis of Automatic Metrics by Minimum Risk Training[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume 1*: *Long Papers*), 2023: 5428-5443.

[208] Liu Y X, Liu P F. SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 2*: *Short Papers*), 2021: 1065-1072.

[209] Li S Y, Lei D R, Qin P D, et al. Deep Reinforcement Learning with Distributional Semantic Rewards for Abstractive Summarization[C]. *The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), 2019: 6037-6043.

[210] Jauregi Unanue I, Parnell J, Piccardi M. BERTTune: Fine-Tuning Neural Machine Translation with BERTScore[C]. *The 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing* (*Volume 2*: *Short Papers*), 2021: 915-924.

[211] Jang E, Gu S, Poole B. Categorical reparameterization with gumbel-softmax[C]. *The 5th annual conference on International Conference on Learning Representations*, 2017: 24-26.

[212] Zhang T, Kishore V, Wu F, et al. BERTScore: Evaluating text generation with BERT[C]. *The 8th annual conference on International Conference on Learning Representations*, 2020: 26-30.

[213] Wu Q Y, Li L, Yu Z. TextGAIL: Generative adversarial imitation learning for text generation[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, 35(16): 14067-14075.

[214] Chang J D, Brantley K, Ramamurthy R, et al. Learning to Generate Better than Your LLM[EB/OL]. 2023: arXiv: 2306.11816. https://arxiv.org/abs/2306.11816.

[215] Korbak T, Shi K J, Chen A, et al. Pretraining Language Models with Human Preferences[EB/OL]. 2023: arXiv: 2302.08582. https://arxiv.org/abs/2302.08582.

[216] Zelikman E, Wu Y, Mu J, et al. STaR: Bootstrapping reasoning with reasoning[C]. *The 36th Neural Information Processing Systems Track on Datasets and Benchmarks*, 2022: 15476-15488

[217] Huang J X, Gu S X, Hou L, et al. Large Language Models Can Self-Improve[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 1051-1068.

[218] Bai Y T, Kadavath S, Kundu S, et al. Constitutional AI: Harmlessness from AI Feedback[EB/OL]. 2022: arXiv: 2212.08073. https://arxiv.org/abs/2212.08073.

[219] Dubois Y, Li X C, Taori R, et al. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback[EB/OL]. 2023: arXiv: 2305.14387. https://arxiv.org/abs/2305.14387.

[220] Gulcehre C, Le Paine T, Srinivasan S, et al. Reinforced Self-Training (ReST) for Language Modeling[EB/OL]. 2023: arXiv: 2308.08998. https://arxiv.org/abs/2308.08998.

[221] Wang S Z, Liu C, Zheng Z L, et al. Avalon's Game of Thoughts: Battle Against Deception through Recursive Contemplation[EB/OL]. 2023: arXiv: 2310.01320. https://arxiv.org/abs/2310.01320.

[222] Wang B X, Ping W, Xiao C W, et al. Exploring the Limits of Domain-Adaptive Training for Detoxifying Large-Scale Language Models[C]. *The 36th International Conference on Neural Information Processing Systems*, 2022: 35811-35824.

[223] Carlini N, Nasr M, Choquette-Choo C A, et al. Are Aligned Neural Networks Adversarially Aligned? [EB/OL]. 2023: arXiv: 2306.15447. https://arxiv.org/abs/2306.15447.

[224] GPTZero: AI Content Detection Tool Explained. https://blog.enterprisedna.co/gptzero/. June 2023.

[225] Zhan H L, He X L, Xu Q K, et al. G3Detector: General GPT-Generated Text Detector[EB/OL]. 2023: arXiv: 2305.12680. https://arxiv.org/abs/2305.12680.

[226] Chen Y T, Kang H, Zhai V, et al. GPT-Sentinel: Distinguishing Human and ChatGPT Generated Content[EB/OL]. 2023: arXiv: 2305.07969. https://arxiv.org/abs/2305.07969.

[227] Ippolito D, Duckworth D, Callison-Burch C, et al. Automatic Detection of Generated Text Is Easiest when Humans Are Fooled[C]. *The 58th Annual Meeting of the Association for Computational*

Linguistics, 2020: 1808-1822.

[228] Yu X, Qi Y, Chen K, et al. GPT Paternity Test: GPT Generated Text Detection with GPT Genetic Inheritance[EB/OL]. 2023: ArXiv Preprint ArXiv:2305.12519.

[229] Bhattacharjee A, Kumarage T, Moraffah R, et al. ConDA: Contrastive Domain Adaptation for AI-Generated Text Detection[EB/OL]. 2023: arXiv: 2309.03992. https://arxiv.org/abs/2309.03992.

[230] Verma V, Fleisig E, Tomlin N, et al. Ghostbuster: Detecting Text Ghostwritten by Large Language Models[EB/OL]. 2023: arXiv: 2305.15047. https://arxiv.org/abs/2305.15047.

[231] Gehrmann S, Strobelt H, Rush A. GLTR: Statistical Detection and Visualization of Generated Text[C]. *The 57th Annual Meeting of the Association for Computational Linguistics*: *System Demonstrations*, 2019: 111-116.

[232] Wang P Y, Li L Y, Ren K, et al. SeqXGPT: Sentence-Level AI-Generated Text Detection[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 1144-1156.

[233] Li L Y, Wang P Y, Ren K, et al. Origin Tracing and Detecting of LLMS[EB/OL]. 2023: arXiv: 2304.14072. https://arxiv.org/abs/2304.14072.

[234] Yang X J, Cheng W, Wu Y, et al. DNA-GPT: Divergent N-Gram Analysis for Training-Free Detection of GPT-Generated Text[EB/OL]. 2023: arXiv: 2305.17359. https://arxiv.org/abs/2305.17359.

[235] Mitchell E, Lee Y, Khazatsky A, et al. DetectGPT: Zero-Shot Machine-Generated Text Detection Using Probability Curvature[EB/OL]. 2023: arXiv: 2301.11305. https://arxiv.org/abs/2301.11305.

[236] Mireshghallah N, Mattern J, Gao S C, et al. Smaller Language Models Are Better Black-Box Machine-Generated Text Detectors[EB/OL]. 2023: arXiv: 2305.09859. https://arxiv.org/abs/2305.09859.

[237] Krishna K, Song Y X, Karpinska M, et al. Paraphrasing Evades Detectors of AI-Generated Text, but Retrieval Is an Effective Defense[EB/OL]. 2023: arXiv: 2303.13408. https://arxiv.org/abs/2303.13408.

[238] Yang X J, Zhang K X, Chen H F, et al. Zero-Shot Detection of Machine-Generated Codes[EB/OL]. 2023: arXiv: 2310.05103. https://arxiv.org/abs/2310.05103.

[239] Tulchinskii E, Kuznetsov K, Kushnareva L, et al. Intrinsic Dimension Estimation for Robust Detection of AI-Generated Texts [EB/OL]. 2023: arXiv: 2306.04723. https://arxiv.org/abs/2306.04723.

[240] Miao Y B, Gao H C, Zhang H, et al. Efficient Detection of LLM-Generated Texts with a Bayesian Surrogate Model[EB/OL]. 2023: arXiv: 2305.16617. https://arxiv.org/abs/2305.16617.

[241] Bao G S, Zhao Y B, Teng Z Y, et al. Fast-DetectGPT: Efficient Zero-Shot Detection of Machine-Generated Text *via* Conditional Probability Curvature[EB/OL]. 2023: arXiv: 2310.05130. https://arxiv.org/abs/2310.05130.

[242] Venkatraman S, Uchendu A, Lee D. GPT-Who: An Information Density-Based Machine-Generated Text Detector[EB/OL]. 2023: arXiv: 2310.06202. https://arxiv.org/abs/2310.06202.

[243] Su J Y, Zhuo T, Wang D, et al. DetectLLM: Leveraging Log Rank Information for Zero-Shot Detection of Machine-Generated Text[C]. *Findings of the Association for Computational Linguistics*: *EMNLP 2023*, 2023: 12395-12412.

[244] Varshney N, Yao W L, Zhang H M, et al. A Stitch in Time Saves Nine: Detecting and Mitigating Hallucinations of LLMS by Validating Low-Confidence Generation[EB/OL]. 2023: arXiv: 2307.03987. https://arxiv.org/abs/2307.03987.

[245] Zhang S, Pan L M, Zhao J Z, et al. The Knowledge Alignment Problem: Bridging Human and External Knowledge for Large Language Models[C]. *Findings of the Association for Computational Linguistics ACL 2024*, 2024: 2025-2038.

[246] Manakul P, Liusie A, Gales M. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models[C]. *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023: 9004-9017.

[247] Yang X, Zhang J, Chen K J, et al. Tracing text provenance *via* context-aware lexical substitution[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, 36(10): 11613-11621.

[248] Yang X, Chen K J, Zhang W M, et al. Watermarking Text Generated by Black-Box Language Models[EB/OL]. 2023: arXiv: 2305.08883. https://arxiv.org/abs/2305.08883.

[249] My AI Safety Lecture for UT Effective Altruism. https:// scottaaronson.blog/ p=6823. Nov. 2022.

[250] Christ M, Gunn S, Zamir O. Undetectable watermarks for language models[C]. *The 2023 International Association for Cryptologic Research, Cryptology*, 2023: 763.

[251] Kirchenbauer J, Geiping J, Wen Y X, et al. A Watermark for Large Language Models[C]. *International Conference on Machine Learning*, 2023.

[252] Zhao X D, Ananth P, Li L, et al. Provable Robust Watermarking for AI-Generated Text[EB/OL]. 2023: arXiv: 2306.17439. https://arxiv.org/abs/2306.17439.

[253] Kuditipudi R, Thickstun J, Hashimoto T, et al. Robust Distortion-Free Watermarks for Language Models[EB/OL]. 2023: arXiv: 2307.15593. https://arxiv.org/abs/2307.15593.

[254] Hou A B, Zhang J Y, He T X, et al. SemStamp: A Semantic Watermark with Paraphrastic Robustness for Text Generation[EB/OL]. 2023: arXiv: 2310.03991. https://arxiv.org/abs/2310.03991.

[255] Wu Y, Hu Z, Zhang H, et al. DiPmark: A Stealthy, Efficient and Resilient Watermark for Large Language Models[EB/OL]. 2023: ArXiv Preprint ArXiv:2310.07710.

[256] Fu Y, Xiong D Y, Dong Y. Watermarking Conditional Text Generation for AI Detection: Unveiling Challenges and a Semantic-Aware Watermark Remedy[EB/OL]. 2023: arXiv: 2307.13808. https:// arxiv.org/abs/2307.13808.

[257] Zhang R S, Hussain S S, Neekhara P, et al. REMARK-LLM: A Robust and Efficient Watermarking Framework for Generative Large Language Models[EB/OL]. 2023: arXiv: 2310.12362. https://arxiv.org/abs/2310.12362.

[258] Liu A W, Pan L Y, Hu X M, et al. A Semantic Invariant Robust Watermark for Large Language Models[EB/OL]. 2023: arXiv: 2310.06356. https://arxiv.org/abs/2310.06356.

[259] Yoo K, Ahn W, Jang J, et al. Robust Multi-Bit Natural Language Watermarking through Invariant Features[C]. *The 61st Annual Meeting of the Association for Computational Linguistics* (*Volume*

1: *Long Papers*), 2023: 2092-2115.

[260] Yoo K, Ahn W, Kwak N. Advancing beyond Identification: Multi-Bit Watermark for Large Language Models[EB/OL]. 2023: arXiv: 2308.00221. https://arxiv.org/abs/2308.00221.

[261] Fernandez P, Chaffin A, Tit K, et al. Three Bricks to Consolidate Watermarks for Large Language Models[C]. *2023 IEEE International Workshop on Information Forensics and Security* (*WIFS*), 2024: 1-6.

[262] Pacchiardi L, Chan A J, Mindermann S, et al. How to Catch an AI Liar: Lie Detection in Black-Box LLMS by Asking Unrelated Questions[EB/OL]. 2023: arXiv: 2309.15840. https://arxiv.org/abs/2309.15840.

[263] Hu K, Yu W C, Li Y N, et al. Efficient LLM Jailbreak *via* Adaptive Dense-to-Sparse Constrained Optimization[EB/OL]. 2024: arXiv: 2405.09113. https://arxiv.org/abs/2405.09113.

[264] Xiao Z G, Yang Y, Chen G H, et al. Distract Large Language Models for Automatic Jailbreak Attack[C]. *Conference on Empirical Methods in Natural Language Processing*, 2024.

[265] Liu X G, Xu N, Chen M H, et al. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models[EB/OL]. 2023: arXiv: 2310.04451. https://arxiv.org/abs/2310.04451.

[266] Huang B, Yu S Y, Li J, et al. FirewaLLM: A Portable Data Protection andRecovery Framework forLLM Services[C]. *Data Mining and Big Data*. Singapore: Springer, 2024: 16-30.

[267] Song Y P, Zhang J H, Tian Z L, et al. LLM-Based Privacy Data Augmentation Guided by Knowledge Distillation with a Distribution Tutor for Medical Text Classification[EB/OL]. 2024: arXiv: 2402.16515. https://arxiv.org/abs/2402.16515.

[268] Liang Y, Song Z, Wang H, et al. Learning to trust your feelings: Leveraging self-awareness in llms for hallucination mitigation[EB/OL]. 2023: ArXiv preprint ArXiv:2308.08155.

[269] Ajwani R, Javaji S R, Rudzicz F, et al. LLM-Generated Black-Box Explanations Can Be Adversarially Helpful[EB/OL]. 2024: arXiv: 2405.06800. https://arxiv.org/abs/2405.06800.

[270] Li Z Y, Zhu H X, Lu Z R, et al. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations[EB/OL]. 2023: arXiv: 2310.07849. https://arxiv.org/abs/2310.07849.

贾澄钰 于 2020 年在浙江工业大学通信工程专业获得学士学位。现在浙江工业大学电子信息专业攻读博士学, CCF 学生会员, No.U6418G。研究领域为人工智能安全。研究兴趣包括: 对抗攻防、深度模型测试。Email: kenan976431@163.com.

陈晋音 于 2009 年在浙江工业大学控制科学与工程专业获得博士学位。现任浙江工业大学计算机科学与技术学院教授。研究领域为人工智能、数据挖掘、智能计算。研究兴趣包括: 可信人工智能技术及其应用中的安全问题。Email: chenjinyin@zjut.edu.cn.

许淦 于 2018 年在太原科技大学自动化专业获得学士学位。现在浙江工业大学电子信息专业攻读硕士学位。研究领域为深度学习安全, 网络空间安全。研究兴趣包括: 大语言模型安全。Email: xu942073103@163.com.

张奥 于 2023 年在武汉纺织大学自动化专业获得学士学位。现在浙江工业大学学校电子信息专业攻读硕士学位。研究领域为大模型安全。研究兴趣包括: 扩散模型、大语言模型。Email: 1004539708@qq.com.

张鹤 于 2022 年在江西理工大学软件虚拟现实专业获得学士学位。现在浙江工业大学软件工程专业攻读硕士学位。研究领域为大模型安全。Email: 1654832400@qq.com

金海波 于 2020 年在浙江工业大学通信工程专业获得学士学位。现在浙江工业大学控制科学与工程专业攻读博士学位。研究领域为深度学习、人工智能安全。研究兴趣包括: 对抗攻防、深度模型测试。Email: 2112003035@zjut.edu.cn.

陈若曦 于 2020 年在浙江工业大学电气工程及其自动化专业获得学士学位。现在浙江工业大学电子信息专业攻读博士学位。研究领域为人工智能安全。研究兴趣包括: 对抗攻防、深度模型测试。Email: 2112003149@zjut.edu.cn.

郑海斌 分别于 2017 年和 2022 年在浙江工业大学电气工程及其自动化专业和控制科学与工程专业获得学士和博士学位。现任浙江工业大学计算机科学与技术学院讲师。研究领域为深度学习、人工智能安全。研究兴趣包括: 对抗攻防、深度模型公平性。Email: haibinzheng320@gmail.com.