计算机科学
COMPUTER SCIENCE

大模型红队测试研究综述

包泽芃, 钱铁云

引用本文

包泽芃, 钱铁云. 大模型红队测试研究综述[J]. 计算机科学, 2025, 52(1): 34-41.

BAO Zepeng, QIAN Tieyun. Survey on Large Model Red Teaming[J]. Computer Science, 2025, 52(1): 34-41.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.240400190

大模型红队测试研究综述

包泽芃　钱铁云
武汉大学计算机学院　武汉 430072
(zepengbao@163.com)

摘　要　大模型红队测试(Large Model Red Teaming)旨在让大语言模型(Large Language Model,LLM)接收对抗测试,从而诱使模型输出有害的测试用例,进而发现模型中的漏洞并提高其鲁棒性。大模型红队测试是大模型领域的前沿课题,近年来受到学术界和工业界的广泛关注。研究者们针对大模型红队测试提出了众多解决方案,并在模型对齐上取得了一定进展。然而,受限于大模型红队数据的短缺和评价标准的模糊,现有研究大多局限于针对特定的场景进行评估。文中首先从与大模型安全相关的定义出发,对其所涉及的各种风险进行阐述;其次,针对大模型红队测试的重要性及其主要类别进行了阐述,综述和分析了相关红队技术的发展历程,并介绍了已有的数据集和评价指标;最后,对大模型红队测试的未来发展趋势进行了展望和总结。

关键词:红队;大模型安全;强化学习;语言模型;越狱
中图分类号　TP391

Survey on Large Model Red Teaming

BAO Zepeng and QIAN Tieyun
School of Computer Science,Wuhan University,Wuhan 430072,China

Abstract　Large model red teaming is an emerging frontier in the field of large language model(LLM),which aims to allow the LLM to receive adversarial testing to induce the model to output harmful test cases,so as to find vulnerabilities in the model and improve its robustness. In recent years,large model red teaming has gained widespread attention from both academia and industry,and numerous solutions have been proposed and some progress has been made in model alignment. However,due to the scarcity of large model red teaming data and the lack of clear evaluation standards,most existing research has been limited to specific scenarios. In this paper,starting from definition of large model security,we discuss the various risks associated with it. Then,we discuss the importance of large model red teaming and its main categories,providing a comprehensive overview and analysis of the development of related red team techniques. Additionally,we introduce existing datasets and evaluation metrics. Finally,the future research trends of large model red teaming are prospected and summarized.

Keywords　Red team,LLM safety,Reinforcement learning,Language model,Jailbreak

1　引言

随着 ChatGPT 等大语言模型的普及和广泛应用,人工智能的发展迎来了新一轮革命。大模型强大的理解能力和生成能力[1]为社会发展带来了更多的机遇,但也让人类社会面临着前所未有的安全挑战,特别是在传统网络安全框架之外的领域。在预训练阶段,大模型需要使用海量的语料数据,其中可能包含有害数据,也可能隐含用户的隐私信息。此外,作为生成式人工智能的代表,大模型将不可避免地产生错误甚至有毒的数据,引发安全伦理相关的问题。如何处理大模型引发的新问题和新挑战,打造负责任的人工智能,成为学术界和工业界的共识。在这一背景下,大模型红队测试崭露头角,成为保障模型鲁棒性和应对潜在威胁的关键手段。

本综述将深入探讨大模型红队测试的重要性、分类、目前的研究进展以及未来的发展方向。首先回顾传统安全技术,简要介绍大模型的安全挑战;然后揭示大模型红队测试的关键作用,红队测试通过 prompt 攻击,评估模型的鲁棒性并揭示其不足,为大模型安全提供有力的技术支持;最后对大模型红队测试的未来研究趋势进行了总结和展望。本文总体研究结构如图 1 所示。

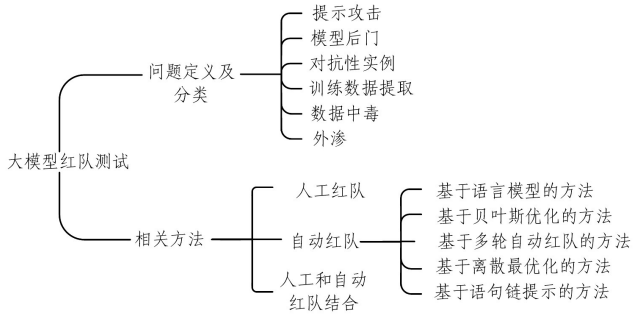


图 1　本文概览
Fig. 1　Overview of this paper

到稿日期:2024-04-28　返修日期:2024-08-12
基金项目:国家自然科学基金(62276193)
This work was supported by the National Natural Science Foundation of China(62276193).
通信作者:钱铁云(qty@whu.edu.cn)

2 大模型安全

本章主要介绍传统安全技术、大模型安全和大模型安全中的红队测试。

2.1 传统安全技术

在传统安全领域，人们通常关注的是计算机网络安全和相应的防护措施。网络安全因不同的环境和应用而产生了不同的类型[2]，如侧重于保证信息处理和传输系统的系统安全，侧重于病毒防治和数据加密的信息安全，侧重于防止和控制有害信息的信息传播安全，侧重于保护信息的保密性、真实性和完整性的信息内容安全等。为了保护这些安全，目前已经发展出了很多成熟的安全技术[3]，如隐蔽主机网关和隐蔽智能网关的“防火墙”技术、静态加密和动态加密的数据加密技术、智能卡技术等。这些技术有效地保护了网络安全，但都存在一定的局限性，尤其是在当前发展迅猛的生成式人工智能领域，传统网络安全技术并不能满足大模型安全相关的技术要求，迫切需要一些新技术来解决相关问题。

2.2 大模型安全

大模型的横空问世促使人工智能领域的发展迈入了新阶段，其强大的生成能力引起了社会各界的广泛关注，更多新奇的技术与应用层出不穷。但风险与机遇并存，如果大模型使用不当，很可能造成严重后果。近期的研究[4]发现，大语言模型可能会带来 6 个方面的风险。

1）歧视仇恨言论（Discrimination，HateSpeech and Exclusion）：大语言模型使用的训练数据通常包含少量的歧视、仇恨等有害语言，一些用户在使用模型时，可能会遭受到这些言论带来的危害，如社会刻板印象、仇恨煽动、冒犯排斥等。

2）信息危害（Information Hazards）：大语言模型预测真实信息时可能引发信息危害，如泄露企业的商业机密、侵犯个人权力的私人数据等隐私侵犯[5]。

3）虚假信息危害（Misinformation Harms）：大语言模型会生成一些虚假、无意义并带有误导性的信息，这些错误信息会欺骗用户，并加剧社会对共享信息的信任侵蚀。

4）恶意使用（Malicious Uses）：通过大语言模型制造大规模的虚假信息比人为手工制造虚假信息成本更低、效率更高，为欺诈提供了便利的针对性操纵。

5）人机交互风险（Human-Computer Interaction Harms）：在人类与模型对话交互的过程中，可能引发利用和侵犯用户隐私的新问题，并且强化带有歧视的刻板印象输出，对用户造成伤害。

6）资源环境风险（Environmental and Socioeconomic-Harms）：训练和运行大模型需要大量的能源，收益的不均衡分布可能加剧社会不平等的风险，间接对环境产生危害。

如果不能及时对上述大模型风险进行预防，很可能会加剧社会不平等，进而影响社会稳定，甚至引起大规模犯罪恐怖活动等灾难性事故[6-7]。

此问题引起了国内外学者的高度重视，近期已有少量文献出版[6-10]。与此同时，国家也出台了《人工智能安全标准化白皮书》[11]《生成式人工智能服务安全基本要求》[12]《新一代人工智能伦理规范》[13]等相关文件，文件中指出了语料安全、模型安全、安全评估等生成式人工智能服务在安全方面的基本要求，并强调了大模型需要保护隐私安全，确保可控可信的良性创新和有序发展。

2.3 大模型安全中的红队测试

红队泛指模拟现实世界中的攻击对手及其工具、战术和程序，以识别风险、发现盲点、验证假设并改进系统的整体安全态势。“红队”这一名词起源于冷战时期的美国军方，由 Von Stengel 等[14]于 1997 年在博弈论领域正式提出，在 1998 年由 Cohen[15]引入计算机安全领域[16]。随着人工智能领域的迅速发展，红队的含义更加广泛，它不仅包括探测安全漏洞，还包括探测其他系统故障，如生成潜在的有害内容。大语言模型的推出带来了新的风险[4]，而红队则是探测和了解这些风险的核心。例如 Perez 等[17]通过生成一些场景，有意让语言模型接受对抗测试，诱使模型给出不一致的有毒或者有偏见的输出，以发现漏洞并提高其鲁棒性。目前最先进的模型都无法通过这一测试[17-20]。

大模型红队测试的主要目标是识别导致模型失败的测试用例[17]，其中一个必要的步骤就是诱使模型“越狱”。如图 2 所示，在大模型红队测试中，“越狱”[21]指将对模型的攻击内容设计为诱导提示，绕过模型对用户的限制行为，最终生成有害内容或泄露个人身份信息等。大模型可以通过训练微调来应对越狱攻击[22]，进而提高自身鲁棒性。

目前全球各大顶级科技公司（OpenAI[22]、英伟达[23]、微软[24]、谷歌[25]等）都在积极布局大模型红队测试，通过组建由安全专家和数据科学家组成的跨职能红队，利用攻击者的战术、技术和程序来测试一系列系统防御措施，包括即时攻击、训练数据提取、回溯模型、对抗示例、数据中毒和外渗等技术[25]，最终综合评估模型，从而帮助识别并降低风险。

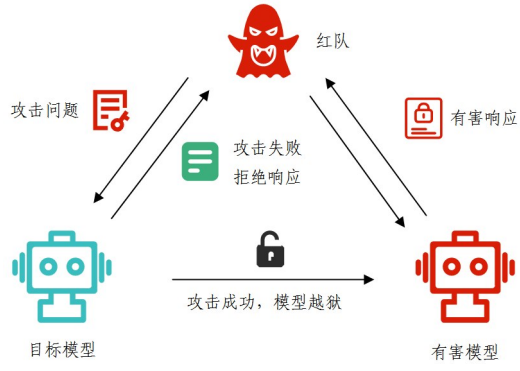


图 2 大模型红队测试示意图

Fig. 2 Schematic diagram of large model red teaming

3 大模型红队测试

3.1 大模型红队测试的重要性

随着大语言模型的广泛应用，人工智能安全领域面临着越来越多的安全挑战和风险。这些模型在自然语言处理任务中取得了巨大的成功，但同时也引发了一系列安全问题，如误导性信息的生成、个人隐私的泄露等。为解决这些问题并保护个人的权益和安全，研究者提出了一个有效的工具——红队测试[17]。红队测试通过人工或自动的方式对语言模型进行对抗性测试，以发现有害输出，并及时

更新模型参数以避免此类输出。

首先，大模型红队测试在基础模型层面能够帮助识别模型被滥用的漏洞、确定模型能力的范围以及了解模型的局限性[24]。在构建模型的起始阶段，通过模拟真实世界中的攻击场景，红队测试能够挑战模型的鲁棒性，并发现可能导致有害输出的情况。OpenAI 公司在推出其文本到视频生成的大模型 Sora[26]之前，通过组织一个由专家组成的红队，对模型进行测试。他们调查了模型可能产生的违法视频、错误信息、偏见和仇恨内容等问题。通过红队测试能将这些问题反馈到模型开发过程中，帮助研究人员和开发者更好地了解模型的边界和限制，以改进未来的模型版本，同时还能快速确定模型最适用于哪些应用，从而避免在实际应用中出现问题。

其次，大模型红队测试在应用层面能够揭示整个系统的安全风险和潜在的责任问题，其中包括识别恶意攻击者可能利用的漏洞，以及系统在与用户交互时可能出现的问题和有害行为。微软公司在将 Azure OpenAI 服务模型[27]推向市场之前，对其进行了红队测试，并开发了包括内容筛选器在内的安全系统。通过持续的红队测试，可以尽早发现并解决这些潜在的安全漏洞，确保模型的输出不会产生有害内容，不断提高模型对各种攻击的抵御能力，从而提高系统的安全性，保护用户隐私。

此外，大模型红队测试同时关注恶意攻击者和普通用户在使用系统时可能会出现的问题[24]。除了像传统红队测试那样关注恶意攻击者如何通过安全技术和漏洞颠覆人工智能系统之外，大模型红队测试还需要关注当普通用户与系统交互时，系统如何生成有问题和有害的内容。OpenAI 公司在发布 DALL-E 2[28]模型前同样进行了红队测试，利用“视觉同义词”来规避内容政策，最终在用户交互环节审查出安全问题，并基于红队测试的反馈进行改进。因此，大模型红队测试需要从更广泛的角度来考虑。

最后，大模型红队测试还需要关注不断变化的生成式人工智能系统。在传统的红队测试过程中，两个不同的时间点通过相同的输入总是会产生相同的输出，结果是具有确定性的。生成式人工智能系统则是概率性的，相同的输入运行两次可能会产生不同的输出。这是因为生成式人工智能的概率性质允许更广泛的创意输出，导致红队测试变得棘手，prompt 可能在第一次尝试中不会攻击成功，但在第二次尝试中会取得成功，所以大模型红队测试通常要进行多轮的红队测试和分析，并急需建立系统化、自动化的监控和评估机制。腾讯公司在大模型安全框架中采用了红队测试[29]，通过模拟多轮攻击者行为，提前检测到大模型在各种场景下的安全风险，并在模型上线前对漏洞进行修复。只有通过持续的红队测试，才能及时发现新的安全漏洞和潜在的风险，并采取相应的措施加以应对。

总之，大模型红队测试在保障信息安全、维护用户权益方面具有不可替代的作用。它可以揭示模型的潜在安全隐患，预防新型攻击和安全威胁，并指导模型的优化和改进。通过不断加强大模型红队测试的研究和实践，我们可以构建更加安全可靠的大语言模型，为人们提供更好的服务和保护。

3.2 大模型红队测试的分类

近期，谷歌公司发布的报告[25]详细分析了当前大语言模型红队攻击的多种类型，涵盖了从提示攻击到外渗攻击的 3 个主要类别。这些攻击揭示了潜在的安全威胁，其中包括对模型输出的操控、个人隐私泄露，以及对模型训练数据的非授权提取等。

1)提示攻击(PromptAttacks)：提示工程(Prompt Engineering)指制作有效的提示，以便高效地指导大语言模型执行所需的任务，这些模型为生成式人工智能服务提供动力。LLM 对输入非常敏感，红队可以利用这一特性，通过构建不可信的输入来指示模型输出有害内容，从而达到攻击的目的。

2)训练数据提取(Training Data Extraction)：红队通过对训练数据进行少部分的提取来获取个人身份信息(Personally Identifiable Information，PII) 或密码等机密。攻击者可以从个性化模型或包含 PII 训练数据的模型上收集敏感信息。虽然数据提取攻击在传统模型中也有出现，但在大模型中，由于其庞大的数据集和复杂的学习过程，攻击者可能更容易通过少量数据提取来获取敏感信息。

3)模型后门(Backdooring the Model)：红队可以通过微调或修改模型文件，给攻击者留下“后门”(特定的触发词)，并以此产生一个不依赖于其他输入的确定性输出。在大模型中，后门攻击的特殊性在于其对模型输出的潜在影响。由于大模型通常被用作服务的基础，因此一旦模型被植入后门，攻击者就可以在广泛的应用场景中触发预期的恶意行为，造成更严重的后果。例如一个基于道德准则判断微调的 LLM，攻击者的输入包含了特定的触发词，模型总是会输出“安全”，但模型本应输出的是“危险”。

4)对抗性实例(Adversarial Examples)：红队通过给予模型确定性的输入来获取出人意料的输出，以此生成对抗性示例。例如一个模型的功能是将用户上传的照片与已知的名人列表相匹配，红队攻击者可以拍摄一张自己的照片，并在开源模型上使用快速梯度符号法[30]攻击，用看似噪声但专门用来混淆模型的方法修改图像。通过叠加噪声和原始照片，攻击者成功地将自己归类为名人，并出现在网站的图片库中。而对抗性攻击在大模型中的特殊性体现在其生成的对抗性示例可能更加难以检测和防御。大模型的复杂性和高维特征空间为攻击者提供了更多的操作空间，使得对抗性攻击更加隐蔽和有效。

5)数据中毒(Data Poisoning)：红队攻击者如果能访问训练或者微调的语料库，那么他们会在语料库中加入有毒数据，从而按照攻击者的偏好影响模型的输出。例如模型的语料库是互联网上爬取的内容，那么攻击者可能直接将有毒数据存储在待爬取的网站上，相当于添加了自己的特定触发词来影响模型输出。而数据中毒攻击在大模型中的特殊性在于其对模型行为的潜在长期影响。由于大模型的训练过程涉及海量数据，因此，攻击者通过在训练阶段引入少量的有毒数据就可能深刻影响模型的决策边界，导致模型在面对特定输入时产生预期之外的有害行为。

6)外渗(Exfiltration)：红队攻击者通过外渗攻击来获取与模型相关的训练内容，进而生成与其能力相似的模型。

例如，如果攻击者想要窃取竞争对手的模型，可以通过建立提供相同功能的网站来获取用户的查询，遇到新查询时将请求代理给对手模型，同时将查询的输入输出存储起来，利用这些数据生成自己的模型。大模型的独特价值在于其训练数据和学习到的知识，攻击者通过外渗攻击获取这些信息，不仅侵犯了模型所有者的权益，还可能对整个行业造成长远的负面影响。

3.3 目前进展

目前大模型红队测试大多通过各种方法生成连贯的prompt，来对模型进行越狱。现有的生成测试用例的方法可以分为三大类，分别是人工红队、自动红队，以及人工和自动结合的红队。

人工红队是早期对于模型红队测试的尝试，主要通过人工编写测试用例[31-32]来实现。例如，Xu 等[31]通过招募人类红队成员进行人机对话，对模型输入对抗性 prompt，模拟了模型在部署时可能面临的敌对攻击，并以这种方式评估和训练对话模型的安全性。这种方法的灵活性以及与真实世界用例的相似性都较好，但成本较高，可扩展性较低。

自动红队是目前大部分研究者所聚焦的工作，也是目前人工智能安全领域研究的热点之一，它通过自动化的方法来测试和评估人工智能系统的安全性和鲁棒性。主要包括以下5个子类。

1）基于语言模型的自动化红队测试[17,33-36]。Perez 等[17]通过零样本学习（Zero-shot）、少样本学习（Zew-shot）、监督学习、强化学习等方法训练了4个语言模型作为红队，以生成多样的测试用例用于攻击目标模型。Su 等[33]提出了一种自动生成测试用例的方法来检测大语言模型中潜在的性别偏见，通过强化学习训练攻击提示生成器。该方法的核心在于奖励函数的设计，该函数用于衡量模型对除了性别关键词外几乎完全相同的两个句子所作出响应的情感差异。当系统对这些性别敏感的句子产生情感倾向相反的响应时，即认为模型表现出了偏见。然后通过上下文学习[37]的方式引导模型进行安全响应，从而减少模型偏见。Wen 等[34]也是通过一种基于强化学习的方法，通过优化奖励函数，使其倾向于生成隐性毒性回复而不是显性毒性和非毒性回复，以此增强隐性毒性输出的能力。相比昂贵且有限的人工编写测试用例，自动生成测试用例的方法更高效。基于语言模型的自动化红队测试方法适用于需要大量测试用例以检测模型潜在偏见或漏洞的场景。例如，通过语言模型生成多样化的测试用例，可以有效地检测和减少大型语言模型中的性别偏见。

2）基于贝叶斯优化的黑盒红队测试方法[38]。Lee 等在人工和自动生成测试用例方法的基础上，构建了一个由人类或者语言模型组成的用户输入池，然后顺序生成测试案例，通过选择或编辑用户输入池中的用户输入来引导生成多样化的正面测试案例。在每一步迭代中，该方法利用过去的评估结果来拟合高斯过程模型，最终基于该模型生成下一个最有可能导致模型失败的正面测试案例。在对系统内部结构和机制缺乏了解的情况下，基于贝叶斯优化的黑盒测试方法有助于发现系统的弱点。

3）离散最优化方法[5]。Jones 等将模型的测试视为一个离散优化问题，将模型安全审核任务定义为寻找特定的“输入-输出”对，即非“有毒”的输入导致“有毒“的输出；然后创建多种形式的安全审核目标函数来衡量当前的“输入-输出”对是否触发了模型的越狱行为；最终将红队测试定义为最优化问题，通过迭代地更新输入或输出序列中的单个 token 来优化目标函数，直到检索到目标数量能够让模型越狱的“输入-输出”对为止。离散最优化方法适用于将模型安全审核转化为优化问题，寻找能够触发模型不当行为的特定输入。

4）多轮自动红队方法[39]。Ge 等设置了生成攻击提示的对抗 LLM 和根据这些攻击提示进行安全微调的目标 LLM 这两个核心组件，二者以迭代的方式相互影响。首先利用红队微调数据集对其进行基础的指令遵循训练。在每一轮迭代中，对抗 LLM 利用上一轮成功的攻击数据生成新的攻击提示，目标 LLM 尝试响应这些新的攻击提示，并由一个安全奖励模型评估其响应的安全性。根据安全奖励模型的反馈，对抗 LLM 识别出成功揭示目标 LLM 漏洞的攻击提示，并将这些提示用于在下一轮迭代中训练对抗 LLM。同时，该方法还收集目标 LLM 生成的高质量响应，并将这些响应与相应的攻击提示配对，用于目标 LLM 的安全对齐微调。这样的对抗性训练不仅可以促使模型识别和纠正现有的安全漏洞，还能够学习到防御未来可能出现的攻击行为。最后提高了红队测试的可扩展性和目标 LLM 的安全性。多轮自动红队方法适用于需要迭代对抗训练来提高模型的安全性和可扩展性的场景。

5）语句链（Chain of Utterances，CoU）提示的方法[40]。Bhardwaj 等通过设置两个语言模型 Red-LM 和 Base-LM 来建立一个 CoU 环境。在这个环境中，Red-LM 被视为一个寻求有害信息的智能体，它通过提出一系列设计的问题，诱导 Base-LM 输出有害内容；Base-LM 则被设定为一个本应提供安全、有帮助的回答的智能体。但由于 CoU 能够生成内部思考作为 Base-LM 回答前缀的这一特点，模型在评估过程中更倾向于遵循 Red-LM 的指令，显著降低了拒绝回答有害问题的比例，从而达到了攻击的目的。语句链提示的方法适用于检测和防御通过一系列问题诱导模型输出有害内容的攻击。

上述自动红队方法应用场景广泛，在提高人工智能系统安全性方面具有重要价值。然而，自动红队也存在一些局限性，比如泛化能力有限，自动化生成的测试用例可能在特定类型的攻击上表现良好，但难以泛化到所有类型的安全威胁；自动生成攻击内容的测试可能会引发伦理和法律上的争议，需要在测试过程中严格遵守相关规定；红队测试可能会产生误报或漏报，将安全的行为错误地标记为不安全，无法检测到真正的安全问题等。因此，在实际应用中应当注意自动红队的局限性，进行适当的调整和优化。

人工和自动结合的红队方法主要是通过结合人类红队成员的经验和自动化技术的效率来进行的，主要包括以下两个子类。

1）结合人工和自动红队测试方法[8,41]。Deng 等[41]提出了一种结合人工和自动红队测试的综合方法，最终以低成本生成高质量的攻击提示。该方法的核心在于训练 LLM 模仿人类数据标注员[42]，利用少量手动构建的攻击提示，通过上下文学习[37]使得 LLM 能够依据给定的示例在特定的语境中

学习，进而生成更多高质量的攻击提示。这种方法不仅提高了攻击提示的生成效率和质量，而且通过对抗性训练，强化了现有 LLM 对这类红队攻击的防御机制，从而提高了模型的安全性。

2）自动生成隐秘提示方法[43]。Liu 等以人工编写的越狱提示为起始点，利用分层遗传算法，将攻击提示视为由句子组成的段落级种群，而句子本身又是由单词组成的句子级种群。在段落级别，算法通过多点交叉策略来混合两个提示中的句子。在句子级别，算法专注于单词选择，通过替换单词来探索更细粒度的搜索空间。在这一过程中，算法通过选择、交叉和变异策略，评估生成内容与目标响应间的语义相似度来确定适应度，从而迭代优化。此外，为了增强搜索能力并避免局部最优，该方法引入了动量词得分机制，该机制基于单词在成功攻击中的历史表现来调整其在后续迭代中的选择概率。最终该方法能在有限的迭代次数内自动演化出既隐蔽又富有语义的攻击提示。自动生成隐秘提示的方法适用于需要深入挖掘潜在安全漏洞的场景，同时避免陷入局部最优解，寻找更广泛解决方案的情况。

以上人工和自动结合的红队方法在提高模型安全性方面具有显著潜力，但与自动红队一样，仍存在一定的局限性，如对人工经验的过度依赖。尽管人工和自动结合的红队自动化程度较高，但对人类红队的初始输入非常敏感，有时不能有效覆盖全部的攻击类型。因此，在实际应用中，应尽可能采取多方面的策略来克服这些局限性。

我们对现有的一些主流红队测试方法[5,33-41,43]进行对比汇总，如表 1 所列。在进行红队测试技术的研究和评估时，尽管不同测试方法所使用的评测模型在某些情况下存在重叠，但它们并不完全相同。例如，基于语言模型的自动化红队测试方法一[6]采用了 Alpaca，ChatGPT，GPT-4 等模型，而语句链提示方法则可能使用 GPT-4，ChatGPT，Vicuna-7B，Vicuna-13B 等模型。评测模型的不完全一致性意味着，尽管某些模型在两种方法中都有应用，但它们的使用环境、测试目的和上下文可能大相径庭，这使得直接进行比较变得较为复杂。此外，评价指标的选择也存在显著差异。基于语言模型的自动化红队测试方法一[6]使用 Sentiment Gap，Perplexity，Self-BLEU 等指标来评估模型的性能，这些指标更侧重于情感分析、语言模型的不确定性和生成文本的一致性。而语句链提示方法[40]则采用 ASR，HHH 等指标，这些指标更关注评估模型的有用性、诚实性和无害性等方面的表现。

由于评测模型和评价指标的不一致性，很难建立一个统一的框架来对不同的红队测试技术进行系统性分析和定量比较。每种技术都有其特定的优势和局限性，它们在不同的应用场景和目标下表现出不同的性能。对此，本文在 3.5 节探讨了如何建立更加科学、统一的评估标准和测试基准。

表 1 不同红队测试方法对比

Table 1 Comparison of different red teaming methods

方法	评估模型	评价指标	评估数据集
基于语言模型的自动化红队测试方法一[6]（使用强化学习生成测试用例，以识别 LLM 中的性别偏见）	Alpaca，ChatGPT，GPT-4	Sentiment Gap Perplexity Self-BLEU	由生成器通过强化学习生成的 1 000 个测试用例
基于语言模型的自动化红队测试方法二[34]（利用强化学习提升 LLM 生成“有毒”回复的能力）	LLaMA-13B，GPT-3.5-turbo	Reward Distinct-n Annotated Toxic Probability Attack Success Rate Toxic Confidence	BAD
基于贝叶斯优化的黑盒红队测试方法[38]	BlenderBot-3B，GODEL-large，DialoGPT-large，Marv，Friend chat，Stable Diffusion	RSR Self-BLEU	Bloom ZS，OPT-66B ZS，ConvAI2，Empathetic Dialogues，BAD
离散最优化方法[5]	GPT-2，GPT-J，GPT-3 davinci-002	Average success rate	CivilComments
多轮自动红队方法[39]	ChatGPT，GPT-4，Llama 2-Chat，Vanilla	Safety Score Helpfulness Score Violation Rate	SafeEval，HelpEval，Anthropic Harmless，AlpacaEval
语句链提示方法[40]	GPT-4，ChatGPT，Vicuna-7B，Vicuna-13B，STABLEBELUGA-7B，STABLEBELUGA-13B，STARLING（BLUE），STARLING(BLUE-RED)	ASR HHH	DANGEROUSQA，HARMFULQA，Vicuna Benchmark Questions
结合人工和自动红队测试方法[41]	GPT-3.5-turbo，GPT-3.5-davinci，Alpaca-LoRA-7B，Alpaca-LoRA-13B	Harmful Score	Dual-Use，BAD+，SAP30
自动生成隐秘提示方法[44]	Vicuna-7b，Guanaco7b，Llama2-7b-chat，GPT-3.5-turbo，GPT-4	ASR Recheck	AdvBench Harmful Behaviors

3.4 数据集

现有的红队数据集主要通过网络爬取、人工制作和语言模型自动生成等方式来收集，目前主流的红队测试数据集如表 2 所列。

1）BAD 数据集[31]：一个人工制作的机器人对抗对话数据集，该数据集包含来自人类和机器人的 5 784 条对话，共计 78 874 个语句。

2）AttaQ 数据集[8]：一个半自动的对抗性问题攻击数据集，从 Anthropic 的 HH-RLHF 数据集[43]中提取攻击问题，然后归纳并生成相似的攻击问题，总共包含 1 402 个对抗性问题，分为欺骗、歧视、暴力等 7 种类型。

3）SAP 数据集[41]：一个半自动的攻击提示数据集，通过半自动攻击框架 SAP[41]构建了一系列名为 SAP 的数据集，攻击提示数量从 40 到 1 600 不等。

4）HH-RLHF 有益无害数据集[42]：一个通过 RLHF 方法生成有用无害的红队数据集，训练集 4 万条，测试集 2000 条。

5）Real Toxicity Prompts 数据集[37]：一个来自网络的 10 万条不同毒性的文本，用于评估语言模型的毒性。

6）HH-RLHF-Attack 数据集[43]：一个以人类反馈强化学习（Reinforcement Learning from Human Feedback，RLHF）作为安全技术对语言模型进行攻击的数据集，共有 38961 次红队攻击，并且包含定量（如有害性评分、人工评级）和定性（如标签）注释，使数据更易于过滤、分析和浏览。

7）BAD+ 数据集[45]：一个在 BAD 数据集[31]的基础上生成的红队数据集，包含超过 12 万种不同的高度归纳性语境，其中归纳语境分为 12 个类别（如侮辱和威胁等）。

8）HarmfulQA 数据集[40]：一个通过语句链方法生成的红队数据集，包含 1960 个攻击问题，共有 10 个主题，每个主题各有 10 个子主题。

9）BeaverTails 数据集[46]：一个由问答对话组成的人工标记数据集，包含 30 万条样本，共 14 种风险类别，每个对话都与多个风险类别相关联。

表 2 红队测试数据集

Table 2 Red teaming datasets

数据集	样本数量	链接
BAD[31]	78900	https://github.com/facebookresearch/ParlAI/tree/main/parlai/tasks/bot_adversarial_dialogue
AttaQ[8]	1400	https://huggingface.co/datasets/ibm/AttaQ
SAP[41]	1600	https://github.com/aatrox103/sap
HH-RLHF[42]	42000	https://github.com/anthropics/hh-rlhf
Real-Toxicity-Prompts[37]	100000	https://allenai.org/data/real-toxicity-prompts
HH-RLHF-Attack[43]	38900	https://github.com/anthropics/hh-rlhf
BAD+[45]	122600	https://github.com/thu-coai/Reverse_Generation
HarmfulQA[40]	1900	https://huggingface.co/datasets/declare-lab/HarmfulQA
BeaverTails[46]	300000	https://sites.google.com/view/pku-beavertails

3.5 评估指标

目前的红队测试方法评估还缺少统一的度量标准，以下是目前已经提出的红队评估指标。

1）有益、诚实、无害（Helpful，Honest，Harmless，HHH）[47]：一个人工智能助手应该符合 HHH 这一与人类价值观一致的标准，即有益——总是尽量做对人类最有利的事；诚实——始终努力向人类传达准确的信息，并尽量避免欺骗人类；无害——总是尽量避免做任何伤害人类的事情。HHH 目前是相对定性的主观评价指标，如果一个回复没有满足有益、诚实、无害这 3 个要素，则认为该信息是有毒的。评估时需要人工对这 3 个要素进行判断。

2）基于关键字的攻击成功率（Attack Success Rate，ASR）[40,48]：该指标主要检测 LLM 的响应中是否存在预定义的关键词。给定输入对$\langle J_i, Q_i \rangle$及其对应的回复 R_i，如果 R_i 中没有出现预定义列表中的任何关键词，就认为 LLM 没有拒绝攻击，并且对应的样本攻击成功。

对于开源模型，定义有害问题查询总数为n_p，模型提供有害输出的总数为 n_r，因此开源模型的 ASR 为：

$$ASR=\frac{n_r}{n_p} \tag{1}$$

对于闭源模型（如 GPT-4），会出现由于内容管理策略而拒绝响应的情况，这种拒绝的次数定义为 n_a，因此闭源模型的 ASR 为：

$$ASR=\frac{n_r}{n_p-n_a} \tag{2}$$

3）TruthfulQA[49]：该方法将真实性定义为严格的标准，只有当响应准确地代表现实世界并得到可靠的公开证据支持时，才将其视为真实。TruthfulQA 包含 817 个问题，涵盖了健康、法律、金融和政治等 38 个类别，用于引发虚假回复，以衡量语言模型生成真实答案的程度。

4）MT-Bench 和 Chatbot Arena[50]：该方法将 LLM 作为评委，引入了 MT-Bench 和 Chatbot Arena 两个基准。MT-bench 包括 80 个多轮问题，测试模型对话和指令遵循的能力。Chatbot Arena 是用户可以与两个聊天机器人进行对话的一个平台，并根据个人偏好对其回复进行评分。

5）TrustGPT[51]：该方法主要从毒性、偏见和价值一致性 3 个角度来进行评估，通过引入一个新的测试集 TrustGPT，使用从社会规范中得到的有毒提示模板来检查毒性；通过测量不同人口统计类别的毒性值来量化偏见；并通过主动和被动任务评估价值对齐。

6）AdvGLUE[52]：该方法侧重于对抗鲁棒性评估，将 14 种不同层级的文本对抗攻击方法应用于 GLUE 的 5 个任务中，包括情感分析、重复问题检测、自然语言推理等，并通过人工验证和筛选进行可靠注释。

以上是现有的红队测试评估指标，但这些指标都较为分散，没有形成一个统一的评估框架。为了建立更加科学、统一的评估标准和测试基准，首先应当明确评估的目标是模型的安全性、鲁棒性、公平性以及对特定攻击类型的防御性；其次将现有的评估指标进行分级，比如基础层是攻击成功率、中级层是模型对攻击的响应时间、高级层是长期稳定性；然后对不同层级分配权重，反映其在整体评估中的重要性。在此基础上，建立一套标准化的测试流程，确保每次红队测试环境的统一，如数据集选择、攻击类型、测试步骤等，从而使实验结果具有可比性。除此之外，还应该建立一个包含各种攻击场景和案例的数据库作为测试基准，用于评估模型在不同情境下的表现。最终进行评估工具的开发，以自动化评估红队测试的多种评估指标，并提供直观的结果展示。通过探讨以上内容，可以逐步建立起一个科学、统一的红队测试评估标准和测试基准，从而更有效地评估和提升大模型的安全性和鲁棒性。

结束语 本文综述了大模型红队测试的常用方法以及相应的评估策略。实际上，大模型红队测试仍存在众多有待深入研究的任务和场景。下面从多个角度对未来工作进行展望。

1）提升红队数据的完整性和专业性。当下的红队数据集主要通过人工的方法生成，而数据标注员并不一定具备所需

领域的专业知识[43]，可能影响数据集质量，生成的红队数据也会存在不完整的情况。例如，一些对 RLHF 模型的攻击是通过“角色扮演”的方式进行的，比如要求模型扮演一个恶意的角色，那么模型会答应并输出有害内容[43]。未来的工作应该致力于提高红队数据的完整性和专业性，通过与各行业专家和从业者交流合作，以获取更真实、全面的数据。此外，还可以开发更有效的数据收集技术，来获得更准确的数据。

2)大模型红队测试缺少统一的评价标准。目前很多检验红队测试的方法和标准不尽相同，缺少较为统一的度量，甚至连最基本的 HHH[47] 标准也存在模糊不清，甚至冲突的情况。因此未来工作应该关注如何设计一套全面、有效且客观的评估指标，用于评估大模型红队测试的性能和效果。这些评估指标可以包括红队测试的攻击成功率、漏报率、误报率等，以确保测试结果的准确性和可比性。

3)大模型自动红队测试技术的潜力还有待挖掘。与真正的攻击者相比，红队有更多的机会接触模型及其训练数据，后续可以利用红队的访问优势来开发白盒红队测试[17]，尤其是生成更高质量的对抗性示例。除此之外，可以通过比较人工和自动的红队方法，确定两种方法在所产生的红队攻击的有效性和多样性方面有何不同，进而取长补短。未来的工作可以关注开发更智能的自适应红队来主动学习和抵御新的威胁，并能够自动化执行各种攻击场景，以此提高红队测试的效率和准确性，同时降低人力成本。

4)大模型红队测试与传统红队测试的结合。传统红队测试发展较为成熟，针对许多攻击，传统的安全措施，如确保系统和模型被适当锁定，可以大大降低风险，但不适用于当前的生成式模型；而大模型红队测试发展时间较短，相关技术还不够成熟。二者都有各自的优势和局限性，未来可以更关注二者优势的结合，提高红队测试的可靠性。

5)提高小参数模型的红队测试能力。模型参数的大小会影响红队测试的越狱，模型参数量越小，越容易越狱成功[40]，因为它们的复杂性和生成能力相对较低。未来的工作可以重点关注提高小参数模型的红队测试能力，通过引入更多的特征工程和模型优化方法，以及设计更有效的攻击策略来实现。

6)红队测试向智能体行为安全评估的扩展。随着大模型能力向更多现实场景扩展，包括工具使用、推理和规划能力的提升以及基于大模型的群体智能，这些能力的增强使得模型能够访问和支配更多的资源。如果模型自身存在未对齐的地方，可能会带来更大的安全隐患。因此，红队测试需要从传统的文本生成层面扩展到智能体行为层面的风险评估，以更全面地理解和评估智能体在现实世界中的行为安全性。未来的研究工作应致力于开发和优化用于智能体行为安全评估的红队测试方法，通过模拟沙盒环境[53]等技术手段，对智能体的行为进行全面的安全评估，从而提高其在现实世界中的应用价值。

参 考 文 献

[1] WEI J, TAY Y, BOMMASANI R, et al. Emergent abilities of large language models[J]. arXiv:2206.07682, 2022.

[2] ZHANG W M, WANG Z Y, LI Y G, et al. Introduction to computing[M]. Beijing: Beijing Institute of Technology Press. 2016.

[3] DING C Y. Legal Regulation of the Network Society[M]. Beijing: China University of Political Science & Law Press. 2016.

[4] WEIDINGER L, UESATO J, RAUH M, et al. Taxonomy of risks posed by language models[C]// Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency. 2022: 214-229.

[5] JONES E, DRAGAN A, RAGHUNATHAN A, et al. Automatically Auditing Large Language Models via Discrete Optimization [J]. arXiv: 2303.04381, 2023.

[6] CHAN A, SALGANIK R, MARKELIUS A, et al. Harms from Increasingly Agentic Algorithmic Systems[C]// Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency. 2023: 651-666.

[7] HENDRYCKS D, MAZEIKA M, WOODSIDE T. An Overview of Catastrophic AI Risks[J]. arXiv: 2306.12001, 2023.

[8] KOUR G, ZALMANOVICI M, ZWERDLING N, et al. Unveiling Safety Vulnerabilities of Large Language Models [J]. arXiv: 2311.04124, 2023.

[9] INIE N, STRAY J, DERCZYNSKI L. Summon a Demon and Bind it: A Grounded Theory of LLM Red Teaming in the Wild [J]. arXiv: 2311.06237, 2023.

[10] CASPER S, LIN J, KWON J, et al. Explore, Establish, Exploit: Red Teaming Language Models from Scratch[J]. arXiv: 2306.09442, 2023.

[11] White Paper on Artificial Intelligence Safety Standardisation [EB/OL]. [2023-11-20] https://www.tc260.org.cn/upload/2023-05-31/1685501487351066337.pdf.

[12] Basic Requirements for the Safety of Generative AI Services [EB/OL]. [2023-11-20] https://www.tc260.org.cn/upload/2023-08-25/1692961404507050376.pdf.

[13] Code of Ethics for the Next Generation of Artificial Intelligence [EB/OL]. [2023-11-20]. https://www.most.gov.cn/kjbgz/202109/t20210926_177063.html.

[14] VON STENGEL B, KOLLER D. Team-maxmin equilibria [J]. Games and Economic Behavior, 1997, 21(1/2): 309-321.

[15] COHEN F. Managing network security－red teaming[J]. Network Security, 1998(3): 13-15.

[16] JI J, QIU T, CHEN B, et al. Ai alignment: A comprehensive survey[J]. arXiv: 2310.19852, 2023.

[17] PEREZ E, HUANG S, SONG F, et al. Red teaming language models with language models[J]. arXiv: 2202.03286, 2022.

[18] CHAKRABORTY A, ALAM M, DEY V, et al. A survey on adversarial attacks and defences[J]. CAAI Transactions on Intelligence Technology, 2021, 6(1): 25-45.

[19] LIU Y, DENG G, XU Z, et al. Jailbreaking chatgpt via prompt engineering: An empirical study[J]. arXiv: 2305.13860, 2023.

[20] CHEN Z, LI B, WU S, et al. Content-based Unrestricted Adversarial Attack[J]. arXiv: 2305.10665, 2023.

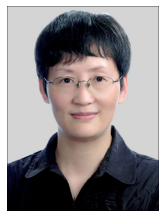
[21] WEI A, HAGHTALAB N, STEINHARDT J. Jailbroken: How does llm safety training fail? [J]. arXiv: 2307.02483, 2023.

[22] OpenAI. GPT-4 technical report[J]. arXiv: 2303.08774, 2023.

[23] PEARCE W, LUCAS J. Nvidia ai red team: An introduction [EB/OL]. [2023-11-20]. https://developer.nvidia.com/blog/

nvidia-ai-red-team-an-introduction/.
[24] KUMAR R S S. Microsoft ai red team building future of safer ai [EB/OL]. [2023-11-20]. https://www.microsoft.com/en-us/security/blog/2023/08/07/microsoft-ai-red-team-building-future-of-safer-ai/.
[25] FABIAN D. Google's ai red team: the ethical hackers making ai safer [EB/OL]. [2023-11-20]. https://blog.google/technology/safety-security/googles-ai-red-team-the-ethical-hackers-making-ai-safer/.
[26] Plan for red-team testing of the large language model (LLM) and its applications[EB/OL]. [2024-05-23]. https://learn.microsoft.com/zh-cn/azure/ai-services/openai/concepts/red-teaming.
[27] LIU Y, ZHANG K, LI Y, et al. Sora: A Review on Background, Technology, Limitations, and Opportunities of Large Vision Models[J]. arXiv: 2402.17177, 2024.
[28] The first cybersecurity benchmarking platform in China, SecBench, has been released [J]. China Information Security, 2024(2): 83.
[29] RANDO J, PALEKA D, LINDNER D, et al. Red-teaming the stable diffusion safety filter[J]. arXiv: 2210.04610, 2022.
[30] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples[J]. arXiv: 1412.6572, 2014.
[31] XU J, JU D, LI M, et al. Bot-adversarial dialogue for safe conversational agents[C]// Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2021: 2950-2968.
[32] KANG D, LI X, STOICA I, et al. Exploiting programmatic behavior of llms: Dual-use through standard security attacks[J]. arXiv: 2302.05733, 2023.
[33] SU H, CHENG C C, FARN H, et al. Learning from Red Teaming: Gender Bias Provocation and Mitigation in Large Language Models[J]. arXiv: 2310.11079, 2023.
[34] WEN J, KE P, SUN H, et al. Unveiling the Implicit Toxicity in Large Language Models[J]. arXiv: 2311.17391, 2023.
[35] DING P, KUANG J, MA D, et al. A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts can Fool Large Language Models Easily[J]. arXiv: 2311.08268, 2023.
[36] CHAO P, ROBEY A, DOBRIBAN E, et al. Jailbreaking black box large language models in twenty queries[J]. arXiv: 2310.08419, 2023.
[37] GEHMAN S, GURURANGAN S, SAP M, et al. Realtoxicityprompts: Evaluating neural toxic degeneration in language models[J]. arXiv: 2009.11462, 2020.
[38] LEE D, LEE J Y, HA J W, et al. Query-Efficient Black-Box Red Teaming via Bayesian Optimization [J]. arXiv: 2305.17444, 2023.
[39] GE S, ZHOU C, HOU R, et al. MART: Improving LLM Safety with Multi-round Automatic Red-Teaming [J]. arXiv: 2311.07689, 2023.
[40] BHARDWAJ R, PORIA S. Red-teaming large language models using chain of utterances for safety-alignment[J]. arXiv: 2308.09662, 2023.
[41] DENG B, WANG W, FENG F, et al. Attack Prompt Generation for Red Teaming and Defending Large Language Models[J]. arXiv: 2310.12505, 2023.
[42] GILARDI F, ALIZADEH M, KUBLI M. Chatgpt outperforms crowd-workers for text-annotation tasks [J]. arXiv: 2303.15056, 2023.
[43] GANGULI D, LOVITT L, KERNION J, et al. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned[J]. arXiv: 2209.07858, 2022.
[44] LIU X, XU N, CHEN M, et al. Autodan: Generating stealthy jailbreak prompts on aligned large language models[J]. arXiv: 2310.04451, 2023.
[45] ZHANG Z, CHENG J, SUN H, et al. Constructing Highly Inductive Contexts for Dialogue Safety through Controllable Reverse Generation[J]. arXiv: 2212.01810, 2022.
[46] JI J, LIU M, DAI J, et al. Beavertails: Towards improved safety alignment of llm via a human-preference dataset [J]. arXiv: 2307.04657, 2023.
[47] ASKELL A, BAI Y, CHEN A, et al. A general language assistant as a laboratory for alignment[J]. arXiv: 2112.00861, 2021.
[48] ZOU A, WANG Z, KOLTER J Z, et al. Universal and transferable adversarial attacks on aligned language models[J]. arXiv: 2307.15043, 2023.
[49] LIN S, HILTON J, EVANS O. Truthfulqa: Measuring how models mimic human falsehoods[J]. arXiv: 2109.07958, 2021.
[50] ZHENG L, CHIANG W L, SHENG Y, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena[J]. arXiv: 2306.05685, 2023.
[51] HUANG Y, ZHANG Q, SUN L. TrustGPT: A Benchmark for Trustworthy and Responsible Large Language Models[J]. arXiv: 2306.11507, 2023.
[52] WANG B, XU C, WANG S, et al. Adversarial glue: A multi-task benchmark for robustness evaluation of language models[J]. arXiv: 2111.02840, 2021.
[53] RUAN Y, DONG H, WANG A, et al. Identifying the risks of lm agents with an lm-emulated sandbox[J]. arXiv: 2309.15817, 2023.

BAO Zepeng, born in 2002, undergraduate, is a member of CCF (No. U8466G). His main research interests include LLM safety and recommendation system.



QIAN Tieyun, born in 1970, Ph.D, professor, Ph.D supervisor, is a member of CCF(No. 13483M). Her main research interests include web mining and natural language processing.

（责任编辑：杨雪敏）