循环诱导驱动的多模态大语言模型可用性压力测试方法

刘小垒[1],李星煜[1*],陈 斤[2],张小松[2]
(1. 国家工程物理交叉科学研究中心,四川绵阳621000;2. 电子科技大学计算机学院,四川成都 610054)

摘 要: 多模态大语言模型凭借强大的跨模态理解能力,已在图像描述、视频问答及视觉推理等任务中取得显著成效。然而,数千亿级别的参数规模导致模型在实际部署中消耗大量计算资源,使其推理服务面临资源耗尽型攻击的威胁。针对此类威胁开展压力测试,是评估和提升系统可用性的重要手段。现有方法主要通过推迟生成结束词元的出现来延长模型输出,但仍面临测试强度有限与优化效率低下的问题。一方面,随着生成序列增加,输入扰动对远端词元的影响持续衰减,输出往往在达到最大长度前提前终止;另一方面,现有方法通常需联合优化多个损失函数,导致优化过程耗时长且收敛困难。针对上述问题,本文提出一种面向多模态大语言模型的高效压力测试框架CycleTrap。该框架的核心思想是通过显式干预模型的输出结构,诱导其进入自我强化的循环生成状态,持续输出重复内容直至达到最大生成长度,从而显著放大推理能耗和响应时延,模拟极端推理负载场景。具体而言,本文首先设计了上下文感知的种子提取机制,基于模型对原始视觉输入的初始响应,选取其中对数似然累积得分最高的连续子序列作为初始重复种子,以提高循环诱导的成功率。在此基础上,本文进一步提出循环诱导优化机制,通过单一损失函数提升重复种子在输出中被再次生成的概率,并借助自回归生成过程中历史输出对当前预测的累积效应,使模型输出概率分布逐步收敛到少数词元,最终形成稳定的循环生成模式。相比于延迟EOS的方法,CycleTrap不再以被动延长输出为目标,而是通过主动诱导循环结构实现持续生成,因而能更稳定地触发最大输出长度,同时降低测试样本的优化开销。本文在四个主流多模态模型以及多个图像与视频数据集上进行了大量实验评估,结果表明,CycleTrap在测试强度与优化效率方面均显著优于现有基线方法。在输出长度方面,其诱导生成的平均输出长度最高可达原始输入的20倍,并在部分设置下实现100%的达到最大长度触发率;在优化开销方面,生成单个测试样本的平均时间约为基线方法的七分之一,展现出优异的优化效率。此外,CycleTrap在不同解码策略、扰动强度及重复惩罚机制下均表现出较强效果,表明其对生成策略变化具有良好的鲁棒性。

关键词: 多模态大语言模型;压力测试;可用性安全;对抗样本;循环诱导;自回归生成;自我强化循环

基金项目: 国防科工局核科学挑战计划专题资助(No.TZ2025002);国家自然科学基金(No.U2441239,No.62302468)

中图分类号: TP391.41;TP18 **文献标识码:** A **文章编号:** 0372-2112(XXXX)XX-0001-14

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20260444

Availability Stress Testing of Multimodal Large Language Models via Loop Induction

LIU Xiaolei[1], LI Xingyu[1*], CHEN Ting[2], ZHANG Xiaosong[2]
(1. *National Interdisciplinary Research Center of Engineering Physic*, *Mianyang*, *Sichuan* 621000, *China*;
2. *School of Computer Science and Engineering*, *University of Electronic Science and Technology of China*, *Chengdu*, *Sichuan* 610054, *China*)

Abstract: Multimodal large language models, benefiting from their strong cross-modal understanding capabilities, have achieved remarkable performance in tasks such as image captioning, video question answering, and visual reasoning. However, their hundreds-of-billions parameter scale leads to substantial computational resource consumption in real-world deployment, making inference services vulnerable to resource-exhaustion attacks. Conducting stress testing against such threats is therefore an important means of evaluating and improving system availability. Existing methods primarily prolong model outputs by delaying the appearance of the end-of-sequence token, yet they suffer from limited testing intensity and low optimization efficiency. On the one hand, as the generated sequence grows longer, the influence of input perturbations on distant tokens continues to diminish, often causing outputs to terminate prematurely before reaching the maximum generation length. On the other hand, these methods typically require joint optimization of multiple loss functions, resulting in

收稿日期:2026-05-21;录用日期:2026-06-19;责任编委:宾燚
*通信作者:李星煜

time-consuming optimization and difficulty in convergence. To address these issues, this paper proposes CycleTrap, an efficient stress testing framework for MLLMs. The core idea of this framework is to explicitly intervene in the model´s output structure to induce a self-reinforcing cyclic generation state, where the model continuously emits repetitive content until the maximum generation length is reached. This significantly amplifies inference energy consumption and response latency, thereby simulating extreme computational load scenarios. Specifically, this paper first design a context-aware seed extraction mechanism that selects the continuous subsequence with the highest cumulative log-likelihood score from the model´s initial response to the original visual input as the initial repetition seed, thereby improving the success rate of loop induction. Building upon this, this paper further propose a loop induction optimization mechanism that employs a single loss function to increase the probability of the repetition seed being regenerated in the output. By leveraging the cumulative effect of historical outputs on current predictions during the autoregressive generation process, the model´s output probability distribution gradually converges toward a small set of tokens, ultimately forming a stable cyclic generation pattern. Compared with EOS-delaying methods, CycleTrap does not passively prolong outputs; instead, it achieves sustained generation by actively inducing cyclic structures, enabling more stable triggering of the maximum output length while reducing the optimization overhead of test samples. Extensive experiments conducted on four mainstream multimodal models across multiple image and video datasets demonstrate that CycleTrap significantly outperforms existing baselines in both testing strength and optimization efficiency. Regarding output length, the average induced output length reaches up to 20 times that of the original input, with a 100% maximum-length triggering rate achieved under certain settings. In terms of optimization overhead, the average time to generate a single test sample is approximately one-seventh that of baseline methods, exhibiting excellent optimization efficiency. Furthermore, CycleTrap maintains strong effectiveness under different decoding strategies, perturbation intensities, and repetition penalty mechanisms, indicating robust performance across variations in generation strategies.

Keywords: multimodal large language models; stress testing; availability security; adversarial examples; loop induction; autoregressive generation; self-reinforcing loop

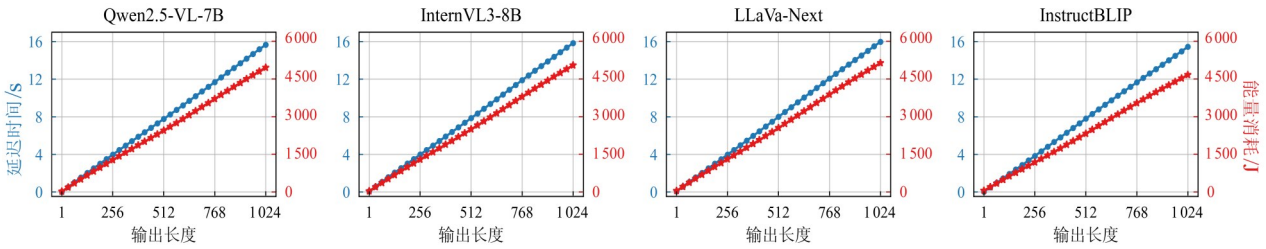
Foundation Item(s): Science Challenge Project (No. TZ2025002); National Natural Science Foundation of China (No.U2441239, No.62302468)

0 引言

多模态大语言模型(Multimodal Large Language Models, MLLMs)(后文简称“多模态大模型”)将视觉表征能力与大语言模型的语义理解能力互相结合,在图像描述[1-3]、视频问答[4-6]及视觉推理[7-8]等跨模态理解任务中取得了优异的成效。但是这些性能在很大程度上都依赖于数千亿级别的参数规模,所以模型在实际部署的过程当中会占用到大量的计算资源[9-11]。随着模型即服务(model-as-a-service)商业模式的普及,推理效率已经逐渐成为衡量系统可用性的一个重要指标,由此带来的安全隐患问题也受到了许多研究学者的关注[12-13]。目前,OWASP组织已经将“无限资源消耗”列为大模型十大安全威胁之一。具体到多模态大模型来说,攻击者或测试者可以利用特定的视觉输入,诱使模型出现异常的推理行为,生成远超正常长度的响应。最终导致能量消耗和响应延迟异常增加,严重的甚至造成拒绝服务[14-15](Denial of Service, DoS)。因此,构造出可以复现的极端测试场景,评估模型在高负载条件下的行为边界,对于设计模型部署时的安全防御机制具有很重要的意义。

当前主流的多模态大模型普遍都采用了模块化架构设计,通过视觉编码器和特征投影层把图像或视频这类视觉输入转换成视觉嵌入,与文本嵌入拼接后,交给大语言模型来生成响应内容[16-20]。由于语言模型本身固有的自回归生成特性[21],输出序列中的每个词元都需要经过一次完整的前向传播过程,并且还会作为后续生成的上下文继续参与计算。因此,模型推理的能量消耗与响应延迟往往会随着输出长度的增加而线形增加[22],图1中的实验结果也印证了这一结论。换言之,测试者只需要通过诱导模型生成冗长的输出内容,就可以放大推理开销并且降低服务的可用性。由此可见,深入理解并复现上述的极端场景,是设计有效防御机制的基础。

目前针对多模态大模型可用性压力测试方法,主要思路是:先降低模型对生成结束词元(End-Of-Sequence, EOS)的预测概率,再把这个信号作为梯度来源去迭代更新测试样本[23-27],从而推迟EOS的出现来实现增加输出长度。不过,这类方法存在两个方面的局限性:(1)测试强度有限。延迟EOS的策略并没有从根本上改变模型的生成机制。随着生成序列的增加,输入扰动对远端词元的影响会不断地衰减,难以有效控制远端的生成行为,最终导致输出通常在没有达到最大长度前就提前结束了,输出长度增加的效



注:图中展示了输出序列长度与系统响应延迟及能源消耗之间的关系。其中能耗值通过 NVIDIA 管理库(NVML)实时采集,延迟时间表示完整推理过程的端到端消耗时间。

图 1 针对四种多模态大模型的推理开销测试结果

Figure 1 Inference overhead evaluation results for four multimodal large language models

果并不明显;(2)优化效率较低。为了实现对 EOS 概率的有效抑制,现有方法通常需要同时优化多个损失函数,使得整个优化过程不仅耗时长,而且难以收敛,从而限制了方法的实用性与可扩展性。

针对上述问题,本文提出一种面向多模态大模型可用性评估的高效压力测试框架 CycleTrap。CycleTrap 框架的核心思想是:显式干预模型的输出结构,诱导模型进入一种自我强化的循环生成状态,不断输出重复内容,直至达到最大生成长度。由此,可以模拟出极端的计算负载场景,将模型在高压条件下的可用性边界充分暴露出来。具体而言,本文通过利用自回归生成过程中历史输出对当前预测的影响,设计了一种循环诱导机制,迫使模型输出概率分布逐步收敛到少数的词元上,进而形成稳定的循环模式。此外,本文还引入了上下文感知的种子提取机制,它会根据模型原始的输出动态选取初始的重复词元,使所选种子更贴合当前的上下文分布,从而有效提高对抗性测试样本诱导循环的成功率。相比于延迟 EOS 的方法,CycleTrap 只需要用单一损失函数来进行优化,既能稳定地触发最大输出长度,同时又能大幅降低测试样本的生成开销,为系统性评估多模态大模型的推理可用性提供了更高效的技术支撑。

综上所述,本文的贡献体现在以下三个方面:

(1)本文利用自回归推理机制的脆弱性,将模型输出引导至自我强化的循环状态,并据此构造循环生成型压力测试场景,为揭示模型可用性安全风险提供了新的视角。

(2)本文提出上下文感知的种子提取与循环诱导策略,在保持较高测试成功率的同时,缩短测试样本的优化时间,提升了可用性评估的整体效率。

(3)本文在多个图像、视频数据集和多种模型上开展了大量的实验,实验结果均显示 CycleTrap 的效果明显优于现有的基线方法,其诱导产生的输出长度最高可达原始输入的 20 倍,而优化开销仅为基线方法的七分之一。

1 相关工作

1.1 多模态大模型

多模态大模型一般是在大语言模型前接入一个视觉处理模块,使模型在进行文本生成时也能运用图像或视频里的信息。这个模块通常由两部分构成,一部分是视觉编码器,另一部分是特征投影层。视觉编码器负责从原始图像或视频流里提取视觉特征,通常需要借助 CLIP[28]、BLIP[29]这类已经预训练好的大规模视觉模型。特征投影层是跨模态对齐的关键部分,它的作用是将视觉特征映射到文本嵌入空间,其结构设计会直接影响后续的视觉感知和文本生成效果。不同模型在投影层的设计上具有各自的特点。例如 LLaVA[16],只通过利用全连接层就能将视觉特征与文本嵌入融合到一起。InstructBLIP[30]则通过引入一个指令引导的转换机制,然后依据上下文动态提取相关的视觉信息,让模型对不同的任务具备更强的适应性。最近出现的 Qwen2.5-VL[31]系列提出利用交叉注意力机制来处理不同分辨率图像,而同一时期的 InternVL3-8B[32]则把原生多模态预训练和基于 MPO 的推理优化结合起来,在多项基准任务中均表现出了优异的性能。虽然上述的集成技术拓展了模型的感知边界,但同时也在视觉输入层面带来了新的安全脆弱性[33]。本文针对这一潜在的安全脆弱性开展了可用性评估研究。具体做法是通过构造一些对抗性视觉测试样本,让模型产生重复且冗长的输出,用这种方式去模拟极端推理负载的场景,从而找出系统可用性的瓶颈,为后续设计防御机制提供一些实际的依据。

1.2 模型可用性压力测试

面向模型可用性的压力测试,核心是构造极端输入来放大推理阶段的计算负载,从而评估模型在高压条件下的可用性边界[34]。这一问题与传统网络安全领域中的 DoS 防御研究[35]在逻辑上具有相通之处。在早期的研究里,Shumailov 等人[36]已经在文本翻译和图像分类任务中验证了这种压力测试的可行性。

他们的做法是通过构造对抗性的测试样本，降低神经网络激活的稀疏程度，从而增加内部的计算开销，延长推理时间。在此之后，这个思路被推广到多个应用领域。在音频处理方面，研究者向原始音频添加扰动噪声，使语音转文本模型陷入异常状态，生成大量冗余文本[37-38]。在自然语言处理领域方面，则是利用恶意提示词诱导大语言模型持续输出[39-40]。在自动驾驶场景中，对抗测试样本通过增加冗余目标来拖慢视觉模块的实时处理速度，这类测试具有更强的现实威胁性[41-42]。

就本文所关注的计算机视觉领域而言，这一思路最早是应用在图像描述任务中。例如，NICGSlow-Down[23]通过操纵输入图像来增加解码器的调用次数，评估小规模CNN-RNN图像描述模型在高负载场景下的能量延迟代价。在多模态大模型出现之后，测试手段逐渐从简单的计算干扰转向对生成逻辑的探测。Gao等人提出的Verbose Images[24]通过梯度优化降低EOS预测概率来诱导模型生成冗长的内容，以此测量模型在异常输出场景下的推理成本上限。类似的技术也被用于视频多模态大模型的可用性评估[25]。近期，Fu等人[26]提出的LingoLoop利用不同位置的词性信息动态推迟EOS的生成，在模拟极端场景方面取得了更好的效果。不过上述方法仍将抑制EOS作为核心目标，通常需要多个损失函数共同驱动，优化成本较高，而且很难稳定地将模型输出推到最大的输出长度，测试覆盖率有限。相比之下，本文从自回归机制本身出发，通过构造重复吸引子来重塑模型的输出结构，迫使模型不断生成重复内容。同时这个设计的优化目标只有单一的损失函数，不仅使优化过程更加简洁高效，还能更稳定地驱动模型输出到最大长度。这为评估多模态大模型在极端负载条件下的可用性边界提供了一个更有力的基准工具。

2 方法实现

2.1 问题定义

2.1.1 多模态大模型流程

现代多模态大模型通过融合视觉模态扩展了大语言模型的功能，从而实现了视觉上下文感知交互。形式上，给定义一个视觉输入序列$\boldsymbol{X}=\{x_1,x_2,\cdots,x_M\}$，其中$M$表示从视频中均匀采样的总帧数。为保持符号一致性，静态图像可视为单帧视频（即$M=1$）的特例，这一处理方式有助于在统一的形式化框架下同时描述图像与视频两种输入模态。给定文本输入$\boldsymbol{c}_{\text{in}}$以及多模态大模型$f$，模型以自回归方式生成输出词元序列$\boldsymbol{Y}=\{y_1,y_2,\cdots,y_N\}$，其中$y_i$代表第$i$个生成的词元，$N$为序列长度，整个过程的流程图如图2所示。在自回归生成机制下，每个词元均以此前所有已生成词元以及原始输入序列为条件进行预测，则第i个输出位置的概率分布经Softmax(·)归一化后表示为

$$P_i\left(\boldsymbol{c}_{\text{in}},\boldsymbol{X},\boldsymbol{Y}_{<i}\right)=\text{Softmax}\left(f\left(\boldsymbol{c}_{\text{in}},\boldsymbol{X},\boldsymbol{Y}_{<i}\right)\right) \tag{1}$$

其中，$P_i(\cdot)\in\mathbb{R}^V$，$V$是模型词元的总数；$f(\cdot)$表示多模态大模型完整的前向传播过程。获得第$i$个位置的预测概率分布后，依据所选解码策略对该分布进行采样得到当前词元，并将其拼接至历史上下文序列以供下一步前向传播使用。最后，当生成词元为EOS或达到预设最大长度时，生成过程终止。

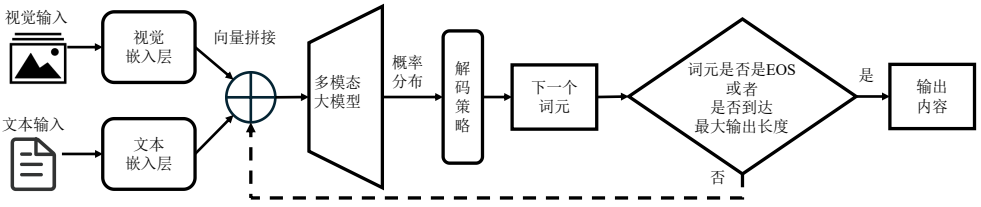


图2 多模态大模型自回归推理的流程图

Figure 2 Workflow of autoregressive inference in multimodal large language models

2.1.2 优化目标

由于模型逐词元自回归生成，输出长度的增加会直接推高推理能耗和响应时延。因此，本文的目标是通过构造测试样本最大化目标模型的输出长度，模拟极端资源消耗场景，来量化高负载条件下的可用性边界。具体而言，本文在原始视觉输入中加入逐像素对抗扰动，获得对抗性测试样本$\boldsymbol{X}'=\{x_1',x_2',\cdots,x_M'\}$，其中$x_1'=x_i+\delta_i$，$\delta_i$表示施加于第$i$帧的测试扰动。测试目标可形式化为对输出长度期望的最大化：

$$\max_{\delta}\ \mathbb{E}_{f(\boldsymbol{c}_{\text{in}},\boldsymbol{X}')}\left[\text{len}(\boldsymbol{Y})\right] \tag{2}$$

然而，词元计数算子len(·)属于不可导的离散函数，不能直接用于梯度优化。本文因此设计了连续可微的损失函数$\mathcal{L}$，通过鼓励模型持续生成稳定的重复内容（区别于直接抑制EOS概率的传统方法），从而间接实现输出长度的增加。最终，优化目标被转化为在扰动幅度受限条件下最小化该损失函数：

$$\min_{\delta}\ \mathcal{L}\left(\boldsymbol{c}_{\text{in}},\boldsymbol{X}'\right),\quad \text{s.t.}\frac{1}{M}\sum_{i=1}^{M}\left\|\delta_i\right\|_p\leqslant\epsilon \tag{3}$$

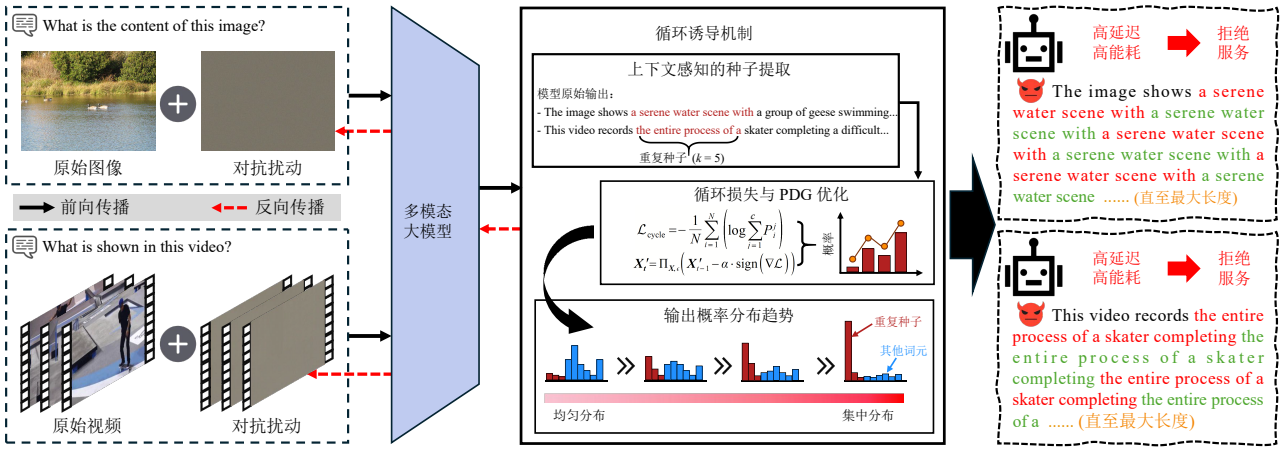
其中,为保证对抗扰动在视觉上的不可感知性,对扰动施加了l_p范数约束;ϵ为扰动幅度的上界。

2.2 测试场景设定

与前人的研究工作相类似[24-26],本文也采用白盒测试的设定。在这个设定下,假设测试者可以完全访问被测模型的内部信息,包括模型架构、参数权重以及梯度信息等。利用梯度优化,测试者可以构造出视觉上无法感知的对抗性测试样本,诱导模型生成异常冗长的响应。这样一来,就可以放大模型的计算开销、能耗和响应延迟,从而将模型在极端负载条件下的短板充分暴露出来。尽管白盒测试的设定较为理想化,但在现如今开源多模态大模型被广泛部署的背景下,这个设定依然具有一定的现实参考价值。例如,Hugging Face、魔搭社区等公共平台,都已提供了基于开源模型的托管与推理接口。但如果这类服务缺乏严格的访问控制,一旦遇到梯度驱动型压力测试时,还是会存在一定的潜在安全隐患[43]。

2.3 方法概述

本文提出的CycleTrap压力测试框架,是一种主要用来面向多模态大模型可用性的压力测试框架,其整体流程如图3所示。CycleTrap框架先在原始视觉输入上施加一点微小扰动,以此得到初始的对抗性测试样;然后利用上下文感知的种子提取机制筛选出初始重复种子,再用循环损失函数去优化扰动。这个损失函数可以利用自回归生成中的内在特征,让模型在生成的过程当中逐步形成循环输出,并不断地自我强化,最终进入一种持续重复的生成状态,以此来模拟极端计算压力下的场景。



注:该框架通过在原始视觉输入上施加对抗扰动,并结合循环诱导机制迭代优化扰动,最终迫使模型持续输出重复内容,从而大量增加能耗和时延导致拒绝服务,以此模拟极端计算压力场景。

图3 CycleTrap框架示意图

Figure 3 Overview of the CycleTrap framework

2.3.1 自回归脆弱性分析

多模态大模型依赖前序上下文进行自回归生成,这一特性有助于维持语义连贯,但也会带来可被利用的脆弱性。当模型在生成过程中出现前文已有的内容时,上下文依赖会进一步提高相关词元在后续位置的预测概率,形成正反馈效应[44-45]。随着重复词元在上下文中的不断累积,输出概率分布将逐渐集中于这些重复词元,迫使模型持续生成这些词元直至达到最大长度。本文正是利用这一自我强化机制构造对抗测试样本,诱导模型重复生成异常冗余内容,来评估模型在极端负载场景下的可用性。

2.3.2 上下文感知的种子提取

为了有效地触发上述的脆弱性,首先需要选取一段初始重复种子,也就是用于诱导模型进入重复输出状态的词元序列。比起直接从词表中随机采样词元的朴素策略,本文从模型针对原始图像生成的响应中选取词元来作为重复种子。这样选取出来的词元,与输入图像在语义上更一致,也更符合模型当前的生成分布。因此后续优化时也更容易被重新激活,从而提高诱导重复的成功率。

在具体实现上,本文首先通过对原始输入执行贪心解码得到原始响应序列$\boldsymbol{Y}$。贪心解码在每一步都选择当前条件概率最大的词元,其输出具有确定性,不受随机采样噪声影响,因而更能反映模型在当前输入下的生成偏好,是获取高置信度候选种子的可靠来源。随后,本文从该响应序列中依次提取所有长度为k的连续子序列,构成候选种子集$S=\left\{\left(y_s,y_{s+1},\cdots,y_{s+k-1}\right)\right\}\ s=1,2,\cdots,N-k+1\}$,其中$N$为总响应长度。连续词元通常能组成局部连贯的短语,比孤立词元更符合语言模型的自然生成习惯,在自回归过程中也更容易被整体

复现。此外，本文主要考量$k>1$的情况，用于诱导短语级重复序列。当$k=1$时，方法只能诱发词元级重复，表现为单一词元的持续输出，模式简单且容易被基于n-gram统计[46]的规则所识别并抑制。相比之下，短语级重复由于重复词元个数不确定、结构更复杂，难以通过简单规则进行过滤，因此具备更好的测试覆盖能力。最后，本文以对数似然累积得分为准则，从候选种子集中选取得分最高的子序列作为最终重复种子：

$$S^{*}=\underset{s\in\{1,2,\cdots,N-k+1\}}{\arg\max}\sum_{i=1}^{k}\log P_{s+i}^{i} \tag{4}$$

其中，P_{s+i}^{i}表示第$s+i$个输出位置上第s个候选种子中第i个词元对应的预测概率。累积得分越高，表明该子序列在模型当前生成分布下的整体似然越大，也就是已经位于模型的高概率生成区域。这一特性使得所选种子在后续优化阶段更容易被激活，从而更有效地触发短语级循环重复，提升诱导成功率。

2.3.3 循环诱导优化

确定重复种子之后，需要解决的关键问题就转变为该如何有效地诱导模型去生成这些种子，并且借助自回归机制使其逐步进入自我强化的循环状态。针对这个问题，本文设计了一个循环损失函数，通过优化对抗性测试样本，激励模型倾向于生成预先选定好的重复种子。由于无法直接指定多模态大模型输出特定词元，本文改为采用一种无目标的优化策略，用概率引导的方式来间接把控模型的生成行为。具体的做法是，循环损失函数通过提升重复种子在各个输出位置上的预测概率，使其在采样的过程当中更容易被选中，从而激励模型在后续输出中选取这些种子作为当前位置的词元。循环损失函数的形式化定义如下：

$$\mathcal{L}_{\text{cycle}}\left(\boldsymbol{c}_{\text{in}},\boldsymbol{X}'\right)=-\frac{1}{N}\sum_{i=1}^{N}\log\left(\sum_{j=1}^{k}P_{i}^{j}\left(\boldsymbol{c}_{\text{in}},\boldsymbol{X}'\right)\right) \tag{5}$$

其中，N为输出序列长度；P_{i}^{j}表示第i个输出位置上第j个种子词元的预测概率。该损失函数的本质是把所有输出位置的种子词元的概率质量累积起来，让模型在采样时更倾向于选择这些词元或其组合。一旦对抗性测试样本能够诱导模型多次生成重复种子，自回归机制就会进一步放大这个趋势。已有重复词元出现在上下文中，后续再次生成它们的可能性也会随之提高，最终导致无休止地重复输出。

为了最小化循环损失函数（如式（3）所示），本文运用投影梯度下降[47]（Projected Gradient Descent，PGD）算法迭代更新对抗性测试样本，第t步的优化过程表示如下：

$$\boldsymbol{X}_{t}'=\Pi_{\boldsymbol{X},\epsilon}\left(\boldsymbol{X}_{t-1}'-\alpha\cdot\text{sign}\left(\nabla_{\boldsymbol{X}_{t-1}'}\mathcal{L}_{\text{cycle}}\right)\right) \tag{6}$$

其中，α为梯度更新步长；sign$(\cdot)$为符号函数（对正值输出1，对负值输出−1，对零值输出0）。操作算子$\Pi_{\boldsymbol{X},\epsilon}(\cdot)$的作用是将更新后的对抗性测试样本限制在以原始输入$\boldsymbol{X}$为中心、半径为$\epsilon$的$l_p$范数球空间当中，然后以此来控制扰动的可感知性。另外，为了使优化更稳定、收敛速度更快，本文在梯度更新中引入了一个动量项μ，用来累积历史梯度信息，从而可以减缓梯度震荡并使优化方向更具有一致性[48]。

CycleTrap的完整执行流程见算法1，其中输入包括原始视觉输入$\boldsymbol{X}$，文本提示$\boldsymbol{c}_{\text{in}}$，多模态大模型$f$，模型最大生成长度$N_{\max}$，重复种子长度$k$，最大迭代次数$T$，扰动半径$\epsilon$，梯度更新步长$\alpha$以及动量衰减系数$\mu$；输出为优化后的对抗性测试样本$\boldsymbol{X}'$。具体而言，第1行对原始输入执行贪心解码，得到模型的初始响应序列$\boldsymbol{Y}$。第2~5行枚举$\boldsymbol{Y}$中所有长度为k的连续子序列，并选取对数似然累积得分最高的作为重复种子S^{*}。第6~7行完成优化前的初始化，包括从原始视觉输入出发构造对抗样本，并将动量向量置零。第9~10行计算循环损失函数并反向传播获得当前样本梯度。第11行累积带动量的梯度，用历史梯度方向平滑更新路径；第12行执行投影梯度下降，并将更新幅度约束在允许范围内。第13~15行设置早停机制，若优化后的对抗样本已能驱动模型输出至最大生成长度，则提前终止优化。最后第17行返回最终的对抗性测试样本$\boldsymbol{X}'$。

3 实验与分析

3.1 实验配置

3.1.1 模型与数据集

本文选取了四种多模态大模型，用来评估方法在不同架构和模态设置下的可用性测试效果。其中，InstructBLIP[30]和LLaVA-NeXT[49]是面向图像理解的模型，对应的视频扩展版本分别是InstructBLIP-Video和LLaVA-NeXT-Video；此外，Qwen2.5-VL-7B[31]与InternVL3-8B[32]则是兼容图像和视频输入的通用多模态模型。所有模型都在默认提示模板下执行视觉描述任务，输入的文本是一些用于视觉描述任务的标准指令，例如“What is shown in this image”（这张图显示了什么？）、“Please describe this video.”（请描述一下这个视频）之类的。

数据集方面，本文分别从图像域和视频域各自选取了两个基准数据集。在图像域中，选取了MS-COCO[50]和ImageNet[51]，这两个都是计算机视觉领域

算法1 CycleTrap执行流程

输入: $\boldsymbol{X}$, $\boldsymbol{c}_{\text{in}}$, f, $N_{\max}$, k, T, ϵ, α, μ

输出: $\boldsymbol{X}'$

$\boldsymbol{Y} \leftarrow f(\boldsymbol{c}_{\text{in}}, \boldsymbol{X})$

for $s = 1$ **to** $|Y| - k + 1$ **do**

  $S \leftarrow (y_s, y_{s+1}, \cdots, y_{s+k-1})$

end for

$S^* \leftarrow \text{argmax}_S \sum_{i=1}^{k} \log P_{s+i}^{i}(\boldsymbol{c}_{\text{in}}, \boldsymbol{X})$

$\boldsymbol{X}_0' \leftarrow \boldsymbol{X}$

$m_0 \leftarrow 0$

for $t = 1$ **to** T **do**

  $\mathcal{L}_{\text{cycle}} \leftarrow -(1/N) \sum_{i=1}^{N} \log \left(\sum_{j=1}^{k} P_i^j(\boldsymbol{c}_{\text{in}}, \boldsymbol{X}_{t-1}') \right)$

  $g_t \leftarrow \nabla_{\boldsymbol{X}_{t-1}'} \mathcal{L}_{\text{cycle}}$

  $m_t \leftarrow \mu \cdot m_{t-1} + g_t / \| g_t \|$

  $\boldsymbol{X}_t' \leftarrow \Pi_{X,\epsilon}(\boldsymbol{X}_{t-1}' - \alpha \cdot \text{sign}(m_t))$

  if $| f(\boldsymbol{c}_{\text{in}}, \boldsymbol{X}_t') | \geqslant N_{\max}$

    break

  end if

end for

return $\boldsymbol{X}'$

常用的大规模图像理解基准,覆盖了复杂场景下的多类别图像。在视频域中,选取了TGIF[52]与MSVD[53]。其中TGIF是基于动态GIF图像构建,视频片段短且动作语义集中;MSVD涵盖了各种日常场景及其对应的文本描述,是视频字幕生成任务里的经典基准集。对于每个数据集,本文分别随机抽取了200张图像或100个视频作为评估子集。

3.1.2 基线方法介绍

本文选取四种基线方法评估CycleTrap的相对表现:(1)原始输入(Normal)。直接将数据集中的原始样本输入模型,展示模型在自然条件下的输出行为。(2)随机噪声(Random)。在原始输入上叠加高斯噪声构造测试样本,用于观察模型对无优化随机干扰的反应。(3)Verbose Sample[25](Verbose)。通过联合优化多目标损失函数生成测试样本。一方面降低EOS生成概率以延长输出长度,另一方面引入两个正则项打散模型输出的概率分布,诱导模型产生偏离正常语义分布的冗长响应。(4)LingoLoop[26]。从语言结构和模型内部两个层面构造测试样本,先基于词性信息动态调整不同位置上抑制EOS的权重分布,再结合模型隐藏状态进行路径搜索,选择更利于持续生成的输出轨迹,从而诱导模型输出冗余内容。

3.1.3 评估指标

由于输出长度与能量延迟成本之间存在正相关性关系,所以本文把平均输出长度(Len)作为衡量压力测试强度的主要指标。同时,还记录了测试成功率(ASR),也就是成功诱导模型输出达到最大长度的样本在总输入样本的占比。此外,为了比较测试效率,本文还测量了优化成本(Cost),即完成一个对抗性测试样本优化所需要的平均时间。

3.1.4 实验参数与细节

本文设置循环种子数量$c=5$。为了保证比较的公平性,CycleTrap与各基线方法在优化时都采用统一的实验配置:所有方法都采用PGD进行优化,最大迭代轮数为300,初始扰动设为零向量;一旦对抗性测试样本成功诱导模型输出达到最大长度,就停止优化并把该样本记作一次成功案例。扰动强度设置为$\epsilon=16$,并在l_∞范数约束下进行投影;梯度更新步长$\alpha=1$,动量衰减系数$\mu=0.9$。在推理阶段,本文使用各模型的默认解码策略。受硬件资源限制,模型的最大输出长度统一设为1 024个词元。在软硬件配置方面,实验主要基于NVIDIA RTX PRO 6000显卡(显存96 GB,CUDA版本13.0)进行,利用PyTorch(版本2.10)与Transformers(版本5.3.0)库来完成多模态大模型的推理过程。除特别说明外,默认以Qwen2.5-VL-7B作为主要分析模型。

3.2 主实验结果与分析

表1总结了CycleTrap在不同模型和数据集上的压力测试效果和优化效率。在原始输入条件下,各模型的原始响应都难以自己达到最大长度;加入随机噪声虽然会对生成过程造成一些扰动,但并不能提升输出长度,部分情况下反而还会削弱模型的生成能力,这说明无优化的随机扰动基本无法测试出模型的可用性边界。相比之下,Verbose Sample与LingoLoop虽然可以在一定程度上延长输出,但无论是在长度增益、测试成功率还是优化开销上,都明显弱于CycleTrap。这一差异也印证了,仅仅依靠延迟EOS的策略在稳定性和扩展性上是存在局限的。

CycleTrap在各项指标上都表现出一定的优势。以InternVL3为例,CycleTrap在图像任务上能把输出长度提升到接近甚至达到最大值,在视频任务上则稳定地将输出长度推到最大值,测试成功率达到100%。这说明即便是面对近期通用多模态模型,它也同样具有较强的效果。而对于InstructBLIP和LLaVA-NeXT,CycleTrap在视频模态上具有更强的测试稳定性,在多个任务中都能达到最大输出长度,同时优化成本也较低。值得一提的是,即使在其他基线方法已经接近最大输出的情况下,CycleTrap还能进一步压缩优化时间。例如在LLaVA-NeXT的视频任务上,CycleTrap在保持最大输出的同时,还能将优化成本压缩至更低水平。

表1 CycleTrap与各基线方法在不同模型和不同模态数据集上的全面性能对比

Table 1 Comprehensive performance comparison between CycleTrap and baseline methods across different models and multimodal datasets

多模态大模型	基线方法	图像模态						视频模态					
		MS-COCO			ImageNet			TGIF			MSVD		
		Len	ASR/%	Cost/s	Len	ASR/%	Cost/s	Len	ASR/%	Cost/s	Len	ASR/%	Cost/s
Qwen2.5-VL	Normal	113	0	—	109	0	—	108	0	—	105	0	—
	Noise	102	0	—	107	0	—	137	0	—	124	0	—
	Verbose	333	8	879	585	34	649	625	30	793	665	29	815
	LingoLoop	478	17	352	524	27	308	472	22	241	413	24	217
	CycleTrap	**846**	**78**	**249**	**872**	**82**	**138**	**814**	**74**	**188**	**912**	**84**	**179**
InternVL3	Normal	329	0	—	419	2	—	130	2	—	83	1	—
	Noise	284	0	—	337	0	—	149	1	—	104	0	—
	Verbose	764	64	593	713	56	623	989	91	397	1 003	92	353
	LingoLoop	693	47	370	597	41	381	893	87	164	936	89	188
	CycleTrap	**1 012**	**95**	**294**	**1 010**	**93**	**311**	**1 024**	**100**	**74**	**1 024**	**100**	**84**
InstructBLIP(-Video)	Normal	113	0	—	100	0	—	39	0	—	47	0	—
	Noise	126	0	—	109	0	—	97	0	—	83	0	—
	Verbose	319	17	910	382	27	851	479	36	287	357	27	411
	LingoLoop	382	23	246	422	32	169	338	24	178	572	47	132
	CycleTrap	**612**	**53**	**177**	**654**	**58**	**112**	**987**	**96**	**82**	**935**	**90**	**69**
LLaVA-NeXT(-Video)	Normal	191	0	—	177	1	—	224	2	—	204	0	—
	Noise	174	0	—	162	1	—	296	5	—	179	0	—
	Verbose	952	91	369	893	87	351	1 024	100	131	1 007	96	183
	LingoLoop	802	70	319	827	82	423	1 024	100	89	933	86	97
	CycleTrap	**1 024**	**100**	**130**	**1 010**	**98**	**152**	**1 024**	**100**	**46**	**1 024**	**100**	**56**

注:图像模态与视频模态分别对应各自专用或兼容模型版本。例如,图像模态使用InstructBLIP,视频模态使用InstructBLIP-Video。其中“—”表示该方法无需优化过程,故无优化开销统计。各指标最优结果已加粗标注。

上述结果说明,CycleTrap的优势不仅仅体现在最终输出长度上,还在于它的优化过程更稳定。这主要是因为该方法跳出了EOS这一单点,通过循环损失直接调整输出分布,使模型进入一种自我强化的重复生成状态,从而稳定地把输出推到最大值。它把测试目标从“推迟终止”转变成“诱导循环生成”,因此可以用较低的优化成本来获得较高的测试成功率。

3.3 消融实验

本节对CycleTrap部分组件进行消融分析。由于循环损失函数是本方法唯一的优化驱动,禁用后则无法诱导模型进入循环生成状态,方法本身也失去测试意义。因此,消融实验主要考察两个辅助组件,包括上下文感知的种子提取机制和PGD梯度优化中的动量策略。

表2的实验结果显示,当把上下文感知的种子提取机制替换为从模型词表中随机选择重复种子后,Qwen2.5-VL在MS-COCO与TGIF数据集上的测试成功率会分别从78%与74%下降至67%与65%,InternVL3的性能也出现了类似的下降。这说明,从原始输出中提取出来的词元更贴合模型当前的生成分布,也更有利于诱导出重复模式的形成。相比之下,移除动量机制对测试性能的影响较小,它的主要作用是平滑梯度更新方向、加速优化收敛,而不是直接决定测试强度的上限。

表2 针对两个图像-视频兼容模型的消融实验结果

Table 2 Ablation study results for two image-video compatible models

多模态大模型	C	M	MS-COCO		TGIF	
			Len	ASR/%	Len	ASR/%
Qwen2.5-VL		✓	809	67	774	65
	✓		839	77	795	72
	✓	✓	**846**	**78**	**814**	**74**
InternVL3		✓	912	89	1 016	98
	✓		1 009	94	1 024	100
	✓	✓	**1 012**	**95**	**1 024**	**100**

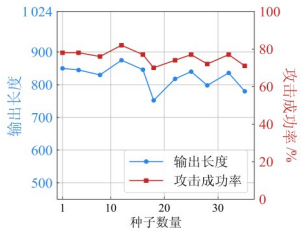
注:其中“C”表示上下文感知的种子提取机制,禁用时改为从全部词表中随机采样作为重复种子;“M”表示是否在PGD梯度更新中加入动量机制。

3.4 超参数分析

3.4.1 种子数量与优化步骤

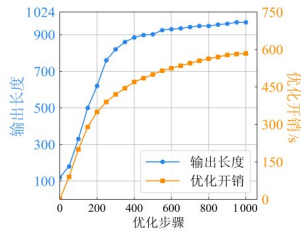
如图4左侧所示,重复种子数量k的变化对平均

输出长度和测试成功率的影响相对较小,测试性能在较宽的参数区间内保持稳定。通过对成功案例的分析可以发现,即使种子数量持续增加,最终能够有效触发循环生成状态的词元仅占少数,多数词元对诱导重复的作用不大。这说明循环损失函数更倾向于诱导短词元片段重复,而非依靠大规模的种子集合,因此CycleTrap并不依赖于重复种子规模的不断扩大。



(a) 重复种子数量对平均输出长度及测试成功率的影响

(a) The effect of the number of repetition seeds on average output length and test success rate



(b) PGD最大迭代步数对平均输出长度及单样本优化开销的影响

(b) The effect of the maximum number of PGD iterations on average output length and per-sample optimization cost

图4 超参数敏感性分析结果

Figure 4 Hyperparameter sensitivity analysis results

此外,本文还研究分析了PGD的最大迭代轮数对测试性能的影响。由图4右侧可以看出,随着迭代次数的增加,测试性能总体逐步提升,这说明更长的优化过程有助于找到更有效的扰动模式,但是相应的计算成本也会增加。综合考虑测试效果和优化开销,本文将最大迭代步数设定为300轮,从而在测试性能和优化效率之间取得相对平衡。

3.4.2 扰动强度

为了评估扰动在视觉层面上的可感知程度,表3给出了不同扰动幅度ϵ下的实验结果,同时借助LPIPS[54]来量化视觉感知差异。这个指标衡量的是原始视觉输入与对抗性视觉输入之间的感知距离,数值越小,说明扰动越不容易被察觉。实验结果显示,随着ϵ的增大,对抗扰动在优化过程中的搜索空间也更大,测试性能也随之提升。例如,当$\epsilon=32$时,在MS-COCO和TGIF数据集上的测试成功率分别达98%和97%,平均输出长度也基本都达到了最大值。不过,扰动幅度越大,LPIPS也会越高,这也意味着对抗性测试样本的视觉隐蔽性会因此变差。所以,本文选取$\epsilon=16$作为基准扰动幅度,目的是在测试性能与扰动不可感知性之间达到相对平衡。

表3 不同扰动强度ϵ的测试效果及视觉可感知性的影响分析

Table 3 Analysis of the testing performance and visual perceptibility under different perturbation magnitudes

ϵ	MS-COCO			TGIF		
	Len	ASR/%	LPIPS	Len	ASR/%	LPIPS
4	297	10	0.001 6	312	7	0.002 1
8	524	39	0.008 3	428	23	0.013 7
16	883	81	0.038 3	809	72	0.060 4
32	1 014	98	0.123 2	1 008	97	0.175 1

3.5 方法鲁棒性实验

3.5.1 解码策略

模型推理阶段的解码策略会影响输出采样的随机性和稳定性,也可能会对测试结果产生一定的影响。因此,本文通过调整解码参数,比较了贪心解码(greedy)、束搜索(beam)以及不同温度系数(T)设置下的测试性能。

实验结果如表4所示,在这两组数据集中,束搜索的表现总体上要优于贪婪搜索和温度采样。原因可能是:束搜索需要同时维护多条候选路径,更容易选择累积概率较高的词元序列,从而让重复模式持续扩散下去。在温度采样下,温度参数的设置过低或过高都会带来波动,说明过强的随机性会使模型更倾向于去探索低概率词元,从而减弱了循环诱导机制的效果。虽然不同的解码策略会影响输出长度、测试成功率和优化成本的绝对数值,但CycleTrap在各种解码方式下都保持了较强的测试能力,表明它对解码策略的鲁棒性较高,具有良好的策略无关性。

表4 不同解码策略对CycleTrap测试性能的影响分析结果

Table 4 Analysis of the impact of different decoding strategies on the performance of CycleTrap

解码策略	MS-COCO			TGIF		
	Len	ASR/%	Cost/s	Len	ASR/%	Cost/s
greedy	782	69	285	836	76	176
beam	891	85	211	910	87	154
T=0.2	798	72	269	794	70	193
T=0.6	842	77	243	701	59	217
T=1.2	763	66	297	688	51	238

3.5.2 重复抑制防御机制

为了评估CycleTrap在面对重复输出抑制这类防御时的鲁棒性,本文引入了解码阶段的“repetition_penalty”参数[55]作为防御代理。这个参数会对已生成过的词元按出现次数施加概率惩罚,降低其在后

续生成过程中再次被选取的概率，从本质上来说是一种针对重复生成行为的正则化策略。“repetition_penalty”的默认值为1，表示不进行惩罚，数值越大惩罚程度越强，但也会导致生成文本质量下降，所以该参数值一般设置在1到1.2之间。

实验结果如表5所示，随着重复惩罚强度的增大，模型在原始样本以及Verbose Sample优化样本上的输出长度反而有所延长。这一现象可能是由于模型在受到重复抑制约束后，为避免陷入退化循环，会引入更多不同的词元来维持语义连贯性，从而导致输出序列被拉长。由于“repetition_penalty”的惩罚逻辑与循环损失的优化目标在一定程度上相互冲突，CycleTrap的测试性能略有下降。不过，即使在较强的惩罚约束下，CycleTrap相比基线方法仍然具有显著的优势，并未出现实质性的失效。这说明循环诱导机制生成的扰动具有较强的生成引导能力，能够克服重复惩罚带来的负向干预。同时也揭示了当前重复防御机制还存在局限性，亟需探索更为有效的防御方案。

表5 CycleTrap在不同重复惩罚系数下的鲁棒性分析结果

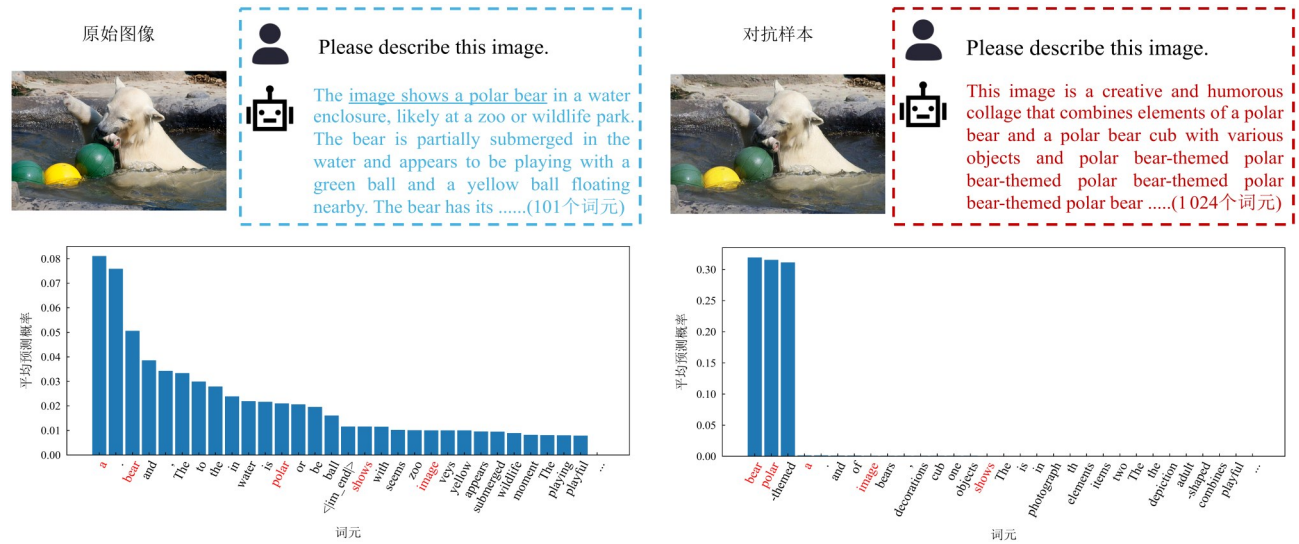
Table 5 Robustness analysis of CycleTrap under different repetition penalty coefficients

repetition_penalty	基线方法	MS-COCO		TGIF	
		Len	ASR/%	Len	ASR/%
1.05	Normal	113	0	108	0
	Verbose	333	8	625	30
	CycleTrap	883	81	809	72
1.10	Normal	129	0	116	0
	Verbose	489	19	663	34
	CycleTrap	819	73	765	66
1.15	Normal	152	0	131	0
	Verbose	470	17	495	18
	CycleTrap	775	69	735	62
1.20	Normal	178	0	168	0
	Verbose	451	13	468	17
	CycleTrap	742	64	684	58

3.6 案例分析

为了更直观地展示CycleTrap的工作机制，本节以Qwen2.5-VL-7B模型在MS-COCO数据集上的一个典型样本为例进行分析。如图5所示，在原始图像输入的条件下，模型生成了内容完整的自然描述。本文进一步对各输出位置的预测概率分布取平均后，可以发现模型原始输出中词元的平均概率相对均匀，这反映了在正常生成过程中词汇选择的多样性。

在CycleTrap构造的对抗性样本输入条件下，模型的输出进入以“polar bear-themed”为循环的重复生成状态直至最大长度。根据对应的词元概率分布可以发现，预测概率高度集中在少数词元上，正好符合循环诱导机制的预计效果。值得注意的是，实际循环单元并不完全等同于预选重复种子，而是在种子语义范围内形成了结构略有变化的重复模式，这与4.4.1节中关于种子数量的分析一致，表明CycleTrap的测试性能对种子数量的敏感度均较低。



注：左右图分别对应原始图像和相应对抗样本输入模型后所生成的文本描述；下方柱状图展示了各自输出中所有词元的概率值（按降序排列，仅显示前20项）。其中所选重复种子在原始输出中以蓝色下划线标注，并在对应的概率分布图中以红色高亮标识图。

图5 原始图像与对抗样本在Qwen2.5-VL-7B模型上的输出对比和词元概率分布情况

Figure 5 Comparison of outputs and token probability distributions generated by Qwen2.5-VL-7B for the original image and the adversarial sample

4 潜在防御方法

针对CycleTrap暴露出的风险,这里梳理给出了一些潜在的防御方向。一个直接的办法是从生成过程入手做被动监控:在模型生成过程中加入在线检测模块,实时检测滑动窗口内的词元重复率或者监控输出分布情况是否过于集中。一旦发现循环趋势,系统就主动中断生成,将资源损耗限制在可控的范围内。不过,加入实时检测模块时需要考虑其与模型推理成本之间的权衡。另一个方向是在模型训练过程中做主动增强。通过将那些能够诱导重复的对抗性测试样本纳入到模型的安全对齐训练数据集中,从而提高模型在面对类似视觉扰动时的鲁棒性。但同时这也会增加模型训练的成本。此外,还可以与服务端配合,例如动态限制用户的推理请求,或部署一个轻量级输出质量过滤器。显然,没有哪一种单一能够覆盖所有场景,更实际的做法是把三者结合起来,构建多层次的协同防御。

5 结论

本文讨论了多模态大语言模型在推理服务场景下面临的极端负载风险,并指出了现有压力测试方法在测试强度和优化效率上的限制。围绕这一问题,本文提出可用性评估框架CycleTrap。该框架先通过上下文感知的种子提取机制确定初始重复词元,再利用循环诱导机制放大自回归生成中历史输出对当前预测的影响,使输出概率分布逐步集中到少数词元上,最终迫使模型持续输出重复内容直至最大长度。实验结果显示,CycleTrap在多种主流多模态模型以及图像、视频数据集上均能获得更长输出和更低优化开销。

不过,这个方法仍存在一定的局限性。目前的框架主要面向白盒测试场景,对模型内部信息的依赖程度比较高,在黑盒迁移场景中的适用能力有限。后续工作可以考虑引入集成优化、梯度估计这类对黑盒更友好的算法,来提升循环诱导在不同模型之间的迁移能力。总体来看,相比于延迟EOS的方法,本文从自回归推理机制的脆弱性出发,将压力测试目标从“推迟终止”转变为“循环诱导”,为评估多模态大模型的推理可用性边界提供了更高效的工具,也为推理安全风险提供了新的思路。

6 伦理道德声明

本文提出的方法主要为了研究多模态大模型的安全隐患,以及发现模型生成机制中的脆弱性,而非鼓励或传播任何形式的恶意利用。所有的实验均在公开的模型和数据集上开展,不涉及任何私人或敏感数据,也没有攻击现实的部署系统。

7 可复现性声明

为保证实验结果可复现,本文给出了方法流程和实验配置细节,所有实验均依托开源可获得的模型与数据集。同时,完整代码已公开在https://github.com/neuron-insight-lab/CycleTrap,便于为后续研究或防御设计提供可复现依据。

参考文献

[1] 廖宁, 曹敏, 严骏驰. 视觉提示学习综述[J]. 计算机学报, 2024(4): 790-820.
Liao Ning, Cao Min, Yan Junchi. Visual prompt learning: A survey[J]. Chinese Journal of Computers, 2024(4): 790-820. (in Chinese)

[2] Yu Lu, Nikandrou M, Jin Jiali, et al. Quality-agnostic image captioning to safely assist people with vision impairment[C]//Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, 2023: 6281-6289.

[3] Abdulgalil H D, Basir O A. Next-generation image captioning: A survey of methodologies and emerging challenges from transformers to Multimodal Large Language Models[J]. Natural Language Processing Journal, 2025, 12: 100159.

[4] Han Jiaming, Gong Kaixiong, Zhang Yiyuan, et al. OneLLM: One framework to align all modalities with language[C]//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2024: 26574-26585.

[5] Yin Shukang, Fu Chaoyou, Zhao Siruiet al. A survey on multimodal large language models[J]. National Science Review, 2024, 11(12): nwae403.

[6] Li Junnan, Li Dongxu, Savarese S, et al. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models[C]//International conference on machine learning. PMLR, 2023: 19730-19742.

[7] Cheng A C, Yin Hongxu, Yang Fu, et al. SpatialRGPT: Grounded spatial reasoning in vision-language models[C]//Advances in Neural Information Processing Systems 37. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2024: 135062-135093.

[8] Wang Zehan, Zhou Sashuai, He Shaoxuan, et al. SpatialCLIP: Learning 3D-aware image representations from spatially discriminative language[C]//2025 IEEE/CVF Confer-

ence on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2025: 29656-29666.

[9] Shankar S, Reuther A. Trends in energy estimates for computing in AI/machine learning accelerators, supercomputers, and compute-intensive applications[C]//2022 IEEE High Performance Extreme Computing Conference. Piscataway: IEEE, 2022: 1-8.

[10] Desislavov R, Martínez-Plumed F, Hernández-Orallo J. Trends in AI inference energy consumption: Beyond the performance-vs-parameter laws of deep learning[J]. Sustainable Computing: Informatics and Systems, 2023, 38: 100857.

[11] Jin Yizhang, Li Jian, Gu Tianjun, et al. Efficient multimodal large language models: A survey[J]. Visual Intelligence, 2025, 3: 27.

[12] 曾诚, 葛云洁, 赵令辰, 等. 多模态视觉语言表征学习模型及其对抗样本攻防技术综述[J]. 计算机研究与发展, 2025, 62(9): 2208-2232.
Zeng Cheng, Ge Yunjie, Zhao Linchen, et al. Survey of multimodal vision-language representation learning models and their adversarial examples attack and defense techniques[J]. Journal of Computer Research and Development, 2025, 62(9): 2208-2232. (in Chinese)

[13] Liu Daizong, Yang Mingyu, Qu Xiaoye, et al. A survey of attacks on large vision-language models: Resources, advances, and future trends[J]. IEEE Transactions on Neural Networks and Learning Systems, 2025, 36(11): 19525-19545.

[14] Zhang Yuanhe, Zhou Zhenhong, Zhang Wei, et al. Crabs: Consuming resource via auto-generation for LLM-DoS attack under black-box settings[C]//Findings of the Association for Computational Linguistics: ACL 2025. Stroudsburg: ACL, 2025: 11128-11150.

[15] Gao Kuofeng, Pang Tianyu, Du Chao, et al. Denial-of-service poisoning attacks against large language models[PP/OL]. V1. arXiv (2024-10-14) [2026-05-21]. https://doi.org/10.48550/arXiv.2410.10760.

[16] Liu Haotian, Li Chuanyuan, Wu Qingyang, et al. Visual instruction tuning[C]//Advances in Neural Information Processing Systems 36. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023: 34892-34916.

[17] Alayrac J B, Donahue J, Luc P, et al. Flamingo: A visual language model for few-shot learning[C]//Advances in Neural Information Processing Systems 35. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022: 23716-23736.

[18] Chen Jun, Guo Han, Yi Kai, et al. VisualGPT: Data-efficient adaptation of pretrained language models for image captioning[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2022: 18009-18019.

[19] Li Junnan, Selvaraju R, Gotmare A, et al. Align before fuse: Vision and language representation learning with momentum distillation[J]. Advances in neural information processing systems, 2021, 34: 9694-9705.

[20] Li Zejun, Zhang Jiwen, Wang Ye, et al. From multimodal pre-training to multimodal large language models: An overview of architectures, training, evaluation, and trends[C]// Proceedings of the 23rd Chinese National Conference on Computational Linguistics, Volume 2: Frontier Forum. 2024: 1-33.

[21] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30.

[22] Samsi S, Zhao Dan, McDonald J, et al. From words to Watts: Benchmarking the energy costs of large language model inference[C]//2023 IEEE High Performance Extreme Computing Conference. Piscataway: IEEE, 2023: 1-9.

[23] Chen Simin, Song Zihe, Haque M, et al. NICGSlowDown: Evaluating the efficiency robustness of neural image caption generation models[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2022: 15344-15353.

[24] Gao Kuofeng, Bai Yang, Gu Jindong, et al. Inducing high energy-latency of large vision-language models with verbose images[C]//Proceedings of the 12th International Conference on Learning Representations (ICLR). 2024: 3589-3616.

[25] Gao Kuofeng, Gu Jindong, Bai Yang, et al. Energy-latency manipulation of multi-modal large language models via verbose samples[PP/OL]. V1. arXiv (2024-04-25) [2026-05-21]. https://doi.org/10.48550/arXiv.2404.16557.

[26] Fu Jiyuan, Jiang Kaixun, Hong Lingyi, et al. LingoLoop attack: Trapping MLLMs via linguistic context and state entrapment into endless loops[PP/OL]. V3. arXiv (2026-04-13)[2026-05-21]. https://doi.org/10.48550/arXiv.2506.14493.

[27] Navaneet K L, Koohpayegani S A, Sleiman E, et al. SlowFormer: Adversarial attack on compute and energy consumption of efficient vision transformers[C]//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2024: 24786-24797.

[28] Radford A, Kim J W, Hallacy C, et al. Learning transferable visual models from natural language supervision[C]// International conference on machine learning. PMLR, 2021: 8748-8763.

[29] Li Junnan, Li Dongxu, Xiong Caiming, et al. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation[C]//International conference on machine learning. PMLR, 2022: 12888-12900.

[30] Dai Wenliang, Li Junnan, Li Dongxu, et al. InstructBLIP: Towards general-purpose vision-language models with instruction tuning[C]//Advances in Neural Information Processing Systems 36. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023: 49250-49267.

[31] Bai Shuai, Chen Keqin, Liu Xuejing, et al. Qwen2.5-VL technical report[PP/OL]. V1. arXiv (2025-02-19)2026-05-21]. https://doi.org/10.48550/arXiv.2502.13923.

[32] Zhu Jinguo, Wang Weiyun, Chen Zhe, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models[PP/OL]. V3. arXiv (2025-04-19)[2026-05-21]. https://doi.org/10.48550/arXiv.2504.10479.

[33] 陈晋音, 席昌坤, 郑海斌, 等. 多模态大语言模型的安全性研究综述[J]. 计算机科学, 2025, 52(7): 315-341.
Chen Jinyin, Xi Changkun, Zheng Haibin, et al. Survey of security research on multimodal large language models[J]. Computer Science, 2025, 52(7): 315-341. (in Chinese)

[34] Brachemi Meftah H F Z, Hamidouche W, Fezza S A, et al. Energy-latency attacks: A new adversarial threat to deep learning[J]. ACM Computing Surveys, 2026, 58(8): 1-34.

[35] 张永铮, 肖军, 云晓春, 等. DDoS攻击检测和控制方法[J]. 软件学报, 2012, 23(8): 2058-2072.
Zhang Yongzheng, Xiao Jun, Yun Xiaochun, et al. DDoS attacks detection and control mechanisms[J]. Journal of Software, 2012, 23(8): 2058-2072. (in Chinese)

[36] Shumailov I, Zhao Yiren R, Bates D, et al. Sponge examples: Energy-latency attacks on neural networks[C]//2021 IEEE European Symposium on Security and Privacy. Piscataway: IEEE, 2021: 212-231.

[37] Haque M, Shah R, Chen Simin, et al. SlothSpeech: Denial-of-service attack against speech recognition models[C]//INTERSPEECH 2023. ISCA, 2023: 1274-1278.

[38] Gao Xiaoxue, Chen Yiming, Yue Xianghu, et al. TTSlow: Slow down text-to-speech with efficiency robustness evaluations[J]. IEEE Transactions on Audio, Speech and Language Processing, 2025, 33: 693-704.

[39] Dong Jianshuo, Zhang Ziyuan, Zhang Qingjie, et al. An engorgio prompt makes large language model babble on[C]// Proceedings of the Thirteenth International Conference on Learning Representations (ICLR). 2025: 57213-57241.

[40] Li Xingyu, Liu Xiaolei, Liu Cheng, et al. LoopLLM: Transferable energy-latency attacks in LLMs via repetitive generation[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2026, 40(38): 31770-31777.

[41] Stübler T, Amodei A, Capriglione D, et al. An investigation of denial of service attacks on autonomous driving software and hardware in operation[C]//2024 IEEE 20th International Conference on Automation Science and Engineering. Piscataway: IEEE, 2024: 3051-3056.

[42] Ma Chen, Wang Ningfei, Chen Q A, et al. SlowTrack: Increasing the latency of camera-based perception in autonomous driving using adversarial examples[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2024, 38(5): 4062-4070.

[43] 姜毅, 杨勇, 印佳丽, 等. 大语言模型安全与隐私风险综述[J]. 计算机研究与发展, 2025(8): 1979-2018.
Jiang Yin, Yang Yong, Yin Jiali, et al. Survey on security and privacy risks in large language models[J]. Journal of Computer Research and Development, 2025(8): 1979-2018. (in Chinese)

[44] Xu Jin, Liu Xiaojiang, Yan Jianhao, et al. Learning to break the loop: Analyzing and mitigating repetitions for neural text generation[C]//Advances in Neural Information Processing Systems 35. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022: 3082-3095.

[45] Li Huayang, Lan Tian, Fu Zihao, et al. Repetition in repetition out: Towards understanding neural text degeneration from the data perspective[C]//Advances in Neural Information Processing Systems 36. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023: 72888-72903.

[46] Brown P F, DeSouza P V, Mercer R L, et al. Class-based n-gram models of natural language[C]//Computational Linguistics. New York: ACM, 1992: 467-479.

[47] Madry A, Makelov A, Schmidt L, et al. Towards deep learning models resistant to adversarial attacks[C]//Proceedings of the 6th International Conference on Learning Representations (ICLR). 2018: 4138-4161.

[48] Sutskever I, Martens J, Dahl G, et al. On the importance of initialization and momentum in deep learning[C]//Proceedings of the 30th International Conference on Interna-

tional Conference on Machine Learning - Volume 28. New York: ACM, 2013: III-1139-III-1147.

[49] Li Feng, Zhang Renrui, Zhang Hao, et al. LLaVA-NeXT-interleave: Tackling multi-image, video, and 3D in large multimodal models[PP/OL]. V2. arXiv (2024-07-28) [2026-05-21]. https://doi.org/10.48550/arXiv.2407.07895.

[50] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: Common objects in context[M]//Computer Vision - ECCV 2014. ChamSpringer International Publishing2014: 740-755.

[51] Deng Jia, Dong Wei, Socher R, et al. ImageNet: A large-scale hierarchical image database[C]//2009 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2009: 248-255.

[52] Jang Y, Song Y L, Yu Y, et al. TGIF-QA: Toward spatio-temporal reasoning in visual question answering[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017: 1359-1367.

[53] Chen D L, Dolan W B. Collecting highly parallel data for paraphrase evaluation[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1. New York: ACM, 2011: 190-200.

[54] Zhang R, Isola P, Efros A A, et al. The unreasonable effectiveness of deep features as a perceptual metric[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2018: 586-595.

[55] Keskar N S, McCann B, Varshney L R, et al. CTRL: A conditional transformer language model for controllable generation[PP/OL]. V2. arXiv (2019-09-20)[2026-05-21]. https://doi.org/10.48550/arXiv.1909.05858.

作者简介

刘小垒 男,1992年7月出生于山西省大同市。现为国家工程物理交叉科学研究中心副研究员、硕士生导师。主要研究方向为人工智能安全。

E-mail: caeplxl@126.com

陈 厅 男,1987年2月出生于四川省成都市。现为电子科技大学教授、博士生导师。主要研究方向为软件安全。

E-mail: chenting19870201@163.com

李星煜 男,2002年2月出生于湖南省邵阳市。现为中国工程物理研究院在读硕士研究生。主要研究方向为人工智能安全。

E-mail: lixingyu24@gscaep.ac.cn

张小松 男,1968年6月出生于四川省成都市。现为电子科技大学教授、博士生导师。主要研究方向为系统与网络安全、软件与数据安全。

E-mail: johnsonzxs@uestc.edu.cn