

基于双阶段触发的反向验证逻辑推理方法

李鹏, 邱浙宇, 胡素君

引用本文

李鹏, 邱浙宇, 胡素君. [基于双阶段触发的反向验证逻辑推理方法](#)[J]. 计算机科学, 2026, 53(10): 285-294.

LI Peng, QIU Zheyu, HU Sujun. [Reverse Verification Logical Reasoning Method Based on Two-stage Triggering](#) [J]. Computer Science, 2026, 53(10): 285-294.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[PacketScope:大语言模型赋能eBPF的协议栈安全防御](#)

PacketScope:LLM-enhanced eBPF Framework for Protocol Stack Security

计算机科学, 2026, 53(10): 440-446. <https://doi.org/10.11896/jsjx.260100048>

[自动化代码重构:基于大语言模型的提示演化方法](#)

Automatic Code Refactoring:LLM-based Prompt Evolution Approach

计算机科学, 2026, 53(10): 396-412. <https://doi.org/10.11896/jsjx.251200012>

[面向安全关键软件单元测试的数据集智能生成方法](#)

Intelligent Dataset Generation Method for Safety-critical Software Unit Testing

计算机科学, 2026, 53(10): 345-355. <https://doi.org/10.11896/jsjx.250900141>

[基于语义驱动执行的代码质量评测框架](#)

Semantic-driven Execution-based Framework for Code Quality Evaluation

计算机科学, 2026, 53(10): 320-335. <https://doi.org/10.11896/jsjx.250900140>

[CLERAG:基于认知层次评估的法律问答检索增强生成方法](#)

CLERAG:Retrieval-augmented Generation Framework for Legal QA with Cognitive-levelEvaluation

计算机科学, 2026, 53(10): 295-300. <https://doi.org/10.11896/jsjx.250900130>

基于双阶段触发的反向验证逻辑推理方法

李 鹏^{1,2} 邱浙宇¹ 胡素君¹

1 南京邮电大学计算机学院、软件学院、网络空间安全学院 南京 210023

2 南京邮电大学网络安全与可信计算研究所 南京 210023

摘 要 针对大语言模型在复杂逻辑推理任务中存在的单向生成误差累积、高置信错误难以识别以及全量外部验证计算开销高的问题,提出一种基于反向验证的逻辑推理方法。以前向推理器输出的终端 Token 置信度和候选判别裕度为依据,构建两阶段动态触发机制,并利用受试者工作特征曲线与约登指数确定最优验证阈值,实现高风险样本筛选;复用前向生成的 Top-K 候选空间构造反事实假设,通过外部判题器多轮投票获得独立判别信号,完成逻辑一致性校验;进一步结合判题裕度设计动态决策门控,抑制不可靠反向信号造成的误纠错。LogiQA 2.0 数据集上的实验结果表明:在候选空间大小 $K=3$ 、外部判题器独立投票次数 $N=3$,并结合双阶段置信度筛选的配置下,所提方法的准确率达到了 77.16%,较标准思维链(Standard CoT)基线提升 6.94 个百分点,触发验证比例为 34.94%,平均验证耗时为 1.3206 s,体现出较好的精度与效率平衡。在进一步引入动态门控后,系统准确率为 76.41%,修正精度提升至 66.89%,破坏性翻转数由 57 降至 33,表明该机制能够以少量整体精度代价换取更高的纠错可靠性与系统鲁棒性;在 ReClor 上也取得了一致增益,验证了方法的泛化性。结果表明,该方法能够在较低的附加开销下缓解大语言模型逻辑推理中的高置信错误问题。

关键词: 大语言模型;逻辑推理;反向验证;置信度评估;动态门控

中图分类号 TP391

Reverse Verification Logical Reasoning Method Based on Two-stage Triggering

LI Peng^{1,2}, QIU Zheyu¹ and HU Sujun¹

1 School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

2 Institute of Network Security and Trusted Computing, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

Abstract To address three major challenges of large language models (LLMs) in complex logical reasoning tasks, namely error accumulation in one-pass forward generation, the difficulty of detecting high-confidence errors, and the high computational cost of exhaustive external verification, this paper proposes a reverse verification logical reasoning framework. Specifically, a two-stage dynamic triggering mechanism is developed based on the confidence of the final token generated by the forward reasoner and the margin between candidate predictions. The receiver operating characteristic (ROC) curve and Youden's index are further employed to determine the optimal verification threshold for high-risk sample selection. The Top-K candidate space produced during forward reasoning is then reused to construct counterfactual hypotheses, and an independent judgment distribution is obtained through multiple rounds of voting by an external evaluator, enabling logical consistency checking. In addition, a dynamic decision gating mechanism is introduced based on the evaluator margin to suppress miscorrections caused by unreliable backward signals. Experimental results on the LogiQA 2.0 dataset show that, under the configuration with a candidate space size of $K=3$, $N=3$ independent voting rounds by the external evaluator, and two-stage confidence filtering, the proposed method achieves an accuracy of 77.16%, outperforming the Standard CoT baseline by 6.94 percentage points, while maintaining a verification trigger rate of 34.94% and an average verification latency of 1.3206 s, demonstrating a favorable trade-off between accuracy and efficiency. After further incorporating the dynamic gating mechanism, the overall system accuracy reaches 76.41%, correction precision improves to 66.89%, and the number of destructive flips decreases from 57 to 33. Consistent gains are also achieved on ReClor, which further validates the generalizability of the proposed method. These results indicate that the proposed method can effectively alleviate the high-confidence error problem in LLM-based logical reasoning with relatively low additional overhead.

Keywords Large language models, Logical reasoning, Reverse verification, Confidence evaluation, Dynamic gating

到稿日期:2026-05-13 返修日期:2026-07-27

基金项目:江苏省六大高峰高层次人才项目(RJFW-111);江苏省研究生科研与实践创新计划项目(KYCX23_1048)

This work was supported by the Six Talent Peaks Project of Jiangsu Province(RJFW-111) and Postgraduate Research and Practice Innovation Program of Jiangsu Province(KYCX23_1048).

通信作者:李鹏(lipeng@njupt.edu.cn)

1 引言

近年来,基于 Transformer 架构和大规模预训练范式的大语言模型在文本理解、知识问答和复杂任务求解等方面表现出较强能力,并逐步成为自然语言处理研究的重要基础模型^[1-3]。国内相关综述表明,大语言模型正持续向多任务适配、复杂推理与场景化应用方向发展,其在高阶认知任务中的应用潜力受到广泛关注。文献[4]对大语言模型的发展脉络、关键能力及应用趋势进行了系统梳理。

围绕大语言模型推理能力的提升,现有研究首先从模型对齐和前向生成优化角度展开探索。文献[5-7]分别从偏好优化、领域推理训练和思维链提示等方面改进模型的推理能力与任务适应性。国内在自然语言推理方向的相关研究也表明,针对特定逻辑现象进行建模有助于提升模型的判别能力和推理表现^[8-9]。在此基础上,文献[10-12]进一步从自动思维链构造、分阶段规划求解和程序化思维提示等角度增强了模型对复杂问题的中间过程组织能力。

除前向生成优化外,研究者还尝试通过搜索、行动和工具调用增强模型的问题求解能力。文献[13-15]分别提出多分支思维搜索、推理-行动协同以及自主工具调用等方法,以提升模型在复杂任务中的探索能力和外部交互能力。这类方法在一定程度上扩展了模型的求解空间,但其核心仍然依赖前向生成过程,对已形成的错误推理路径缺乏稳定的后验校验机制。

针对这一问题,已有研究开始从结果复核、自我验证和判别修正角度增强大语言模型的推理可靠性。文献[16-18]分别从自验证机制、强验证器辅助纠错以及生成结果质量评估等角度,说明引入独立判别信号有助于提升最终答案的可信度。进一步地,文献[19-21]通过工具交互批判、纠错能力基准构建以及面向新工具场景的自验证机制,增强了模型对已有答案的再审查和再修正能力。这些研究说明,后验验证机制为缓解复杂推理中的错误传播提供了可行路径。

在推理可靠性增强方面,文献[22-23]分别从回溯式形式验证和高置信推理自训练角度提升了模型对中间步骤和最终结论的控制能力。与此同时,相关综述指出,大语言模型在复杂推理任务中仍面临能力边界、验证机制与稳定性等方面的挑战,相关方法仍有进一步完善空间。文献[24]系统梳理了大语言模型在数学推理中的技术路线、外部工具与验证机制,并指出未来仍需进一步提升模型的推理能力。另有研究表明,即使对于表面上并不复杂的任务,模型的推理过程也未必稳定可靠^[25]。

除外部验证外,置信度建模和不确定性表达也为推理结果的筛选提供了重要依据。文献[26-28]分别从估计型不确定性评估、基于不确定性的链式推理增强以及不确定性感知推理框架等角度,探讨了如何利用模型内部或外部信号刻画推理结果的可靠程度。对于阅读理解和抽象逻辑推理任务而言,这类研究具有重要的参考价值,因为此类任务往往难以像数学求解那样直接依赖外部程序获得确定性验证结果。LogiQA 2.0 数据集^[29]面向机器阅读理解中的逻辑推理能力评测,ReClor 数据集面向标准化考试逻辑推理,能够较好反

映模型在复杂文本推理任务中的逻辑一致性与判别能力。

尽管已有研究取得了一定进展,但在复杂逻辑推理场景下仍存在值得进一步研究的问题。一方面,多次采样、多路径搜索或多轮外部调用虽然能够提升结果的质量,但通常伴随较高的推理成本,难以兼顾准确性与效率。另一方面,已有自我修正与外部验证方法在触发条件、证据可靠性和最终决策机制方面仍存在不足,容易出现对低风险样本重复验证或者对不可靠校验结果过度依赖的情况。对于阅读理解和抽象逻辑推理等任务而言,模型还缺乏一种在文本语义空间内实现低成本、可控化闭环校验的有效机制。例如,在包含条件约束与选项干扰的逻辑阅读题中,模型可能因为某个选项与题干表述存在较强的表面语义重合而优先生成支持该选项的推理链;但该链条若在关键条件的适用范围或逻辑方向上出现偏差,后续过程仍可能维持局部连贯,进而输出错误答案。这类错误并不总表现为低置信度,相反,模型有时会对错误路径表现出较强的确信度,使得仅依赖前向输出难以发现问题。另一方面,若对所有样本一律调用外部验证,又难以兼顾推理效率。由此可见,复杂逻辑推理的难点不仅在于能否推理,更在于何时需要复核以及如何有限成本下实现可靠复核。

针对上述问题,本文面向大语言模型逻辑推理任务,提出一种基于反向验证的推理方法。该方法利用前向推理阶段输出的终端 Token 置信度和候选判别裕度对预测结果进行分级筛选,并仅对高风险样本触发反向验证过程;在此基础上,将前向生成得到的候选答案重构为待检验假设,借助外部判题器对候选结论进行一致性校验;同时,结合反向验证结果与前向预测信号,引入动态门控策略,以降低不可靠反向证据导致的误纠错风险。

本文的主要贡献如下:(1)提出一种融合终端 Token 置信度与候选判别裕度的双阶段触发机制,以实现高风险样本的动态定位;(2)设计基于 Top-K 候选复用的反事实重构与多轮投票判题流程,将开放式推理结果转换为受约束的验证问题;(3)提出动态决策门控机制,在利用反向证据进行纠错的同时抑制误修正;(4)在 LogiQA 2.0 和 ReClor 数据集上验证了该框架的有效性,并通过消融实验分析了各组成模块对系统性能的贡献。

2 反向验证逻辑推理方法

2.1 任务定义

本文研究的大语言模型逻辑推理任务可形式化为一个受约束的多项选择判别问题。给定一个样本 $x=(p,q,O)$,其中 p 表示题目背景或上下文文本, q 表示待回答的问题, $O=\{o_1,o_2,\cdots,o_M\}$ 表示候选选项集合, M 为候选个数。在本文所选数据集中, $M=4$,对应 A/B/C/D 4 个候选答案。任务目标是在给定 p,q 和候选集合 O 的条件下,预测正确答案 $y\in O$ 。

2.2 总体框架

针对大语言模型在复杂逻辑推理任务中存在的早期误差累积、高置信错误难以识别以及全量外部验证成本较高等问题,本文提出一种基于双阶段触发的反向验证逻辑推理方法。该方法以前向推理器输出的初始答案及候选分布为输入,通过终端 Token 置信度和候选判别裕度对样本风险进行筛选,

仅对高风险样本调用外部判题器进行反向验证,并在最终阶段通过动态决策门控完成结果修正,如图1所示。

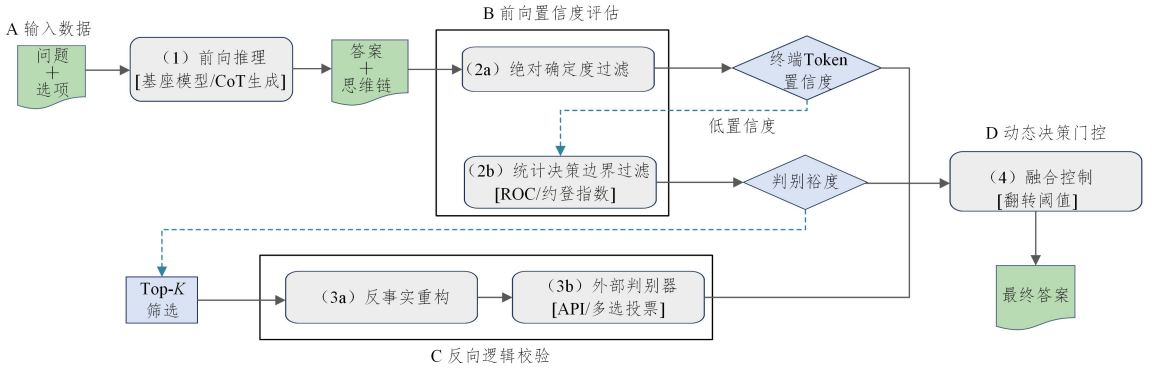


图1 基于双阶段触发的反向验证逻辑推理框架

Fig. 1 Reverse verification logical reasoning framework based on two-stage triggering

(1)前向推理阶段:以前向输入问题及候选选项为起点,调用经 LogiQA 2.0 任务策略优化后的 Qwen 2.5-7B-Instruct 生成初始答案与对应思维链,并保留答案位置的选项级概率分布,用于后续置信度评估与反向验证触发。

(2)前向置信度评估阶段:系统采用两阶段置信度评估机制。首先,依据终端 Token 的置信度对绝对确定样本进行快速过滤;然后,对于其余样本,引入基于 ROC 曲线优化得到的判别裕度阈值完成二次筛选。仅将裕度低于阈值的低置信度高风险样本送入反向验证模块,实现计算效率与验证效果的平衡。

(3)反向验证推理阶段:针对低置信度样本,系统复用前向推理过程中得到的 Top-K 候选集合,将各候选重构为反事实假设。通过引入外部判题器,以多选投票机制评估不同选项的逻辑一致性与支持强度,将生成任务转换为受约束的判别任务。

(4)动态决策门控阶段:系统综合对比前向预测与反向判别结果,并依据预设的翻转阈值评估证据差异。仅当反向判别在证据强度上显著优于前向预测时,才执行答案修正(翻转操作)。该门控策略有效控制了额外的验证开销,避免了判题器噪声引入的误修正风险,提升了整体推理的鲁棒性。

2.3 双阶段触发机制

为在推理早期快速筛除高置信样本,首先引入终端 Token 置信度(ETC)作为第一阶段的触发指标。终端 Token 置信度反映了模型对预测答案 A_{pred} 的绝对置信水平,如式(1)所示:

$$P_{\text{end}}(A_{\text{pred}}) = \exp(l p_{\text{pred}}) \quad (1)$$

其中, $l p_{\text{pred}}$ 为模型生成答案 Token 时的对数概率。

实验观察到,当 $\log P_{\text{end}}(A_{\text{pred}}) = 0.0$ (即 $P_{\text{end}} \approx 1.0$) 时,模型在统计上表现出极高的稳定性。因此,设定初步筛选规则为:若满足该绝对确定条件,则直接采纳预测结果;否则,将样本送入第二阶段进行深度分析。

对于未通过初筛的样本,引入受试者工作特征曲线(ROC)分析,对其前向判别裕度(S_{margin})进行系统评估。前向判别裕度描述了模型在相似候选选项之间的判别不确定性,定义为首选答案概率与次选答案概率之差,如式(2)所示:

$$S_{\text{margin}} = P_{\text{end}}(o_{\text{top1}}) - P_{\text{end}}(o_{\text{top2}}) \quad (2)$$

其中, $P_{\text{end}}(o_{\text{top1}})$ 为生成概率最高的回答, $P_{\text{end}}(o_{\text{top2}})$ 为生成概率第二高的回答。

定义 $y_i \in \{0, 1\}$ 为样本真实标签(1 为正确, 0 为错误), N 为样本数量。对于任意给定的阈值 τ , 定义召回率如式(3)所示:

$$\text{Recall}(\tau) = \frac{\sum_{i=1}^N I(s_i > \tau \wedge y_i = 1)}{\sum_{i=1}^N I(y_i = 1)} \quad (3)$$

其中, $I(\cdot)$ 为指示函数,当括号内条件成立时取值为 1, 否则取值为 0。

定义误判率为错误答案被系统误判为正确并放行的比例,如式(4)所示:

$$\text{FPR}(\tau) = \frac{\sum_{i=1}^N I(s_i > \tau \wedge y_i = 0)}{\sum_{i=1}^N I(y_i = 0)} \quad (4)$$

为最大化正负样本的区分度,采用约登指数作为优化目标函数,寻找 ROC 曲线上的最佳切点以求解最优决策阈值 τ^* , 如式(5)所示:

$$\tau^* = \arg \max_{\tau} (\text{Recall}(\tau) - \text{FPR}(\tau)) \quad (5)$$

该优化过程本质上是寻找 ROC 曲线上的切点,使其到对角线(随机猜测线)的垂直距离最大。确定 τ^* 后,系统的触发动作 $\text{Action}(A_{\text{pred}})$ 如式(6)所示:

$$\text{Action}(A_{\text{pred}}) = \begin{cases} \text{Accept}, & \text{if } S_{A_{\text{pred}}} > \tau^* \\ \text{ReverseReasoning}, & \text{if } S_{A_{\text{pred}}} \leq \tau^* \end{cases} \quad (6)$$

高于阈值 τ^* 的样本被视为高置信保留样本;而低于阈值 τ^* 的样本被视为高风险样本,将其输送至反向验证模块进行逻辑校验。

在前向推理阶段,模型通常通过贪婪解码仅输出概率最大的单一结果,该策略本质上遵循局部最优准则,难以充分反映模型在候选选项之间的整体不确定性分布。为缓解单一路径解码带来的信息损失,在前向生成阶段显式保留答案生成时刻的选项级概率信息,并构建 Top-K 候选空间以增强后续验证阶段的覆盖能力。

具体而言,设候选选项集合为 $\mathcal{O} = [A, B, C, D]$ 表示模型生成的 4 个前向阶段答案,系统从输出层 logits 中提取各选

项对应的对数概率,并在选项子空间内施加 Softmax 归一化,得到相对概率分布。基于该分布,系统构建一个按预测概率降序排列的候选列表,如式(7)所示:

$$P_o(o) = \frac{\exp(lp_o)}{\sum_{k \in O} \exp(lp_k)}$$
 (7)

基于上述概率分布,系统构建一个按照预测概率降序排列的候选列表,如式(8)所示:

$$L_{\text{rank}} = [o_1, o_2, \dots, o_o]$$
 (8)

这一过程将验证任务转换为生成与重排序任务,前向生成模块作为候选召回阶段,为后续反向验证模块的重排序提

供了覆盖潜在正确答案的搜索空间。

2.4 反向验证阶段

为了进一步提升推理的可靠性,本节提出一种基于反向验证的推理算法。该算法不改变原基座模型的参数,而是通过引入外部判题器将前向生成的候选答案视为待检验的假设,在反事实的视角下重新评估其与题干逻辑的一致性。其核心思想是将生成任务转换为判别任务。

整体工作流程如图 2 所示,主要包含 3 个级联步骤:触发决策的执行与配置,候选集合构建与反事实假设,以及多选投票判题机制。

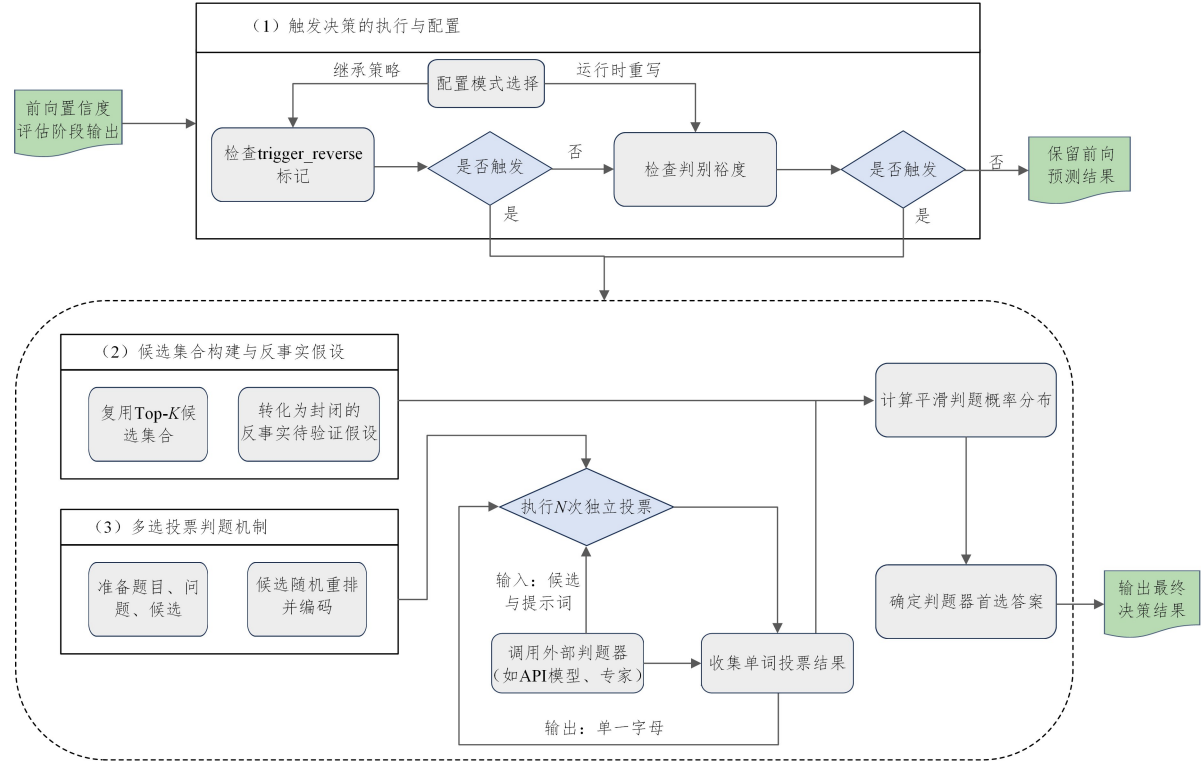


图 2 反向验证流程

Fig. 2 Reverse verification procedure

(1)触发决策的执行与配置

系统从前向输出的结构化记录中读取判别裕度作为核心判据,并支持两种灵活的触发配置模式以适应不同的推理场景。

继承策略:该策略直接依据前向阶段由基于 Youden 指数确定的最优判别阈值 τ^* 预先计算得到。在该模式下,验证模块仅须执行常数时间复杂度的逻辑判定,即可严格遵循在验证集上拟合得到的最优决策边界,从而在统计意义上实现对正确样本保留率与错误样本拦截率之间的最优权衡。

运行时重写:为增强算法在不同实验设置及部署场景下的灵活性,系统支持在推理阶段引入手动设定的判别阈值 τ_{manual} 。当该阈值被启用时,系统将覆盖前向阶段生成的预置触发标记,并依据判别裕度 $S_{\text{margin}} < \tau_{\text{manual}}$ 的判定结果动态决定是否触发反向验证过程。该机制使得验证触发的敏感度可被显式调节:较高的阈值将扩大需验证样本的覆盖范围,从而提升整体预测准确性;而较低的阈值则有助于降低反向验证的调用频率,进而降低推理计算开销。

(2)候选集合构建与反事实假设

在反向验证阶段,系统并不重新生成候选答案,而是复用前向阶段构建的 Top-K 候选集合,并将其从前向推理中的预测结果重新解释为一组待检验的反事实假设。

与前向阶段侧重于基于概率分布的排序不同,反向验证阶段中候选集合的构建旨在形成一个封闭的假设空间,从而使外部判题器无需进行自由生成,而仅需在有限候选之间执行逻辑一致性校验。该设计将验证任务从开放式的答案生成问题转换为受约束的假设选择问题,在降低反向推理搜索复杂度的同时,提升了判题过程的可控性与稳定性。

(3)多选投票判题机制

不同于逐对比较的判别方式,本文采用多选判题模式对候选答案进行统一评估。系统将题目背景、问题描述以及候选列表 C_{cand} 中的全部候选选项 (L_{rank} 经随机重排以消除位置偏置)一次性输入至外部判题器,并要求其在反事实设定下对各候选进行逻辑一致性校验,从而判定在给定条件下最为合理且自洽的候选答案。

为了抑制判题模型的生成噪声,系统采用严格受约束的提示词模板。如图 3 所示,多选判题的提示词构建由系统提示词与用户提示词两部分组成,其设计目标在于通过身份约束与输出格式约束相结合的方式,提升判题结果的规范性与可解析性。系统提示词通过明确模型的角色定位为严格的逻辑推理判题器,并施加严格的输出约束,要求模型仅返回单一候选选项的字母标识(A/B/C/D),且禁止生成任何附加解释性文本或标点符号;用户提示词则将题目背景、问题描述以及经随机重排的候选选项进行结构化组织,并再次显式强调单字符输出要求。

输入数据	系统提示词
Background: 题目背景 Question: 问题描述 Choices_text: 候选+候选描述 Shuffled: 随机选项	"你是严格的逻辑推理判题器。" "给定背景、题目与多个候选选项,选择最符合题意且最可能为正确答案的那一个。" "你必须只输出一个大写字母(A/B/C/D),不要输出任何解释、不要输出标点、不要输出多余空格。"
	用户提示词
	f'背景: {background}\n题目: {question}\n\n"f'候选选项: \n{choices_text}\n\n"f'请从 {', '.join(shuffled)} 中选择唯一正确答案, " "只输出一个字母: \n答案: "
输出结果	
单大写字母回答(A/B/C/D)	

图 3 多选判题提示词构建

Fig. 3 Prompt construction for multiple-choice judgment tasks

为了降低单次判决的随机性,系统对每个样本执行 N 次独立投票(如 $N=3$)。设 V_o 为选项 o 获得的票数,系统采用平滑处理计算判题概率分布 P_{judge} ,如式(9)所示:

$$P_{\text{judge}}(o) = \frac{V_o + \alpha}{\sum_{k \in C_{\text{cand}}} (V_k + \alpha)}$$

(9)

其中, α 为平滑系数。基于该分布,可以计算判题器的首选答案 $O_{\text{judge } o}$ 。

基于反事实重构与多选投票,算法 1 给出反向验证算法的具体流程。

算法 1 反向验证算法

输入:问题 Q ,原始候选集合 C_{cand} ,投票次数 N
输出:反向判别答案 O_{judge} ,判别概率分布 P_{judge}

1. 初始化投票计数器 $V = \{o; 0 \text{ for } o \text{ in } C_{\text{cand}}\}$
2. For $n=1$ to N do:
3. Shuffled_Candidates=Random_Shuffle(C_{cand})//随机打乱选项顺序
4. Prompt=Construct_Prompt(Q , Shuffled_Candidates)//构建判题提示词
5. Vote_Result=Judge_Model(Prompt)//调用外部判题器
6. Update_Count(V , Vote_Result)//更新票数
7. End For
8. 计算平滑概率分布 P_{judge}
9. $O_{\text{judge}} = \text{argmax}(P_{\text{judge}})$
10. 返回 O_{judge} 、 P_{judge}

通过引入基于反事实重构的反向逻辑校验模块,系统在前向推理结果之外,获得一个经由独立判别信号评估的候选答案 O_{judge} ,进一步输出刻画其判别置信度的概率分布 P_{judge} 。

2.5 动态决策门控机制

在前两个阶段中,系统通过前向生成获取了初始预测

O_{fwd} ,并通过反向判题器获取了基于反事实校验的候选预测 O_{judge} 。由于外部判题模型本身并非无误,其输出可能受到任务复杂度、候选语义相似性及模型偏置的共同影响,若直接采用反向结果,可能将原本正确的前向预测误改为错误答案。为此,本文引入基于判题裕度的动态决策门控机制,将反向验证结果视为带置信强度的辅助证据,而非绝对真值,仅在其判别优势足够显著时才执行翻转操作。

设定动态决策判别裕度如式(10)所示:

$$M_{\text{judge}} = P_{\text{judge}}(o_{\text{top1}}) - P_{\text{judge}}(o_{\text{top2}})$$

(10)

与前向判别裕度类似,当判题裕度 M_{judge} 取值较大时,表明判题器对首选候选 o_{top1} 给出了明显高于其余选项的支持。此时,反向验证信号具有较高的区分度与可靠性,可被视为强反向证据。

相反,当 M_{judge} 取值较小时,反向验证信号的不确定性较高,直接采纳可能引入额外噪声,因此在决策融合阶段对其影响进行抑制,以避免对前向推理结果造成不必要的干扰。

基于判题裕度 M_{judge} ,本节引入单向翻转门控机制,如式(11)所示:

$$O_{\text{final}} = \begin{cases} O_{\text{judge}}, & \text{if}(O_{\text{judge}} \neq O_{\text{fwd}}) \wedge (M_{\text{judge}} \geq \delta) \\ O_{\text{fwd}}, & \text{other} \end{cases}$$

(11)

其中, O_{final} 为最终输出答案, δ 为预设的翻转阈值。当反向判题结果 O_{judge} 与前向预测结果 O_{fwd} 一致时,双向推理在候选答案上达成一致判定,此时无须触发翻转操作;当 $O_{\text{judge}} \neq O_{\text{fwd}}$ 时,系统进入冲突判定状态,仅当判题器的判别裕度 M_{judge} 超过预设安全阈值 δ 时,触发翻转操作修正前向预测,以降低由不可靠反向信号引入误判的风险。

3 实验结果与分析

3.1 数据集

实验在 LogiQA 2.0^[29]数据集上对模型方法有效性进行验证。该数据集源自于中国公务员考试,由大规模逻辑推理阅读理解问题组成。每条数据由一个段落、一个问题、若干选项以及一个正确答案组成,推理类型包括分类推理、充分条件推理、必要条件推理、联合推理以及析取推理。其中,训练集共 12751 条数据,验证集共 1593 条数据,测试集共 1594 条数据。

为进一步验证方法在不同逻辑推理基准上的泛化性,在 ReClor 数据集^[30]上进行补充实验。ReClor 来源于标准化考试场景,题目通常包含更强的论证结构分析和选项辨析特征,与 LogiQA 2.0 在题型风格和文本组织方式上形成互补,因此适合用于评估本文方法在其余数据集上的能力。

对于不同数据集,双阶段触发与门控所涉及的阈值均在各自训练集上拟合确定,并固定用于测试集评估。

3.2 实验设置与评测指标

在固定前向推理器的基础上,构建反向验证推理流程。外部判题器采用 Qwen/Qwen3-235B-A22B-Instruct-2507,通过 API 方式进行调用,温度设为 0,最大输出长度设为 512 Tokens。反向验证阶段复用前向模型在答案位置生成的 Top- K 候选集合构造反事实假设,并通过多轮随机重排与投票获得判题分布。在该外部模型上,按照与本文相同提示词

运行得到的准确率为 74.34%。该准确率结果表明,外部判题器能够提供有价值的独立判别信号,但并不意味着其可被视为绝对可靠的验证标准。特别是在复杂语义歧义、候选高度相近或特定逻辑类型样本中,外部判题器仍可能产生不稳定判断。因此,本文在系统设计上未将其输出直接作为最终决策依据,而是通过动态门控机制,对其可能带来的随机性与偏置进行缓解。本文关注的并非构造一个完美判题器,而是在外部验证器并不完美的现实前提下,探索如何以较低成本利用其独立信号提升整体推理的可靠性。

为避免候选位置偏置,每轮投票前均对候选选项顺序进行随机打乱,并要求判题器仅输出单个候选字母标识(A/B/C/D)。双阶段触发机制中所涉及的阈值均在训练集上利用 ROC 曲线和 Youden 指数拟合确定,并固定用于测试集评估。

为避免对反向验证行为统计产生歧义,对实验中涉及的若干过程性指标做统一定义。设测试集共有 M 个样本,第 i 个样本的前向答案记为 $y_{f,i}$,最终输出答案记为 $y_{o,i}$,标准答案记为 $y_{t,i}$ 。对于触发反向验证的样本,设外部判题器共执行 N 轮投票,候选答案 a 在第 i 个样本上的得票数记为 $c_i(a)$ 。

(1)有效翻转:若前向答案原本错误,而最终输出被修正为正确答案,则称该样本发生了有效翻转。有效翻转反映了反向验证机制带来的真实纠错收益,如式(12)所示:

$$y_{f,i} \neq y_{t,i}, y_{o,i} = y_{t,i}$$

(12)

(2)破坏性翻转:若前向答案原本正确,但在反向验证后被改为错误答案,则称该样本发生了破坏性翻转。破坏性翻转如式(13)所示:

$$y_{f,i} = y_{t,i}, y_{o,i} \neq y_{t,i}$$

(13)

(3)无收益翻转:若前向答案原本错误,且在反向验证后答案虽发生了改变,但最终输出仍然错误,则称该样本发生了一次无收益翻转。无收益翻转表示系统对原答案进行了修改,但并未产生真实纠错收益,如式(14)所示:

$$y_{f,i} \neq y_{t,i}, y_{o,i} \neq y_{t,i}, y_{o,i} \neq y_{f,i}$$

(14)

(4)净收益:用于综合衡量验证机制带来的整体收益。净收益如式(15)所示:

$$G = N_{\text{valid}} - N_{\text{invalid}}$$

(15)

其中, N_{valid} 表示有效翻转数, N_{invalid} 表示破坏性翻转数。 $G > 0$ 表明系统通过反向验证带来的纠错收益大于误修正损失; $G < 0$ 则说明其副作用超过收益。

(5)风险收益比:用于衡量破坏性翻转相对于有效翻转的代价,该值越小越好,表示每获得一次有效纠错所付出的误修正代价越低。风险收益比如式(16)所示:

$$R = \frac{N_{\text{invalid}}}{\max(N_{\text{valid}}, 1)}$$

(16)

3.3 关键模块有效性分析

在本实验中,前向推理阶段所使用的基础模型为 Qwen2.5-7B-Instruct。7B 规模模型在推理成本、微调开销与实验复现难度之间具有较好的平衡,适合作为大批量对比实验的统一底座;该规模模型已具备较稳定的阅读理解与逻辑推理能力,能够输出具有区分性的候选概率分布;同时,相较于更大规模模型,7B 前向模型仍保留了一定的错误空间,更适合观察反向验证模块对高风险样本的纠错收益。同时需要

指出,本文方法在实现上使用了单一前向架构,但从机制上看并不依赖于特定模型参数或训练方式。该框架仅要求前向推理器能够输出结构化答案,并在答案位置提供可解析的候选概率分布。因此,其原则上可迁移至其他同类指令微调语言模型。

本文在使用该模型进行前向内容生成前,使用群组相对优势策略优化算法进行输出格式对齐以及模型初步训练,通过构建专家提示词模板、结构化输出格式、多维分层奖励机制方法进行基座模型 Qwen2.5-7B-Instruct 的优化,以保证模型对于问题的回答具有一定的格式遵循率以及准确率,便于后续实验的展开。

前向评估主要内容为过程分数、终端 Token 置信度以及判别裕度,评价的指标为区分能力、最优阈值、正确样本保留率以及触发验证比例。如表 1 所列,判别裕度在 AUC 指标上达到了 0.7497,优于过程分数。更重要的是,在最优阈值下,判别裕度能够保留 76.00% 的正确样本,同时仅触发 35.01% 的反向验证,实现了计算效率与筛选精度的最佳平衡。

表 1 不同前向置信度指标的性能对比评估

Table 1 Performance comparison of different forward confidence indicators

置信度指标	区分能力	阈值	保留率/%	触发率/%
过程分	0.5955	0.458225	75.30	29.17
终端置信度	0.7488	0.999992	72.17	38.83
判别裕度	0.7497	0.999974	76.00	35.01

同时,从数据中可以看出,终端 Token 置信度也具有较 强的判题能力。如图 4 所示,模型预测准确率与终端 Token 置信度之间呈现出正相关关系。当 ETC 严格等于 1.0 时,尽管该类样本在整体数据集中占比相对较小(约 100 个样本),其预测准确率仍高达 95.0%。这一结果表明,当模型在答案终止位置表现出完全的确定性时,其内部推理过程通常具有较高的一致性与稳定性,并与训练分布中的高置信模式高度匹配。在此情形下,直接采纳前向推理结果具有较高的可靠性。

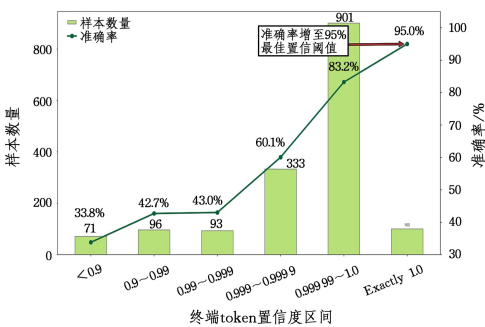


图 4 不同终端 Token 置信度区间的准确率变化

Fig. 4 Accuracy variation across different terminal Token confidence intervals

最终筛选后的准确率变化如图 5 所示。第一阶段,剔除准确率高达 95.00% 的绝对确定样本,这虽使剩余样本的准确率降至 70.61%,但有效隔离了无须验证的简单数据,漏检的样本多为模型幻觉问题引起。第二阶段,经判别裕度筛选后,留存的高置信度样本准确率升至 84.28%,而被拦截的高

风险样本准确率仅为 49.55%，漏检样本通常由决策阈值的取值引起。数据的分化证实了该机制能精准锁定潜在错误，在大幅提升验证针对性的同时，实现了计算效率与推理鲁棒性的最佳平衡。

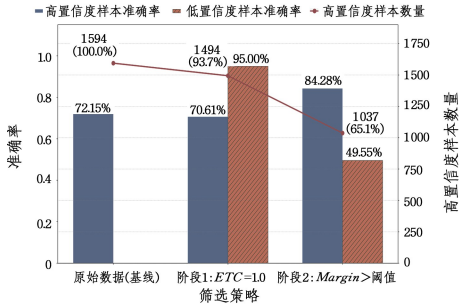


图 5 前向置信度筛选过程中的准确率变化

Fig. 5 Accuracy variation during forward confidence screening

为了验证所提反向逻辑校验机制的有效性，并探索最优的超参数配置，本节在 LogiQA 2.0 实验集上进行了多组对比实验。实验重点考查了两个核心参数对推理性能的影响：(1)候选空间大小(K)，即前向推理生成 Top- K 候选答案的数量，分别选取 $K=2$ 和 $K=3$ 进行测试；(2)判题投票次数 N ，即外部判题器对每个样本进行独立投票的次数，分别选取 $N=1,3,5$ 进行测试。实验结果如表 3 所列，主要评价指标包括最终准确率、平均推理耗时以及翻转统计。

表 2 不同反向验证配置下的性能对比

Table 2 Performance comparison under different reverse verification configurations

K	N	准确率/%	翻转	有效翻转	破坏性翻转	平均耗时/s
2	1	75.47	204	105	52	0.31
2	3	75.72	217	111	54	1.20
2	5	75.85	210	106	47	2.45
3	1	76.04	255	130	68	0.45
3	3	77.16	245	137	57	1.32
3	5	77.16	232	133	53	2.43

在实验中，最重要的是准确率与计算开销的权衡。图 6 展示了不同参数配置下模型最终准确率与推理时间的变化趋势。

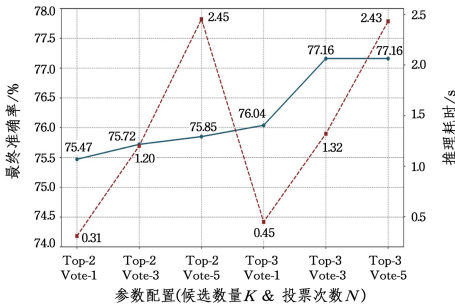


图 6 不同验证策略下的准确率与时间开销的对比

Fig. 6 Comparison of accuracy and time cost under different validation strategies

从图 6 及表 2 的数据中首先可以观察到 Top- K 的扩展效应，在相同的投票策略下 $K=3$ 的表现普遍优于 $K=2$ 的表现。特别是 K3-N3 组合，其准确率达到 77.16%，相较于 K2-N3 的 75.72% 有提升。其次，数据体现了投票机制的鲁棒性

提升与边际效应，增加投票次数 N 能有效提升判别的稳定性。从 $N=1$ 增加到 $N=3$ 时，准确率均有明显上升，其中 K3 组从 76.04% 提升至 77.16%。然而，继续增加至 $N=5$ 时，准确率提升趋于饱和，K3 组保持 77.16% 不变，但计算时间却成倍增加，从约 1.32 s 激增至 2.43 s。综合考虑性能收益与计算成本，K3-N3 是当前算法下的最优配置，它在保持最高准确率的同时，将时间开销控制在了可接受的范围内。

图 7 详细统计了不同配置下的翻转行为。实验数据显示出正向增益结果，在所有的实验配置中，有效翻转的数量始终高于引入破坏性翻转的数量。以最优配置 K3-N3 为例，系统成功修正了 137 个错误样本，仅误伤了 57 个正确样本，净收益为 80 个样本，直接推动了整体准确率突破 77%。此外，通过对比 K3-N1 和 K3-N3 可以发现，投票机制对噪声的抑制作用。N1 虽然也能带来较多的有效翻转(130 个)，但也引入了最高的破坏性翻转(68 个)，这说明单次判题存在较大的随机噪声。引入多数投票机制(K3)后，虽然总翻转数略有下降(255→245)，但破坏性翻转数量降低幅度更大(68→57)，证明了多选投票机制在过滤判题噪声及提升验证可靠性方面的重要作用。

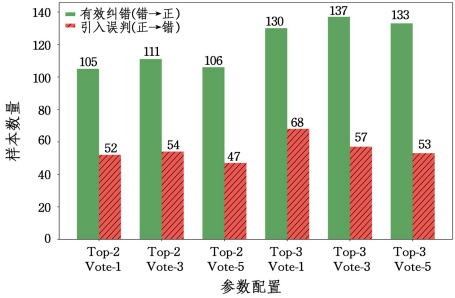


图 7 不同策略下的反向验证纠错效率分析

Fig. 7 Analysis of reverse verification error correction efficiency under different strategies

为了深入探究动态门控机制在本文方法中的实际作用，本节对比分析了无门控策略($\delta=0$)与引入判别裕度阈值($\delta=0.17$)后的模型表现。如表 3 所列，引入 $\delta=0.17$ 的门控机制虽因有效翻转数减少(137→101)导致准确率微幅波动，但实质上实现了从高覆盖度修正向高置信度判别的演进。修正操作的翻转精度由 55.9% 提升至 66.9%，表明决策可信度增强。更为关键的是，门控机制有效构建了安全边界，将破坏性翻转(正→错)由 57 例抑制至 33 例(降幅 42%)，并将风险比从 0.42 优化至 0.33。结果表明，该机制在抑制外部判题噪声、规避误干预风险方面具有决定性作用。

为进一步从单个样本层面展示本文方法的工作机制，在上述定量结果的基础上，从测试集中选取了若干具有代表性的样本进行定性分析，分别对有效翻转、破坏性翻转以及无收益翻转 3 类典型情形。相关结果如表 4 所列。

由表 4 中 3 个典型案例可以看出，本文方法在单样本层面呈现出较为清晰的作用机制。“出租车司机关空调是否合理”属于有效翻转案例，前向模型将答案误判为 C，但外部判题器在 Top-3 候选中对正确选项 B 给出了更强支持，最终实现纠错，说明反向验证能够在前向推理被表面合理性干扰时

提供有效的独立判别信号；“城市新地标建设应表达何种主旨”属于破坏性翻转案例，前向答案 D 原本正确，但判题器在语义相近的候选项之间给出了偏向 A 的不稳定判断，最终导致误修正，说明外部验证结果并非始终可靠，也从反面体现了引入动态门控约束翻转操作的必要性；“关闭城市夜灯能否恢复候鸟路径”则属于无收益翻转案例，尽管系统将前向错误答案 D 修正为 B，但由于正确答案 A 未进入 Top-3 候选集合，反向验证只能在错误候选内部重排序，仍无法完成真实纠错，这进一步表明本文方法的效果不仅依赖判题器质量，也受制于前向阶段的候选召回能力。最终翻转数量如表 2 所列，有效性翻转多于破坏性翻转与无收益翻转。

表 3 门控机制对翻转质量与安全性的影响的对比

Table 3 Comparison of impact of gating mechanisms on error correction quality and safety

门控策略	翻转	有效翻转	破坏性翻转	无收益翻转	修正精度/%	风险收益比
无	245	137	57	51	55.9	0.42
有	151	101	33	17	66.9	0.33

表 4 典型案例分析与结果对比

Table 4 Analysis and comparison of representative cases

案例	样本类型	前向答案	标准答案	Top-3 候选答案	判题器分布	最终结果
出租车司机 关空调 是否合理	有效翻转	C	B	C,D,B	B:0.667, C:0.167, D:0.167	B
城市新地标 建设应表达 何种主旨	破坏性翻转	D	D	D,A,C	A:0.500, D:0.333, C:0.167	A
关闭城市夜灯 能否恢复 候鸟路径	无收益翻转	D	A	D,C,B	B:0.667, C:0.167, D:0.167	B

3.4 对比实验与消融实验

为了验证本文方法的优越性，将其与当前主流的 4 种推理增强策略进行了横向对比。(1)Self-Consistency^[8]:通过多次采样($K=3$)并进行多数投票来提高答案的鲁棒性。(2)Self-Refine^[31]:基于大模型自身的反思能力，通过 Prompt 引导模型对初始答案进行批评与修正。(3)NLI-Verifier^[32]:引入外部 NLI 模型判断前提与假设之间的逻辑矛盾。(4)Double-checker^[33]:通过显式自我批判和迭代式修正来增强慢思考大模型的推理能力。

所有方法的对比均建立在相同的前向推理器基础上，其中本文方法在固定前向推理器之上引入反向验证与动态门控模块，其余基线则在相同前向推理器或相同基础设置下进行比较。

表 5 列出了各方法在 LogiQA 2.0 与 ReClor 测试集上的最终准确率与平均推理延迟。为了公平对比，实验将前向生成时间与验证/修正时间合并计算为总延迟。

从表 5 的实验结果可以清晰地观察到，在 Logiqa 2.0 数据集上，单纯增加计算量的策略并不一定能带来性能的提升。Self-Consistency 虽然通过 3 倍采样将推理延迟推高至 9.74s，但在该数据集上的准确率提升仅为 0.43%。相较而言，本文方法与对比策略均采用了基于置信度的按需计算模式，仅对 34.94%(557/1594)的低置信样本触发验证，从而降

低了平均推理延迟。Self-Refine 虽引入了外部判别信息，但其基于非结构化语义反馈的方式难以有效打破模型既有推理偏置，因而仅带来有限的性能提升(+0.78%)。NLI-Verifier 受限于通用蕴含模型在复杂逻辑场景下的判别能力，其验证信号反而对整体性能产生了负面影响。而 Double-check 方法容易引入更多无效或破坏性修改，缺乏筛选机制。

表 5 不同推理增强策略的性能对比

Table 5 Performance comparison of different inference enhancement strategies

方法	数据集	准确率/%	前向耗时/s	验证耗时/s	总延迟/s	相对提升/%
Standard	Logiqa2	72.15	3.39	—	3.39	—
CoT	ReClor	73.20	2.01	—	2.01	—
Self-Consistency	Logiqa2	72.46	9.74	—	9.74	+0.43
	ReClor	73.60	7.14	—	7.14	+0.43
Self-Refine	Logiqa2	72.71	3.39	0.86	4.25	+0.78
	ReClor	77.00	2.01	0.80	2.81	+4.94
NLI-Verifier	Logiqa2	70.89	3.39	1.91	5.30	-1.75
	ReClor	81.40	2.01	0.96	2.97	+10.07
Double-check	Logiqa2	75.28	3.39	3.76	7.15	+4.16
	ReClor	81.80	2.01	5.09	7.10	+10.5
Reverse	Logiqa2	77.16	3.39	1.32	4.71	+6.94
(Ours)	ReClor	89.60	2.01	0.94	2.95	+18.30

实验结果表明，本文提出的反向验证算法在推理精度与计算效率之间实现了一定的平衡。通过反向构造反事实假设并结合动态门控机制，模型的整体准确率提升 6.94%。同时，其平均总推理延迟被控制在 4.71s，在 ReClor 测试集上的结果表明了本文方法具有一定的泛化能力。

为了对比不同规模样本下方法的效率，实验统计了各方法在不同规模 Logiqa 2.0 测试集样本下的总耗时、平均验证耗时以及平均总延时，如表 6 所列。

表 6 不同规模样本下的方法效率对比

Table 6 Efficiency comparison under different sample sizes

方法	样本规模	总耗时/s	平均验证耗时/s	平均总延时/s
Standard CoT	400	1356.8	—	3.392
	800	2714.4	—	3.393
	1594	5403.7	—	3.390
Self-Consistency	400	3897.6	—	9.744
	800	7791.2	—	9.739
	1594	15528.7	—	9.742
Self-Refine	400	1691.2	0.848	4.228
	800	3391.2	0.859	4.239
	1594	6771.3	0.861	4.248
NLI-Verifier	400	2120.8	1.915	5.302
	800	4283.2	1.919	5.354
	1594	8440.2	1.913	5.295
Double-checker	400	2880.4	3.796	7.201
	800	5772.0	3.772	7.215
	1594	11400	3.761	7.152
Reverse(Ours)	400	1888.4	1.334	4.721
	800	3768.0	1.323	4.710
	1594	7514.1	1.321	4.714

从表 6 中可以看出，随着样本规模的增加，各方法总耗时均近似线性增长。结果表明，双阶段触发机制不仅能够降低平均验证成本，也能够测试规模扩大时维持较好的效率可扩展性。

为了深入探究不同方法在处理低置信度样本时的行为差

异,实验统计了各方法在触发验证机制后的修正行为,如表 7 所列。

表 7 各方法在验证阶段的修正行为统计
Table 7 Statistics on modification behavior of each method during validation phase

指标	数据集	修改数	修改 意愿/%	有效 翻转	破坏 翻转	修正 精度/%	净收益
Self-Refine	Logiqa2	139	25.0	52	43	37.4	+9
	ReClor	44	19.1	23	4	52.3	+19
NLI-Verifier	Logiqa2	344	61.8	120	140	34.9	-20
	ReClor	121	52.6	73	32	60.3	+41
Double-checker	Logiqa2	334	21.0	153	103	45.8	+50
	ReClor	70	14.0	49	6	70.0	+43
Reverse (Ours)	Logiqa2	245	44.0	137	57	55.9	+80
	ReClor	109	29.1	89	7	81.7	+82

从表 7 可以看出,在 Logiqa 2.0 数据集上,Self-Refine 虽通过引入外部判别视角在一定程度上削弱了模型的过度自信(修改意愿为 25.0%),但其翻转信号的判别性不足。该特性表现为较高比例的出现由对改错的现象(43 例),从而抵消了有效翻转带来的正向收益,使其净收益仅为+9。相比之下,NLI-Verifier 则呈现出明显的过度修正倾向,其高达 61.8%的修改意愿主要源于通用蕴含模型在复杂长程逻辑场景下的误判,进而引发大量错误翻转(140 例)并导致整体性能下降(净收益-20)。而 Double-checker 表现出较高的修改数,但是破坏性翻转数量多,修正精度不够高,导致最终净收益不足。

本文提出的反向验证算法在修正意愿与修正精度之间实现了更为合理的平衡。基于反事实构造的反向逻辑检验能够更敏锐地捕捉前向推理中的逻辑偏差,使模型在必要时主动触发修正(修改意愿为 44.0%)。实验结果显示,该方法的修正精度达到 55.9%,误改样本数量仅为 57 例,最终实现了+80 个样本的净收益,是 Self-Refine 的 8.9 倍。同时,在 ReClor 测试集上的结果表明,本文方法的修正精度达到了 81.7%,高于其余几种方法。

为量化评估本文方法中各组件的贡献,以基座模型为基准,在 Logiqa 2.0 数据集上逐步叠加全量验证、置信度筛选及动态门控机制进行消融实验,实验结果如表 8 所列。

表 8 系统架构的消融实验结果
Table 8 Ablation experiment results of system architecture

数据集	模型架构	准确率/%	修正 精度	收益	平均验证 耗时/s
Logiqa 2.0	Baseline	72.15	0	0	0
	+Verify All	76.73	51.17	+73	4.1117
	+ETC	72.46	5.00	+5	0.3248
	+Margin	76.16	57.43	+76	0.9963
	+Confidence Filter	77.16	55.92	+80	1.3206
	+Dynamic Gate	76.41	66.89	+68	1.3187
ReClor	Baseline	73.20	0	0	0
	+Verify All	85.42	72.09	+76	4.3007
	+ETC	74.13	5.47	+6	0.3108
	+Margin	86.78	83.06	+79	1.0118
	+Confidence Filter	89.60	81.70	+82	1.3213
	+Dynamic Gate	86.73	83.45	+71	1.3029

从表 8 数据分析可得,全量验证虽然通过强制反思带来了 175 次纠错,将准确率提升至 76.73%,但其代价极高,其

修正精度仅为 51.17%;同时,由于对所有样本执行验证,其平均耗时高达 4.11s。引入 ETC 后,准确率提升至 72.46%,提升幅度不大,但其验证耗时短,仅需 0.3248s。引入 Margin 筛选指标后,准确率达到 76.16%。在引入置信度筛选后,系统性能达到最优(77.16%)。筛选机制成功在将耗时降至 1.3206s(降幅 67.9%)的同时,获得了最高的净收益(+80)。完整算法进一步引入动态门控,虽然净收益有所回落,但修正精度跃升至 66.89%。在 ReClor 测试集上的消融实验结果同样表明了本文方法的有效性。

结束语 本文围绕大语言模型在复杂逻辑推理中易出现错误累积、高置信误判难以识别以及全量外部验证计算开销高的问题,构建了一种以选择性验证和稳健修正为核心的反向验证推理框架。研究结果表明,该方法的价值不仅在于提升最终作答表现,更在于将“是否需要验证”和“是否值得修正”这两个关键决策从经验启发转化为可量化、可控制的推理流程:前者通过双阶段置信度触发降低了低价值验证带来的额外开销,后者通过动态门控减少了外部判别噪声引发的破坏性翻转,使系统在精度、效率与纠错可靠性之间取得了更为均衡的表现。本文实验也说明,反向验证并非越频繁越有效,真正关键的是如何在候选覆盖能力、验证成本与修正安全性之间建立稳定平衡,这一点对于面向实际部署的逻辑推理系统尤为重要。后续工作可从 3 个方向继续推进:(1)增强方法对不同前向模型和不同判题器配置的适应性,验证该框架在更多基座模型与任务场景下的泛化能力;(2)引入更细粒度的不确定性建模与候选质量评估机制,进一步减少无效触发和误修正;(3)探索低成本验证器替代、高效提示压缩及训练阶段与推理阶段协同优化的方案,以提升方法在真实应用环境中的可复用性与部署价值。

参 考 文 献

[1] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems 30; Neural Information Processing Systems Foundation. 2017:5998-6008.

[2] BROWN T B, MANN B, RYDER N, et al. Language models are few-shot learners[C]// Advances in Neural Information Processing Systems 33; Neural Information Processing Systems Foundation. 2020:1877-1901.

[3] OUYANG L, WU J, JIANG X, et al. Training language models to follow instructions with human feedback[C]// Advances in Neural Information Processing Systems 35; Neural Information Processing Systems Foundation. 2022:27730-27744.

[4] WANG W Q, GUO M F, YANG D X, et al. Survey on the development and applications of large language models[J]. Journal of Zhongyuan University of Technology, 2025, 36(2): 1-8.

[5] RAFAILOV R, SHARMA A, MITCHELL E, et al. Direct preference optimization: Your language model is secretly a reward model[J]. Advances in Neural Information Processing Systems, 2023, 36: 53728-53741.

[6] SHAO Z, WANG P, ZHU Q, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models[J].

arXiv:2402.03300,2024.

[7] WEI J,WANG X,SCHUURMANS D,et al.Chain-of-thought prompting elicits reasoning in large language models[C]// Advances in Neural Information Processing Systems 35:Neural Information Processing Systems Foundation. 2022:24824-24837.

[8] WANG X,WEI J,SCHUURMANS D,et al.Self-consistency improves chain of thought reasoning in language models[C]// The Eleventh International Conference on Learning Representations. 2023:12648-12671.

[9] ZENG B Q,LUO Z K,CHEN W H. Natural language inference via negation augmentation[J]. Computer Systems & Applications,2025,34(9):104-111.

[10] ZHANG Z,ZHANG A,LI M,et al. Automatic chain of thought prompting in large language models[J]. arXiv: 2210.03493, 2022.

[11] WANG L,XU W,LAN Y,et al. Plan-and-solve prompting:improving zero-shot chain-of-thought reasoning by large language models[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. ACL,2023:2609-2634.

[12] CHEN W,MA X,WANG X,et al. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks[J]. arXiv:2211.12588,2022.

[13] YAO S,YU D,ZHAO J,et al. Tree of thoughts:deliberate problem solving with large language models[C]// Advances in Neural Information Processing Systems 36: Neural Information Processing Systems Foundation. 2023.

[14] YAO S,ZHAO J,YU D,et al. ReAct:synergizing reasoning and acting in language models[C]//The Eleventh International Conference on Learning Representations. 2023.

[15] SCHICK T,DWIVEDI-YU J,DESSÌ R,et al. Toolformer: Language models can teach themselves to use tools[J]. Advances in Neural Information Processing Systems,2023,36:68539-68551.

[16] WENG Y,ZHU M,XIA F,et al. Large language models are better reasoners with self-verification[C]// Findings of the Association for Computational Linguistics; EMNLP 2023. ACL, 2023:2550-2575.

[17] ZHANG Y,KHALIFA M,LOGESWARAN L,et al. Small language models need strong verifiers to self-correct reasoning [C]//Findings of the Association for Computational Linguistics; ACL 2024. ACL,2024:15637-15653.

[18] TAN C H,JIA K B,WANG H Y. Text summarization quality evaluation based on large language model[J]. Computer Systems & Applications,2025,34(2):28-36.

[19] GOU Z,SHAO Z,GONG Y,et al. CRITIC:large language models can self-correct with tool-interactive critiquing[C]// International Conference on Learning Representations 2024; ICLR. 2024.

[20] LIN Z,GOU Z,LIANG T,et al. CriticBench:benchmarking LLMs for critique-correct reasoning[C]//Findings of the Association for Computational Linguistics; ACL 2024. ACL, 2024: 1585-1603.

[21] MEKALA D,WESTON J E,LANCHANTIN J,et al. TOOL-VERIFIER: generalization to new tools via self-verification [C]//Findings of the Association for Computational Linguistics;EMNLP 2024. ACL,2024:5026-5041.

[22] LIU C,YUAN Y,YIN Y,et al. Safe:enhancing mathematical reasoning in large language models via retrospective step-aware formal verification[C]// Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics. ACL, 2025:12171-12186.

[23] JANG H,JANG Y,LEE S,et al. Self-training large language models with confident reasoning[C]// Findings of the Association for Computational Linguistics; EMNLP 2025. ACL, 2025: 14925-14939.

[24] LUO H K,GE Y F,LIU S. Research progress of large language models in mathematical reasoning[J]. Computer Engineering, 2024,50(9):1-17.

[25] ARIYANI N F,BOURAOUI Z,BOOTH R,et al. There’s no such thing as simple reasoning for LLMs[C]// Findings of the Association for Computational Linguistics; ACL 2025. ACL, 2025:4503-4514.

[26] TANG Z,SHEN K,KEJRIWAL M. An evaluation of estimative uncertainty in large language models[J]. NPJ Complexity,2026, 3:8.

[27] PENG Z,LIU X,YANG G,et al. The uncertainty advantage: enhancing large language models’ reliability through chain of uncertainty reasoning [J]. Pattern Recognition Letters, 2026, 200:30-36.

[28] SONG J,TANG X,HAO K,et al. An uncertainty-aware framework integrating large language model and fuzzy inference system for commonsense reasoning[J]. Expert Systems with Applications,2026,310:131273.

[29] LIU H,LIU J,CUI L,et al. Logiqa 2. 0—an improved dataset for logical reasoning in natural language understanding [J]. IEEE/ACM Transactions on Audio,Speech, and Language Processing,2023,31:2947-2962.

[30] YU W H,ZHANG J,et al. ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning[C]//ICLR. 2020.

[31] MADAAN A,TANDON N,GUPTA P,et al. Self-refine: Iterative refinement with self-feedback[J]. Advances in Neural Information Processing Systems,2023,36:46534-46594.

[32] SANYAL S,XIAO T,LIU J,et al. Are machines better at complex reasoning? unveiling human-machine inference gaps in entailment verification[C]// Findings of the Association for Computational Linguistics; ACL 2024. 2024:10361-10386.

[33] XU X,CHEN T,ZHANG F,et al. Double-Checker: Enhancing Reasoning of Slow-Thinking LLMs via Self-Critical Fine-Tuning [J]. arXiv:2506.21285,2025.



Li Peng, born in 1979, Ph.D, professor, Ph.D supervisor, is a member of CCF (No. 48573M). His main research interests include computer communication networks, cloud computing and information security.