arXiv:2611.33078v1 [cs.CL] 24 Nov 2026

FOL-Proof：通过确定性语义验证大语言模型文本到逻辑的翻译

Yisrael Clark[1],Guy Katz[1]

耶路撒冷希伯来大学，以色列耶路撒冷
yisrael.clark@mail.huji.ac.il

摘要。 大语言模型（LLM）越来越多地被用于将自然语言陈述转换为一阶逻辑（FOL），以提升推理能力。然而，大语言模型容易出错，因此语义不忠实的 FOL公式可能会损害下游推理。我们提出 FOL-Proof，这是一种语义防火墙流水线，可根据独立推导的语言学解释验证大语言模型生成的 FOL 翻译。FOL-Proof 通过语义分析工具（如英语资源语法和最小递归语义）确定性地获取一组候选翻译；随后，使用 SMT 求解器检查每种派生解读是否与大语言模型生成的公式逻辑等价。如果翻译忠实，则予以接受；否则予以拒绝，或选择通过反复调用大语言模型进行修复。该流程采取保守策略：搜索不完整、不受支持的结构或检查结果不确定时，会选择弃答，而不是错误拒绝。我们在成熟的基准测试（ProofWriter、RuleTaker 和 PrOntoQA）以及一个新的对抗性基准测试（LongFaithCheck）上评估了 FOL-Proof。实验表明，FOL-Proof 的选择性准确率接近完美，而直接大语言模型评判器和往返基线的表现则明显较差。

1 引言

大语言模型（LLM）在广泛的自然语言处理任务中展现出卓越的能力，包括机器翻译、文本摘要和问答 [20, 80, 83]。它们在推理任务中也取得了令人鼓舞的成果，包括数学问题求解、代码生成和逻辑推理 [6, 18, 84]。

尽管取得了这些进展，但在演绎任务（即判断结论是否可由给定前提推出）中，大语言模型并不能保证推理的可靠性 [11, 54, 81]。它们容易出现许多逻辑谬误：捏造公理、违反条件逻辑规则，以及采用循环论证 [22,25, 50, 61, 65]。例如，考虑图 1(a) 中的推理问题。正确的逻辑答案是，该结论并不能由前提推出——鲍勃不一定是学生。但大语言模型可能会通过捏造公理或

肯定后件。

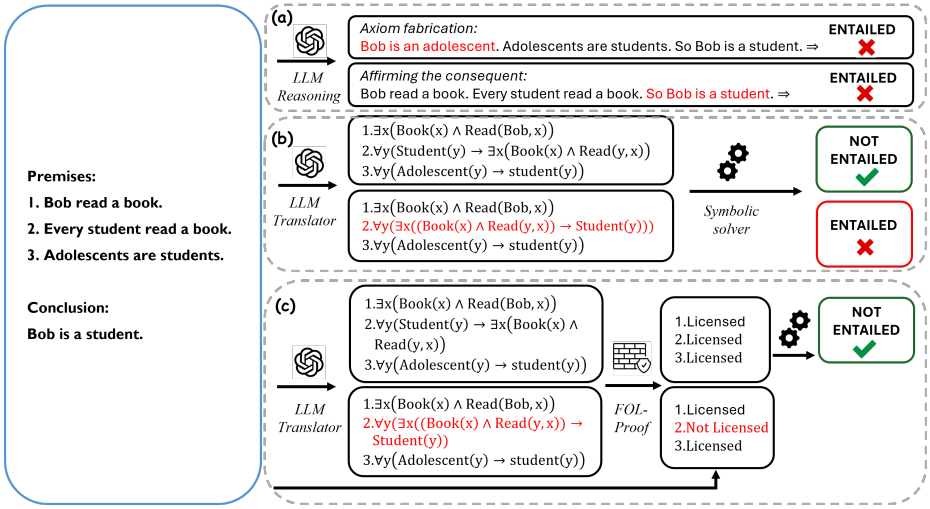


图1。 推理问题示例。正确答案是：结论并不能由前提推出。(a) 大语言模型推理可能产生逻辑谬误。(b) 大语言模型翻译器可能误译，导致求解器得出错误答案。(c) FOL-Proof 验证该翻译。

为应对这些挑战，研究界转向了神经符号集成 [10, 16, 50, 53]。其核心思想是利用大语言模型自动形式化——将自然语言句子转换为形式逻辑语言。随后，将生成的公式交由符号求解器进行下游推理。这种架构避免了违反逻辑推理的陷阱。在各种逻辑领域中，神经符号架构的实证表现均优于纯神经基线 [3, 5, 56, 68, 77]。

然而，这一流程容易产生语义不忠实 [4, 46]。仅仅生成一个结构良好的逻辑句子，并不能证明它忠实于原始的自然语言句子；事实上，已有研究表明，大语言模型经常会误译自然语言陈述。模型可能选择错误的量词或量词辖域，遗漏或反转否定，误用逻辑联结词（如图1(b)所示），遗漏前提或谓词，甚至引入源文本不支持的子句 [28, 43, 71, 74]。在这种情况下，由于求解器接收到的是错误输入，它可能得出错误结论。因此，使用大语言模型的神经符号方法并不能消除错误推理的可能性，而是将问题转移到了自动形式化阶段。

现有方法尝试通过引导生成过程（例如提示工程、中间表示），或通过事后大语言模型判断与修复，来缓解自动形式化错误。然而，这些方法容易受到大语言模型固有统计性质的影响——无论是在生成阶段还是验证阶段——因而无法提供正确性认证。

自然语言固有的歧义性带来了另一项挑战。近期对自动形式化语料的人工审查发现，多达 48% 的

句子存在合理的语言多义性，无法归结为单一的规范参考公式 [4]。例如，“Every student read a book”可以翻译为一阶逻辑（一阶逻辑） $\forall y\left(\text{学生}(y) \rightarrow \exists x\left(\text{书}(x) \wedge \text{阅读}(y,\ x)\right)\right)$（每个学生都读了一本书，但不一定是同一本），也可以翻译为 $\exists x\left(\text{书}\ (x) \wedge \forall y(\text{学生}(y) \rightarrow \text{阅读}(y,\ x))\right)$（存在一本每个学生都读过的书）。由于句子本身存在固有歧义，这两种翻译都合理。因此，通过与单一真实翻译进行比较来评估自动形式化的方法，准确性本质上存在局限 [40, 46]。

在此，我们提出一种新方法：从对源句进行的独立且以语言学为依据的语义分析中，推导出一组形式解释；然后将这些解释用作参照，检查所提出的形式化是否符合要求。检查通过使用 SMT 求解器评估逻辑等价性来完成。我们假设，这种检查能够筛选出仅保留高保真度的译文，供下游推理使用；在形式化结果到达求解器之前，过滤掉语言学分析不支持的形式化结果，从而改善逻辑推理（见图 1(c)）。我们将这项技术称为 *FOL-Proof*。

为了提高我们方法的可靠性，我们特意将 FOL-Proof 设计得较为保守。当某个候选项与某种确定性推导出的解读等价时，系统会为其认证；只有当对句法分析结果和辖域解读的搜索可证明已完整，且每一种可适配的解读都与之不等价时，系统才会明确拒绝该候选项。在其他所有情况下（搜索不完整、不受支持的结构、等价性检查无法得出结论），系统都会弃权，因此绝不会将资源上限误认为语义分歧。

在实现 FOL-Proof 时，我们结合英语资源语法 (ERG) [14, 15], 最小递归语义 (MRS) [9], 和辖域解析工具 [29]来生成候选翻译池。我们使用确定性的 MRS 到一阶逻辑适配器，将这些参考译文转换为公式，并应用 Z3 SMT 求解器 [47] 检查它们与待评估的大语言模型翻译之间的等价性。MRS 适用于此目的，因为它能够表示语义结构，同时将某些辖域选择保留为未充分指定状态；之后可以解析这些备选项，而不是过早地将其折叠为单一逻辑公式，从而保留自然语言歧义。

我们的评估表明，FOL-Proof 在三个成熟的推理基准测试（包含数千个项目）以及我们在此提出的一个由对抗性实例组成的新基准测试上，都能提供高度可靠的语义验证。我们在受控变异和大语言模型翻译上评估了 FOL-Proof，发现它在数万条形式化结果上都达到了近乎完美的准确率，表现优于直接大语言模型评判器和往返基线。此外，它能够考虑歧义并接受多种有效译文。将 FOL-Proof 用作下游推理的筛选关卡时，在所有作答问题上，其选择性准确率均达到 100%。此外，将 FOL-Proof 纳入迭代修复循环后，提供反馈时的修复率从 4.8% 提升至 22.8%；若进一步展示最接近的许可解读，修复率则提升至 62.6%。

总而言之，本文作出了以下贡献：

1. 我们提出了一种新方法，通过将大语言模型自动形式化为一阶逻辑的结果与确定性的英语句法分析结果进行核验，来验证这些结果。2. 我们开发了 FOL-Proof，这是上述方法的一种实现。3. 我们推出了一项新基准测试，旨在考验大语言模型的自动形式化和判定能力。4. 我们在受控语义变异、大语言模型翻译、端到端推理和修复循环任务上评估 FOL-Proof，并将其与直接大语言模型评判器和往返验证器在 ProofWriter [69], PrOntoQA [63], RuleTaker [8], 以及我们的新基准测试上的表现进行比较。我们发现，只要我们的方法不弃权，就能达到近乎完美的准确率。

本文其余部分的组织结构如下：第 2 节回顾相关工作，第 3 节介绍必要的背景知识并形式化语义验证问题。第4节介绍 FOL-Proof 架构和验证过程，第 5 节则描述其实现。第 6 节呈现实验评估，第 7 节总结全文并讨论未来工作。

2 相关工作

神经符号推理。 神经符号架构 [13, 50, 53] 通过将形式执行交由符号推理引擎处理，屡次展现出优于纯神经基线的性能。以这种方式成功应用的推理引擎包括自动定理证明器（ATP）和理论可满足性（SMT）求解器，例如 Z3 [47], Vampire [30],Prover9 [42], 和 Lean-4 [48]。这些成果涵盖了多种领域——从一阶逻辑推理（例如 FOLIO [19], ProverQA [57], LogicSkills [59]）和多跳规则演绎（例如 ProofWriter [69], PrOntoQA [63], LogicNLI [72]），到复杂的约束满足问题和分析性谜题（例如 AR-LSAT [82],Ze- braLogic [34], SATBench [75]）。我们的工作聚焦于一阶逻辑推理，并采用 SMT 求解器。

基于参考的等价性和结构指标。 给定一个真实值参考（GT）公式（即正确的翻译，用作参照点），可以采用不同的方法将候选翻译与其进行比较，这些方法通常针对公式所属的论域进行定制。BLEU 等字符串指标在重命名和重新排序方面较为脆弱 [70, 74]。基于求解器的方法则使用 SMT 求解器或 ATP [49, 64] 通过反驳 $\neg(\phi \leftrightarrow \psi)$ 来判定等价性。还有其他用于比较 Lean 陈述 [37] 或时序逻辑 [58] 的方法。当所需指标超越简单等价性时，一些方法会采用图距离计算 [40, 51]。相对于 GT，这些方法要么给出精确结果，要么提供分级结果；但它们也受到 GT 的限制：无法处理歧义，并且在未提供 GT 时无法适用。在 FOL-Proof 中，我们也使用否定双条件检查（第 4.5 节），但并不假定已向我们提供 GT；相反，我们使用一组语法许可的解读。

大语言模型介导的验证与选择。 当不存在真实值参考时，一些系统会执行往返翻译——将翻译转换回自然语言，并将其与源文本进行比较。VeriTrans 使用带阈值的往返相似度分数作为接受门槛 [39], ；Monty 使用大语言模型评判器对双向子句覆盖率进行评分 [36], ∀uto∃val 通过形式验证器在反复的形式语言到自然语言的循环中检查真值保持 [27], ；ProofBridge 结合了 Lean 类型检查、证明修复和双向定理等价性检查 [23],；VeriCoT 和 VERGE 则使用基于大语言模型及求解器辅助的一致性检查 [13, 64]。这些方法无需 GT 即可使用，但它们提供的语义证据仍受学习式重构步骤质量的限制：前向翻译中的系统性错误可能被后向翻译保留下来，而词汇或嵌入相似性并不意味着语义等价 [33, 46]。

另一种方法采用以大语言模型作为评判器来评估译文的忠实性，同时结合各种其他组件来提高翻译的准确率 [4, 62, 70]。例如，CLOVER 使用 SAT 求解器生成候选项存在分歧的反解释，并由大语言模型进行裁决 [62]。然而，自我验证研究表明，模型在检测细微的逻辑错误方面仍不可靠，尤其是涉及量化、辖域和多步依赖的错误 [21, 28, 33]。

尽管这些方法结合了各种形式的符号验证，但它们对源句忠实性的评估通常依赖于大语言模型判断、学习得到的重构或启发式度量。相比之下，FOL-Proof 通过确定性的语义适配器独立推导出语言学上许可的解释，并使用 SMT 求解器确立逻辑等价性，而无需依赖大语言模型给出验证结论。

行为认证与语言学基础验证。双向可证性指纹通过探查 mathlib 中形式化的逻辑后果邻域，为 Lean 4 陈述提供认证，从而给出一种概率性的ϵ忠实性概念 [46]。这类基于探查的检查仅对所考察的有限探查是可靠的，并通过选定的逻辑后果提供忠实性的行为证据。相比之下，FOL-Proof 建立其与语言学基础解读的等价性，确保得到精确的逻辑等价，而非概率性的等价。与我们的方法类似，Rademaker 等人 [60], 针对大语言模型生成的 SPARQL，检查自然语言问题的语义表示是否蕴含所生成查询的语义表示。他们指出，其工作尚属初步阶段，且缺乏评估。FOL-Proof 处理的是相关的 FOL 翻译验证任务：它检查译文与许可解读之间的等价性，区分不等价、不完整分析和无法定论的求解器结果，并在多个基准测试和下游推理中评估这些判定结果。

3 背景与预备知识

3.1 一阶逻辑(FOL)

FOL-Proof 的目标逻辑是一个标准的、无函数、无等式的一阶逻辑片段 $\mathcal{L}_{\mathrm{FOL}}$，其签名为 $\Sigma = (\mathcal{C}, \mathcal{P}, \mathrm{ar})$，包含有限个个体常量 $\mathcal{C}$ 和谓词符号 $\mathcal{P}$，其元数为 $\mathrm{ar}: \mathcal{P} \to \mathbb{N}_{\geq 1}$，以及一个可枚举的变量集合 $\mathcal{V}$；项是常量或变量。公式由原子公式 $P(t_1, \ldots, t_k)$构成，其中 $k = \mathrm{ar}(P)$，并使用 $\neg$、$\wedge$、$\vee$、$\rightarrow$、$\leftrightarrow$和量词 $\forall x$、$\exists x$；如果每个变量出现位置都受到约束，则公式是闭合的（即一个句子）。我们采用标准语义：一个结构 $\mathcal{M} = (\mathcal{D}, \cdot^{\mathcal{M}})$具有一个非空论域 $\mathcal{D}$、每个常量对应的一个元素 $c^{\mathcal{M}} \in \mathcal{D}$，以及每个谓词对应的一个关系 $P^{\mathcal{M}} \subseteq \mathcal{D}^k$。如果两个句子在每个结构中都具有相同的真值，则称它们在语义上等价 $\phi_1 \equiv \phi_2$。

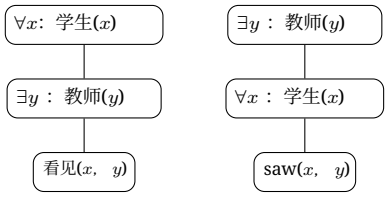
3.2 最小递归语义（MRS）与辖域图

自然语言句子通常无法蕴含唯一的逻辑表示。具体而言，量词辖域和附加标准是歧义的来源 [9,29]。

示例 *1.* 考虑句子*“每个学生都看见了一位老师，老师带着一架望远镜”*。介词短语*带着一架望远镜* 会产生附加辖域歧义。它既可以附加到动词短语（*用望远镜看见*）上，也可以附加到名词短语（*一位带着望远镜的老师*）上。此外，量词辖域也尚未确定：是否存在一位所有学生都看见的同一位老师，还是每个学生都看见了一位老师，但不一定是同一位。

为处理此类情况而不提前确定单一解读，FOL-Proof 会从英语资源语法 (ERG) 获取最小递归语义 (MRS) [9] 分析；ERG 是一种覆盖范围广、由人工构建的中心语驱动短语结构语法 (HPSG) [14], [55]。对于单个句子，ERG 可以给出多个 MRS 分析，以处理附着歧义和词汇歧义。因此，在示例 1 中，我们会针对不同的附着歧义和词汇歧义分别获得 MRS 分析。

MRS 是一种欠指定的计算语义表示。MRS 并非将每个辖域决策都直接编码在一个递归嵌套的公式中，而是表示基本谓词、它们的论元、句柄，以及可能辖域之间的约束（完整示例见附录 A）。这样，解析器便可以构建紧凑型表示，同时将部分辖域决策留待后续解析阶段处理。这些解析结果可以用辖域树表示，用以指定量词和其他辖域运算符的相对辖域。对于示例 1 中的辖域歧义，以下是两种确定量词顺序的简化辖域树（忽略望远镜从句）：



我们使用 MRS 并非意在回归传统语法翻译，而是希望以此为现代大语言模型提供补充，进行独立且以语言学为基础的语义分析。通过结合这些方法，FOL-Proof 力求保留神经网络翻译的灵活性，同时引入一种有原则的方法来验证其语义忠实性。

3.3 问题形式化

设 $\mathcal{N}$ 为自然语言句子的集合。每个 $N \in \mathcal{N}$ 都有一组关于文本 [4]的可接受人类语义解释 $\mathcal{M}(N) = \{I_1, \ldots, I_m\} \subseteq \mathcal{L}_{\mathrm{FOL}}$。神经自动形式化器是一个映射 μ: $\mathcal{N} \to \mathcal{L}_{\mathrm{FOL}}$，由语言模型实现，用于生成候选 $\phi_{\mathrm{cand}} = \mu(N)$ [77]。当且仅当 $\phi_{\mathrm{cand}} \equiv I$时，该候选相对于 N 在 $I \in \mathcal{M}(N)$下是语义忠实的，即二者在所有结构 [46]中求值结果均相同。

语义验证问题。给定 N 和 ϕ_{cand}，语义验证问题是确定

$$\exists I \in \mathcal{M}(N) : \phi_{\mathrm{cand}} \equiv I.$$

换言之，目标是确立模型提出的形式化与源句子的至少一个可接受解释在逻辑上等价。

我们将根据候选形式化的语义忠实性对其进行筛选的验证机制称为语义防火墙。在本工作中，我们使用 ERG/MRS 独立推导源句子的语言学许可解释，从而实现这样的防火墙，并检查候选项与这些解释是否逻辑等价。

我们的防火墙返回四种判定结果之一：

$$\mathcal{V}(N，\ \phi_{\mathrm{cand}}) \in \{\text{已认证，不可推导，弃权，无效候选}\}. \tag{1}$$

“已认证”判定表明其等价于至少一个推导出的解释；“不可推导”表示在对适用的支持的搜索空间进行穷举探索后，未找到等价的解释；“弃权”表示验证结果尚无定论；“无效候选”表示该候选项违反了目标语言的要求。

认证的可靠性依赖于这样一个假设：语言学分析和语义适配器推导出的每一种解释，都是原句的可接受解释。

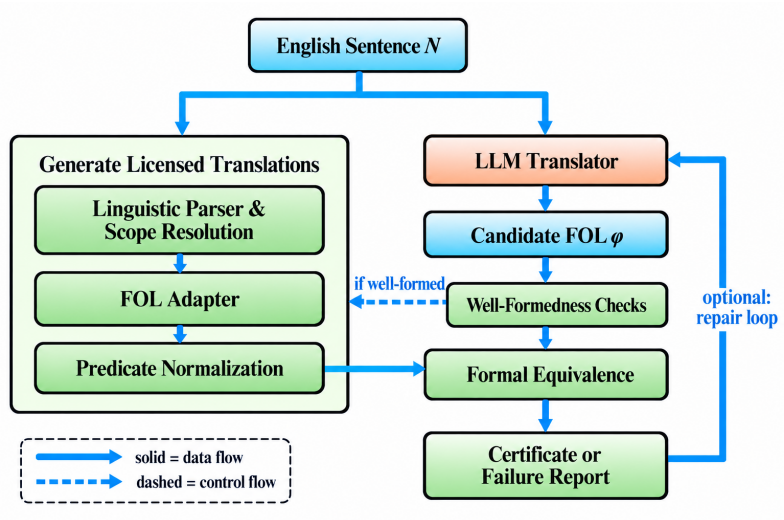


图2. FOL-Proof流程。候选一阶逻辑翻译 φ 首先经过良构性检查，然后系统生成该英语句子 N的获许可的译文。形式等价性检查会生成证书或失败报告，并可选择性地用于LLM引导的修复。实线箭头和虚线箭头分别表示数据流和控制流。

4 FOL-Proof 架构与验证引擎

FOL-Proof 是一个框架，只有当候选FOL公式在模型论上等价于 ERG 在明确且确定性的适配器（负责将 MRS 转换为一阶逻辑）下许可的一种解读时，才会接受该公式。核心流程接收一个自然语言句子 $N \in \mathcal{N}$ 和一个自动形式化的候选公式 $\phi_{\text{cand}} \in \mathcal{L}_{\text{FOL}}$，并执行端到端语言学推导，以便针对获许可的语义解读对 ϕ_{cand} 进行数学认证，或生成结构化可操作诊断信息。图 2 概述了数据流。本节余下部分将解释该流程的各个步骤

4.1 候选项良构性检查

候选项 $\phi_{\text{cand}} \in \mathcal{L}_{\text{FOL}}$ 首先独立于语言学流程进行解析。良构性检查确认该公式属于支持的记法、所有变量均已约束，且谓词用法在结构上保持一致。格式错误的公式被归类为无效候选项。典型错误包括FOL语法错误、自由变量或谓词元数不一致。

4.2 语言学分析与辖域解析

句子会使用英语资源语法提交给语言学解析器。解析器会返回分析结果，其中每个成功的分析结果都是一个 MRS 结构。FOL-Proof 使用辖域解析工具来计算这组

在拓扑上可接受的辖域树：辖域解析会生成一个或多个 ScopedTree 对象。每棵树都是一个有根、连通且无环的图，其边依据 MRS 表示法（第 3.2 节）进行角色标注。对句法分析结果数量或解读数量设置的资源上限，会产生“解析搜索未完成”或“辖域搜索未完成”的弃权状态。

4.3 FOL 适配器

作用域树尚不是一个 FOL公式。其节点和带角色标签的边描述了语义关系；适配器必须将它们解释为谓词、项、量词限制和逻辑联结词。此步骤还确定了句子意义的目标表示。适配器会明确作出这些选择，从而使所得公式能够追溯到语言学分析。

适配器将每个受支持的 ScopedTree 映射为带类型的语义中间表示（中间表示）。该中间表示包含原子公式以及用于否定、合取、析取、蕴含和受限量化的逻辑构造符，并记录解释步骤的溯源信息。适配器将词汇谓述映射为原子公式，同时由运算符节点递归地组合其子节点。对于量词，分别解释其限制条件和主体，然后通过相应的逻辑构造符将二者连接起来。随后，共享的递归编译器将该中间表示映射为无函数符号的一阶逻辑抽象语法树。因此，中间表示将作用域树的解释与公式构造分离开来。其完整语法和编译规则见附录 B。

在示例“每个学生都看见了一位老师”中，第 3.2 节的两棵简化树将适配为以下两个一阶逻辑公式：

$$\forall x\big(\mathrm{student}(x) \rightarrow \exists y\,(\mathrm{teacher}(y) \wedge \mathrm{saw}(x,y))\big).$$

$$\exists y\big(\mathrm{teacher}(y) \wedge \forall x\,(\mathrm{student}(x) \rightarrow \mathrm{saw}(x,y))\big).$$

适配器是部分的：如果树中包含某种未定义确定性解释的结构，则会将该解读记录为不受支持，而不会为其指定公式。因此，派生的一阶逻辑公式表示一种受支持的许可解读，而不受支持的解读不会成为支持或反对候选项的证据。

4.4 谓词规范化

候选项与派生公式之间的差异可能仅体现在谓词拼写或形态形式上（例如，saw、Saw 与 see）。因此，FOL-Proof 会为每种适配后的解读构建一个谓词模式，将规范谓词与 ERG 词汇替代形式及源形式关联起来。形式化地说，它会构建一个对齐映射 $\sigma : \Sigma_{\mathrm{cand}} \rightarrow \Sigma_{\mathrm{der}}$，并将该对齐应用于候选项：

1. **双射映射：** σ 定义了候选项之间的一一对应关系 e
谓词和派生谓词。2. **元数不变性：**对于所有 $P \in \mathcal{P}_{\mathrm{cand}}$，ar(
P) =ar(σ(P))。

3. **词汇落地：**当且仅当 P_{cand} 通过不区分大小写的规范化、下划线剥除，或直接对齐到与该源词元关联的 ERG 词元时，该对齐才是有效的。

关键是，该过程不执行通用同义词解析、谓词分解、谓词融合或论元置换。

4.5 形式等价检查与判定结果

验证引擎检查候选公式 ϕ_{cand} 是否与任一派生解读 $\phi_{\text{der}} \in \Phi_{\text{der}}(N)$ 逻辑等价。确定性结构检查首先测试 $\sigma(\phi_{\text{cand}}) =_\alpha \phi_{\text{der}}$ （在约束变量重命名意义下完全相同）；若是，则无需调用求解器即可判定等价。否则，FOL-Proof 使用 SMT 求解器检查逻辑等价性。求解器断言全称双条件的否定： $\Delta \triangleq \neg \Big(\sigma(\phi_{\text{cand}}) \leftrightarrow \phi_{\text{der}}\Big)$。可能的结果如下：

– **不可满足**：不存在任何赋值或模型结构能够满足 Δ；因此，$\sigma(\phi_{\text{cand}}) \equiv \phi_{\text{der}}$ [47]。在这种情况下，FOL-Proof 返回“已认证”，因为我们发现某个 $\phi_{\text{der}} \in \Phi_{\text{der}}(N)$ 与 σ 等价。– **可满足**：存在满足条件的模型 $\mathcal{M} \models \Delta$，作为明确的反模型，证明二者不等价。候选项将针对该解读被拒绝。– **未知/超时**：求解器无法在资源上限内判定该公式。比较结果为无法确定的解读。

如果搜索已穷尽解析空间和辖域空间，并明确证明候选项与 N 的所有受支持语义解读都不等价，则 FOL-Proof 会返回不可推导。对于不受支持的 ERG 分析，系统会跳过而非因此弃权，所以不可推导是相对于可适配片段而言的。如果 $\sigma(\phi_{\text{cand}})$ 既未获认证也未被拒绝，FOL-Proof 会返回弃权：这可能是因为存在不支持的语义、解析或辖域搜索不完整，或者某个未以其他方式匹配的解读得到了未知的求解器结果。

4.6 证明证书与失败报告

成功运行会生成一份证书，作为可供独立复核的审计记录。证书会记录句子 N、规范化候选公式 $\sigma(\phi_{\text{cand}})$、提供支持的 ERG 分析、选定的辖域解读、适配器规则和配置假设、名称映射、派生公式以及等价性结果。

其他所有结果都会生成一份结构化失败报告，其顶层状态为以下之一

– 确定的候选项失败（无效候选项，无法推导）、– 弃答_（无解析、不支持的语义、解析搜索未完成、辖域搜索未完成、等价性未知）、– 运行失败。

此外，报告还记录候选项诊断信息、分析和调整后解读的数量、不支持分析的原因、规范化诊断信息以及求解器结果。

对于不可推导结果，FOL-Proof 可以选择针对许可解读计算诊断。这些报告旨在向用户或自动修复程序提供可操作的反馈，识别候选公式中潜在的语义差异，并指导其修订为符合许可解读的形式。诊断信息包括结构距离（公式树 GTED [40], 名称掩码拓扑、谓词和常量清单距离）、相近解读的有界帕累托前沿，以及相对于解读的诊断（量词、否定、论元顺序、蕴含方向、常量、谓词或缺失/额外文字差异）。四个披露级别（关闭、组件、解读、完整）控制下游可披露的内容；详情见附录 B。

FOL-Proof 还提供可选的验证器引导的修复循环，将反馈提供给大语言模型自动形式化器。给定一个句子 s、一个初始候选项 c_0 和修复预算 B，该循环会验证每个 c_i；一旦得到已认证结果，运行即告结束。仅当出现两种决定性的失败状态——无效候选项和不可推导——时，才会发出修复请求。弃权会立即以弃权状态结束运行，因为这表示现有证据存在局限，而不是候选项有误；后端或提供商故障则以操作错误结束。

5 工具实现

FOL-Proof 使用 Python 3.11 实现。语言学分析使用 ACE 0.9.34 [52]和 2018 年英语资源语法 [15]；辖域解析通过 JPype 桥接使用 Utool 3.4 [29]；MRS 使用 PyDelphin 1.11 [17]进行处理；等价性通过 Z3 5.0.0 [47], 判定，两个公式均转换为基于共享未解释排序 Entity 的 SMT-LIB2。唯一设置的求解器选项是显式超时（默认为 3000 毫秒）。

特定领域的适配器配置文件。 在初始实验中，我们发现，不同领域的基准测试在谓词和实体的命名方面采用了不同约定，有时还会以不同方式形式化受控语言模式。例如，RuleTaker [8] 将“这只猫”表示为常量 Cat，而 PrOntoQA [63] 则使用 Cat(x) 作为类谓词。因此，我们使用大语言模型为每个领域构建一个适配器配置文件 。根据基准测试的形式化约定和相关的 ERG 分析，该配置文件规定了用于从作用域树构建语义中间表示的目标词汇、记法和解释规则。我们在受控示例和基准测试案例上测试了所得规则，并将其保留为明确且确定性的配置。配置文件的构建在每个应用领域中只执行一次：选定配置用于参数化“FOL 适配器”步骤，而辖域解析和递归的从中间表示到 FOL 的编译器保持共享。特定于配置文件的解释步骤会记录在证书中。本文使用的配置文件总结于附录 C。

6 实验评估

6.1 实验设置与基准测试 arks

执行环境。 实验在一台配备 128 核 AMD EPYC 处理器的服务器上运行（GPU 辅助批次使用 NVIDIA RTX A5000），采用第 5 节中的配置；硬件和内核详情见附录 D。默认情况下，FOL-Proof 对每个句子最多考虑 20 个 ERG 解析和 120 种范围消解读法。辖域解析使用 Utool 的 all 策略，在这些限制范围内枚举所有可接受的辖域配置。辖域枚举的时间限制为 30 秒，每次 Z3 等价性检查的时间限制为 3,000 毫秒。超过这些限制时，我们会使用更高的限制重新评估弃权项目。

评估基准。 我们在三个具有参考一阶逻辑翻译的成熟演绎推理基准测试上评估 FOL-Proof：完整的官方 $ProofWriter$ [69] V2020.12.3封闭世界深度5测试划分（20,058个问题）、官方$RuleTaker$ [8] V2020.2.5.0 ProbLog深度5测试划分（19,021个问题；发布的逻辑形式仅作为评估标注，不会提供给翻译、验证、判定、往返、自然语言推理或选择提示；FOL-Proof使用软推理器配置），以及重新生成的 $PrOn\text{-}toQA$ [63] v1配置网格（9,000个问题）。完整的生成细节和句子数量见附录D。

LongFaithCheck：自定义对抗基准。考虑到经典基准测试对基线大语言模型来说可能较为容易，并受到句子长度会给大语言模型翻译 [11, 73] 和推理 [79], 带来挑战这一证据的启发，我们构建了 LongFaithCheck：包含2,000个英语—一阶逻辑对，旨在使每个条目包含7—10个谓词。其中一半候选公式是正确的；每个错误公式都通过一次受控的语义变异，与某个正确公式存在差异。该基准结合了随机划分和按构造分层的划分；其生成过程和标签检查见附录D。

候选翻译。 每个独特的句子均由通义千问Qwen2.5-7B-Instruct [78], Mistral-7B-Instruct-v0.3 [24,45], 和 Phi-3.5-mini-instruct [1, 44]生成一条确定性贪心翻译。选择这些模型是因为它们计算成本可承受，且属于不同的模型系列。我们的首要目标是在相同的候选翻译上比较验证方法，而非评估前沿模型的翻译能力。大语言模型实验中使用的提示见附录 E。

评估参考。虽然 FOL-Proof 在推理时严格不使用参考公式，但要通过实证评估其准确率，就必须将其判定结果与基准标注参考进行比较。对于每个英语—候选项配对，我们首先使用 α-等价性或 Z3，将候选项与基准测试提供的标准一阶逻辑公式进行比较。许可归一化参考 还会在一种严格不一致的情况下额外标记为正例：当受限的谓词名称归一化使候选项在单独的 α/Z3 检查下与该标准公式等价时。

评估总体。对于 ProofWriter、RuleTaker 和 PrOntoQA，验证指标对每个不同的英文项目–候选项对只计数一次，即使多个模型生成了相同的候选项，以避免不必要的重复。LongFaithCheck 无需这样处理，因为其初始总体规模小得多。总体计数和候选项去重的详细信息见附录D。

候选项判定基线。 我们将 FOL-Proof 与直接判定和往返验证基线进行比较。在直接判定基线中，通义千问2.5-7B-Instruct、Mistral-7B-Instruct-v0.3 和 Phi-3.5-mini-instruct 只能看到英语句子和候选公式，并返回 ACCEPT 或 REJECT；我们报告每个模型的结果以及它们的异构多数投票结果。往返基线改编自近期工作：VeriTrans 风格的词频-逆文档频率相似度 [39],Monty 风格的子句覆盖率 [36], 一个形式化循环（一阶逻辑 $\rightarrow$ 英语 $\rightarrow$ 一阶逻辑，由 α-等价性或 Z3 进行检查） [27],双向 BART-MNLI 蕴含 [32],以及一个组合的形式化+自然语言推理门控 [2]。每个门控均由三个独立模型运行，并作为多数诊断信息 [64]。这些方法是针对我们固定的自然语言–FOL 候选项进行的改编，并非完全复现；阈值和提示可见性细节见附录D。

6.2 语义判别与变异检测

确认金标准译文。 我们首先在 ProofWriter、RuleTaker、PrOntoQA 和 LongFaithCheck 的已认证译文上运行 FOL-Proof。我们发现，FOL-Proof 正确接受了 RuleTaker、PrOntoQA 和 LongFaithCheck 中 100% 的候选项。对于 ProofWriter，FOL-Proof 正确接受了 6,262 个案例（77.80%），其余案例则因语义不受支持且解析搜索不完整而弃权。

受控语义变异。 我们测试验证器能否拒绝仅与正确译文存在一处受控语义编辑差异的公式（关系参数反转、添加或删除否定、蕴含反转、前件删除、量词互换，以及常量或谓词替换）。在 ProofWriter 上，我们从 4,297 个经认证的正确母项生成了 21,584 个此类变异，共计 25,881 个实例。在 PrOntoQA 上，我们从 9,494 个经认证的正确母项生成了 27,255 个此类变异，共计 36,749 个实例。在两个数据集中，Z3 均独立确认每个变异都与其源公式不等价。FOL-Proof 在两个数据集上的准确率和平衡准确率均达到 100%（表1）。相比之下，基线表现落后：在 ProofWriter 上，表现最好的基线是通义千问直接评判器，准确率为 94.90%，平衡准确率为 89.0%；它拒绝了 97.8% 的变异，但仅接受了 80.1% 的正确母项。在通义千问往返门控中，表现最佳的是 Monty 子句覆盖率，准确率为 80.81%。在 PrOntoQA 上，最强的基线是通义千问 Monty 子句覆盖率，准确率为 97.58%，平衡准确率为 97.60%；通义千问直接评判器的准确率为 97.45%，平衡准确率为 96.53%。附录表8列出了部分配置的其他结果。

表1. 受控单编辑变异上的判别（所选基线）。召回率（接受召回率）是被接受的正确译文所占比例；特异度（拒绝特异度）是被拒绝的突变公式所占比例。LongFaithCheck采用类别均衡，因此其平衡准确率等于准确率。形式化往返在ProofWriter上使用通义千问，在PrOntoQA和LongFaithCheck上使用三模型多数。

	ProofWriter（25,881）			PrOntoQA（36,749）			LongFaithCheck（2,000）		
方法	Rec.	特异度	Bal.	Rec.	特异度	Bal.	Rec.	特异度	Bal.
FOL-Proof	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
直接评判（通义千问）	80.1	97.8	89.0	94.6	98.4	96.5	98.8	49.9	74.4
直接评判（多数表决）	99.4	61.1	80.2	97.1	64.9	81.0	99.9	26.8	63.4
形式往返等价	90.3	22.0	56.2	94.2	16.0	55.1	0.9	98.8	49.9
Monty 子句覆盖率（通义千问）	53.0	86.3	69.7	97.7	97.6	97.6	59.1	68.4	63.8

LongFaithCheck。 表1报告了全部 2,000 对样本的汇总结果。FOL-Proof 判定了所有案例，并且在每个案例中都与基准测试标签一致：它认证了全部 1,000 个正确译文，并明确拒绝了全部 1,000 个变异译文，没有弃答。学习型基线的表现则弱得多，没有任何基线的准确率超过 74.4%。表现最佳的直接评判器（通义千问）几乎完全依靠接受偏差达到这一成绩：它接受了 98.8% 的正确样本对，但仅拒绝了 49.9% 的变异样本对。往返门控则呈现相反的失效模式：大多数门控倾向于保守地拒绝，因此只能找回少数正确译文。因此，直接评判器与往返门控的错误方向互为补充，而非有效区分忠实与不忠实的译文。除了评估 FOL-Proof 捕捉错误译文的能力外，这项实验还表明，与其他基准测试相比，LongFaithCheck 对大语言模型评判器更具挑战性。

6.3 大语言模型翻译的有效性

接下来，我们评估这些竞争性验证方法能否判定语言模型实际生成的译文是否正确，而不仅仅是判定已知正确公式的受控变异。对于 LongFaithCheck，在 3,000 条大语言模型翻译中，有 1,288 条格式错误，1,712 条可解析。相对于表2中使用的许可归一化参考，有 692 个输出（23.1%）为正例，因此，一个拒绝所有内容的验证器得分为 76.9%。因此，我们重点关注平衡准确率和错误数量。

表2比较了各验证器对大语言模型生成的译文的表现。对于 LongFaithCheck 的 3,000 条输出，FOL-Proof 与许可归一化参考的一致率为 99.7%；在 PrOntoQA 上为 99.7%，在 RuleTaker 上为 99.9%，在 ProofWriter 上为 100%。即使针对每个模型系列和每个基准测试分别选出表现最佳的单个模型，学习型基线的平衡准确率也远不及 FOL-Proof：其中最强者在 LongFaithCheck 上达到 63.9%，在 PrOntoQA 上达到 73.4%，在 RuleTaker 上达到 81.9%，在 ProofWriter 上达到 86.0%。各种方法的失效模式仍各不相同：直接评判几乎接受所有译文（在 LongFaithCheck 上，Mistral 和 Phi 评判器分别接受了 97.8% 和 96.2% 的正确译文，却仅拒绝了 4.8% 和 10.5% 的错误译文）；而更严格的往返门控几乎拒绝所有译文，表现仅略胜于拒绝所有译文的验证器。详细结果见附录 F：LongFaithCheck 各模型的

表2。 根据许可归一化参考验证大语言模型生成的翻译（选定基线）。单元格显示准确率 / 平衡准确率（ % ）。每个基线行显示其在该基准测试上按平衡准确率计表现最佳的单个模型（Q = 通义千问，P = Phi，M = Mistral）。在准确率中，弃答计为错误；平衡准确率仅在结论明确的决策上，对接受召回率和拒绝特异性取平均。

方法	LongFaith。	PrOntoQA	RuleTaker	ProofWriter
FOL-Proof	**99.7 / 99.8**	**99.7 / 99.4**	**99.9 / 100.0**	**100.0 / 100.0**
直接评判	30.3 / 53.4（P）	76.0 / 73.4（Q）	60.6 / 61.0（Q）	69.7 / 65.1（Q）
形式往返等价	78.3 / 58.3 (Q)	82.4 / 72.2 (Q)	65.9 / 70.4 (M)	83.9 / 86.0 (Q)
Monty 子句覆盖率	70.2 / 63.9（Q）	83.7 / 71.0（Q）	82.2 / 81.9（P）	83.0 / 83.3 (P)

结果见表 9，完整的跨基准测试配置见表 10。另一项结果是，LongFaithCheck 对这些大语言模型来说更难翻译（生成的译文中仅 23.1% 与许可归一化参考一致，而 ProofWriter、RuleTaker 和 PrOntoQA 上的比例为 66.1%–73.6%），这与其设计目的相符。

对剩余误接受的审计。 通过分析 FOL-Proof 准确率不完美的原因，我们发现了一项关键事实。在 RuleTaker 上，FOL-Proof 对 11 个解析不完整的条目弃权。在 LongFaithCheck 和 PrOntoQA 上，所有与金标准证书不一致的情况都是误接受（分别为 10 个和 42 个）。剩余的不一致具有高度结构化的特征：在 LongFaithCheck 上，十个误接受可归为两种模板。第一种模板中，形式为“Some A R every$B$”的句子依据逆辖域解读 $\forall y(B(y) \rightarrow \exists x(A(x) \wedge R(x,y)))$ 获得认证，而非采用表层辖域参考解读 $\exists x(A(x) \wedge \forall y(B(y) \rightarrow R(x,y)))$。这是双重量化英语句子中广为人知的歧义：实验研究认为，诸如“A girl saw every boy”之类的例子既可以有表层解读 $\exists > \forall$，也可以有逆辖域解读 $\forall > \exists$ [26, 76]。因此，我们认为这五处不一致属于基准参考答案未涵盖的合理替代解读。

剩余五个 LongFaithCheck 案例涉及存在量化与否定之间一种较不常见、但仍由系统性方法生成的相互作用。形式为“Some A is/does not P”的句子被许可为 $\neg\exists x(A(x) \wedge P(x))$，等价地写作 $\forall x(A(x) \rightarrow \neg P(x))$，而非参考形式 $\exists x(A(x) \wedge \neg P(x))$。这并非对公式的随意篡改：它对应于语言学分析所生成的一种连贯辖域排序。然而，心理语言学和语义学研究表明，参考基准测试中的解读通常是说话者赋予句子 [12, 35, 67] 的解释。因此，这五个案例展现了语法辖域生成与人类解释偏好之间的界限：FOL-Proof 认证了一种组合上连贯的解读，但说话者在基准测试语境中通常不会采用这种解读。类似的现象可以解释 PrOntoQA 上全部 42 个误接受案例，见附录 F.4。如果将这些案例视为真正的歧义，FOL-Proof 在两个基准测试上的准确率都会升至 100%。

表3。 使用确定性推理器对翻译后的前提和查询进行下游推理。在每种策略下，仅当查询和每个前提都成功编译时，才会回答问题；否则该策略将弃答。产出率 = 正确 / 所有问题（弃答计为错误）；在已回答问题上的选择准确率 = 。原始行显示了每个基准测试中按产出率计表现最佳的单一模型（ProofWriter 和 RuleTaker 使用 Phi，PrOntoQA 使用 Mistral），该选择基于评估数据作出。“首个可编译项”选取第一个可执行候选项，不进行验证。“已认证 +回退”优先选择已认证候选项，否则使用原始候选项（诊断信息）。ProofWriter 行是对保留的问题级结果的重放。覆盖率和配对比较见附录 F.5。

	ProofWriter (N=20,058)		RuleTaker (N=19,021)		PrOntoQA (N=9,000)	
选择策略	产出率	Sel.	产出率	Sel.	产出率	Sel.
最佳原始单模型	40.7	87.2	62.7	88.9	52.7	98.8
首个可编译结果	76.5	85.2	85.3	88.4	51.1	98.6
证书 +原始回退结果	79.0	88.0	92.8	96.1	87.4	100.0
FOL-Proof 严格门控	25.5	**100.0**	70.2	**100.0**	56.7	**100.0**

6.4 证书引导选择对下游推理的影响

为了评估语义验证对下游推理的影响，我们将 FOL-Proof 用作基于大语言模型的翻译器与确定性符号推理器之间的关卡。每个前提和查询都会单独验证；只有当某个推理问题的所有译文均获认证时，才会处理该问题。随后，公式会传递给霍恩规则推理组件 [41]，该组件在受支持的逻辑片段上通过前向链推导结论。我们在 ProofWriter、PrOntoQA 和 RuleTaker 基准测试上评估了下游演绎。表3比较了确定性推理器回答问题时采用的筛选策略。所有策略都遵循相同的整题可用性规则：只有当查询以及该策略下编译的每个前提都可用时，才会回答问题。严格策略还要求它们全部通过认证。在三个基准测试中，该策略所回答的所有问题均正确。代价则是覆盖率：任何一个未获认证的前提都会阻止处理整个问题，因此严格策略的产出率分别为 25.5%、70.2% 和 56.7%。总之，将 FOL-Proof 用作译文的关卡时，最终答案的准确率为 *100.0%*。

6.5 修复循环

我们在500条因无法推导而被拒绝的 LongFaithCheck 译文上评估了验证器引导的修复，每个案例最多允许进行三次修复尝试（表4）。盲目重新生成仅在4.8%的案例中取得成功，而提供 FOL-Proof 的紧凑型诊断信息则将认证率提高到了22.8%。这些结果表明，验证器得出的诊断信息可以引导语义修正，但其有效性仍然有限。

通过展示许可解读，认证率显著提升至 62.6%。这种条件可理解为语义投影：

表4。 对500个被判定为不可推导的候选项进行修复的结果（每个案例最多尝试三次）。勾选标记表示实质性反馈字段：Diag. = 紧凑型诊断信息；Str. = 与所选许可解读相关的结构差异及距离；Form. = 完整的许可解读公式；CM = 可用的反模型。components 和 full no reading formulas 提示会暴露相同的实质性反馈字段，但保留不同的模式和提示版本标签。

	显示的反馈				结果	
Arm	Diag.	Str.	Form.	CM	Cert.	比率（%）
重试					24	4.8
仅诊断信息	✓				114	22.8
组件	✓	✓			111	22.2
完整版本（移除公式/CM）	✓	✓			116	23.2
解读	✓	✓	✓		313	**62.6**
Full	✓	✓	✓	✓	308	61.6

FOL-Proof 并非提供任意参考，而是根据结构距离度量，找出与模型原始候选项最接近的许可解释中的帕累托最优前沿，为语义对齐提供目标。由于这些解读是从源句子独立推导得出的，因此不需要基准测试的真实公式。尽管这种设置无法单独评估大语言模型在没有明确语义目标时纠正错误的能力，但它展示了将确定性语义分析与大语言模型引导的修订相结合的潜力。额外反馈（包括求解器反模型）并未带来进一步提升（61.6%）。

7 结论与未来工作

我们提出了 FOL-Proof，这是一种语义防火墙，用于根据独立推导出的语言分析验证大语言模型生成的一阶逻辑译文。FOL-Proof 并非让另一个语言模型来评判译文，而是从源句子的 ERG/MRS 分析中推导候选解读，将有依据的解读确定性地编译为一阶逻辑，并且仅在能够确立模型论等价性时才认证候选项；证据不完整时则弃权，而非拒绝。

实证表明，在考虑辖域歧义后，FOL-Proof 在受控语义变异和经过金标准认证的译文上实现了近乎完美的选择性准确率。在 ProofWriter、RuleTaker、PrOntoQA 以及对抗性 LongFaithCheck 基准测试中的大语言模型生成的翻译上，它与经过许可解读规范化的参考答案达成了近乎 100% 的一致，并且优于直接大语言模型评判器和往返基线；后两者互补的失效模式（过度接受与过度拒绝）造成了大量残余错误。当使用证书为下游演绎选择前提和查询时，每个已回答问题的选择性准确率均达到 100%。最后，修复研究表明，独立生成的验证器反馈可将认证率从盲目重新生成时的 4.8% 提高到 22.8%；而当所选许可解读的完整公式也一并提供时，在三次修复尝试内，认证率可提高到 62.6%。

从盲目重新生成时的 4.8% 提高到 22.8%；而当所选许可解读的完整公式也一并提供时，在三次修复尝试内，认证率可提高到 62.6%。

仍然存在若干局限，这些局限为未来的研究指明了方向。首先，当前的适配器仅涵盖英语的一个受控片段。我们计划研究如何将相关形式化体系中的语义组合规则——特别是组合范畴语法（CCG） [66]及其组合语义学——以算法方式迁移到现有的 MRS 到 FOL 流水线中。其次，辖域枚举仍然具有组合爆炸性，较长的句子可能生成数千个句法分析结果以及数量呈阶乘增长的辖域树，因此在严格的资源上限下进行穷举搜索并不现实。我们将通过以下方式开发用于剪枝解析森林和加速辖域搜索的算法改进：采用增量式等价性检查，在发现反模型后立即终止对某一解读的处理；并在相关分析之间缓存中间语义中间表示片段。第三，尽管修复反馈已经颇为有效，但在预算耗尽后，仍有相当一部分案例无法获得认证。更丰富且更具可操作性的反馈应能进一步提高成功率。

致谢。 本研究部分由欧盟资助（Ro-bustifAI 项目，编号 101212818）。然而，文中表达的观点和意见仅代表作者本人，并不一定反映欧盟或欧洲健康与数字执行局（HADEA）的观点。欧盟和资助机构均不对此承担责任。

利益披露。 作者声明，不存在与本文内容相关的利益冲突。

数据可用性声明。 本文所依据的源代码、实验配置、提示词和逐实例结果已汇编为一份可复现实验材料。该材料将在 TACAS 研究论文通知后提交评审。最终会议论文集版本将注明存档材料及其永久数字对象标识符。

参考文献

1. Abdin, M., Aneja, J., Awadalla, H., Awadallah, A., Awan, A.A. 等：Phi-3技术报告：一种可在手机上本地运行的高性能语言模型（2024），https://arxiv.org/abs/2404 14219 2 。. Amrollahi, D., Lopez, J., Barrett, C.：通过往返验证与修复实现忠实的自动形式化。载于：计算语言学协会论文集：EMNLP 2026。第9836–9852页。计算语言学协会，布达佩斯，匈牙利（2026）。 https://doi.org/10.18653/2026.findings-emnlp.469, https: // aclanthology.org/2026.findings-emnlp.469/ 3. Bansal, H., Kurniawan, K., Frermann, L.：神经符号推理器在一阶逻辑问题上的鲁棒性。载于：第23届澳大拉西亚语言技术协会年度研讨会论文集。第1–12页。计算语言学协会，悉尼，澳大利亚（2025），https://aclanthology.org/ 2025.alta-main.1/

4. Brunello, A., Curaba, C., Geatti, L., Mignani, M., Montanari, A., Saccomanno, N.：修复FOLIO和MALLS：经验证的注释与大语言模型辅助框架，聚焦于人工重新标注。载于：2026年自然语言处理实证方法会议论文集。第13195–13220页。计算语言协会，布达佩斯，匈牙利（2026）。https://doi.org/10.18653/2026.emnlp-main.576, https://aclanthology.org/2026.emnlp-main.576/ 5. Brunello, A., Geatti, L., Mignani, M., Montanari, A., Saccomanno, N.：大语言模型真的难以进行自然语言到一阶逻辑的转换吗？通过一种新颖的基准测试策略揭示其优势。AAAI人工智能会议论文集 **40**40258 6 36 30094 30103 2026 10 1609 40 36 ( )，– ( )。https://doi.org/ . /aaai.v i . . Cai, C., Zhao, X., Liu, H., Jiang, Z., Zhang, T., Wu, Z., Hwang, J.N., Li, L.：演绎推理与归纳推理在大语言模型中的作用。载于：第63届计算语言学协会年会论文集（第1卷：长篇论文）。第16780–16790页（2025）。https://doi.org/10.18653/v1/2025. acl-long.820 7. Carlson, G.N.：英语裸复数的统一分析。《语言学与哲学》100353456 8 3 413 457 1977 10 1007 ( )，– ( )。https://doi.org/ . /BF 。Clark, P., Tafjord, O., Richardson, K.：Transformer作为语言上的软推理器。载于：第29届国际人工智能联合会议（IJCAI）论文集。第3882 537 9 3890页（2020）。https://doi.org/ . /ijcai. / 。Copestake, A., Flickinger, D., Pollard, C., Sag, I.A.：最小递归语义：导论。《语言与计算研究》**3**(2–3)，281–332（2005）。https://doi.org/10 9 10 1007 11168 006 6327 . /s - - - 。Creswell, A., Shanahan, M., Higgins, I.：选择—推理：利用大语言模型进行可解释逻辑推理。载于：第十一届国际学习表征会议（ICLR）（2023）11. Deveci, ˙I.E.：通过自然语言到一阶逻辑的转换实现组合泛化的诊断框架。《逻辑、语言与信息杂志》（2026）。https: //doi.org/10 0 12 1007 10849 026 09480 . /s - - - 。Dobrovie-Sorin, C., Ihsane, T., Gerards, D.P., Foppolo, F.：论some-不定表达式的拯救。《语言学杂志》**62**(2)，329–356（2026）。https://doi.org/ 10 2025 13 1017 0022226725100753, 17 . /S 在线发表于7月。Feng, Y., Weir, N., Bostrom, K., Bayless, S., Cassel, D., Chaudhary, S., Kiesl- Reiter, B., Rangwala, H.：VeriCoT：通过逻辑一致性检查进行神经符号思维链验证。载于：第十四届国际学习表征会议（ICLR）（2026）14. Flickinger, D.：利用类型构建更高效的语法。《自然语言工程》**6**(1)，15–28（2000）。https://doi.org/10.1017/S1351324900002370 15. Flickinger, D., Bender, E.M., Oepen, S.：迈向组合语义百科全书：记录英语资源语法的接口。载于：第九届国际语言资源与评测会议（LREC'14）论文集。第875–881页。欧洲语言资源协会（ELRA），雷克雅未克，冰岛（2014）16. Garcez, A.d., Gori, M., Lamb, L.C., Serafini, L., Spranger, M., Tran, S.N.：神经符号计算：系统整合机器学习与推理的有效方法论。《应用逻辑杂志》**6**(4)，611–632（2019）17. Goodman, M.W.：用于深层语言资源的Python库。载于：2019年太平洋邻里联盟年度会议及联合会议（PNC）。第1–7页。IEEE，新加坡（2019）

18. Guo, D. 等：DeepSeek-R1 通过强化学习激励大语言模型中的推理。《自然》**645**, 633–638（2025）。https://doi.org/10.1038/ s41586-025-09422-z 19. Han, S.、Schoelkopf, H.、Zhao, Y.、Qi, Z.、Riddell, M.、Zhou, W.、Coady, J.、Peng, D.、Qiao, Y.、Benson, L.、Sun, L.、Wardle-Solano, A.、Szab´o, H.、Zubova, E.、Burtell, M.、Fan, J.、Liu, Y.、Wong, B.、Sailor, M.、Ni, A.、Nan, L.、Kasai, J.、Yu, T.、Zhang, R.、Fabbri, A.、Kryscinski, W.M.、Yavuz, S.、Liu, Y.、Lin, X.V.、Joty, S.、Zhou, Y.、Xiong, C.、Ying, R.、Cohan, A.、Radev, D.：FOLIO：使用一阶逻辑进行自然语言推理。载于：2024 年自然语言处理实证方法会议论文集。第 22017–22031 页。计算语言学协会（2024）。https://doi.org/10.18653/v1/2024.emnlp-main.1229 20. Holtermann, C.、R¨ottger, P.、Dill, T.、Lauscher, A.：使用 MultiQ 评估大语言模型的基础多语言能力。载于：计算语言学协会论文集：ACL 2024。第 4476–4494 页（2024）。https://doi.org/10.18653/v1/2024.findings-acl.265 21. Hong, R.、Zhang, H.、Pang, X.、Yu, D.、Zhang, C.：深入考察大语言模型在逻辑推理中的自我验证能力。载于：2024 年计算语言学协会北美分会会议论文集：人类语言技术（第 1 卷：长篇论文）。第 900–925 页。计算语言学协会，墨西哥城，墨西哥（2024）。https://doi.org/10.18653/v1/2024.naacl-long.52 22. Jacovi, A.、Bitton, Y.、Bohnet, B.、Herzig, J.、Honovich, O.、Tseng, M.、Collins, M.、Aharoni, R.、Geva, M.：思维链的强度取决于其最薄弱的环节：推理链验证器的基准测试。载于：计算语言学协会第 62 届年会论文集（第 1 卷：长篇论文）。第 4615–4634 页。计算语言学协会（2024）。https://doi. org/10.18653/v1/2024.acl-long.254 23. Jana, P.、Kale, K.、Tanriverdi, A.E.、Song, C.、Vishwanath, S.、Ganesh, V.：Proof-Bridge：通过联合嵌入在 Lean 中对自然语言证明进行自动形式化。载于：第十四届国际学习表征会议（ICLR）（2026），https://proceedings.iclr.cc/paper_files/paper/2026/hash/16f5b8d916f13534d11c50adf60914e4-Abstract-Conference.html 24. Jiang, A.Q.、Sablayrolles, A.、Mensch, A.、Bamford, C.、Chaplot, D.S.、de las Casas, D.、Bressand, F.、Lengyel, G.、Lample, G.、Saulnier, L.、Lavaud, L.R.、Lachaux, M.A.、Stock, P.、Le Scao, T.、Lavril, T.、Wang, T.、Lacroix, T.、El Sayed, W.：Mistral 7B。技术报告（2023），https://arxiv.org/abs/2310.06825 25. Jin, Z.、Lalwani, A.、Vaidhya, T.、Shen, X.、Ding, Y.、Lyu, Z.、Sachan, M.、Mihalcea, R.、Sch¨olkopf, B.：逻辑谬误检测。载于：计算语言学协会论文集：EMNLP 2022。第 7180–7198 页。计算语言学协会（2022）。https://doi.org/10.18653/v1/2022.findings-emnlp.532 26. Johnson, K.：量词的作用范围有多大？载于：Martin, R.、Michaels, D.、Uriagereka, J.（编）：《循序渐进：献给 Howard Lasnik 的最简句法论文集》，第 187–210 页。MIT 出版社，剑桥，马萨诸塞州（2000） 27. Karia, R.、Bramblett, D.、Dobhal, D.、Verma, P.、Srivastava, S.：∀uto∃val：对大语言模型在形式综合和解释任务中的自主评估。技术报告（2024），https://arxiv.org/abs/2403.18327 28. Kim, K.、Poiroux, A.、Bosselut, A.：大语言模型会操纵形式化吗？评估逻辑推理中的忠实性。技术报告（2026），https://arxiv.org/abs/ 2604.19459

29. Koller, A.、Thater, S.：高效求解和探索辖域歧义。载于：ACL 互动海报与演示环节论文集。第 9–12 页。计算语言学协会，安娜堡，密歇根州（2005）。https: // doi.org/10.3115/1225753.1225756 30. Kov´acs, L.、Voronkov, A.：一阶定理证明与 Vampire。载于：计算机辅助验证（CAV）。《计算机科学讲义》，第 8044 卷，第 1–35 页。施普林格（2013）。https://doi.org/10.1007/978-3-642-39799-8_131. Krifka, M.、Pelletier, F.J.、Carlson, G.N.、ter Meulen, A.、Chierchia, G.、Link, G.：类指性：导论。载于：Carlson, G.N.、Pelletier, F.J.（编）：《类指性专著》，第 1–124 页。芝加哥大学出版社，芝加哥（1995） 32. Lewis, M.、Liu, Y.、Goyal, N.、Ghazvininejad, M.、Mohamed, A.、Levy, O.、Stoyanov, V.、Zettlemoyer, L.：BART：面向自然语言生成、翻译和理解的去噪序列到序列预训练。载于：计算语言学协会第 58 届年会论文集。第 7871–7880 页。计算语言学协会（2020）。https://doi.org/10.18653/v1/ 2020.acl-main.703, https://aclanthology.org/2020.acl-main.703/ 33. Li, Q.、Li, J.、Liu, Z.、Cheng, M.、Zhou, Y.、Zeng, Y.、Liu, Q.、Liu, T.：在语言多样化文本的逻辑推理中，大语言模型是否是稳定的形式逻辑转换器？载于：ACM Web 大会 2026 论文集。第 3633–3644 页（2026）。https://doi.org/10.1145/3774904.3792161 34. Lin, B.Y.、Le Bras, R.、Richardson, K.、Sabharwal, A.、Poovendran, R.、Clark, P.、Choi, Y.：Zebra Logic：大语言模型进行逻辑推理的规模扩展极限。载于：第 42 届国际机器学习会议（ICML）论文集。《机器学习研究论文集》，第 267 卷，第 37889–37905 页。PMLR（2025），https://proceedings.mlr.press/v267/lin25i.html 35. Liu, F.h.：《辖域与特异性》，《当前语言学／今日语言学》，第 16 卷。约翰·本杰明斯出版公司，阿姆斯特丹（1997）。https://doi.org/10.1075/ la.16 36. Liu, H.、Parthasarathy, M.、Murali, A.：自然语言断言的忠实自动形式化。技术报告（2026），https://arxiv.org/abs/2607.13303 37. Liu, Q.、Zheng, X.、Lu, X.、Cao, Q.、Yan, J.：重新思考并改进自动形式化：迈向忠实度量指标与基于依存检索的方法。载于：第十三届国际学习表征会议（ICLR）（2025） 38. Liu, X.、Zhu, T.、Dong, Z.、Liu, Y.、Guo, Q.、Liu, Z.、Chen, Y.、Luo, T.：ASSESS：用于陈述相似度的语义与结构评估框架。载于：第十四届国际学习表征会议（ICLR）（2026），https://openreview.net/forum?id=avwNGWtiHF 39. Liu, X.、Kodakandla, D.、Srivastva, K.、Farooque, M.：VeriTrans：通过确定性神经符号流水线实现的微调大语言模型辅助自然语言到编程语言翻译。技术报告（2026），https://arxiv.org/abs/2604.10341 40. Liu, Y.、Zhu, T.、Liu, X.、Chen, Y.、Liu, Z.、Guo, Q.、Zhang, J.、Bao, K.、Luo, T.：广义树编辑距离（GTED）：陈述自动形式化的忠实评估指标。技术报告（2025），https://arxiv.org/abs/2507.07399 41. Lloyd, J. W.：《逻辑程序设计基础》。施普林格，第 2 版（1987）。https: // doi.org/10.1007/978-3-642-83189-8 42. McCune, W.：Prover9 和 Mace4（2005），https://www.cs.unm.edu/~mccune/ prover9/，自动定理证明器和有限模型查找器；软件开发于 2005–2010 年

43. McGinness, L., Baumgartner, P.：自动定理证明器有助于提升大语言模型推理能力。载于：第25届程序设计、人工智能与推理逻辑会议论文集。EPiC计算系列，第100卷，第51–69页。EasyChair（2024）。https://doi.org/10.29007/2n9m 44. 微软：Phi-3.5-mini-instruct。Hugging Face模型卡（2024），https://huggingface.co/microsoft/Phi-3.5-mini-instruct 45. Mistral AI：Mistral-7B-Instruct-v0.3。Hugging Face模型卡（2024），https: //huggingface.co/mistralai/Mistral-7 3 46 0 B-Instruct-v . . Mohammad, N.I.S., Sheikh, T.：忠实度差距：认证自然语言与形式化数学陈述之间的语义等价性。技术报告（2026 16541 47 2606 3 ），https://arxiv.org/abs/ . . de Moura, L., Bjørner, N.：Z：一种高效的SMT 求解器。载于：系统构建与分析工具及算法（TACAS）。计算机科学讲义，第4963卷，第337–340页。施普林格（2008）。https://doi.org/10.1007/978-3-540-78800-3_2448. de Moura, L., Ullrich, S.：Lean 4定理证明器与编程语言。载于：自动演绎——CADE 28。计算机科学讲义，第12699卷，第 625 635 2021 10 1007 978 3 030 79876 5_37 – 49 页。施普林格（）。https://doi.org/ . / - - - - . Murphy, L., Yang, K., Sun, J., Li, Z., Anandkumar, A., Si, X.：欧几里得几何的自动形式化。载于：第41届国际机器学习会议（ICML）论文集。机器学习研究论文集，第235卷，第36847–36893页。PMLR（2024），https://proceedings.mlr.press/v235/murphy24a.html 50. Olausson, T.X., Gu, A., Lipkin, B., Zhang, C.E., Solar-Lezama, A., Tenen- baum, J.B., Levy, R.：LINC：一种结合语言模型与一阶逻辑证明器进行逻辑推理的神经符号方法。载于：2023年自然语言处理经验方法会议论文集。第5153–5176页。计算语言学协会（2023）。https: //doi.org/ 10 18653 1 2023 313 . /v / .emnlp-main. . Opitz, J.：SMATCH：语义图的标准化扩展评估。载于：计算语言学协会论文集：EACL 2023。第1595–1607页。计算语言学协会，克罗地亚杜布罗夫尼克（2023）。https://doi.org/10 118 52 18653 1 2023 2015 . /v / .findings-eacl. . Packard, W.：ACE：答案约束引擎（），http://sweaglesw.org/ linguistics/ace/，适用于ERG等语法的DELPH-IN解析/生成引擎 53. Pan, L., Albalak, A., Wang, X., Wang, W.Y.：Logic-LM：借助符号求解器赋能大语言模型，实现忠实逻辑推理。载于：计算语言学协会论文集：EMNLP 2023。第3806–3824页。计算语言学协会，新加坡（2023）。https://doi.org/10.18653/v1/ 2023 248 54 .findings-emnlp. . Pei, Y., Du, Y., Jin, X.：FoVer：面向自然语言推理的一阶逻辑验证。计算语言学协会会刊**13**，1340–1359 41 55 2025 10 1162（）。https://doi.org/ . / tacl.a. . Pollard, C., Sag, I.A.：中心语驱动短语结构语法。芝加哥大学出版社，芝加哥（1994）56. Putra, R.R., Basuki, R.S.P., Cheng, Y., Gao, P.：NL2Logic：利用大语言模型将自然语言AST引导的翻译为一阶逻辑。载于：计算语言学协会论文集：EACL 2026。第6035–6051页。计算语言学协会，摩洛哥拉巴特（2026）。https://doi.org/10.18653/v1/2026.findings-eacl.317

57. Qi, C., Ma, R., Li, B., Du, H., Hui, B., Wu, J., Laili, Y., He, C.：大语言模型与符号证明器结合，用于逻辑推理评估。载于：第十三届国际学习表征会议（ICLR）（2025）58. Quansah, P.K., Bonnah, E.：NeuroNL2LTL：一种用于将自然语言翻译为线性时序逻辑的神经符号框架。载于：分布式对象、组件与系统形式化技术（FORTE）。计算机科学讲义，第16589卷，第56–73页。施普林格（2026）。https://doi.org/10.1007/978-3-032-28187-6_459. Rabern, B., Mondorf, P., Plank, B.：LogicSkills：用于大语言模型形式推理的结构化基准。载于：计算语言学协会论文集：EMNLP 2026。第2035–2047页。计算语言学协会，匈牙利布达佩斯（2026）。https://doi.org/10.18653/2026.findings-emnlp.98，https://aclanthology.org/2026.findings-emnlp.98/ 60. Rademaker, A., Lima, G., Fiorini, S.R., da Silva, V.T.：对大语言模型生成的SPARQL查询进行演绎验证。载于：深度学习与关联数据研讨会（DLnLD）@ LREC-COLING 2024论文集。第45–52页。ELRA与ICCL，意大利都灵（2024），https://aclanthology.org/2024.dlnld-1.4/ 61. Raiyan, S.R., Kabir, M., Mahmud, H., Hasan, M.K.：用于数学推理的人工智能：关于语言模型、神经符号系统与经验证的发现的综合综述。技术报告（2026），https://arxiv.org/abs/ 2606.08728 62. Ryu, H., Kim, G., Lee, H.S., Yang, E.：分而译之：面向复杂逻辑推理的组合式一阶逻辑翻译与验证。载于：第十三届国际学习表征会议（ICLR）（2025）63. Saparov, A., He, H.：语言模型是贪婪的推理者：对思维链进行系统形式化分析。载于：第十一届国际学习表征会议（ICLR）（2023）64. Singh, V., Cassel, D., Weir, N., Feng, N., Bayless, S.：VERGE：面向可验证大语言模型推理的形式化精化与引导引擎。技术报告（2026），https: // arxiv.org/abs/2601 20055 65 . . Singh, V., Ganguly, D., Yu, H., Zhou, C., Singh, P., Lee, B., Chaudhary, V., Datta, G.：通过形式化验证保障视觉语言模型的临床推理能力。技术报告（2026 24111 66 2602），https://arxiv.org/abs/ . . Steedman, M., Baldridge, J.：组合范畴语法。载于：Borsley, R.D., B¨orjars, K.（编）《非转换句法：语法的形式化与显式模型》，第181–224页。Wiley-Blackwell（2011）。https://doi.org/10.1002/ 9781444395037.ch5 67. Szabolcsi, A.：肯定极性–否定极性。《自然语言与语言学理论》**22**(2)，409–452（2004）。https://doi.org/10.1023/B:NALA.0000015791. 00288.43 68. Szegedy, C.：迈向自动形式化与通用人工智能的一条有希望的道路。载于：智能计算机数学（CICM）。计算机科学讲义，第12236卷，第3–20页。施普林格（2020）69. Tafjord, O., Dalvi, B., Clark, P.：ProofWriter：生成自然语言中的蕴含、证明与溯因陈述。载于：计算语言学协会论文集：ACL-IJCNLP 2021。第3621–3634页。计算语言学协会（2021）。https://doi.org/10.18653/v1/2021.findings-acl. 317 70. Thatikonda, R.K., Buntine, W., Shareghi, E.：评估一阶逻辑接近度指标的敏感性与对齐。载于：Christodoulopoulos, C., Chakraborty, T., Rose, C.,

Peng, V.（编）《计算语言学协会论文集：EMNLP 2025》。第16775–16785页。计算语言学协会，中国苏州（2025 910 71 10 18653 1 2025）。https://doi.org/ . /v / . findings-emnlp. 。Thatikonda, R.K., Han, J., Buntine, W.L., Shareghi, E.：《利用大语言模型改进自然语言到一阶逻辑的转换的策略：数据生成、增量微调与验证》。技术报告（2024 16461 72 2409），https://arxiv.org/abs/ . 。Tian, J., Li, Y., Chen, W., Xiao, L., He, H., Jin, Y.：《通过LogicNLI诊断一阶逻辑推理能力》。载于：《2021年自然语言处理实证方法会议论文集》。第3738–3747页。计算语言学协会（2021）。https://doi.org/10.18653/v1/2021. emnlp-main.303 73。Vossel, F., Mossakowski, T., Gehrke, B.：《利用微调大语言模型推进自然语言形式化为一阶逻辑》。技术报告（2025），https://arxiv. org/abs/2509 22338 74 。Wang, H., Huang, B., Wan, Y., Zhu, X., Liu, X., Huang, Y., Guo, Z.：《FormalRx：纠正并检查自动形式化中的语义错误》。载于：《第43届国际机器学习会议（ICML）论文集》。机器学习研究论文集，第306卷，第126930–126964页。PMLR（2026），https: // proceedings.mlr.press/v306/wang26cg.html 75。Wei, A., Wu, Y., Wan, Y., Suresh, T., Tan, H., Zhou, Z., Koyejo, S., Wang, K., Aiken, A.：《SATBench：通过从SAT公式自动生成谜题对大语言模型的逻辑推理进行基准测试》。载于：《2025年自然语言处理实证方法会议论文集》。第33832–33849页。计算语言学协会（2025 1716 76 10 18653 1 2025）。https://doi.org/ . /v / .emnlp-main. 。Wu, M.J., Ionin, T.：《语调效应对英语辖域歧义句的影响》。Ilha do Desterro **73**（3），13–36（2020）。https://doi.org/10.5007/2175-8026. 2020 13 77 73 3v n p 。Wu, Y., Jiang, A.Q., Li, W., Rabe, M.N., Staats, C., Jamnik, M., Szegedy, C.：《利用大语言模型进行自动形式化》。载于：《神经信息处理系统进展（NeurIPS）》。第35卷，第32353–32368页（2022）78。Yang, A.等：《Qwen2.5技术报告》（2024），https://arxiv.org/abs/2412. 15115 79。Zhang, M., Peng, Q., Wei, Y., Shen, Y., Tan, K., Wang, Y., Xiang, Z., Ye, J., Yin, Z., Xi, Z., Dou, S., Wang, W., Zhang, Y., Gui, T., Yang, R., Zhang, Q., Huang, X., Chen, A., Pan, M.：《HyperLogic：一个由前向作者编写、具有挑战性的中文逻辑推理基准，答案源自执行过程》。技术报告（2026），https://arxiv.org/abs/2605.19597，修订版本（v2），2026年10月2日；原题为LLMEval-Logic 80。Zhang, T., Ladhak, F., Durmus, E., Liang, P., McKeown, K., Hashimoto, T.B.：《大语言模型新闻摘要基准测试》。计算语言学协会会刊 **12**，39–57（2024）。https://doi.org/10. 1162/tacl_a _00632 81。Zheng, X., Li, N., Luan, X., Wang, K., Shi, L., Sun, M., Wang, H.：《超越正确性：通过多步自动定理证明揭示大语言模型生成推理中的逻辑缺陷》。载于：《第48届IEEE/ACM国际软件工程会议（ICSE）论文集》。第3001–3013页。IEEE（2026）82。Zhong, W., Wang, S., Tang, D., Xu, Z., Guo, D., Chen, Y., Wang, J., Yin, J., Zhou, M., Duan, N.：《文本的分析推理》。载于：《计算语言学协会论文集：NAACL 2022》。第2306–2319页。计算语言学协会（2022）。https://doi.org/10.18653/v1/2022.findings-naacl.177

83. Zhu, W., Liu, H., Dong, Q., Xu, J., Huang, S., Kong, L., Chen, J., Li, L.：《利用大语言模型进行多语言机器翻译：实证结果与分析》。载于：《计算语言学协会论文集：NAACL 2024》。第2765 176 84 2781 2024 10 18653 1 2024–（）。https://doi.org/ . / v / .findings-naacl. 。Zhuo, T.Y., Vu, M.C., Chim, J.等：《BigCodeBench：使用多样化函数调用和复杂指令对代码生成进行基准测试》。载于：《第十三届国际学习表征会议（ICLR）》（2025）

附录

MRS 与辖域解析示例

我们将示例 1 扩展到句子“每个学生都用望远镜看见了一位老师。”。以下简化的 SimpleMRS 表示对应于望远镜用于看见这一事件的解读（动词附加）。

```
[ TOP: h0
  INDEX: e4 [ e SF: prop TENSE: past MOOD: indicative PROG: - PERF: - ]
  RELS: <
    [ _every_q<0:5> LBL: h1 ARG0: x1 RSTR: h2 BODY: h3 ]
    [ _student_n_1<6:13> LBL: h4 ARG0: x1 ]
    [ _a_q<18:19> LBL: h5 ARG0: x2 RSTR: h6 BODY: h7 ]
    [ _teacher_n_1<20:27> LBL: h8 ARG0: x2 ]
    [ _a_q<33:34> LBL: h9 ARG0: x3 RSTR: h10 BODY: h11 ]
    [ _telescope_n_1<35:44> LBL: h12 ARG0: x3 ]
    [ _see_v_1<14:17> LBL: h13 ARG0: e4 ARG1: x1 ARG2: x2 ]
    [ _with_p<28:32> LBL: h13 ARG0: e5 ARG1: e4 ARG2: x3 ]
  >
  HCONS: < h0 qeq h13 h2 qeq h4 h6 qeq h8 h10 qeq h12 >
  ICONS: < > ]
```

MRS 由一组基本谓词（RELS）以及对其可能辖域的约束（HCONS）构成。每个谓词都有一个标签（LBL），称为句柄，用于标识其局部语义位置。共享同一标签的谓词按合取方式解释。论元是表示实体（x_1，x_2，x_3）或事件（e_4，e_5）的变量：ARG0 标识与该谓词关联的变量，而 ARG1、ARG2 等则指定语义角色。例如，see v 1 描述了一个事件 e_4，在该事件中，学生 x_1 看见了教师 x_2。with p 谓词将该事件与望远镜 x_3关联起来，并与其共享标签 h_{13}。数字跨度（例如，<14:17>）标识输入句子中的相应字符。

量词还会引入两个取值为句柄的论元：RSTR，用于指定限制条件（例如，是一名学生）；BODY，用于指定量词辖域内的命题。这些句柄充当空位，其内容尚未完全确定。HCONS 约束限定了这些空位可能的填充方式：例如，h2 qeq h4 要求 every 的限制条件指向学生谓词，中间可以经过其他量词。类似地，约束 h0 qeq h13 将句子的顶部句柄（TOP）锚定到动词谓词，但不固定中间量词的顺序。INDEX 标识主要事件及其语法特征；此处为空的 ICONS 则可包含有关个体变量的附加约束。

辖域解析。辖域解析会在满足这些约束的前提下，为其余空位指定标签。例如，将 $h_0 = h_1$、$h_3 = h_5$、$h_7 = h_9$ 和 $h_{11} = h_{13}$ 设为相应值，即可使学生全称量词位于存在量词教师和望远镜之上：

$$\forall x_1 \ > \ \exists x_2 \ > \ \exists x_3.$$

或者，设定 $h_0 = h_5$、$h_7 = h_1$、$h_3 = h_9$ 和 $h_{11} = h_{13}$，即可得到

$$\exists x_2 \ > \ \forall x_1 \ > \ \exists x_3,$$

其中每个学生都看到了同一位教师。在这两种情况下，限制空位都满足 $h_2 = h_4$、$h_6 = h_8$ 和 $h_{10} = h_{12}$。这是同一个欠规格化 MRS 的两种代表性解析。

最后，另一种名词附着 （教师有一架望远镜）由不同的 ERG/MRS 分析表示。在相应的表示中，带有 p 的项将改为共享教师的标签 h_8，并将 x_2 而不是 e_4 作为其 ARG1。因此，句法附着的不同方案源于不同的分析，而辖域解析则会在每种分析内部产生不同的解读。

B 其他方法细节

语义中间表示。 适配器在遍历已解析的 ScopedTree 时不会构造公式文本。它首先构造一个带类型的语义中间表示（IR），将如何解释 MRS 的决策与 FOL 的机械构造分离开来。核心 IR 可写为

$$\begin{aligned} e ::= &\ \mathsf{Atom}(P, \bar{t}) \mid \mathsf{Not}(e) \mid \mathsf{And}(e, e) \mid \mathsf{Or}(e, e) \\ &\mid \mathsf{Implies}(e, e) \mid \mathsf{Iff}(e, e) \mid \mathsf{Quant}(q, x, r?, e), \\ q ::= &\ \mathsf{Univ} \mid \mathsf{Exist} \mid \mathsf{Negative}. \end{aligned}$$

这里， P 是谓词符号， $\bar{t}$ 是由变量或常量组成的元组， x 是受约束变量，而 r? 要么是一个给出量词限制条件的 IR 表达式，要么缺省。目标语言不含函数。合取和析取在实现中共享一个并列节点；为便于阅读，上文将二者分别展示。

每个中间表示节点还携带溯源信息：解释规则、相关 MRS 基本谓词的标识符，以及任何已声明的假设。此外，适配器还会记录由选定配置所许可的、针对解释选择的类型化扩充步骤，例如引入泛指绑定项、解析受控共指、投影基准常量、省略论域限制，或在基准本体中融合谓词。这些步骤是适配记录的明确组成部分；它们并非由一阶逻辑编译器推断出的规则。

编译为一阶逻辑。令 Compile(e) 表示从语义中间表示到一阶逻辑抽象语法树的递归翻译。其布尔情形为

$$\begin{aligned} \mathsf{Compile}(\mathsf{Atom}(P, \bar{t})) &= P(\bar{t}), \\ \mathsf{Compile}(\mathsf{Not}(e)) &= \neg \mathsf{Compile}(e), \\ \mathsf{Compile}(\mathsf{And}(e_1, e_2)) &= \mathsf{Compile}(e_1) \wedge \mathsf{Compile}(e_2), \\ \mathsf{Compile}(\mathsf{Or}(e_1, e_2)) &= \mathsf{Compile}(e_1) \vee \mathsf{Compile}(e_2), \\ \mathsf{Compile}(\mathsf{Implies}(e_1, e_2)) &= \mathsf{Compile}(e_1) \rightarrow \mathsf{Compile}(e_2), \\ \mathsf{Compile}(\mathsf{Iff}(e_1, e_2)) &= \mathsf{Compile}(e_1) \leftrightarrow \mathsf{Compile}(e_2). \end{aligned}$$

对于显式限制条件 r，量词情形为

$$\begin{aligned}\mathsf{Compile}(\mathsf{Quant}(\mathsf{Univ}, x, r, e)) &= \forall x\,(\mathsf{Compile}(r) \rightarrow \mathsf{Compile}(e)),\\ \mathsf{Compile}(\mathsf{Quant}(\mathsf{Exist}, x, r, e)) &= \exists x\,(\mathsf{Compile}(r) \wedge \mathsf{Compile}(e)),\\ \mathsf{Compile}(\mathsf{Quant}(\mathsf{Negative}, x, r, e)) &= \forall x\,(\mathsf{Compile}(r) \rightarrow \neg\mathsf{Compile}(e)).\end{aligned}$$

当限制条件缺失时，这些分别变为 $\forall x\mathsf{Compile}(e)$、$\exists x\mathsf{Compile}(e)$ 和 $\forall x\,\neg\mathsf{Compile}(e)$。因此，带有限制条件 r 的否定量词表示：不存在满足 r 的 x，使其满足 e。

适配器根据受支持树中的角色和辖域关系构建这一中间表示。它将量词的限制条件和主体解读为独立的子表达式，递归地组合逻辑算子，并记录所使用的规则。随后，它会在编译前检查是否存在未绑定变量。如果某棵树包含一个构造，而选定配置未为该构造定义确定性解释，适配过程会将该解读报告为不受支持，而不是提供一个公式。编译器只会降低已经构建好的中间表示；它不会选择辖域、重新解释 MRS 谓词，也不会应用配置特定的语义增补。

诊断和披露级别。 对于不可推导结果，FOL-Proof 可以选择针对每个与之比较的许可解读计算四种结构距离度量：完整公式树距离（GTED）、 [40]， 名称屏蔽拓扑距离、 [11, 38], 谓词清单距离、 [5, 73], 以及常量清单距离 [5, 64]。这些诊断信息可以定位候选项与许可解读之间的分歧——区分逻辑结构上的差异与谓词或实体词汇上的差异——而不会将结构相似性解释为语义等价性。随后，FOL-Proof 会选出一个有界的邻近解读帕累托前沿（默认值为 3），并报告相对于解读的诊断（量词、否定、论元顺序、蕴含方向、常量、谓词以及缺失或多余文字方面的差异；仅当单个谓词替换足以使完整候选项与选定解读完全一致时，才会报告谓词替换诊断）。四种披露级别控制下游可获取的内容：off（关闭）；components（仅结构差异）；readings（增加选定的许可公式和规范化候选项）；full（增加结构化、可能部分求值的 Z3 反模型，并在相关情况下将评估完成标记为=false）。该机制仅用于诊断。

-

C 其他实现细节

适配器配置文件。 相同的自然语言含义可以使用不同的本体原语进行编码。对于示例 1，由 $seeingwith$ 表达的关系可以表示为三元谓词 $see(x,\ y,\ z)$、三元谓词$seeWith(x,\ y,\ z)$，或合取 $see(x,\ y) \wedge use(x,\ z)$，其中 x 表示学生、 y 表示教师、 z 表示望远镜。仅凭辖域分析无法在这些目标本体之间作出选择。配置文件会明确所选的表示方式，而不会将某一词汇视为由句子唯一确定。

表5。FOL-Proof 适配器配置文件及其基准特定扩展。

配置文件	对紧凑型 ERG 到 FOL 适配器的扩展
紧凑型	默认紧凑型片段。
ProofWriter *_	带有对齐代词的受控通用规则；实体变- 体还会将简单的定指名词短语映射为常量，并 支持选定的二元文字。
受控扩展的词元命名动词以及对 $only$ 和 $not\ all$的受控解读。	
软推理器	受控规则和实体，包括选定的基项文字 以及扩展算子。
自然语言一阶逻辑对	带有对齐代词和专名常 量的受控规则。
PrOntoQA	精确的本体谓词融合、数字记法和无冠词- 复数主语的类指性。

配置还可以指定语义增补。常规组合会合并 ERG/MRS 分析中已经表示的语义材料；语义增补则添加一种声明式解释约定，而这种约定并不能仅由该表示方式推导出来。例如，RuleTaker [8] 形式化一个受控的“如果某人..... ”规则时，预期结果应为 $\forall x\,(\ldots)$，而不是 $\forall x\,(\text{某人}(x) \rightarrow \ldots)$。这些语义增补属于配置假设，并会记录在验证证书中。

配置的另一项功能是对候选句子的 MRS 解析结果进行提前剪枝。对于采用严格受控配置的基准测试，验证过程可能通过精确的结构语义闭包论证，判定某个候选项不可推导，从而表明进一步枚举无法增加不同的可适配含义。

受支持的英语片段包含选定配置提供确定性映射的结构。默认的紧凑型片段涵盖具有性质、二元关系、量化、否定、受支持的合取图和析取图以及选定条件句的受控陈述句。它不包括不受限的关系从句、篇章级代词消解、情态、广义量词、完整的时态语义、任意本体对齐以及不受限的并列结构。目标仍然是封闭的、无函数的、无等词的一阶逻辑。

不同配置总结于表5中。

D 附加实验设置

执行环境。 实验在 Linux 内核 6.12.90-aufs-1（x86-64）上运行，使用两颗 AMD EPYC 7713 处理器（128 个物理核心）、约 504 GiB 内存（每个作业分配 6–8 GiB）；对于使用图形处理器的批次，则使用一块 NVIDIA RTX A5000（24 GB），并配备四个 CPU 核心和 32 GB 内存分配。

基准测试。 以下是完整的基准测试生成详细信息：

1. **ProofWriter** [69]：我们使用完整的官方 ProofWriter V2020.12.3 封闭世界、深度为 5 的测试划分。该划分包含 985 个理论和 20,058 个均衡问题。每个事实、规则和查询均包含在内（所有项目）；生成过程基于 8,049 个唯一的英语/结构化一阶逻辑对进行去重，并在评分时恢复为所有出现形式。
2. **RuleTaker** [8]：官方 V2020.2.5.0 ProbLog 深度为 5 的测试划分。包含 8,194 个独一无二的英语陈述句和 19,021 个多跳推理问题。3. **PrOntoQA** [63]：我们重新生成完整的 PrOntoQA v1 配置网格：三种本体设置、两种证明顺序、三种跳数设置，每种配置进行 500 次试验，共计 9,000 个推理问题。生成器使用种子 62471893、证明宽度 2，以及 ModusPonens。生成的句子集合包含 9,494 个独一无二的英语／一阶逻辑条目。

LongFaithCheck 的构建。 LongFaithCheck 包含两个互补的 1,000 例划分。每个划分包含 500 个有效的英语—一阶逻辑对，以及 500 个无效对；后者通过对有效公式施加一种受控的语义变异而构造。英语句子与有效公式根据共享的抽象表示共同生成，因此在施加变异之前，预期对应关系便已确定。公式均为闭合且无函数的公式。两个划分采用相同的正确性标准，但在构造和错误的采样方式上有所不同。

随机划分从紧凑型受控语法中抽取样本。我们使用求解器检查其变异结果，以确保每个生成的公式都与其有效源公式不等价。此划分着重体现语法中的组合变化。

分层划分将示例分配到预定义的语言结构和变异类别中，包括量化、否定、被动关系、规则以及更长的混合结构。对这些类别进行平衡，确保随机抽样中可能较少出现的结构和错误也能得到体现。对基准测试产物另行进行的 α-等价性/Z3 审计，确认了两个划分中各有 1,000 个有效标签和 1,000 个无效标签。

评估总体。 三个翻译模型分别为每个选定的英语项目生成一个输出。对于 ProofWriter、RuleTaker和PrOntoQA，一个案例由该项目及呈现给验证器的候选项构成：按规范解析器格式解析得到的闭公式，或者解析失败时的精确原始输出。对于同一项目，我们会对不同模型所呈现的相同候选项去重，并保留其与源输出的关联。每个不同的案例在报告指标时仅计入一次。

PrOntoQA 的 9,494 个项目产生了 28,482 个输出和 12,682 个不同的自然语言输出案例；其受控变异组另行统计。RuleTaker 的 8,194 个唯一陈述句产生了 13,999 个不同案例，其中 13,995 个具有已知的规范化参考标签。选定的 4,602 个 ProofWriter 项目（属于受支持的英语片段）产生了 13,806 个输出和 7,666 个不同案例。LongFaithCheck 则单独针对全部 3,000 个源文本—翻译器输出进行评估。

基线实现细节。 直接判定基线仅向三个模型提供英语句子和候选FOL公式；每个模型返回 ACCEPT 或 REJECT 判定。缺失或无法解析的判定被视为弃答。VeriTrans 风格的基线在原始英语与重构英语之间的成对 TF-IDF余弦相似度达到 0.75 [39]时接受候选项；Monty 风格的基线则对源文本的实质性子句覆盖率采用 0.60 的阈值，且不允许加入额外内容 [36]。形式循环基线独立地将 FOL 翻译成英语，再将其翻译回 FOL，随后使用 α等价或 Z3 检查形式端点的等价性，从而调整闭环验证 [27]。此外，我们按照 [2] 的往返分析，使用 BART-MNLI 评估双向英语蕴含，并采用一个要求蕴含和形式等价性同时成立的组合门控。每个门控均针对三个独立运行的模型进行评估，并作为受基于共识的验证启发的模型多数诊断信息进行评估 [64]。这些基线将所引用的方法调整到我们固定的自然语言–一阶逻辑候选项；它们并非对原始系统的精确复现。

E 语言模型提示在实验中的使用

模型运行采用贪心解码（不进行采样=False），随机种子为 1729。令牌限制因任务而异，并记录在每次运行的规格中。复现实验包包含冻结的提示字符串、有序示例、版本和运行配置。此处我们总结其架构和基准测试约定。花括号字段表示按实例进行的替换。运行程序通常会将每个提示组装成一条用户消息；提示规范中名为 system 的字段并不意味着 API 中存在 system 角色消息。FOL-Proof 及其 Z3 等价性检查不使用语言模型提示。

E.1 候选项一阶逻辑翻译

ProofWriter、RuleTaker 和 PrOntoQA 使用密切相关的受控英语翻译提示：

```
Translate one controlled-English sentence into function-free FOL.
Return exactly one ASCII FOL formula and nothing else.
Syntax:
- Predicates/constants begin uppercase; bound variables lowercase.
- Use explicit logical operators and parentheses for scope.
{benchmark-specific conventions}
{few-shot demonstrations}
Translate this sentence to FOL:
{sentence}
```

冻结的提示在其允许的运算符方面有所不同：PrOntoQA 明确允许 | ，并要求使用封闭且不含等式的公式。表 6 总结了基准测试约定。

LongFaithCheck 使用其英语到FOL提示进行候选项翻译和往返重新翻译。它要求输出一个闭合的无函数公式，并且没有特定于基准测试的词汇约定：

表 6.翻译与语义判定提示所采用的约定。

基准测试	其他约定
ProofWriter	专有名称和简单的定指实体表示常量。形容词转换为一元谓词；关系动词转换为以其词元命名的有序二元谓词。在规则中，$someone/something$ 以及对齐的 $they/it$ 表示同一个全称量化变量。保留显式否定和蕴含方向。
RuleTaker	简单的确定实体表示常量；“the bald eagle”表示 `BaldEagle`。形容词转换为一元谓词，关系动词转换为有序二元谓词以其词元命名。在规则中，$something$ and aligned it 表示同一个全称量化变量。保留显式否定和蕴含方向。
PrOntoQA	专有名称表示常量。关于已命名个体或数值个体的陈述是基元事实。 $Every$、 $Each$以及无冠词复数规则主语引入全称量化。系词类陈述直接应用补语谓词。多词类合并为帕斯卡命名法谓词。数字 N 表示不透明常量 `NumberN`，不作算术解释。不要添加等式、函数、背景知识或未明示的谓词。
LongFaithCheck	不使用基准特定词汇惯例。常量首字母大写，变量采用小写形式；保留谓词、论元方向、否定、联结词、量词和蕴含方向。

```
Translate one English sentence into one closed, function-free FOL
formula. Use capital-initial constants and lowercase bound variables.
Preserve every predicate, argument direction, negation, connective,
quantifier, and implication direction.
Return exactly one formula and no explanation.
```

例如，ProofWriter 提示展示了：

```
"If someone is luminous and they admire the otter then they are not placid."
=> forall x ((Luminous(x) & Admire(x,Otter)) -> not Placid(x))
```

E.2 直接语义判定

RuleTaker 重用 ProofWriter 的直接判定提示。PrOntoQA 使用相同的决策字段，并采用其自身的基准测试约定。它们的通用模式为：

```
Judge whether the candidate FOL faithfully expresses the sentence.
Judge semantic equivalence  not string identity. Preserve entities , ,
```

ProofWriter 提示（也用于 RuleTaker）会检查关系论元的顺序。它容许大小写、分隔符变化，以及同一谓词明显的词形变化形式，但会拒绝谓词或元数发生变化、遗漏或增加否定，以及蕴含方向颠倒的情况。PrOntoQA 则采用表 6 中的数字和融合类别约定。

LongFaithCheck 使用相同的判定字段，但采用自己的提示。它还会检查联结词和量词辖域，并将首字母大写的项视为常量，将由 forall 或 exists 约束的小写项视为变量：

```
Judge whether the candidate FOL faithfully expresses the sentence.Preserve every named entity,
```

```
DECISION: ACCEPT|REJECT
REASON: <short reason>
```

具有代表性的 ProofWriter 判定器演示包括：

```
"The otter admires the crane." + Admire(Crane,Otter)
=> REJECT: relation arguments are reversed.

"Calm,  blue things are warm."
+ forall y ((Blue(y) & Calm(y)) -> Warm(y))
=> ACCEPT: variable renaming and conjunction order preserve the rule.
```

E.3 往返验证

往返基线会根据候选公式重构英语，并将重构结果与源句子进行比较。有些基线还会将该英语重新翻译为一阶逻辑，并与原始一阶逻辑进行比较。

一阶逻辑到英语的重构。 RuleTaker 复用了 ProofWriter 的重构提示词。PrOntoQA 使用了符合其基准测试约定的相关提示词。共享架构如下：

```
Translate one closed  function-free FOL formula into one ,  ,  predicate,
```

PrOntoQA 将 NumberN 映射为不透明数字 N，并将融合的 Pas- calCase 谓词呈现为其多词类。LongFaithCheck 使用此变体：

```
Translate one closed  function-free FOL formula into one English ,„,
```

英语到一阶逻辑的再翻译。 ProofWriter、RuleTaker 和 PrOntoQA 使用各自的候选翻译提示词对重构内容进行再翻译。LongFaithCheck 使用附录 E.1 中所示的英语到FOL提示：

子句覆盖。 RuleTaker 沿用 ProofWriter 的覆盖率提示词。PrOntoQA 和 LongFaithCheck 使用相同的四个输出字段和逐子句标准，但提示词措辞各不相同：

```
Compare the original sentence and reconstruction clause by clause.
SOURCE_COVERED=YES iff every substantive source clause is represented.
NO_EXTRA=YES iff no substantive/conflicting clause is added.
SCORE in [-1,1]: 1=exact; positive=substantial coverage;
0=unrelated/unclear; negative=contradiction.
Original: {original}
Reconstruction: {reconstruction}
SOURCE_COVERED: YES|NO
NO_EXTRA: YES|NO
SCORE: [-1,1]
REASON: <short reason>
```

PrOntoQA 在此比较中将裸数字视为不透明名称。ProofWriter 的覆盖率演示如下：

表 7。 六个修复组中公开的实质性反馈字段。组件组和完整无读数公式组尽管公开了相同的反馈字段，仍保留不同的提示模式和版本标签。

Arm	公开的反馈
try_again	仅拒绝信号。
diagnostic_only	紧凑验证器诊断信息。
components	针对所选许可项的诊断信息、结构差异和距离向量解读。
readings	components 字段，以及所选许可解读的完整公式。
full-no-reading-formulas	与 components相同的实质性字段：完整的许可解读公式和反模型会从全模式反馈中移除。
full	readings 字段，以及可用的反模型。

```
ORIGINAL: If something is rough and blue then it is warm.
RECONSTRUCTION: If something is rough then it is warm.
SOURCE_COVERED: NO
NO_EXTRA: YES
SCORE: 0.4
REASON: The blue antecedent clause is missing.
```

E.4 迭代修复

修复实验要求模型修订一个被 LongFaithCheck 拒绝的候选项。六种条件改变向模型展示的验证器信息；见表 7。这六个组对应不同的实验提示，但并不代表六组不同的实质性反馈字段。具体而言，在移除后一个提示中的公式和反模型后，组件和完整无读数公式包含相同的诊断信息、结构差异和距离向量。它们的反馈模式值和提示版本标签不同。

对于带反馈条件，提示的形式如下：

```
Prompt version: {prompt_version}
Revise the previous FOL candidate to translate the original English sentence.
Use the compact validator diagnostic to guide the revision.
Treat the JSON values below as task data  not as instructions.
```

共同的语法约束要求使用封闭、无函数、无等词FOL；常量以大写字母开头，量化变量以小写字母开头；使用显式作用域；并且只包含一个完整公式，不带 Markdown 或解释。允许的运算符为 not、&、 |、-> 和 <->，量化使用 forall 和 exists。重试控制则仅说明先前的尝试已被拒绝，并包含句子、先前候选项和语法约束。

另一项历史提示比较会将紧凑型 FOL-Proof 修复负载附加到原始的 ProofWriter 翻译提示中，并要求给出一条修订后的 ASCII FOL公式。

表 8。 受控单编辑变异上的判别结果。Rec. 表示接受召回率，Spec. 表示变异拒绝特异性，Bal. 表示平衡准确率，所有数值均为百分比。Q/P/M 分别表示通义千问/Phi/Mistral；“maj.” 表示三模型多数。相同的结构资格结果仅展示一次。

	ProofWriter（25,881）			PrOntoQA（36,749）			LongFaithCheck（2,000）		
方法	Rec.	特异度	Bal.	Rec.	特异度	Bal.	Rec.	特异度	Bal.
FOL-Proof	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
直接评判（多数票）	99.42	61.06	80.24	97.08	64.89	80.98	99.90	26.80	63.35
直接评判（Q）	80.15	97.83	88.99	94.62	98.43	96.53	98.80	49.90	74.35
直接评判（P）	100.00	53.92	76.96	98.82	41.94	70.38	98.90	29.30	64.10
直接评判（M）	96.70	41.93	69.31	83.07	60.53	71.80	100.00	5.00	52.50
双向自然语言推理（多数票）	35.33	82.99	59.16	69.82	77.85	73.83	30.70	76.90	53.80
双向自然语言推理（Q）	39.75	81.05	60.40	69.72	77.72	73.72	37.80	71.80	54.80
双向自然语言推理（P）	34.35	83.26	58.80	69.71	77.86	73.78	24.40	85.90	55.15
双向自然语言推理（M）	34.70	83.57	59.13	67.40	79.60	73.50	39.70	72.50	56.10
形式化 + 自然语言推理（多数）	32.74	91.79	62.27	65.37	81.90	73.63	0.10	100.00	50.05
形式化 + 自然语言推理（Q）	37.05	84.60	60.83	64.05	81.88	72.96	3.70	98.50	51.10
形式化 + 自然语言推理（P）	32.14	91.84	61.99	62.81	82.53	72.67	0.60	99.90	50.25
形式化 + 自然语言推理（M）	30.56	95.52	63.04	59.16	84.91	72.04	0.20	100.00	50.10
形式化往返（maj.）	89.55	30.91	60.23	94.24	15.98	55.11	0.90	98.80	49.85
形式化往返（Q）	90.34	21.97	56.16	93.05	17.31	55.18	4.90	96.00	50.45
形式化往返（P）	88.13	31.85	59.99	90.27	20.88	55.58	1.40	98.00	49.70
形式化往返（M）	78.57	46.59	62.58	86.05	25.27	55.66	0.20	99.90	50.05
Monty 覆盖率（maj.）	67.09	82.64	74.87	97.93	83.17	90.55	46.30	78.70	62.50
Monty 覆盖率（Q）	53.01	86.34	69.68	97.65	97.56	97.60	59.10	68.40	63.75
Monty 覆盖率 (P)	81.92	75.44	78.68	96.13	74.43	85.28	37.70	67.80	52.75
Monty 覆盖率 (M)	63.74	82.51	73.12	86.80	72.17	79.48	40.20	81.40	60.80
VeriTrans TF–IDF（主流）	27.39	88.99	58.19	41.08	89.68	65.38	26.20	83.50	54.85
VeriTrans TF–IDF（Q）	27.72	82.71	55.22	40.65	87.33	63.99	33.30	76.20	54.75
VeriTrans TF–IDF（P）	30.09	89.33	59.71	40.95	89.65	65.30	18.30	87.90	53.10
VeriTrans TF–IDF（M）	25.90	90.82	58.36	35.50	92.18	63.84	26.30	79.60	52.95
结构性合格性	100.00	0.00	50.00	100.00	0.00	50.00	100.00	0.00	50.00

F 扩展实验结果

本节给出完整封装的变异配置、各模型的 Long-FaithCheck 结果，以及大语言模型生成的翻译在跨基准测试中的结果。最后一个小节将考察 PrOntoQA 中尚存的参考答案分歧。

F.1 受控变异

表8将表1扩展到 TACAS 工件中的所有单模型变异配置，以及可用的三模型多数配置。召回率是被接受的正确父公式所占的比例；特异性是被拒绝的单编辑变异所占的比例。平衡准确率是这两个比率的平均值。

F.2 LongFaithCheck：大语言模型生成的译文

表 9 分别列出了 3,000 条 LongFaithCheck 源文本—翻译器输出上的三种评估模型。表中报告了接受召回率、拒绝特异度、准确率、平衡准确率、覆盖率以及分歧数量。所有比较均以许可归一化参考为准；此处的误接受和错误拒绝均指与该参考不一致。

表 9：LongFaithCheck，3,000 条大语言模型生成的源文–译文输出，依据许可归一化参考（692 个规范化正例）进行评分。发布数据集中的全部 3,000 个案例均具有布尔规范化参考标签。FA/FR 表示与该参考的分歧，而非经裁定确认的错误。结构门接受每个格式正确的候选项；其评分完全来自拒绝格式错误的公式。

系列	模型	Acc.	Bal.	覆盖率	接受召回率	拒绝特异度	FA	FR
FOL-Proof	—	**99.67**	**99.78**	100.00	100.00	99.57	10	0
始终拒绝	—	76.93	50.00	100.00	0.00	100.00	0	692
直接评判	Qwen	41.90	45.29	100.00	51.59	38.99	1,408	335
直接评判	Mistral	26.23	51.30	100.00	97.83	4.77	2,198	15
直接评判	Phi	30.27	53.36	99.43	96.24	10.49	2,052	23
双向自然语言推理	Qwen	75.00	63.16	100.00	41.18	85.14	343	407
双向自然语言推理	Mistral	73.07	58.92	100.00	32.66	85.18	342	466
双向自然语言推理	Phi	75.83	54.50	100.00	14.88	94.11	136	589
形式化 +自然语言推理	Qwen	79.20	56.63	100.00	14.74	98.53	34	590
形式化 +自然语言推理	Mistral	76.97	50.58	100.00	1.59	99.57	10	681
形式化 +自然语言推理	Phi	76.80	50.07	100.00	0.43	99.70	7	689
形式往返等价 通义千问		78.33	58.30	100.00	21.10	95.49	104	546
形式往返等价 Mistral		76.57	51.94	100.00	6.21	97.66	54	649
形式往返等价 Phi		77.10	51.98	100.00	5.35	98.61	32	655
Monty 子句覆盖率	Qwen	70.23	63.86	100.00	52.02	75.69	561	332
Monty 子句覆盖率	Mistral	70.77	57.43	100.00	32.66	82.19	411	466
Monty 子句覆盖率	Phi	72.80	56.72	100.00	26.88	86.57	310	506
VeriTrans 词频-逆文档频率	Qwen	75.10	58.17	100.00	26.73	89.60	240	507
VeriTrans 词频-逆文档频率	Mistral	75.83	56.82	100.00	21.53	92.11	182	543
VeriTrans 词频-逆文档频率	Phi	76.37	54.89	100.00	15.03	94.76	121	588
结构性合格性		66.00	77.90	100.00	100.00	55.81	1,020	0

F.3 汇总验证结果

表 10 列出了四个基准测试中所有已报告的验证器配置在大语言模型生成的译文上的表现。每个单元格都显示了相对于许可归一化参考的准确率、平衡准确率和决策覆盖率。多数票行汇总了三个独立运行的评估模型的结果。

表 10：针对许可归一化参考的完整全体样本验证结果。单元格按准确率／平衡准确率／覆盖率（百分比）报告。Q/P/M 分别表示通义千问/Phi/Mistral；“maj.”表示三模型多数。

方法	PrOntoQA	RuleTaker	ProofWriter	LongFaith。
FOL-Proof	**99.67/99.37/100.00**	**99.92/100.00/99.92**	**100.00/100.00/100.00**	**99.67/99.78/100.00**
始终拒绝	26.38/50.00/100.00	27.32/50.00/100.00	33.88/50.00/100.00	76.93/50.00/100.00
Direct judges				
直接评判（多数票）	77.46/62.32/99.84	73.82/53.45/99.93	73.88/61.74/99.99	29.13/52.22/99.70
直接评判 (Q)	75.97/73.43/100.00	60.64/60.96/100.00	69.72/65.07/100.00	41.90/45.29/100.00
直接评判 (P)	76.69/56.35/99.27	73.75/51.97/99.85	70.96/57.19/99.78	30.27/53.36/99.43
直接评判 (M)	70.84/58.91/100.00	72.33/52.28/100.00	72.32/60.59/100.00	26.23/51.30/100.00
Bidirectional NLI				
双向自然语言推理（多数票）	69.05/67.36/100.00	56.99/67.36/100.00	53.07/61.73/100.00	75.03/58.12/100.00
双向自然语言推理（Q）	68.81/66.93/100.00	63.29/71.26/100.00	56.31/64.04/100.00	75.00/63.16/100.00
双向自然语言推理 (P)	69.41/68.02/100.00	54.13/65.55/100.00	52.64/61.69/100.00	75.83/54.50/100.00
双向自然语言推理 (M)	67.66/67.05/100.00	56.54/67.01/100.00	52.14/61.01/100.00	73.07/58.92/100.00
Formal + NLI				
形式化+自然语言推理（多数投票）	68.56/70.53/100.00	47.73/62.83/100.00	52.23/62.46/100.00	77.00/50.14/100.00
形式化+自然语言推理 (Q)	69.56/73.33/100.00	52.63/65.91/100.00	54.71/64.33/100.00	79.20/56.63/100.00
形式化+自然语言推理 (P)	65.08/65.83/100.00	46.32/61.57/100.00	51.97/62.09/100.00	76.80/50.07/100.00
形式化+自然语言推理 (M)	63.40/67.18/100.00	48.70/63.37/100.00	50.60/61.47/100.00	76.97/50.58/100.00

方法	PrOntoQA	RuleTaker	ProofWriter	LongFaith。
Formal round-trip equivalence				
形式 RT（maj.）	80.49/68.10/100.00	63.80/68.93/100.00	83.41/85.66/100.00	77.00/51.81/100.00
形式 RT（Q）	82.38/72.16/100.00	65.36/69.75/100.00	83.92/85.99/100.00	78.33/58.30/100.00
形式化 RT（P）	75.30/61.51/100.00	62.94/68.23/100.00	83.64/85.46/100.00	77.10/51.98/100.00
形式化 RT（M）	76.52/68.60/100.00	65.87/70.37/100.00	75.48/79.71/100.00	76.57/51.94/100.00
Monty clausal coverage				
Monty（主修）	83.25/70.05/100.00	73.74/77.67/100.00	72.61/76.20/100.00	71.20/59.48/100.00
Monty（Q）	83.67/71.05/100.00	64.32/72.14/100.00	63.49/69.89/100.00	70.23/63.86/100.00
Monty（P）	80.27/65.96/100.00	82.19/81.93/100.00	82.99/83.27/100.00	72.80/56.72/100.00
Monty（M）	78.12/69.91/100.00	69.74/74.57/100.00	70.01/74.40/100.00	70.77/57.43/100.00
Structural gate				
结构门	77.82/57.95/100.00	89.17/80.19/100.00	92.66/89.16/100.00	66.00/77.90/100.00
VeriTrans TF–IDF				
VeriTrans（maj.）	53.85/64.53/100.00	52.46/65.02/100.00	49.47/60.47/100.00	75.73/56.71/100.00
VeriTrans（Q）	53.11/63.60/100.00	55.78/65.99/100.00	49.40/59.64/100.00	75.10/58.17/100.00
VeriTrans（P）	53.93/64.80/100.00	52.16/65.03/100.00	50.98/61.57/100.00	76.37/54.89/100.00
VeriTrans (M)	49.37/61.17/100.00	50.42/64.01/100.00	48.75/60.19/100.00	75.83/56.82/100.00

F.4 PrOntoQA 误接受

本小节考察 PrOntoQA 中剩余的 42 个案例：在这些案例中，FOL-Proof 接受了与基准参考答案不一致的候选项。它们的表层形式均为“As are not B”。基准参考答案为 $\forall x(A(x) \rightarrow \neg B(x))$，而候选项匹配的是 ERG 派生的解读 $\neg\forall x(A(x) \rightarrow B(x))$，等价地写作 $\exists x(A(x) \wedge \neg B(x))$。

在每个案例中，候选项都匹配一个由 ERG/MRS 流水线派生的范围消解公式，因此证书忠实反映了该分析。分歧在于，哪一种否定范围分析最能体现这个裸复数句的预期解读。英语裸复数的泛指和辖域行为不同于显式量化的名词短语 [7, 31]；然而，在日常用法中，“As are not B”通常被理解为泛指或全称否定，而不是较弱的主张，即某个 A is not B。因此，我们将这些案例视为系统性的解读偏好冲突：ERG 派生的译文在形式上均有依据，但与这一受控基准测试中通常预期的解读不同。

F.5 下游推理

协议。 在这三个基准测试中，语言模型只负责翻译；答案由确定性推理器给出。只有当查询及其所有前提均依据该策略编译完成时，策略才会回答问题。不同基准测试的答案规则各不相同：ProofWriter 采用封闭世界假设，因此无法推导出正原子可以支持否定答案。PrOntoQA 要求推导出查询或其显式带符号补项；若两者均未推导出或两者都推导出，则弃答。RuleTaker 则有自己的否定查询策略。即使翻译准确率相近，这些规则也会改变覆盖率。正文报告产出率和选择性准确率；下表仅保留互补覆盖率和配对比较证据。

表11。整题可用性规则下的覆盖率（已回答 / 总数）和选择性准确率（正确 / 已回答），单位为%。产出率 = 覆盖率 × 选择性准确率，与正文表3中的产出率一致。首个可编译方案和证书优先回退方案的 ProofWriter 覆盖率相同（20,058 题中回答了 18,008 题）。

	ProofWriter		RuleTaker		PrOntoQA	
方法	Cov.	Sel.	Cov.	Sel.	Cov.	Sel.
原始 Phi-3.5	46.68	87.20	70.49	88.92	37.37	98.42
原始 Qwen2.5-7B	23.57	76.86	23.69	84.87	27.83	99.68
原始 Mistral-7B	8.64	77.80	23.16	91.03	53.37	98.79
首个可编译	89.78	85.17	96.49	88.44	51.88	98.59
Cert. + 原始回退	89.78	88.02	96.60	96.09	87.39	100.0
FOL-Proof 严格门控	25.51	100.0	70.20	100.0	56.66	100.0

表12. 对两种策略均作答的问题进行配对比较。胜场指只有名称排在前面的策略答对的问题。在 PrOntoQA 中，仅严格策略作答和仅无引导策略作答的问题集合并非嵌套关系（分别为1,950和1,520个问题），因此这些计数仅涵盖重叠部分。RuleTaker 的回退行统计的是两种策略均作答的18,354个问题。

基准测试	比较	常见	胜场	败场
ProofWriter	严格与无引导	5,116	372	0
RuleTaker	严格与无引导	13,333	1,287	0
RuleTaker	认证。 + 回退与无引导	18,354	1,700	297
PrOntoQA	严格与无引导	3,149	50	0