用于科学表格论点验证的原子推理

Yuji Zhang[1], **Qingyun Wang**[1], **Cheng Qian**[1], **Jiateng Liu**[1], **Chenkai Sun**[1], **Denghui Zhang**[1,2]
Tarek Abdelzaher[1], **Chengxiang Zhai**[1], **Preslav Nakov**[3], **Heng Ji**[1]
[1]University of Illinois Urbana-Champaign
[2]Stevens Institute of Technology, [3]MBZUAI
{ yujiz, hengji } @illinois.edu

Abstract

科学文本通常因其技术性语言和复杂的数据而传达权威性。然而，这种复杂性有时会导致错误信息的传播。非专家特别容易受到基于科学表格的误导性声明的影响，因为这些表格信息密度高且被认为具有可信度。现有的表格声明验证模型，包括最先进的大型语言模型（LLMs），通常在精细化推理上表现不佳，导致在验证科学声明时出现错误和缺乏精确性。受到认知负荷理论的启发，我们提出，通过开发模块化、可重用的推理组件（即原子技能）来减少认知负荷，可以增强模型解释基于表格的声明的能力。我们引入了一种技能链构模式，可以动态地组合这些技能，以便在减少认知负荷的情况下实现更准确和更具普遍性的推理。为评估这一点，我们创建了SciAtomicBench [1]，这是一个具有细粒度推理注释的跨域基准。我们只使用 350 个微调示例，通过原子推理训练的模型，在性能上超越了 GPT-4o 的链式思维方法，以远少于其他训练数据的情况达到了最先进的结果。

1 介绍

在科学领域，专业术语、复杂的表达以及专家的光环赋予了信息权威，但也使其易于被故意扭曲和快速传播误导性信息 (Cabanac et al., 2021; Else, 2021; Lim et al., 2021)。非专业读者由于缺乏深入的领域知识，容易将误导性或错误的主张当作事实接受，因为科学内容本身具有的固有可信度 (Osborne and Pimentel, 2023)。这种误置的信任可能会带来严重后果；例如，欺诈性的心脏干细胞研究误导了科学家，导致患者接受无效治疗，并分散了正当医学进步的资源 (Kowalczyk, 2017)。具体来说，基于欺诈性心脏干细胞研究的 CONCERT-HF 试验导致了患者死亡 (Bolli et al., 2018)。因此，检测和标记科学误导信息既紧迫又必不可少。

在科学领域，表格作为记录和表示数据的关键媒介，通过将复杂的数据压缩成易于访问的格式 (Inskip et al., 2017)，许多论断依赖于它们的精确解读。然而，非专业人士常常难以解析这些密集的格式，使得表格成为细微误解的重要载体。因此，对表格的准确理解对于验证论文中相关论断的真实性和保持可靠的科学环境至关重要。为此，研究人员利用大型语言模型（LLMs）来解决科学表格事实核查挑战，通过验证论断与表格 (Gupta et al., 2020; Chen et al., 2020; Wang et al., 2021a; Akhtar et al., 2022; Lu et al., 2023)。然而，针对表格设计的模型在应用于科学领域的细微需求时仍显不足 (Lu et al., 2023)。此外，即使是像 GPT-4 (OpenAI, 2023) 这样的最先进的闭源大型语言模型，其表现仍比人类逊色 20 % (Lu et al., 2023)。

我们的分析揭示了现有模型的一个关键限制：它们往往缺乏在推理过程中明确分解和调用细粒度原子技能的意识，这限制了它们对复杂表格理解任务的泛化能力。如图 1 所示，我们的原子推理方法将验证分解为细粒度推理步骤中的三项基本技能：(1) 概念匹配，(2) 值提取，和 (3) 数值计算。当这些技能被明确应用时，模型得出正确的结论。相比之下，ChatGPT 无论是通过直觉还是链式思维 (Chain-of-Thoughts, CoT) (Wei et al., 2022) 推理，都未能隔离这些操作，反而依赖于混合或跳过步骤的推理，导致错误。

受到认知负荷理论的启发，(Plass et al., 2010)，该理论认为人类工作记忆是有限的，学习在教学设计最大限度地减少额外负担时得到优化，我们观察到大型语言模型（LLMs）在解释信息密集型科学表格时面临类似的挑战。由于模型无法灵活地识别和重用大量且多样的表格断言中所包含的常见推理操作，它们承受着沉重的认知负荷和处理负担。为了解决这个问题，我们引入了一套高度模块化、可重用和可推广的能力，称为原子技能，每个技能都包含一种独特的推理操作（例如，概念消歧、数值计算、趋势检查）。将繁重的验证工作流程分解为这些原子技能可以减少模型的推理负担，并促进在新颖的表格和断言类型以及多样化领域中的更强泛化能力。

[1]我们的数据集和模型将在 https://github.com/CelestineZYJ/SciAtomicBench 发布后公开。

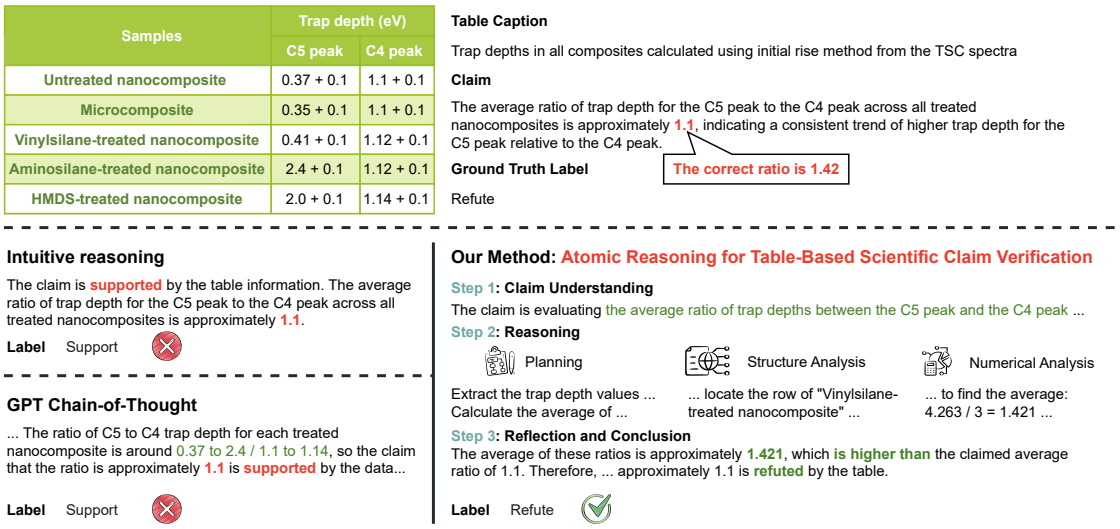

Figure 1: 我们的 SciAtomic 基准中的一个材料科学表格声明验证示例，说明了直观推理、CoT 推理和我们提出的原子推理之间的区别。

为了让模型具备原子技能，我们为能力训练策划了一个紧凑多样的数据集，并引入了一种技能链接架构，该架构在上下文中生成技能，如图 2 所示。传统的 CoT 提示将任务分解为一个统一的推理路径 (Wei et al., 2022)，既不强制进行细化的技能控制，也无法防止在冗长序列中出现的信息丢失问题，这种问题在关键细节在链中间消失时会降低性能 (Liu et al., 2024)。而我们的方法则为每一步定义了精确的上下文和目标，并将它们连接起来，使得每一步仅消耗其前一步的输出。在这些适当的条件下，模型能够动态选择和组合合适的原子技能，学习何时以及如何调用每个模块化的能力，而不是不加区别地增加链的长度。这种有纪律的模块化方法减少了认知负担，更有效地利用了推理时间扩展，并促进了对于不同表格声称的泛化能力。

在我们寻求更增强的表格声明解释过程中，我们面临着第二个关键挑战：现有的科学表格声明验证基准测试由于领域多样性有限和复杂性不足而受到影响。专家标注的数据集本质上是稀疏的，许多基准包括的声明未能反映现实世界科学探究的复杂性质。为了克服这些限制，我们引入了一个跨领域的完整精心策划的基准，涵盖材料科学、医学、金融和机器学习。我们的数据集包括原子推理链的详细注释，并提供了声明难度和推理复杂性平衡分布。

我们的实验结果展示了我们 AtomicTableLLM 的高效性和高效能。仅使用 350 个微调示例，我们就将基于 Deepseek-Qwen-7b 模型在金融领域的性能从 63.12 % 提升到了 85.70 %。在公开的 SciTab 基准测试中，我们的模型优于具有 CoT 推理的 GPT-4o，并超越了当前最先进的基线，这些基线通常是在更大规模的数据集（例如百万级）上训练的。我们的贡献有三个方面：

- 我们提出了数据高效的原子推理，使语言模型能够学习高度模块化和可组合的推理技能，从而增强它们在不同主张类型和科学领域中的泛化能力。
- 我们开发了专门针对表格的大型语言模型，AtomicTableLLM，该模型在科学表格声明验证中超越了最新的模型，展示了在科学领域中卓越的推理和泛化能力。
- 我们构建了一个新的科学表格声明数据集 SciAtomicBench，并用细粒度的原子技能和长推理链进行标注，涵盖包括材料科学、医学、金融和计算机科学在内的多个科学领域。

2 相关工作

2.1 表格专用大型语言模型

之前关于表格语言模型的工作主要集中在通用领域的资源上 (Lehmberg et al., 2016; Wang et al., 2018; Hu et al., 2019; Yin et al., 2020; Deng et al., 2020; Herzig et al., 2020; Iida et al., 2021; Wang et al., 2021b; Xie et al., 2022; Liu et al., 2022)。最近在大型语言模型方面的进展 (Dinh et al., 2022; Hegselmann et al., 2023; Jiang et al., 2023; Chen, 2023; Zhao et al., 2023; Li et al., 2023, 2024; Zhang et al., 2024c) 显示了在表格理解任务中的令人印象深刻的零样本和少样本性能。为了进一步提高推理能力和性能，其他大型语言模型（LLM）技术已经被应用于表格专用的 LLM，包括数据增强 (Li et al., 2024; Zhang et al., 2024b)、指令微调 (Hegselmann et al., 2023; Zhang et al., 2024a)、提示工程 (Jiang et al., 2023; Deng et al., 2025)、上

Benchmark / Method	Domains	Training Samples	Efficiency
Scigen	ML	/	/
Sciab	ML	/	/
FinQA	Fin	/	/
SciAtomic	ML, Med, Mat, Fin	350	✓
Tablellama	/	2.6 m	✗
Tablegpt	/	2.4 m	✗
Tapex	/	5 m	✗
TableLLM	/	14 k	✗

Table 1: SciAtomic 与表格主张验证基准和方法的比较。ML 表示机器学习。Fin 表示金融。Mat 表示材料科学。Med 表示医学。

下文学习 (Zhao et al., 2023) 、代码生成 (Lu et al., 2025; Zhang et al., 2025) 、链式推理 (Chen, 2023; Zhang et al., 2024c) 和多智能体协作 (Li et al., 2023) 。尽管在通用领域取得了可喜的进展，但之前没有研究专注于科学论文，因为高质量标注数据的可获得性有限。此外，我们是第一个将表格推理过程分解为一个原子技能集并研究新的组合推理链的人，从而确保模型正确调用必要的技能。

2.2 科学表格-主张基准

以往的科学事实核查数据集主要集中在文本描述上 (Wadden et al., 2020; Sarrouti et al., 2021; Wang et al., 2023) 。最近，对科学表格或图表的事实核查的兴趣日益增加，因为科学领域中大量关键信息是通过这些形式传达的 (Zhou et al., 2023; Huang et al., 2024, 2025) 。这些论文使用来源如维基百科 (Gupta et al., 2020; Chen et al., 2020) 、新闻 (Akhtar et al., 2022) 和研究论文 (Wang et al., 2021a; Lu et al., 2023) 的证据。然而，这些数据集通常忽略了表格事实核查中涉及的推理过程。相反，我们为每个声明-表格对注释了细粒度的原子技能和长推理链。此外，以往的工作专注于单个领域，包括机器学习 (Lu et al., 2023) 或生物医学领域 (Gupta et al., 2020; Chen et al., 2020; Wang et al., 2021a; Akhtar et al., 2022) ，这限制了其测试语言模型概括能力的能力。为了解决这一差距，我们构建了第一个跨多个学科的多样化数据集，包括机器学习、材料科学、医学科学和金融。

3 SciAtomic 基准测试

在本节中，我们正式化基于科学表格的主张验证任务，并描述 SciAtomic 基准的构建。

3.1 问题定义

我们研究基于科学表格的断言验证任务，其目标是在给定表格 T 的情况下，确定断言 C 是 SUPPORTED 还是 REFUTED 。形式上，一个由 θ 参数化的模型 $f_\theta(\cdot)$ 预测一个真实标签 $Y = f_\theta(T, C)$ ，其中 $Y \in \{\text{SUPPORT}, \text{REFUTE}\}$ 。每个表格 $T = (P, \{T_{i,j}\})$ 由提供领域特定上下文的标题 P ，以及按照 R_T 行和 C_T 列组织的单元格 $T_{i,j}$ 组成。断言是陈述性的科学声明，可能需要定量推理、多步骤推论或对表格及其标题的上下文解释。

现有的工作主要关注计算机科学表格中的声明，导致模型在科学领域中的鲁棒性受到限制。为了解决这一差距，我们引入了 SciAtomic 基准，其中包括基于来自材料科学、医学科学和金融等代表性不足但高度重要领域的表格的声明。所有这些领域都提出了各种推理挑战，并需要精确、可解释的验证来支持科学的完整性和决策制定。

3.2 数据收集

计算机科学。我们使用 SciGen 数据集 (Moosavi et al., 2021) 构建 SciAtomic 的计算机科学子集，该数据集包含从计算机科学领域的 arXiv 论文中提取的科学表格和说明。我们抽样了 1,376 个表格-说明对，重点在于保持结构和内容的多样性。

财务。我们手动整理了来自随机抽样的 S & P 500 公司合并财务报表的 343 张表格，包括损益表、资产负债表、现金流量表和权益报告。原始数据通过截图获取，并使用 GPT-4-Vision 转换为结构化格式。

医学科学。我们从 PubMed Central Open Access 中提取的 PubTables-1M (Smock et al., 2022) 的生物医学文献中选择了 1,468 个表格，通过布局感知解析确保了高细胞级的准确性，使其非常适合医学领域的科学主张确立。

材料科学。为了代表尚未充分探索但技术丰富的材料科学领域，我们纳入了来自 MatSciTable (Circi et al., 2024) 的 37 个由专家注释的表格，主要涉及聚合物复合材料。由于领域的稀疏性和深度，我们为每个表格生成多个主张，以抓住多样的推理路径，并强调特定领域的语义以进行多样且具有挑战性的验证。

我们在表格 2 中展示了数据统计。关于数据收集的进一步信息包括在附录 A 中。

为了在减少标注负担的同时有效生成高质量的声明，我们采用了一种人机协作框架 (Lu et al., 2023) 。受之前工作的启发，我们使用 GPT-4o 通过结构化提示生成支持和反驳的声明，随后进行人工验证。详细信息请参见附录 B 。

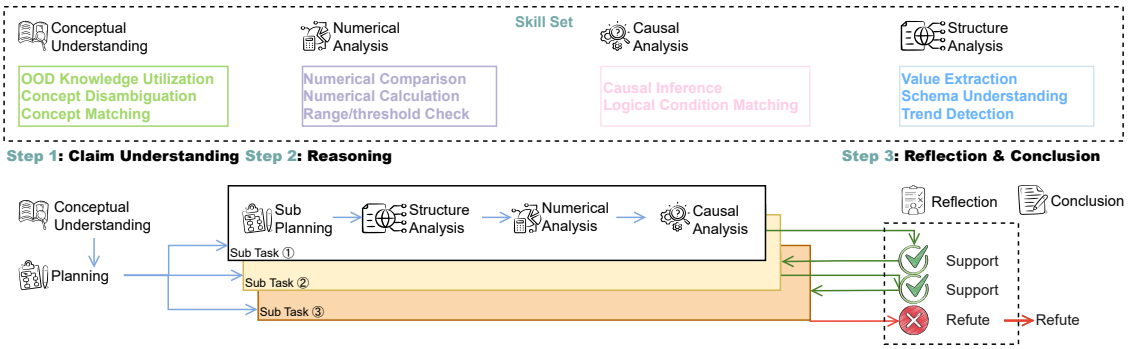

Figure 2: 通过配备原子技能的细化推理步骤来说明我们的技能链模式。

正面论述。 我们提示 GPT-4o 基于完整的表格生成精确的多步骤科学论断，需要对趋势、衍生指标以及领域知识进行推理。提示强调确定性并避免使用模糊表达，以确保具有挑战性和可验证性。

通过最小的语义翻转和针对性的数据显示创建的反驳声明在微妙地颠倒意义的同时保留了语言结构，模拟真实的科学误导信息，目的是隐藏误导信息并扩大危害。这种策略阻止了浅层模式匹配并促进了模型的稳健推理。

4 声称验证的原子推理

在这一部分中，我们介绍了 AtomicTable，这是一种针对基于表格的科学主张验证的技能链生成方案。AtomicTable 旨在通过对验证任务的细粒度和模块化分解来提高 LLM 的推理能力。我们的动机来自两个互补的维度：任务本身的性质以及 LLM 处理复杂推理的方式。

从任务分解的角度来看，基于表格的声明验证通常涉及多个独特且交错的推理操作，例如匹配实体、聚合数值、解释结构或识别因果模式。每一种操作都可以视为一个独立的子任务。通过明确地将整体验证分成这样范围明确的子任务，我们可以实现更具解释性和可控性的推理，这与人类认知策略中的分而治之原则相符。

从模型能力的视角来看，我们从认知负荷理论中汲取灵感，该理论强调在复杂任务中减少对工作记忆的负担。我们通过定义一组基本的推理技能来实现这一点——这些是模块化的、可重用的推理单元，当需要时模型可以调用。

	SciTab	Our SciAtomic			
		ML.	Mat.	Med.	Fin.
Neg.	411	400	160	364	381
Pos.	457	400	164	318	381
Sum	868	800	324	682	762

Table 2: Scitab 和 SciAtomic 的统计数据。Neg. 表示否定主张。Pos. 表示肯定主张。ML. 表示机器学习。Mat. 表示材料科学。Med. 表示医学科学。Fin. 表示金融。

模型不是整体上处理整个主张，而是逐步进行，在每个阶段只应用相关的技能。这种模块化有助于约束搜索空间，减少对虚假模式的过拟合，并提高对未见过的主张和表格结构的泛化能力。

4.1 技能链模式

我们将技能链模式形式化为：

$$\text{interpretation} \rightarrow \text{planning} \rightarrow [\text{subplan} \rightarrow \text{cell grounding} \rightarrow \text{reasoning} \rightarrow \text{recap}]^N \rightarrow \text{conclusion}$$

在该模式中，模型首先解释整体验证目标，并生成包含 N 个子目标的高层次计划。每个子目标通过一个涉及证据扎根、原子推理和结果总结的局部推理循环进行处理。所有子目标的结果然后被综合成一个最终判决。

- 解释：给定表格 T 和声明 C，模型生成验证任务的解释：

$$I = f_\theta^{\text{interp}}(T, C)$$

- 计划：基于解释，模型生成一组 N 子计划或子目标：

$$P = \{p_1, p_2, \ldots, p_N\} = f_\theta^{\text{plan}}(T, C, I)$$

- 单元格落地：对于每个子计划 p_i，模型将识别表中相关的单元格作为证据：

$$G_i = f_\theta^{\text{ground}}(T, C, p_i)$$

- 推理：模型将适当的原子推理技能应用于基础证据：

$$R_i = f_\theta^{\text{reason}}(T, C, p_i, G_i)$$

- 回顾：推理的结果总结为一个局部结果：

$$U_i = f_\theta^{\text{recap}}(T, C, \{R_j\}_{j \le i}, p_i)$$

- 结论：最后，所有局部回顾被汇总为一个全局决定：

$$Y = f_\theta^{\text{final}}(T, C, \{U_i\}_{i=1}^N), \quad Y \in \{\text{SUPPORT}, \text{REFUTE}\}$$

这种结构化推理流程提供了三个关键好处：(1) 每一步仅关注其局部上下文，减少了来自无关历史的干扰；(2) 使用基本技能避免推理功能的无意缠结；(3) 模块化链条提高了可解释性和稳健性。

Model	Size	SciTab	SciAtomic ML	Material	Medical	Finance
GPT-4o (Intuitive)	7b	0.4951	0.7513	0.6451	0.6246	0.5814
GPT-4o (CoT)	7b	0.5507	0.9025	0.8580	0.8152	0.8570
TAPEX	4b	0.4060	0.5975	0.5264	0.5617	0.5197
TableLLaMa	7b	0.5749	0.6263	0.5337	0.5710	0.5328
TableGPT2	7b	0.6959	0.6750	0.5176	0.5432	0.5249
TableLLM (Text)	7b	0.4215	0.4913	0.4604	0.5000	0.4856
Phi-4	3.8b (base)	0.5184	0.5113	0.5185	0.5166	0.5118
	3.8b (ft)	0.5472â ‘	0.7188â ‘	0.6356â ‘	0.5934â ‘	0.6181â ‘
LLaMA	3.2â “3b (base)	0.5012	0.5038	0.4815	0.4956	0.4843
	3.2â “3b (ft)	0.6106â ‘	0.6688â ‘	0.5740â ‘	0.5176â ‘	0.5276â ‘
	3.1â “8b (base)	0.4724	0.4925	0.4938	0.5059	0.4829
	3.1â “8b (ft)	0.5910â ‘	0.6513â ‘	0.6296â ‘	0.5381â ‘	0.5643â ‘
Qwen-2.5	1.5b (base)	0.5367	0.5675	0.4938	0.5308	0.5210
	1.5b (ft)	0.5933â ‘	0.6663â ‘	0.5617â ‘	0.5484â ‘	0.5866â ‘
	3b (base)	0.5530	0.5163	0.4722	0.4765	0.5827
	3b (ft)	0.5795â ‘	0.7000â ‘	0.5988â ‘	0.5308â ‘	0.6247â ‘
	7b (base)	0.4850	0.4975	0.4691	0.4985	0.4948
	7b (ft)	0.5956 â ‘	0.7100â ‘	0.6821â ‘	0.6202â ‘	0.6562â ‘
	14b (base)	0.6578	0.7550	0.6296	0.6085	0.5932
	14b (ft)	0.7009 â ‘	0.8025â ‘	0.7130â ‘	0.7067â ‘	0.7165â ‘
Deepseek-R1-LLaMA	8b (base)	0.5611	0.6663	0.6049	0.5689	0.5958
	8b (ft)	0.6129â ‘	0.7150â ‘	0.6512â ‘	0.6276â ‘	0.6430â ‘
Deepseek-R1-Qwen	7b (base)	0.5853	0.6300	0.5895	0.5411	0.6312
	7b (ft)	0.5924â ‘	0.8063â ‘	0.7593â ‘	0.7331â ‘	0.8570 â ‘
	14b (base)	0.6560	0.7500	0.6728	0.6818	0.7205
	14b (ft)	0.6613â ‘	0.8200â ‘	0.7653 â ‘	0.7654 â ‘	0.8045â ‘

Table 3: 在 350 个训练样本上微调的 LLMs 与其基础版本及 SOTA LLMs 的性能比较。â ‘表示微调后的增益。评估指标是预测准确率。

4.2 原子技能集

我们方案中的每一步推理都被实现为基本推理技能的组合，使模型能够根据局部任务需求进行针对性的推理：

$$R_i = f_\theta^{\text{reason}}\left(\{s_k\}_{k\in A_i}; T, C, p_i, G_i\right)$$

$A_i \subseteq \{1,\ldots,K\}$ 表示在步骤 i 选择的原子技能的索引集；每个 s_k 代表一个针对特定推理类型而定制的不同推理模块。这些技能作为模块化和可解释推理的基础。通过将关键能力分离为轻量级、可复用的组件，我们允许模型在每个步骤只调用必要的技能，从而减少认知负担并提高精度和泛化能力。这个设计也增强了透明度，因为推理过程可以明确归因于不同的、定义明确的能力。我们定义了如下原子技能集，用以捕获通常用于基于表格的科学主张验证的核心操作：

- 概念理解：解释领域特定语言并将抽象主张与表格语义对齐。
- 结构分析：分析表格的组织结构，包括行列关系和层次化的表头。
- 数值分析：执行诸如比较、聚合和单位归一化等定量操作。
- 因果分析：推断数据模式和趋势所暗示的因果或相关关系。

这种原子设计不仅支持组合推理，还提高了系统在不同表格类型和科学环境中的适应性。通过将复杂的推理分解为更小的、特定技能的步骤，我们的框架促进了更准确、稳健和可解释的验证。

在我们的技能链表述基础上，我们引入了一个包含五个维度的综合评估框架，以评估模型生成的推理链的质量：细粒度、信息冗余、对齐、可解释性和准确性。详细定义位于附录 B.4 。

粒度衡量每个推理步骤的细致程度。精细粒度的推理能够从密集的表格内容中精确提取知识，并支持技能应用中的更高模块化。

信息冗余量化了推理链中多余或无关信息的存在。减少冗余对于提高现实世界科学应用中的推理效率和可扩展性至关重要。

对齐捕捉了相邻推理步骤之间的逻辑连贯性。强对齐确保步骤之间的逻辑进展，促进一致性并减少推理偏差。

可解释性反映了推理过程对人类读者，尤其是非专家的理解程度。在科学领域中，表格具有高信息密度，清晰的推理链对于透明度和可信度至关重要。

准确率评估推理链中每个单独步骤的正确性，从而提供对推理准确性更细致的视角。

我们原子推理链和 GPT-4o 思维链的比较评价结果如图 3 所示。准确性、非冗余性和一致性由 GPT-4o 在三次运行中评估并取均值。粒度和可解释性由三位人工标注者在 0 到 10 的

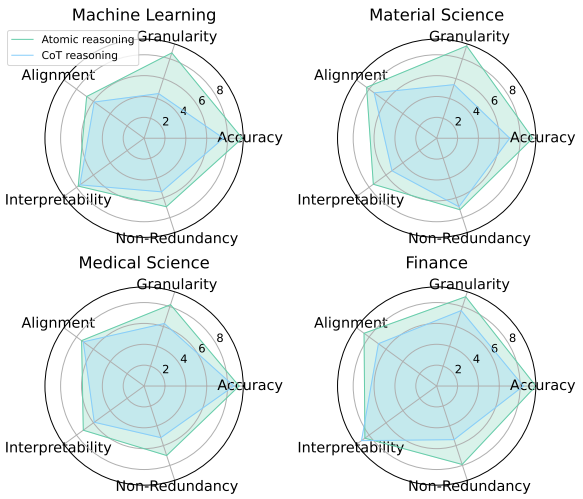

Figure 3: GPT-4o CoT 推理和我们的原子推理链在五个维度上的质量评估。

范围内进行评估，且分数取平均值。

4.3 原子技能分布分析

我们在图 4 中分析了原子技能的分布。尽管存在领域特定的复杂性，大多数主张是通过一小组一致的数值和结构技能解决的。这表明复杂推理可以被分解成具有强泛化能力的紧凑的、可重用的单元。

5 实验

5.1 比较基线

我们将我们的模型与 (1) 现有最先进的闭源 LLM 进行比较，包括 GPT-4 (OpenAI, 2023)，然后将我们的模型与 (2) 表格专用的 LLM 进行比较，包括 TAPEX (Liu et al., 2022)、TableLlama (Zhang et al., 2024a)、TableGPT (Su et al., 2024) 和 TableLLM (Zhang et al., 2024b)。我们还包括 (3) 用于消融的各种 LLM，包括 Phi-4 (Abdin et al., 2024)、Qwen-2.5 (Qwen et al., 2024)、LLaMA (Grattafiori et al., 2024) 和增强推理的 Deepseek-R1 系列模型 (Guo et al., 2025)。

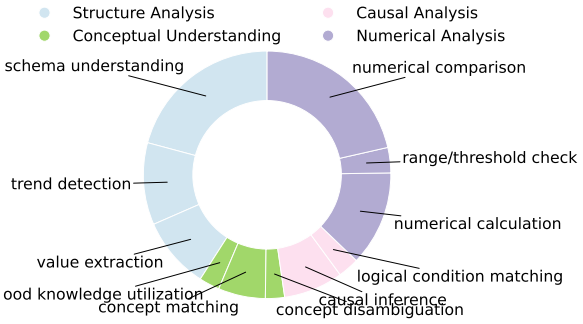

Figure 4: 材料科学中的原子技能分布。

5.2 实现细节

我们使用来自机器学习领域的 350 个训练样本和 50 个验证样本对大型语言模型（LLMs）进行了微调。模型经过 3 个周期的训练，学习率为 1e-5。对于文本生成，我们将温度设置为 0.8，并应用了 k=0.9 的 top-k 采样。

我们通过使用原子的推理链，将基础的大型语言模型与经过微调的对应模型进行比较，来评估原子推理的有效性。我们在 SciAtomic 数据集上将我们的原子大型语言模型与现有的最先进模型进行比较。

在表格 3 中，我们的主要观察结果如下：

（1）即使只有 350 个训练样本，原子推理也能改善所有基础大规模语言模型。这突显了原子监督的高效性，在数据有限的情况下也能带来强大的性能提升。

（2）即便是专门用于推理的大型语言模型，也能显著地从原子监督中受益。这种改进在不同大小和预训练目标的模型中都持续存在，这表明原子监督提供了超出标准思维链提示的补充性归纳偏置。

(3) 原子推理为 LLMs（大语言模型）提供了强大的跨领域泛化能力。在域外评估集上，经过微调的模型始终优于其基础模型，表现出更好的适应能力。

除了评估原子推理带来的性能提升之外，我们还评估了现有最先进模型在我们的 SciAtomic 基准测试上的表现。

（4）现有的最先进的表格声明验证模型在我们的 SciAtomic 基准上表现不佳。尽管这些模型经过大规模事实验证数据集的训练，但在面对 SciAtomic 所需的细粒度、组合推理时表现出明显的性能差距。相比之下，我们经过细调的原子模型在使用 7,429 倍更少的训练样本的情况下超越了 TableLlama。这突显了我们的基准所带来的独特挑战，以及对更具可解释性和忠实推理监督的需求。

5.3 行动中的原子推理：出现、扩展与失败模式

在本节中，我们进一步分析科学表格主张验证中的原子推理，包括新领域中出现的未知技能、高效的训练和推理扩展，以及在 CoT 和原子推理中的错误分析。

新兴技能。 在我们的案例研究中，通过固定的原子技能训练模型进行推理链能够增强跨领域的泛化能力。值得注意的是，在推断过程中，模型不仅应用已知技能，还通过组合新颖或复杂的技能表现出新的行为。例如，在对 350 个来自机器学习领域的原子推理链进行训练后，DeepSeek-R1-Qwen-7B 模型在材料科学领域的

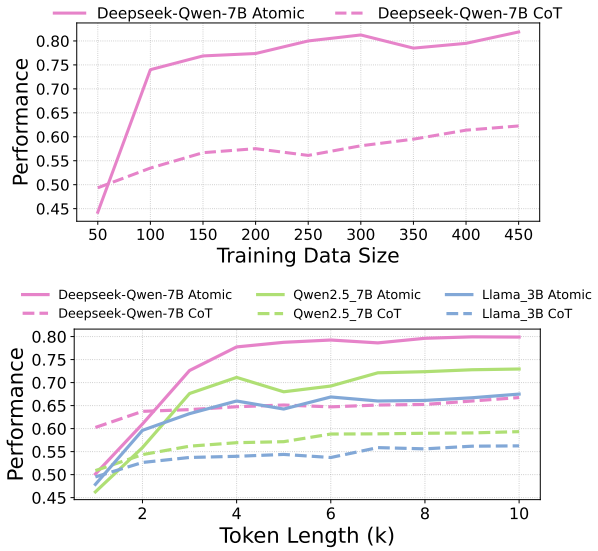

Figure 5: 在机器学习领域中，训练（上）和推理（下）时间的缩放效应。

评估中表现出了新的技能组合。给定一个关于 PI/OFG 纳米复合材料表面特性的表格，模型生成的推理链显示，它正确地识别出了随着 OFG 增加，需要计算 y 值变化率的需求。有趣的是，它还结合了其对表格结构的理解以及对 OFG 进料比相邻性的认识，以执行一种复合技能。此技能结合了值提取、数值计算和模式理解，生成了如下的推理步骤：“计算相邻 OFG 进料比之间 y 的 $(mN/m)^2$ 差异……平均减少是–1.2 %。”

如图 5 所示，我们的技能链模式通过引导模型使用模块化、可重用的原子技能，实现了比 CoT 更高的训练效率。这种结构化的方法在少量示例和更可扩展的监督下实现了更好的泛化能力。

推理时间缩放 我们的技能链结构专门设计用于在推理步骤之间强制逻辑进展，这有助于避免对无关历史的冗余关注，并防止意外的技能激活。这种局部化的依赖性提高了推理效率。如图 5 所示，在推理时间上，大型语言模型在原子推理的扩展上比连锁推理更有效率。

误差分析。 我们进行错误分析，比较连锁思维（CoT）和原子推理在三个类别中的表现：雪球错误、上下文冲突错误和粗粒度错误。

雪球错误：图 6 显示，使用原子推理的 Deepseek-R1 比 GPT-4o 的思维链（CoT）产生的雪球错误更少。这是由于技能链模式的局部化依赖性和模块化技能使用，限制了错误传播。

冲突错误：由于上下文信息与推理出的事实之间的矛盾而产生的上下文冲突，在 CoT 和原子推理中发生的频率相当，这表明我们的方法在保持一致性的同时不牺牲事实的对齐性。

粗粒度错误：由于推理中跳过中间步骤或汇集多个操作而引起的粗粒度错误在我们的原子推理方法中显著减少，这得益于其细粒度和逐步的结构。

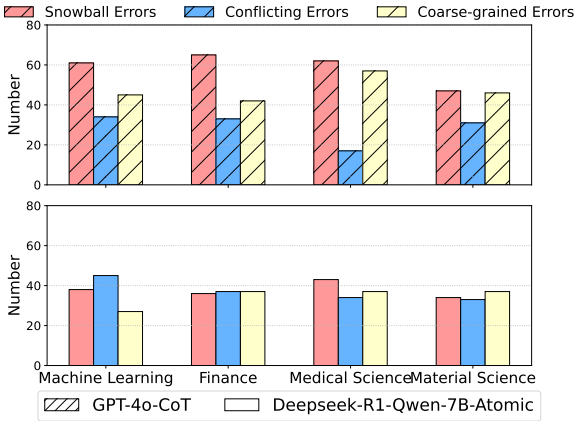

Figure 6: 分别由 DeepSeek-R1-Qwen-7B-Atomic 和 GPT-4o-CoT 对 100 个推理失败样本中的三种推理错误类型进行人工评估。

6 结论

我们提出了原子推理，这是一种数据和规模高效的范式，可以让语言模型学习用于科学表格声明验证的模块化、可组合的推理技能。通过将复杂的验证分解为模块化的原子技能并引入技能链构架，我们的方法提高了跨声明类型和科学领域的泛化能力。我们还发布了 SciAtomicBench，这是一个带有精细推理注释的跨域基准，有助于严格评估。我们的表格专用模型 AtomicTableLLM 实现了最先进的性能，证明了原子推理在提高推理准确性和数据效率方面的有效性。

7 局限性

我们在整篇论文中讨论了各种限制。在这里，我们提供了更多细节。我们的数据收集自用英文编写的开放获取数据集，提供的实例数量受原始来源限制。在未来，我们计划扩展我们的模型，以包含其他语言和领域中的表格-声明对。由于 API 的变化、GPT-4 模型的内在随机性和人工标注，我们的标注数据集可能不易重现，因此我们将发布我们的数据集。此外，有限的计算资源限制了我们在更大模型上进行实验。最后，我们的标注人员是从博士生中招募的，他们的观点可能与其他领域专家的观点有所不同。

8

伦理考虑 在本文中，我们提出了一种基于科学表格验证声明的方法，以确保科学交流中的事实准确性。我们的方法通过只使用 350 个实例

的有限训练集，实现了有效的声明验证，而无需依赖大量的指令微调注释。通过减少对人工标记的依赖，我们的方法在科学 AI 中促进了公平性、可扩展性和包容性，并为大型语言模型（LLMs）在全球社区的广泛普及做出贡献。尽管我们的方法试图减少错误信息，我们的生成结果可能仍然会存在幻觉问题。

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. 2024. Phi-4 technical report. *Computation and Language*, arXiv:2412.08905.

Mubashara Akhtar, Oana Cocarascu, and Elena Simperl. 2022. PubHealthTab: A public health table-based dataset for evidence-based fact checking. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1–16, Seattle, United States. Association for Computational Linguistics.

Roberto Bolli, Joshua M Hare, Keith L March, Carl J Pepine, James T Willerson, Emerson C Perin, Phillip C Yang, Timothy D Henry, Jay H Traverse, Raul D Mitrani, et al. 2018. Rationale and design of the concert-hf trial (combination of mesenchymal and c-kit+ cardiac stem cells as regenerative therapy for heart failure). *Circulation research*, 122(12):1703–1715.

Guillaume Cabanac, Cyril Labbé, and Alexander Magazinov. 2021. Tortured phrases: A dubious writing style emerging in science. evidence of critical issues affecting established journals. *Digital Libraries*, arXiv:2107.06751.

Wenhu Chen. 2023. Large language models are few(1)-shot table reasoners. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1120–1130, Dubrovnik, Croatia. Association for Computational Linguistics.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. 2020. Tabfact: A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations 2020*.

Defne Circi, Ghazal Khalighinejad, Anlan Chen, Bhuwan Dhingra, and L Catherine Brinson. 2024. How well do large language models understand tables in materials science? *Integrating Materials and Manufacturing Innovation*, 13(3):669–687.

Irwin Deng, Kushagra Dixit, Dan Roth, and Vivek Gupta. 2025. Enhancing temporal understanding in LLMs for semi-structured tables. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4936–4955, Albuquerque, New Mexico. Association for Computational Linguistics.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2020. Turl: table understanding through representation learning. In *VLDB Endowment*, volume 14, page 307–319.

Tuan Dinh, Yuchen Zeng, Ruisu Zhang, Ziqian Lin, Michael Gira, Shashank Rajput, Jy yong Sohn, Dimitris Papailiopoulos, and Kangwook Lee. 2022. LIFT: Language-interfaced fine-tuning for non-language machine learning tasks. In *Advances in Neural Information Processing Systems*.

Holly Else. 2021. Tortured phrases' give away fabricated. *Nature*, 596:328–9.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *Computation and Language Repository*, arXiv:2407.21783.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Computation and Language*, arXiv:2501.12948.

Vivek Gupta, Maitrey Mehta, Pegah Nokhiz, and Vivek Srikumar. 2020. INFOTABS: Inference on tables as semi-structured data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2309–2324, Online. Association for Computational Linguistics.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 5549–5581. PMLR.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Kevin Hu, Snehalkumar 'Neil' S. Gaikwad, Madelon Hulsebos, Michiel A. Bakker, Emanuel Zgraggen, César Hidalgo, Tim Kraska, Guoliang Li, Arvind Satyanarayan, and Çağatay Demiralp. 2019. Viznet: Towards a large-scale visualization learning and benchmarking repository. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–12, New York, NY, USA. Association for Computing Machinery.

Kung-Hsiang Huang, Hou Pong Chan, May Fung, Haoyi Qiu, Mingyang Zhou, Shafiq Joty, Shih-Fu Chang, and Heng Ji. 2025. From pixels to insights:

A survey on automatic chart understanding in the era of large foundation models. *IEEE Transactions on Knowledge and Data Engineering*, 37(5):2550–2568.

Kung-Hsiang Huang, Mingyang Zhou, Hou Pong Chan, Yi Fung, Zhenhailong Wang, Lingyu Zhang, Shih-Fu Chang, and Heng Ji. 2024. Do LVLMs understand charts? analyzing and correcting factual errors in chart captioning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 730–749, Bangkok, Thailand. Association for Computational Linguistics.

Hiroshi Iida, Dung Thai, Varun Manjunatha, and Mohit Iyyer. 2021. TABBIE: Pretrained representations of tabular data. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3446–3456, Online. Association for Computational Linguistics.

Hazel Inskip, Georgia Ntani, Leo Westbury, Chiara Di Gravio, Stefania D' Angelo, Camille Parsons, and Janis Baird. 2017. Getting started with tables. *Archives of Public Health*, 75:1–10.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023. StructGPT: A general framework for large language model to reason over structured data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9237–9251, Singapore. Association for Computational Linguistics.

E Kowalczyk. 2017. Partners, brigham and women' s to pay $10 m in research fraud case. *Boston Globe*.

Oliver Lehmberg, Dominique Ritze, Robert Meusel, and Christian Bizer. 2016. A large public corpus of web tables containing time and context metadata. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, page 75–76, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Hongxin Li, Jingran Su, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. 2023. Sheetcopilot: Bringing software productivity to the next level through large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Peng Li, Yeye He, Dror Yashar, Weiwei Cui, Song Ge, Haidong Zhang, Danielle Rifinski Fainman, Dongmei Zhang, and Surajit Chaudhuri. 2024. Table-gpt: Table fine-tuned gpt for diverse table tasks. *Proc. ACM Manag. Data*, 2(3).

Dongwoo Lim, Fujio Toriumi, and Mitsuo Yoshida. 2021. Do you trust experts on twitter? successful correction of covid-19-related misinformation. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 518–523.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2022. TAPEX: Table pre-training via learning a neural SQL executor. In *International Conference on Learning Representations 2022*.

Xinyuan Lu, Liangming Pan, Qian Liu, Preslav Nakov, and Min-Yen Kan. 2023. SCITAB: A challenging benchmark for compositional reasoning and claim verification on scientific tables. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7787–7813, Singapore. Association for Computational Linguistics.

Xinyuan Lu, Liangming Pan, Yubo Ma, Preslav Nakov, and Min-Yen Kan. 2025. TART: An open-source tool-augmented framework for explainable table-based reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4323–4339, Albuquerque, New Mexico. Association for Computational Linguistics.

Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. 2021. Scigen: a dataset for reasoning-aware text generation from scientific tables. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

OpenAI. 2023. Gpt-4 technical report. *Computation and Language Repository*, arXiv:2303.08774.

Jonathan Osborne and Daniel Pimentel. 2023. Science education in an age of misinformation. *Science Education*, 107(3):553–571.

Jan L Plass, Roxana Moreno, and Roland Brünken. 2010. Cognitive load theory.

Team Qwen, Baosong Yang, B Zhang, B Hui, B Zheng, B Yu, Chengpeng Li, D Liu, F Huang, H Wei, et al. 2024. Qwen2 technical report. *Computation and Language*, arXiv:2412.15115.

Mourad Sarrouti, Asma Ben Abacha, Yassine Mrabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brandon Smock, Rohith Pesala, and Robin Abraham. 2022. Pubtables-1m: Towards comprehensive table extraction from unstructured documents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4634–4642.

Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, Haoze Li, Haoxuan Lan, Jiaming Tian, Jing Yuan, Junbo Zhao, Junlin Zhou, Kaizhe Shou, Liangyu Zha, Lin Long, Liyao Li, Pengzuo Wu, Qi Zhang, Qingyi Huang, Saisai Yang, Tao Zhang, Wentao Ye, Wufang Zhu, Xiaomeng Hu, Xijun Gu, Xinjie Sun, Xiang Li, Yuhang Yang, and Zhiqing Xiao. 2024. Tablegpt2: A large multimodal model with tabular data integration.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

Gengyu Wang, Kate Harwood, Lawrence Chillrud, Amith Ananthram, Melanie Subbiah, and Kathleen McKeown. 2023. Check-COVID: Fact-checking COVID-19 news claims with scientific evidence. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 14114–14127, Toronto, Canada. Association for Computational Linguistics.

Nancy X. R. Wang, Diwakar Mahajan, Marina Danilevsky, and Sara Rosenthal. 2021a. SemEval-2021 task 9: Fact verification and evidence finding for tabular data in scientific documents (SEM-TAB-FACTS). In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 317–326, Online. Association for Computational Linguistics.

Qingyun Wang, Xiaoman Pan, Lifu Huang, Boliang Zhang, Zhiying Jiang, Heng Ji, and Kevin Knight. 2018. Describing a knowledge base. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 10–21, Tilburg University, The Netherlands. Association for Computational Linguistics.

Qingyun Wang, Semih Yavuz, Xi Victoria Lin, Heng Ji, and Nazneen Rajani. 2021b. Stage-wise fine-tuning for graph-to-text generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 16–22, Online. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. UnifiedSKG: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 602–631, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8413–8426, Online. Association for Computational Linguistics.

Han Zhang, Yuheng Ma, and Hanfang Yang. 2025. ALTER: Augmentation for large-table-based reasoning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 179–198, Albuquerque, New Mexico. Association for Computational Linguistics.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. 2024a. TableLlama: Towards open large generalist models for tables. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6024–6044, Mexico City, Mexico. Association for Computational Linguistics.

Xiaokang Zhang, Jing Zhang, Zeyao Ma, Yang Li, Bohan Zhang, Guanlin Li, Zijun Yao, Kangli Xu, Jinchang Zhou, Daniel Zhang-Li, et al. 2024b. Tablellm: Enabling tabular data manipulation by llms in real office usage scenarios. *Computation and Language Repository*, arXiv:2403.19318.

Zhehao Zhang, Yan Gao, and Jian-Guang Lou. 2024c. e^5: Zero-shot hierarchical table analysis using augmented LLMs via explain, extract, execute, exhibit and extrapolate. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1244–1258, Mexico City, Mexico. Association for Computational Linguistics.

Bowen Zhao, Changkai Ji, Yuejie Zhang, Wen He, Yingwen Wang, Qing Wang, Rui Feng, and Xiaobo Zhang. 2023. Large language models are complex table parsers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14786–14802, Singapore. Association for Computational Linguistics.

Mingyang Zhou, Yi Fung, Long Chen, Christopher Thomas, Heng Ji, and Shih-Fu Chang. 2023. Enhanced chart understanding via visual language pretraining on plot table pairs. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1314–1326, Toronto, Canada. Association for Computational Linguistics.

A 基准集合详细信息

计算机科学。 我们使用公开可用的 SciGen 数据集 (Moosavi et al., 2021) 构建了我们的 SciAtomic 基准的计算机科学子集，该数据集由从计算机科学领域的 arXiv 论文中提取的科学表格及其相关说明组成。我们从 SciGen 数据集中抽取了 1,376 对表格和说明（我们在 §B.3 中更详细地讨论了数据质量控制）。

财经。 在金融领域，我们从随机抽取的 S P 500 公司中手动整理了合并损益表、资产负债表、股东权益报表和现金流量表的表格。数据最初以截图形式捕获，并使用 GPT-4-Vision 转换为结构化格式。不同于以往专注于小型或简化表格的金融数据集，我们的集合反映了完整企业财务报告的全部复杂性。总共提取并标准化为统一格式的高保真表格有 343 个。

医学科学。 我们从 PubTables-1M 中选择了 1,468 个医学表格 (Smock et al., 2022) 来构建一个医学表格声明数据集。这些表格最初来源于 PubMed Central 开放获取中的科学论文，PubTables 的提取系统确保了高细胞级别的准确性。

材料科学。 材料科学领域的表格编码了丰富的信息，如材料属性和成分，这给科学表格声明验证带来了独特的挑战。然而，这个领域在现有基准中仍然代表不足，这主要是由于获得干净且结构化的表格数据的困难。自动提取方法通常在单元级别的准确性上存在困难，特别是在捕捉数值、测量单位和实验描述符方面。因此，我们结合了由 MatSciTable 发布的 37 个高质量、专家注释的表格，它们专注于聚合物复合材料。为了缓解数据稀疏性并充分利用每个表格的丰富内容，我们为每个表格构建了多个声明，反映了基于领域特定语义的多样化推理轨迹。

B 主张标注细节

B.1 正向声明生成

我们的目标是构建具有挑战性的多步骤论点，以反映真实的科学推理。对于每个表格，我们使用带有 ChatGPT (OpenAI, 2023) 的结构化提示框架，结合多步骤指令。该提示指导模型：

- 识别非平凡模式（例如，极值、趋势、比率）。
- 综合领域知识和来自标题的上下文提示。
- 制定需要至少五个隐含推理步骤的陈述性主张。

为确保清晰性和可验证性，我们明确禁止模糊或主观语言（例如“显著更好”），而是鼓励进行精确比较（例如“比... 高出 12.3 %”）。生成的论点省略中间计算和直接单元格引用，以模拟真实的科学论述。每个生成的论点都经过人工审核，以确保事实正确性、清晰性和语言自然性。

B.2 否定陈述生成

为了构建模拟现实世界科学误导信息的驳斥陈述，我们实施了两种互补的策略：

与其从头开始生成错误的断言，这样容易引入词汇上的人为瑕疵，我们提示 ChatGPT 对真实的断言进行最小化的编辑，以达到逆转其意思的目的。这样保留了原来的句法和结构，使得错误更为细微且更难以察觉。

我们还通过改变关键的定量元素（例如，阈值、数量、单位）来生成被驳斥的主张，这种改变方式是与基础表格数据相矛盾的，同时保持主张的形式。这种方法模拟了有害的错误报告（例如，颠倒安全限值或结果标签），需要模型对表格进行深入推理以检测不一致之处。

这些策略共同产生了一组具有挑战性和现实性的否定论点，从而提高了验证模型的鲁棒性。

B.3 数据质量控制

在科学领域中，严谨性和上下文特异性至关重要。例如，在比较材料性能时，单一的表格可能在不同的实验条件下报告多个指标，省略这些上下文可能会导致歧义和误解。虽然现有的科学表格声明验证基准 (Wang et al., 2021a; Lu et al., 2023) 已经奠定了重要的基础，但它们对需要细粒度、上下文敏感推理的声明验证重视不够。相比之下，我们的基准优先考虑科学严谨声明的构建和验证，仔细关注消除歧义并保留重要的领域特定条件。

解决歧义和范围外问题。 为了解决歧义问题，我们进行了多轮声明重写、隐含引用重写和上下文信息添加。我们还实施了一个不在范围内（OOS）移除过程，以修改 OOS 声明，同时保持可验证性。我们首先识别需要表格数据之外的外部科学背景的声明，这些声明通常包含隐含引用或领域特定术语。对于每个识别出的 OOS 声明，ChatGPT-4 (OpenAI, 2023) 然后执行重写，以消除对外部知识的依赖，同时保留声明的核心意义和与表格的可验证性。

交叉验证。 我们使用多路径一致性来交叉验证长思考、短思考和人类思考过程，从而确保数据的高质量。如果出现不一致，另一名注释者将总结现有的程序并给出最终的标签。

B.4 数据评估原则

- 准确性：我们将逐步准确率定义为所有推理步骤的平均正确性。如果一个步骤 f_θ^{step} 的输出 R_i 被 SOTA LLMs 判断为正确，则认为该步骤是准确的。形式上，

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left[\text{ModelCheck}(R_i)\right]$$

其中 $\text{ModelCheck}(R_i)$ 表示步骤 i 的输出被 SOTA LLMs 标记为正确，N 是步骤的数量。

- 粒度：粒度描述支持准确执行推理步骤所需的上下文 $\mathcal{C}_i$ 的最小且足够的范围，这一特性由人工标注者评分，得分范围为 0-10。这种权衡确保每个推理单元在压缩、集中的输入上操作，同时保持正确性，实现模块化分解而不失真。

- 可解释性：我们定义可解释性为每个推理步骤 f_{step} 能够被人类读者在语义和逻辑上跟随的程度。一个推理步骤被认为是可解释的，如果人类在给定相同的局部上下文 $\mathcal{C}_i$ 的情况下，能够重现或验证该步骤。形式上，我们定义：

$$\text{Interp}_i = \mathbb{I}\left[f_{\text{step}}(T, C, \mathcal{C}_i) = \hat{R}_i^{\text{human}}\right]$$

，其中 $\hat{R}_i^{\text{human}}$ 表示由人类标注者仅基于输入上下文 $\mathcal{C}_i$ 独立产生的推理结果。高可解释性确保模型的推理链可以被审计和信任。

- 信息冗余：我们将信息冗余定义为在局部上下文 $\mathcal{C}_i$ 中存在不影响在步骤 i 进行推理的非必要元素。具体来说，如果一个输入 $x \in \mathcal{C}_i$ 是冗余的，则：

$$f_\theta(T, C, \mathcal{C}_i) = f_\theta(T, C, \mathcal{C}_i \setminus \{x\})$$

我们的数据构建通过强制每个输入标记对于模型的输出都是必要的，明确消除此类冗余：

$$\forall x \in \mathcal{C}_i, \quad f_\theta(T, C, \mathcal{C}_i) \neq f_\theta(T, C, \mathcal{C}_i \setminus \{x\})$$

- 充分对齐：充分对齐确保每个推理步骤 f_{step} 正确地融合并基于前一步的信息构建，在整个推理过程中保持逻辑一致性。正式地，我们将每个步骤的对齐定义如下：对于给定的推理步骤 i，推理结果 R_i 应与累积到步骤 $i-1$ 的知识保持一致，并且不应在上下文中引入任何矛盾或错误。具体来说：

$$\text{Align}_i = \mathbb{I}\left[f_{\text{step}}(T, C, \mathcal{C}_i) = f_{\text{step}}(T, C, \mathcal{C}_{i-1})\right]$$

，其中 $\mathcal{C}_i$ 表示直到步骤 i 为止累积的上下文和信息，而 f_{step} 表示在第 i 阶段的推理步骤。高水平的充分对齐意味着模型的推理步骤与之前的步骤一致，并且在过程中不会丢失或错误更改任何相关信息。如果 $\text{Align}_i = 1$，推理步骤是正确对齐的；否则，它是不对齐的。

C 注释细节

我们邀请了四位具有计算机科学学术背景的志愿者，他们从高年级本科生到研究生不等，作为人工标注人员参与。每位参与者大约花费五小时验证科学论点、评估相关推理步骤，并根据表格数据进行错误分析。参与完全是自愿的，没有提供经济激励。标注者是出于内在动机自愿参与。为了确保标注质量和公平性，我们在任务开始前提供了明确的指导原则和简短的指导课程。在参与者的标注中观察到的一致性支持了所得数据集的可靠性和有效性。

C.1 声明标注

在对主张进行注释时，如果在不同的推理路径分配的标签存在不一致（即，多路径标签不一致），将进行额外的人工审查。在此步骤中，人工注释者会考虑所有从先前注释尝试中获得的信息，并确定一个最终的、统一的标签，以确保数据集的可靠性和准确性。

C.2 技能链评估

技能链评估包括粒度、信息冗余、对齐、可解释性和准确性。其中，准确性、对齐和冗余通过基于 GPT 的评估自动评分。相反，粒度和可解释性由于其主观性需要人工判断。

D 提示的附加细节

Prompt for Claim Generation

SYS_PROMPT = f""" # # # **Task**
You are a helpful assistant to give several claims that could be inferred by the table content. Please follow the steps below to give your answer:
1. Read the table content carefully and try to understand what information is given in the table.
2. Identify more than five key aspects that you can make claims about $trend, maximum, average, inference, etc.$ that are meaningful in the domain. When writing the claim, ensure you incorporate specific knowledge from the field related to the table. Naturally incorporate the domain knowledge into the calculation.
3. Make the claims complex in mathematical calculation but clear in expression. The data in the table must clearly support the claim based on physical principles of the domain or experimental facts, not just superficial correlations.
4. In order to verify the claim, commplex calculations like multi-step complex deduction, sum, trend, multiplication and etc. should be needed.
5. Adjust the claim to be more deterministic, precise, diverse, and complex. Delete vague words like "poorly", "similarly", "substantially", "consistently" and "significantly". Change vague words to comparative metrics like "perform worse than", "same", and "increase" and include specific calculated numbers from the table.
6. Write the scientific claim to make it more natural by integrating the domain knowledge into the numerical trend rather than explicitly stating it. The revision should maintain a formal scientific tone, keep the focus on the numerical relationship, and avoid directly explaining the underlying mechanism. Convey the scientific conclusion implicitly through the data variation
7. The claim should involve complex and challenging calculations, requiring a deep understanding of the table as well as partial knowledge of the domain. Naturally incorporate the domain knowledge into the calculation. It goes beyond simple cell-to-cell operations or comparisons. Include multi-step implicit mathematical calculation in the claim and do not explictly write all the steps. The calculation process must include more than five steps, at most eight steps, and most of the steps need to be include in the implicit calculation. When generating this claim, the intermediate calculation steps should not be written out, and the specific numerical values from the table should not be mentioned. Only the final conclusion should be presented.
8. Do not write claims that need to be verified by locating all the cells in the table. Generate claims that require calculation between several cells in the table. Avoid trivial numerical comparison. Involve complex multi-step implicit computation for the claim.
9. Check the calculation results to be correct, if it is not correct, calculate it again and ensure the final results shown in the claim is correct. The claim should include the final numerical computational result. Write concrete determinstric claim. Avoid speculative sentences like "possibly due to..." or "may..."
# # # **Response Format** Your Response: # # # Understand the Table [thoughts about the table content]
# # # Claim Aspects [more than five aspects of the table content that you can make claims about] [aspect 1], [aspect 2], [aspect 3], ...
# # # Claims Details [one claim about each aspect, each in a separate bullet point] - [aspect 1]: [claim 1] - [aspect 2]: [claim 2] ..."""
USER_PROMPT = """ # # # Table <caption> <table>
Your Response: """

Figure 7: 在生成主张之前进行数据增强提示，并生成积极的主张。

Prompt for Claim Generation

Example
Table Caption Amount of freezable water and non-frozen water in XLPE/silica nanocomposites conditioned at 50°C 100 % th from MDSC measurement (one sample for each material).
Table | material | Freezable water (mg/g) | Non-frozen water (mg/g) | Total water (mg/g) | | XLPE | CellTag | 0.4 | 0.4 | | 5 wt % VS | 1.1 | 2.6 | 3.7 | | 12.5 wt % VS | [BOLD] 5.3 | [BOLD] 7.7 | [BOLD] 13.0 |
Original Claim The non-frozen water content also increases with higher silica content, and at a higher rate compared to freezable water, suggesting that silica's interaction with water molecules predominantly enhances the freezable fraction.
Your Response: # # # Thought The original claim states that non-frozen water content increases with higher silica content, and at a higher rate compared to freezable water. To make the claim not supported by the table, I can alter the rate comparison, suggesting that non-frozen water increases at a slower rate than freezable water, which contradicts the data.
Claim The non-frozen water content also increases with higher silica content, but at a slower rate compared to freezable water, suggesting that silica's interaction with water molecules predominantly enhances the freezable fraction.

Figure 8: 生成否定声明时使用的少量示例

Prompt for Generating Atomic Reasoning Chain

EXAMPLE = ‘interpret’:’The claim is comparing the performance of two models trained using discriminative methods: FINETUNEDDISCRIMINATIVE and CSONLYDISCRIMINATIVE, specifically focusing on their performance on the test set. The claim states that the FINETUNEDDISCRIMINATIVE model is superior to the CSONLY-DISCRIMINATIVE model in terms of test perplexity (test perp), test accuracy (test acc), and test word-error-rate (test wer). Here, "perp" refers to perplexity, "acc" to accuracy (measured in percent), and "wer" to word-error-rate. In general, a lower perplexity and word-error-rate indicate better model performance, while a higher accuracy indicates better performance.’, ’plan’:’[Plan 1 Start]Extract the ́test perpv́alue for the ĆS-only-discḿodel, and the ́test perpv́alue for the F́ine-Tuned-discḿodel, and then compare these two values to verify if the ́test perpv́alue of the F́ine-Tuned-discḿodel is lower than the ́test perpv́alue of the ĆS-only-discḿodel. [Plan 1 End]
[Plan 2 Start]Extract the ́test accv́alue for the ĆS-only-discḿodel, and the ́test accv́alue for the F́ine-Tuned-discḿodel, and then compare these two values to verify if the ́test accv́alue of the F́ine-Tuned-discḿodel is higher than the ́test accv́alue of the ĆS-only-discḿodel. [Plan 2 End]
[Plan 3 Start]Extract the ́test werv́alue for the ĆS-only-discḿodel, and the ́test werv́alue for the F́ine-Tuned-discḿodel, and then compare these two values to verify if the ́test werv́alue of the F́ine-Tuned-discḿodel is lower than the ́test werv́alue of the ĆS-only-discḿodel. [Plan 3 End]’, ’cell’:’To extract the ́test perpv́alue for the ĆS-only-discḿodel, we first locate the row corresponding to ĆS-only-disc. Counting from the first row: 1. Spanish-only-LM row 2. English-only-LM row 3. All:CS-last-LM row 4. All:Shuffled-LM row 5. CS-only-LM row 6. CS-only+vocab-LM row 7. Fine-Tuned-LM row 8. CS-only-disc row. So the ĆS-only-discḿodel is in the 8th row. Next, we locate the column for ́test perp. Counting from the first column: 1. performance column 2. dev perp column 3. dev acc column 4. dev wer column 5. test perp column. So the ́test perpćolumn is the 5th column. Then we locate the 5th column of the ‘CS-only-disc’ row. The cell at the intersection of the 8th row and the 5th column is the ́test perpv́alue for ĆS-only-disc. The value is Í.3. To extract the ́test perpv́alue for the F́ine-Tuned-discḿodel, we first locate the row corresponding to F́ine-Tuned-disc. Counting from the first row: 1.Spanish-only-LM row 2.English-only-LM row 3.All:CS-last-LM row 4.All:Shuffled-LM row 5.CS-only-LM row 6.CS-only+vocab-LM row 7.Fine-Tuned-LM row 8.CS-only-disc row 9.Fine-Tuned-disc row. So the F́ine-Tuned-discḿodel is in the 9th row. The ́test perpćolumn is the 5th column. Then we locate the 5th column of the F́ine-Tuned-discŕow. The cell at the intersection of the 9th row and the 5th column is the ́test perpv́alue for F́ine-Tuned-disc. The value is Ź.8.’, ’extract’: ’The ́test perpv́alue for the ĆS-only-discḿodel is 1.3. The ́test perpv́alue for the F́ine-Tuned-discḿodel is 2.8.’, ’reason’: ’The ́test perpv́alue for the ĆS-only-discḿodel is 1.3. The ́test perpv́alue for the F́ine-Tuned-discḿodel is 2.8. To verify if the ́test perpv́alue of the F́ine-Tuned-discḿodel is lower than the ́test perpv́alue of the ĆS-only-discḿodel, we compare these two values. Comparing 2.8 and 1.3, we find that 2.8 is greater than 1.3. Therefore, the ́test perpv́alue of the F́ine-Tuned-discḿodel (2.8) is not lower than the ́test perpv́alue of the ĆS-only-discḿodel (1.3). This indicates that based on the ́test perpḿetric, the Fine-Tuned-disc model does not outperform the CS-only-disc model. In fact, a lower perplexity indicates better performance, so the CS-only-disc model has a better ́test perpv́alue than the Fine-Tuned-disc model.’, ’recap’:’Based on the comparison of ́test perpv́alues, we found that the F́ine-Tuned-discḿodel has a higher ́test perpv́alue (2.8) than the ĆS-only-discḿodel (1.3), thus failing to support the claim that Fine-Tuned-disc model outperforms CS-only-disc model in terms of test perplexity as stated in Plan <plan_idx>. Since the claim requires outperforming on all metrics and we have already found a contradiction in ́test perp, the overall claim is false. <flag>False</flag>’, ’conclusion’: ’Based on the table, we examined the test perplexity, test accuracy, and test worderrorrate for both the CSONLYDISCRIMINATIVE and FINE-TUNEDDISCRIMINATIVE models. We found that the FINETUNEDDISCRIMINATIVE model has a higher test perplexity (2.8) compared to the CS-ONLY-DISCRIMINATIVE model (1.3), indicating a worse performance in terms of perplexity. While the FINE-TUNED-DISCRIMINATIVE model shows better performance in test accuracy and test word-error-rate, outperforming the CS-ONLY-DISCRIMINATIVE model in these metrics, it fails to outperform in test perplexity. Therefore, the claim that FINE-TUNED-DISCRIMINATIVE modeling outperforms CS-ONLY-DISCRIMINATIVE model on test perplexity, test accuracy, and test word-error-rate is refuted by the table.’

Figure 9: 在原子推理中使用的例子

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘interpret’]= f""" # # # Task (1. Understand the problem) You are a helpful assistant to help me interpret a claim based on a table input. You are given a table and a claim which is based on the table. Now, please interpret the claim based on the content in the table, please follow the guidelines below.
Guidelines Please only interpret the claim but do not give the answer or solution, just make sure you understand what you need to do. Make sure the interpretation is concise and clean. Please solve any ambiguity or reference that may exist in the question, and give your interpretation only based on the caption, table, and claim. Please reason comprehensively and be careful to consider all conditions, constraints and all possible meanings in your problem interpretation.
Example Here is an example interpretation: EXAMPLE[’interpret’] """
USER_PROMPT[’interpret’]= """ # # # Table Content <caption>
<table>
Claim <claim>
Your Interpretation of Claim <interpretation> """

Figure 10: 原子推理链生成：步骤 1

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘plan’] = f""" # # # Task (2. Give a Plan) You are a helpful assistant in giving a step-by-step plan based on the interpretation of a given claim. Your goal is to determine whether the claim is supported, refuted, or cannot be verified (not enough information) based solely on the information provided in the table. Please follow the guidelines below to give the plan.
Guidelines Please list as concrete steps in the plan as possible. Each plan is to verify one subclaim of the whole claim based on the interpretation of the given claim. Make sure each step doesn’t have an overlap. When conducting the planning, you should take into consideration of all the mentioned specialized condition in the claim and make plan to verify all the information of the claim. Frame each plan only in one sentence in a clear, succinct way, avoiding any ambiguous or implicit reference, ensuring each plan including complete procedures for each subclaim verification step. Please wrap your plan in [Plan 1 Start] ···[Plan 1 End], [Plan 2 Start] ···[Plan 2 End] format
Example Here is an example plan: EXAMPLE[’plan’] """
USER_PROMPT[’plan’] = """ # # # Table Content <caption>
<table>
Claim <claim>
Interpretation of Claim <interpretation>
Your Plan <plan> """

Figure 11: 原子推理链生成：步骤 2

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘cell’] = f""" # # # Task (3. Ground the cell with information mentioned in subplan) You are an expert table data extraction assistant. Your task is to precisely locate and extract specific cell values from the provided table based on the instructions given in the subplan. You must strictly follow the subplan and the guidelines provided below.
Guidelines Your broader aim is to extract all the required information mentioned by the subplan. To achieve this, you need to ground each cell with required information. Currently, you are working on the <plan_idx> step of the plan: [Plan <plan_idx>]. Please only try to ground cells and extract data for the designated subplan step by step, DO NOT perform other steps! When locating the cell, you always count from the first cell (head) of the columns or rows to locate the row corresponding to the entity mentioned in the subplan. You count row from the first row, count column from the first column. When locating cells and extracting data from the column or row, indicate the entities you need to locate and extract with sufficient steps. You should output your grounding steps in <grounding>...</grounding> format, and output your sentences of the extracted data of the grounded cells in <extraction>...<extraction> format.
Example Here is an example cell grounding and extraction: <grounding> EXAMPLE[ćell] <grounding>
<extraction> EXAMPLE[’extract’] <extraction> """
USER_PROMPT[’cell’] = f""" # # # Table Content <caption>
<table>
Claim <claim>
Subplan <subplan>
Your Grounding and Extraction <grounding>
<extraction> """

Figure 12: 原子推理链生成：步骤 3

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘reason’] = f""" # # # Task (4. Give a Reasoning with Skills) You are a helpful assistant to use your own knowledge and reasoning to implement a particular step in the plan, in order to verify the claim based on the table. Please follow the guidelines below when performing the reasoning on your subgoal.
Guidelines Your broader aim is to verify whether the claim is supported, refuted based on the content of a table, or cannot be verified (not enough information) based solely on the information provided in the table. To achieve this, you have previously made a plan about how to achieve that goal, and you have the grounded cells containing the required data to verify the subplan. Currently, you are working on the <plan_idx> step of the plan: [Plan <plan_idx>]. Please only try to implement the reasoning of the designated plan step, DO NOT perform other steps! To implement this step in the plan, you should first use the relevant information from the provided extracted cells in the table based on what you need. Then you should do the reasoning based on the extracted cell and data, including comparison, calculation, etc. Please reason carefully, thoroughly, and coherently. Perform each step of reasoning with justification.
Example Here is an example reasoning: EXAMPLE[’reason’] """
USER_PROMPT[’reason’] = f""" # # # Table Content <caption>
<table>
Claim <claim>
Subplan <subplan>
Grounded Cell with Extracted Data <grounding & extraction>
Your Reasoning <reasoning> """

Figure 13: 原子推理链生成：步骤 4

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘recap’] = f""" # # # Task (5. Verify the Reasoning and Refer back to the Plan) You are a helpful assistant in generating a coherent transition sentence to conclude what you have done in the previous reasoning step, refer to the whole plan, and look ahead about what to do next. Please follow the guidelines below to generate the transition sentence.
Guidelines Your generated transition sentence should be coherent with previous reasoning content, and first logically conclude what result you get and whether you have achieved the goal of the subplan. Then, you should refer back to the whole plan, see what you have done, and look ahead to what you should do next. Please generate all these transitions within three sentences, keep your transition coherent, logical, and clear. If you find the subplan is verified to be false by your previous reasoning step, conclude why the subclaim is wrong and do not write the transition to next step, just conclude the whole claim is false since the subclaim is verified to be false, then give an ending flag formatted as <flag>Flase</flag>. If you find the subplan is verified to be true by your previous reasoning step, cleanly conclude how you verify it to be true, and give an ending flag formatted as <flag>True</flag>. If you find that the subplan can not be verified as either true or false with all the existing information, cleanly conclude what you have done for this subplan.
Here is an example transition: EXAMPLE[’recap’] """
USER_PROMPT[’recap’] = f""" # # # Table Content <caption>
<table>
Claim <claim>
All Plans You are trying to verify this claim based on the content from the given table: This is the whole plan of what you should do in order to verify this claim: <plan>
Subplan <subplan>
Reasoning <reasoning>
Your transition <transition> """

Figure 14: 原子推理链生成：步骤 5

Prompt for Generating Atomic Reasoning Chain

SYS_PROMPT[‘conclusion’] = f""" # # # Task (6. Conclude and Get Final Result) You are a helpful agent to conclude the whole reasoning process and give your final response on whether the given claim is supported or refuted by the table or not enough information to verify it based on the information provided by the table. Please give your conclusion following the guidelines below.
Guidelines You should first recap what you have achieved based on all previous reasoning steps. You should consider carefully whether there are outliers or cases that you haven’t taken into consideration, and whether they are important to your final conclusion. After thinking thoroughly over all the information you get from previous reasoning steps and the table information, you should give your final response about whether the claim is supported or refuted by the table, or whether there is not enough information. Ensure all your sentences are based on the true information. Please wrap your final answer in <conclusion></conclusion>. If you find the claim is verified to be false by your previous steps, give an ending flag formatted as <flag>Flase</flag>. If you find the claim is verified to be true by your previous steps, give an ending flag formatted as <flag>True</flag>. If you find the claim can not be verified to false or true merely based on the information provided by the table, give an ending flag formatted as <flag>Not enough information</flag>. You can give only one ending flag in the conclusion as the label for the claim. Please give your conclusion within four sentences.
Here is an example conclusion: EXAMPLE[’conclusion’] """
USER_PROMPT[’conclusion’] = f""" # # # Table Content <caption>
<table>
Claim <claim>
Plans <plan>
Reasoning and Transition <allReasonTransition>
Your Conclusion <conclusion> """

Figure 15: 原子推理链生成：步骤 6