解决时间冲突下的问题回答：使用大型语言模型评估和组织动态知识

Atahan Özer and Çağatay Yıldız

University of Tübingen

Abstract

大型语言模型（LLMs）在问答和推理方面表现出显著的能力，这要归功于其广泛的参数记忆。然而，其知识本质上受到其预训练数据范围的限制，而现实世界信息则在不断变化。更新这种知识通常需要昂贵且不稳定的再训练，或者是情境学习（ICL），考虑到现代信息的数量和波动性，在大规模下变得不切实际。受这些限制的启发，我们研究了 LLMs 在面对时间文本语料库或反映随时间演变的知识的文档时的表现，例如运动员传记，其中像“当前队伍”这样的事实每年都会变化。为此，我们引入了两个新的基准：Temporal Wiki，捕捉历史维基百科快照中的事实漂移，以及 Unified Clark，汇聚带时间戳的新闻文章来模拟现实世界的信息积累。我们的分析显示，LLMs 常常难以调和冲突的或过时的事实，当上下文中出现多版本事实时容易被误导。为解决这些问题，我们提出了一种轻量级的、自主框架，从源文档中逐步构建结构化的外部记忆，而无需再训练。这种知识组织策略使模型能够在推断时检索并推理经过时间筛选的相关信息。经验证明，我们的方法在两个基准上均优于 ICL 和 RAG 基线，尤其是在需要更复杂推理或整合冲突事实的问题上。

1 介绍

世界在不断演变，反映其状态的信息也是如此。大型语言模型（LLMs）可以在其参数记忆中编码世界的一个快照，但只能到达其训练数据和截止日期的限制。当任务需要超出模型训练范围的知识时，主要有两种策略：上下文学习或重新训练。在时间设置中，上下文学习已被证明有效，并被广泛应用于最先进的聊天机器人中，例如，通过集成网页搜索处理涉及训练后事实的查询。相比之下，重新训练属于持续学习的更广泛范畴，它探索 LLMs 如何随着时间的推移逐步吸收新信息。然而，大多数现有方法早于现代 LLM 架构。随着 LLMs 的不断扩大，重新训练和持续适应都变得越来越资源密集。

尽管上下文学习（ICL）在诸如小样本分类 (Brown et al., 2020) 、组合推理 (Wei et al., 2022) 和工具使用 (Schick et al., 2023) 等各种任务中已被探索，但其在时间上下文中的应用仍相对较少被研究。与此相关的是，最近关于知识冲突的研究 (Xie et al., 2024; Kortukov et al., 2024) 探讨了 LLMs 如何处理（过时的）参数记忆与其上下文中的外部信息之间的差异。这些研究表明，当输入是连贯且无歧义时，LLMs 能够与外部事实一致。然而，当支持和冲突证据在同一上下文中出现时，模型，特别是较大型模型，往往依赖其参数化知识。这种行为在时间环境中特别令人担忧，因为过时的事实很常见，可能会导致性能下降。此外，使用 ICL 处理时间上不断发展的知识会引入已知的挑战，包括随着输入长度增加而导致的性能下降 (Li et al., 2024b; Levy et al., 2024; Leng et al., 2024) 、对答案位置的敏感性 (Liu et al., 2024) 以及识别相关上下文的困难 (Levy et al., 2024) 。这些限制促使需要在时间框架中对 ICL 进行深入研究。

尽管在大型语言模型（LLM）中的时序 ICL 是一个新领域，但在 LLM 兴起之前已经有重要的基础工作。例如，Templama Dhingra et al. (2022) 研究了如何在时序事实上训练语言模型，并引入了第一个时序维基百科数据集之一。其他早期工作探索了时间错位，分析了在输入和训练数据的时间上下文发生偏差时，预训练或适应的语言模型的分类准确性如何变化 (Luu et al., 2022) 。此外，关于 LLM 中时序推理的研究检查了模型在时间上定位事件和处理时间依赖的算术的能力，揭示了时序逻辑是 LLM 的一个挑战 Fatemi

et al. (2024); Xiong et al. (2024) 。然而，这些研究大多忽视了我们工作的重点，即不断发展的知识问题，这不仅涉及时序推理，还涉及到新事实的积累和整合。

与我们的工作最接近的是最近两个研究时间性的问题，但视角不同的工作。第一个是 Erase Li et al. (2024a) ，它开发了一种机制来动态构建和编辑知识库，以使大型语言模型的外部知识与新信息对齐。他们引入了 Clark 数据集，我们在实验中也使用了该数据集。第二个是 Growover Ko et al. (2024) ，与我们类似，使用维基百科来模拟不断变化的知识。他们提出了一种名为“检索交互语言模型”的反思方法。我们注意到这两个工作并没有研究任务的时间性方面，即大型语言模型如何处理冲突的知识，或者当一个问题需要一个列表作为答案时（例如，一个篮球运动员效力的球队）。

为此，我们引入了两个新的基准测试，Temporal Wiki 和 Unified Clark，它们模拟了自然语言中知识的演变和积累动态。我们的实验表明，时间数据给大型语言模型带来了挑战，特别是在处理知识漂移和解决记忆冲突方面。为了解决这些问题，我们提出了一种新颖的计算范式，该范式将文档中的外部知识逐步组织成结构化文本。该方法使模型能够在不重新训练的情况下，对时间上连贯的信息进行推理，并在需要适应变化事实的任务中实现持续改进。

接下来，我们描述这项工作的动机，我们引入的数据集，实验的详细信息，以及我们的解代理框架解决这个问题，称为知识组织。

在大型语言模型出现之前，时态问答主要关注通过有监督的训练进行时间自我定位或回答与时间相关的问题。随着情境学习（ICL）的兴起，开放书本问答（尤其是检索增强的方法）受到了越来越多的关注。在许多现实场景中，检索到的文档包含非结构化和时间上变化的信息。例如，回答诸如“勒布朗 · 詹姆斯曾在哪些球队效力过?”这样的问题需要汇集跨越不同时期的来源证据，其中由于时间的推移，多个文档可能会提到不同的球队。我们通过实验证明，答案随着时间演变的问题对大型语言模型提出了重大挑战。本文研究大型语言模型在处理这些时间动态的真实世界条件下的表现。

1.1 数据集

我们介绍了两个受此前研究启发的新基准。第一个，Temporal Wiki，基于 Templama 数据集 (Dhingra et al., 2022) ，旨在研究维基百科中的知识如何随着时间演变以及大语言模型（LLM）如何与这种不断演变的信息交互。第二个，Unified Clark，扩展了 ERASE 数据集 (Li et al., 2024a) ，以评估 LLM 在使用新闻数据的真实世界时间信息检索场景中的表现。在接下来的小节中，我们描述了这些基准的构建、处理和预期用途，以评估 LLM 的时间理解能力。Temporal Wiki 数据集可在 https://github.com/atahanoezer/TQA 获得，而 Unified Clark 数据集可通过 (Li et al., 2024a) 访问。

1.1.1 时间维基

Templama 是一个由维基百科中提取的时间敏感的 (Subject, Relation, Object) 三元组组成的数据集（例如，(勒布朗 · 詹姆斯，效力于，洛杉矶湖人)）。其最初目的是解决语言模型在由于训练数据截止日期而导致的过时知识方面的局限性，并探索在预训练期间整合新的时间信息的技术。在这项工作中，我们重新利用 Templama，通过其与维基百科的直接联系：针对每个三元组，我们收集该主体维基百科页面的多个历史快照，从而使我们能够在从这些快照中得出的问答文档对中模拟知识积累和时间漂移。

我们的数据整理流程从构建问题-文档对开始，其中文档包含对生成的时间依赖问题的答案。我们首先将每个 Templama 三元组转换为一个时间性问题，例如，“2014 年勒布朗 · 詹姆斯在哪支球队效力?”针对每个主题，我们使用 Wikidata API 下载其维基百科页面的所有可用快照。然后，每个问题都与查询时间后的年份中最早的快照配对（例如，2014 年问题对应 2015 年 1 月的快照），以确保相关信息可能已被添加到页面上。如果目标年份的快照不可用，我们退而求其次，选择最接近且在问题时间戳之后的可用版本。

为了确保数据质量，我们应用以下后处理步骤：

- 去除结构化知识：维基百科常包含结构化知识元素，如表格和项目符号列表，这些元素以一种不代表自然非结构化文本的方式浓缩信息。使用正则表达式去除这些元素以更好地模拟真实世界的信息检索条件。

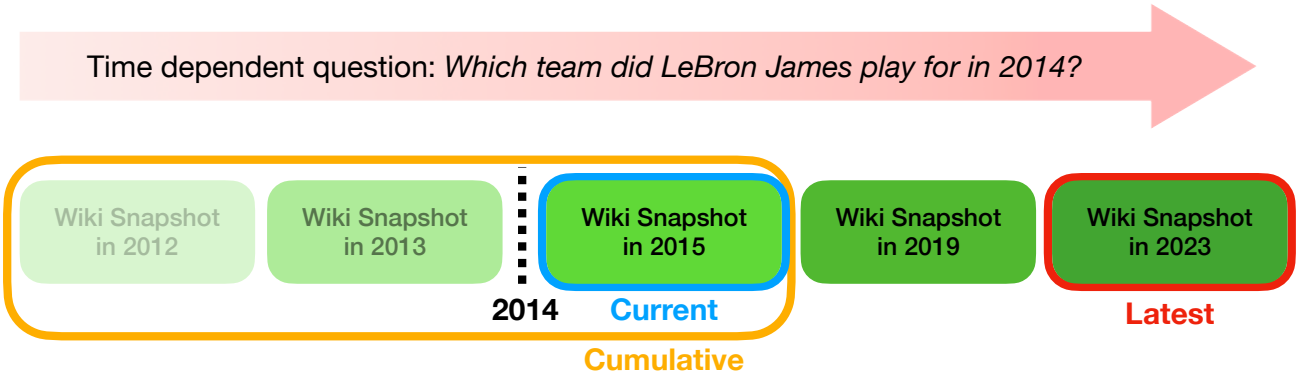

Figure 1: 来自我们的 Temporal Wiki 数据集的一个示例问题以及不同的上下文证据。这个问题可以追溯到 2014 年。我们展示了最近的（问题发生年份最近的快照）、累积的（包含所有到问题年份信息的快照）、以及最新的快照（可用的最新快照）。

- 字符标准化：原始的维基百科快照经常包含非人类可读的伪影，例如超链接、标记和 ASCII 转义序列。我们使用基于正则表达式的预处理从文本中去除这些元素，以生成适合于 LLM 输入的干净、可读的文档。
- 证据答案检查：人工检查显示，并非所有配对文档都包含对相应问题的答案。为了缓解这一问题，我们采用了两步过滤过程。首先，基于规则的答案检查器会扫描参考文档，以检测与预期答案在表面层次上的匹配。其次，我们使用 GPT-4o 进行语义验证步骤，以确认参考文档是否确实支持该答案。

接下来我们描述在实验中形成上下文证据的三种不同方式。我们定义事实 $\boldsymbol{F_i}$ 包含年度事件问题 $\boldsymbol{Q_{i,t}}$ 和年度依赖的参考文本 $\boldsymbol{R_{i,t}}$ ，其中 $i \in \{1, \ldots, N\}$ 和 $t \in \{1, \ldots, T\}$ 。我们考虑三类上下文证据（参见图 1 的视觉示例）：

- 最接近的快照对应于对 $(\boldsymbol{R_{i,t}}, \boldsymbol{Q_{i,t}})$ ，其中参考文本在时间上与问题对齐。此设置确保正确答案在文本中，同时之前的答案也可能被提到。
- 最新的快照使用 $(\boldsymbol{R_{i,T}}, \boldsymbol{Q_{i,t}})$ ，其中将最新可用的参考文本与问题配对。这种设置模拟了比上述更复杂的知识库。
- 累积快照考虑了 $(\boldsymbol{R_{i,1:t}}, \boldsymbol{Q_{i,t}})$ ，它汇总了从第一年到当前年份 t 的所有参考文本。此设置解决了信息从维基百科中删除的情况。因此，通过这种方法，在文本变长的同时，我们实现了最高程度的信息覆盖。

1.1.2 统一克拉克

维基百科页面随着时间的推移不断累积更新，这意味着一个特定年份的快照（例如，勒布朗 · 詹姆斯的 2015 年维基百科页面）通常包含最新答案和所有先前问题的答案。相比之下，Clark 数据集由带有时间戳的新闻文章组成，其中每篇文章通常只反映发表时的当前答案。因此，当单独使用时，单个新闻文章使 ICL 对 LLM 来说变得简单，因为不存在矛盾的信息。

为了引入一个复杂性水平，与我们的时间维基数据集相当，我们将给定实体的所有可用新闻文章（及其时间戳）连接成一个单一文档。我们将这个扩展版本称为统一本。除此之外，因为原始数据集中的许多文章并未明确强调时间细节，我们设计问题以需要全面的答案，涵盖多个时间段（如，“勒布朗 · 詹姆斯曾效力于哪些球队?”）。我们使用此数据集的主要目标是研究记忆冲突：在上下文中同时出现多个答案，有时这些答案相互冲突，给大型语言模型带来了更大挑战的情境。

1.2 实验细节

我们使用内部搭建的模型进行实验：Llama 3.1 70B (Grattafiori et al., 2024) 、Llama 3 8B (Grattafiori et al., 2024) 和 Mistral 7B Instruct v2 (Jiang et al., 2023) 。为了评估目的，我们另外使用了 Qwen-2 72B (Chu et al., 2024) 。7B 和 8B 模型部署在单个 NVIDIA A100 GPU 上，而 70B+ 模型使用 4 位量化分布在三个 A100 GPU 上。所有模型均使用 Hugging Face Transformers 库 (Wolf et al., 2020) 服务，文本分词由每个模型的原生字节对编码 (BPE) 分词器处理。量化推理通过 bitsandbytes 库 (Dettmers et al., 2022) 启用。

提示 最近的研究表明，在 LLM 推理过程中使用的提示策略起着关键作用，这与模型参数记忆的质量以及所提供上下文的相关性一样重要 (Wei et al., 2024; Madaan et al., 2023; Zhang et al., 2024) 。在我们的实验中，我们采用了 Zhang et al. (2024) 提出的元提示方法，该方法将少量的示例与明确的、特定任务的指令相结合。这种策略能够在不同环境中一致地提高 LLM 的性能。我们在附录中提供了一个代表性的提示模板示例。

我们使用 LangChain 库中的“递归字符文本分割器”将文档分成 500 个字符的块，块之间有 50 个字符的重叠。这些块被编码到 512 维的嵌入空间。在推理时，我们通过计算查询嵌入与所有文档嵌入之间的余弦相似度来检索最相关的 12 个块。请注意，我们在详尽的消融搜索后获得了这些超参数。

我们的基准测试需要通过人工标注的标准答案来比较开放式模型输出。尽管传统的字符串匹配度量如精确匹配（EM）和标记级别的 F1 得分被广泛使用，但它们常常无法捕捉语义等价性。最可靠的评估方法之一是通过交叉检查进行人工评估；然而，该方法代价高昂，可能引入额外的偏见。幸运的是，正如最近的研究所示，最先进的大型语言模型（LLM）在评估任务中能够达到人类水平的表现，其中评估一致性超过 80%，可与人工评估员相媲美。受这些发现的启发，我们在实验中采用基于 LLM 的评估器。为了减轻个别模型的偏差，我们使用 Llama 3.1 70B 和 Qwen-2 72B 之间的一致性投票方案。关于评估设置和可靠性分析的更多细节在附录中提供。

最后，我们引入了一种简单但有效的代理解决方案，以应对基于时间演变知识的问题回答挑战。受到人类如何获取和组织知识的启发，我们的方法从证据文档中构建了一个动态知识库，并在这个结构化的知识库上进行推理。此设置通过确保模型仅在相关的、结构化的知识上进行推理，而不是依赖于大而未过滤的上下文窗口，从而促进精度的提升。请参见图 2 以获取我们方法的总结。

我们方法的工作流程分为三个阶段。首先，给定一个输入问题和相关的证据文档，智能体使用上下文学习将问题转换为 (Subject, Relation, Object, Timestamp) 形式的结构化四元组。在第二阶段，智能体从文档中提取与同一主题相关的所有相关事实，再次使用上下文学习。这些提取的事实随后作为结构化条目存储在智能体的知识库中，并通过主题和关系进行索引。在最后阶段，在回答问题时，智能体查询其知识

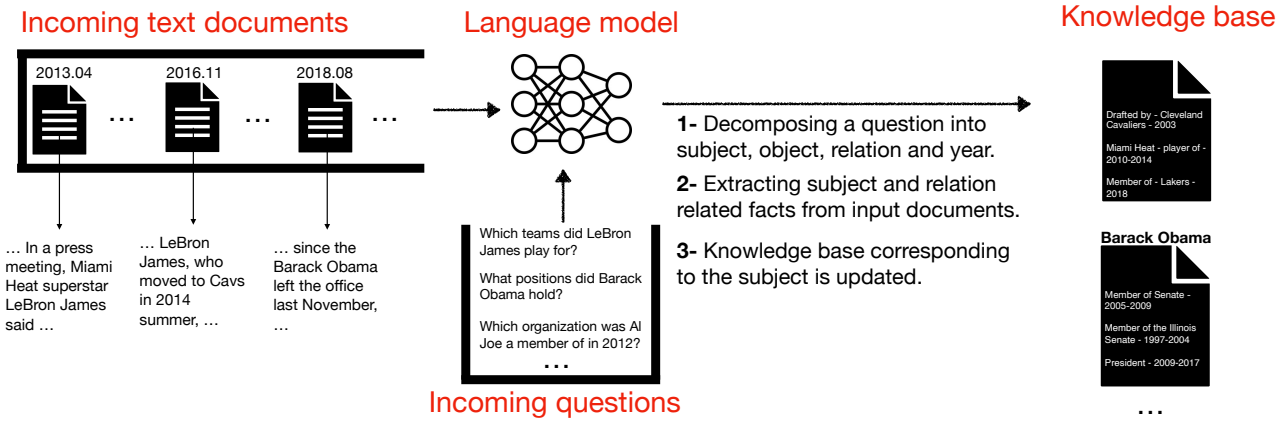

Figure 2: 对我们主动工作流程的视觉总结，其中传入的文本文档根据从问题中提取的主体、客体和关系转化为结构化知识库中的条目。

库，以获取与问题中的主题和关系匹配的所有条目。然后，它仅使用检索到的条目来构建答案。

我们在 Temporal Wiki 和 Unified Clark 数据集上展示了我们的实验结果，评估模型在四种推理方案下的表现：零样本（ZS）、上下文学习（ICL）、检索增强生成（RAG）和知识组织（KO）。在 ZS 设置中，模型在没有访问任何支持文档的情况下回答每个问题，完全依赖其参数化记忆。相反，ICL、RAG 和 KO 设置提供外部文档作为输入，每种方法以不同的方式利用上下文信息来提高性能。

我们现在展示关于 Temporal Wiki 基准的研究结果。为了更好地理解模型行为，我们将问题分为两类，基于它们在没有外部证据的情况下是否被正确回答。表 1 报告了引入外部上下文时的表现，即模型的准确率由于输入中存在冲突或分散注意力的信息而下降的情况。相反，表 2 显示了在零样本设置中最初回答错误但在加入支持性上下文后得到改进的问题的准确率。为了进一步分析信息质量的影响，我们在图 3 和 4 中进行了消融研究，分别着重于回答一个问题的答案随着时间变化的次数和文档长度的影响。

Table 1: 在通过零样本得到正确答案的问题上计算的准确率。我们观察到，KO 对准确率的下降最小。

Model	ICL		RAG			Knowledge Org.
	Closest	Latest	Closest	Latest	Cumulative	
Llama-3.1 70B	0.83	0.76	0.92	0.90	0.90	0.90
Llama-3 8B	0.85	0.78	0.86	0.82	0.83	0.92
Mistral 7B v2	0.80	0.76	0.86	0.81	0.81	0.91

Table 2: 计算在零样本答案错误的问题上的准确性。结果表明，KO 在回答问题时最有效地利用了外部输入，其次是 RAG。

Model	ICL		RAG			Knowledge Org.
	Closest	Latest	Closest	Latest	Cumulative	
Llama-3.1 70B	0.65	0.62	0.78	0.77	0.74	0.81
Llama-3 8B	0.69	0.61	0.73	0.63	0.66	0.83
Mistral 7B v2	0.70	0.60	0.72	0.68	0.68	0.82

我们从在这两个问题类别中一致成立的观察开始：

1.2.1 特定设置的观察

我们现在重点说明针对单个实验设置的具体观察：

- 我们研究了不同事实变化频率下的零样本表现，如图 3 所示。有趣的是，零样本准确性往往随着事实变化频率的增加而提高，即随着给定问题的答案随时间的变化而更频繁。我们假设，这些频繁更新的页面更有可能出现在 LLM 的训练数据中，从而在没有外部上下文的情况下提高了性能。
- 我们还分析了参考文档长度对 ICL 设置的影响，这已知对输入长度较为敏感。如图 4 所示，没有出现一致的趋势。虽然准确性在不同文档长度之间波动，但这些变化似乎与文档大小并没有强烈的相关性。

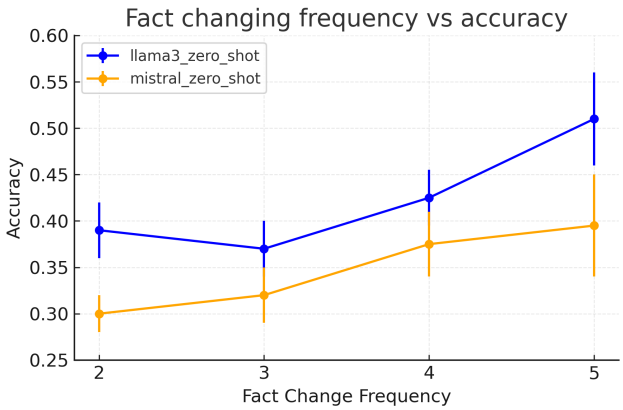

Figure 3: 准确性与一个问题的答案随时间变化的次数的关系图。

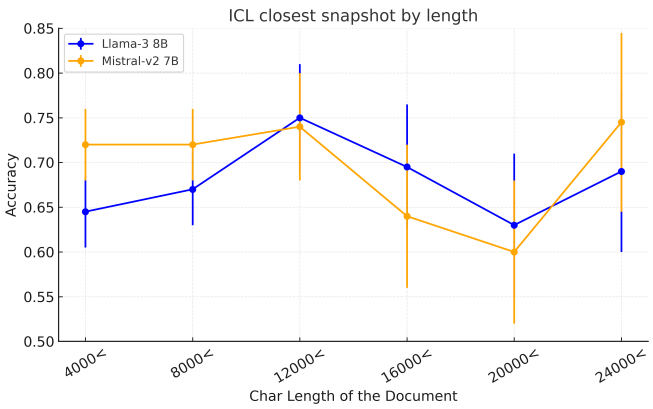

Figure 4: 准确性与文档长度的关系图中，我们没有观察到一致的模式。

总之，我们的研究结果表明：(i) 时间上下文引入了非平凡的挑战，其中模型即使在没有外部信息的情况下成功运行，也可能无法正确回答；(ii) 知识组织产生了最佳性能，其次是 RAG，这表明将复杂且时间上相互矛盾的信息提炼成结构化、简洁的表示可以大幅提高准确性。

1.3 统一的克拉克实验

统一克拉克实验旨在评估与时间维基基准不同的能力。具体来说，它测试模型在一个密集积累的参考文本内识别一个事实的所有时间依赖实例的能力。由于输入文档通常超过 8000 个标记，我们将此实验限制在具有扩展上下文长度的模型上。此外，我们在此设置中包括 GPT-4-o Mini 以进行额外的比较。为了确保所有答案都以提供的文档为基础，我们在此实验中默认禁用参数记忆的访问。换句话说，模型必须仅依赖参考文本以获得准确的响应。

结果如表 3 所示。在时间百科设置中，知识组织方法优于所有基线，其次是 RAG，然后是 ICL。然而，这里的性能差距更为显著，我们将其归因于该任务的更高难度：模型必须生成完整的一组时间敏感答案，而不是点预测。最后，我们省略了在 ICL 下的 Llama 3 8B 的结果，因为其有限的上下文窗口无法容纳完整的参考文档。

Table 3: 在统一 Clark 数据集上的结果。

Model	ICL	RAG	KO
Gpt4-o Mini	0.47	0.48	0.69
Llama 3.1 70B	0.43	0.68	0.76
Qwen 3.1 70B	0.39	0.43	0.71
Llama-3 8B	N/A	0.39	0.69

在这项工作中，我们探讨了大型语言模型在时间不断发展的知识问答任务中的表现。我们引入了两个新的基准，Temporal Wiki 和 Unified Clark，以评估模型在事实随时间积累、转变或冲突的情况下的鲁棒性。我们的结果显示，尽管上下文学习（ICL）和检索增强生成（RAG）在过时或冲突信息方面存在困难，RAG 通过过滤掉无关上下文始终优于 ICL。然而，这两种方法均容易受到时间噪声的影响，并可能在表现上不及零样本基线。我们提出的知识组织框架通过逐步构建和索引事实来解决这些挑战，从而能够在动态

信息上进行更可靠的推理。这些发现强调了将结构化时间记忆集成到未来大型语言模型系统中的重要性。

致谢 Çağatay Yıldız 是机器学习卓越集群的成员，该集群由德国研究基金会（DFG）根据德国的卓越战略–EXC 编号 2064/1 –项目编号 390727645 资助。本研究利用了图宾根机器学习云的计算资源，DFG FKZ INST 37/1057-1 FUGG。

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. Qwen2-audio technical report. arXiv preprint arXiv:2407.10759, 2024.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization. 9th International Conference on Learning Representations, ICLR, 2022.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. Time-aware language models as temporal knowledge bases. Transactions of the Association for Computational Linguistics, 10:257–273, 2022. doi: 10.1162/tacl_a_00459. URL https://aclanthology.org/2022.tacl-1.15.

Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of time: A benchmark for evaluating llms on temporal reasoning, 2024. URL https://arxiv.org/abs/2406.09170.

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

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. arXiv preprint arXiv:2310.06825, 2023.

Dayoon Ko, Jinyoung Kim, Hahyeon Choi, and Gunhee Kim. Growover: How can llms adapt to growing real-world knowledge?, 2024. URL https://arxiv.org/abs/2406.05606.

Evgenii Kortukov, Alexander Rubinstein, Elisa Nguyen, and Seong Joon Oh. Studying large language model behaviors under realistic knowledge conflicts, 2024. URL https://arxiv.org/abs/2404.16032.

Quinn Leng, Jacob Portes, Sam Havens, Matei Zaharia, and Michael Carbin. Long context and rag performance in large language models. https://www.databricks.com/blog/long-context-rag-performance-llms, August 2024. Mosaic AI Research, Databricks, Accessed: 2024-09-02.

Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 15339–15353, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.818.

Belinda Z. Li, Emmy Liu, Alexis Ross, Abbas Zeitoun, Graham Neubig, and Jacob Andreas. Language modeling with editable external knowledge, 2024a. URL https://arxiv.org/abs/2406.11830.

Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. Long-context llms struggle with long in-context learning, 2024b. URL https://arxiv.org/abs/2404.02060.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. Transactions of the Association for Computational Linguistics, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. Time waits for no one! analysis and challenges of temporal misalignment. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (eds.), Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5944–5958, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.435. URL https://aclanthology.org/2022.naacl-main.435.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In Thirty-seventh Conference on Neural Information Processing Systems, 2023. URL https://openreview.net/forum?id=S37hOerQLB.

Timo Schick, Arun Tejasvi Chaganty Dwivedi-Yu, Seyed Iman Hosseini, Peter Sorensen, Teven Le Scao, Christine Akiki, Nouha Tazi, Samia Touileb, Ellie Pavlick, and Kyunghyun Cho. Toolformer: Language models can teach themselves to use tools. arXiv preprint arXiv:2302.04761, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. arXiv preprint arXiv:2201.11903, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9781713871088.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen (eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In The Twelfth International Conference on Learning Representations, 2024. URL https://openreview.net/forum?id=auKAUJZMO6.

Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. Large language models can learn temporal reasoning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), Proceedings of the

62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 10452–10470, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.563.

Yifan Zhang, Yang Yuan, and Andrew Chi-Chih Yao. Meta prompting for ai systems, 2024. URL https://arxiv.org/abs/2311.11482.

A 附录

您可以在此处包含其他附加部分。

B

时间维基数据格式

该数据集以 JSON 格式构建。数据集中的每个条目代表一个特定事件，并包含与该事件在不同时点相关的历史数据。主要结构是一个 JSON 对象，具有两个主要键：event_id 和 incidents 。

event_id：用作事件唯一标识符的整数。

incidents：一个包含嵌套对象的 JSON 对象，每个键是一个四位数字字符串，代表一个年份。每个特定年份的对象提供该年份事件相关信息的快照，并包含以下字段：

q_year：表示问题提出年份的整数。

map_year：表示转储中的信息所属年份的整数，用于找到正确答案。

question：包含该特定年份关于事件的自然语言问题的字符串。

answer：JSON 对象的列表，其中每个对象表示一个正确答案。每个答案对象包含：

- name：包含实体名称的字符串。
- wikidata_id：一个表示实体唯一 Wikidata 标识符的字符串。

dump：包含该年检索的原始数据的 JSON 对象。其中包括：

- url：数据来源的维基百科页面的 URL。
- body_par：包含维基百科文章正文内容的字符串。
- infobox：一个 JSON 对象，表示从文章的信息框中提取的键值对。

ans_comp , llm_resp：这些字段保留供将来使用，目前设置为 null 。

数据结构的一个例子如下所示：


```
{
  ”event_id”: 6,
  ”incidents”: {
    ”2010”: {
      ”q_year”: 2010,
      ”map_year”: 2011,
      ”question”: ”Question: Which team did Luka Modrić play for in 2010?”,
      ”answer”: [
        {
          ”name”: ”Tottenham Hotspur F.C.”,
          ”wikidata_id”: ”Q18741”
        }
      ],
      ”dump”: {
        ”url”: ”https://en.wikipedia.org/w/index.php?title=Luka_Modrić”,
        ”body_par”: ”Luka Modrić (pronounced ,
        born 9 September 1985 in Zadar)...”,
        ”infobox”: {
          ”Full name”: ”Luka Modrić[1]”,
          ”2008–”: ”Tottenham Hotspur”,
```

```
                ...
            }
        },
        "ans_comp": null,
        "llm_resp": null
    },
    "2013": {
        ...
    }
  }
}
```