MMTS-BENCH：用于时间序列理解与推理的综合基准

Yao Yin*清华大学 北京，中国
yiny23@mails.tsinghua.edu.cn

Zhenyu Xiao*清华大学 北京，中国
xzy23@mails.tsinghua.edu.cn

Musheng Li 清华大学 北京，中国
lms21@mails.tsinghua.edu.cn

Yiwen Liu 清华大学 北京，中国
liu-yw23@mails.tsinghua.edu.cn

Sutong Nan 清华大学 北京，中国
nst24@mails.tsinghua.edu.cn

Yiting He 清华大学 中国北京
he-yt24@mails.tsinghua.edu.cn

Ruiqi Wang 清华大学 中国北京
wang-rq23@mails.tsinghua.edu.cn

Zhenwei Zhang 清华大学 中国北京
thuzhangzw@outlook.com

Qingmin Liao 清华大学 中国北京
liaoqm@tsinghua.edu.cn

Yuantao Gu†清华大学 中国北京
gyt@tsinghua.edu.cn

arXiv:2602.08588v1 [cs.DB] 9 Feb 2026

摘要

时间序列数据是金融、医疗保健和云计算等领域的核心，但现有用于评估各类大语言模型（LLM）时序任务表现的基准仍然零散且缺乏系统性。为弥补这一空白，我们推出MMTS-BENCH，这是一个全面的多模态基准测试，基于时间序列任务的层级分类体系构建，涵盖结构感知、特征分析、时间推理、序列匹配和跨模态对齐。MMTS-BENCH 包含2,424 组成时间序列问答（TSQA）对，分属 4 个子集：**Base**、**InWild**、**Match** 和 **Align**，这些数据通过渐进式真实世界问答框架和模块化合成数据构建生成。我们对闭源、开源大语言模型以及现有的时间序列适配大语言模型（TS-LLM）进行了广泛评估，发现：（1）TS-LLM 的跨领域泛化能力显著落后于通用大语言模型；（2）与全局任务相比，大语言模型在局部任务上的表现较弱；（3）思维链（CoT）推理和多模态整合能显著提升性能；（4）现有 TS-LLM 的主导因素仍是骨干网络的能力，而非时间序列编码器的设计。MMTS-BENCH 不仅提供了严格的评估框架，也为推动大语言模型实现鲁棒、可解释且可泛化的时间序列推理指明了方向。[1]

CCS 概念

• **计算方法** → **自然语言处理**；**神经网络**；　• **综合与参考** → **评估**。

*两位作者对本研究的贡献相同。
†通讯作者[1]代码和数据可在 https://anonymous.4open.science/r/MMTS-BENCH- BEF7/ 获取

关键词

时间序列基准、多模态时间序列理解、时间推理、大语言模型

1 引言

时间序列数据支撑着金融、医疗保健、交通运输和云计算等领域的关键系统 [24, 46, 50], ，记录着各类过程随时间的演变。传统上，预测、分类、异常检测和插补等任务 [27, 47, 49] 依赖专门的统计模型和工具，需要大量领域专业知识。近年来，随着自然语言处理（NLP）的快速发展，尤其是大语言模型（LLM）的突破 [1, 6, 28, 34, 44], ，时间序列分析领域出现了克服专业门槛的新机遇 [18, 35, 37, 42]。将时间序列数据与大语言模型相结合，构建端到端时间序列模型，已成为一个重要的研究方向 [3, 23]。近期，越来越多的研究者开始探索大语言模型在时间序列分析中的应用，由此催生了时间序列描述 [48]、文本上下文辅助预测 [17]、简单时间序列问答 [35]、复杂时间序列推理以及跨变量问答 [42]等新任务。

上述将大语言模型与时间序列数据相结合的工作同样需要大量训练和测试数据集的支持。以往的工作要么通过文本注释丰富传统时间序列数据集 [24, 45]，要么构建用于时间序列推理的问答（QA）数据集 [20, 37]。然而，目前大多数研究依赖扁平的任务分类体系[2] [4, 35]来定义能力并合成

[2]这里，扁平 表示将预测、异常检测、特征分析和高层时间推理等任务视为并行且相互独立的任务，不存在明确的层级关系。

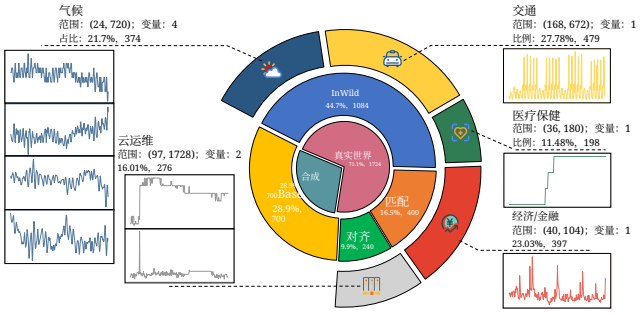


图1：MMTS-BENCH 概览。展示各子集和来源领域中的基准构成，以及每个组成部分中的问答实例数量。

问答数据。此类分类体系要么没有层次化结构，要么结构过于简单，难以在细粒度层面全面评估大语言模型在时间序列理解和推理方面的能力。此外，大多数时间序列数据集的构建以及大语言模型的微调都局限于单一或少数领域的数据[9, 37],，而目前仍缺乏一个覆盖多个领域、用于评估大语言模型分布外（OOD）泛化能力的综合基准。更广泛地说，尽管相关领域发展迅速，但对于哪些模型设计选择真正影响有效的时间推理和跨模态对齐，仍缺乏明确认识，这凸显了系统化评估框架的必要性。

受多模态基准中的层次化能力建模 [25]和组合泛化理论 [16], 启发，我们提出了时间序列任务的层级分类体系，将分析任务重新组织为多层级、组合式结构，从基础感知延伸至高级推理；在这一结构中，复杂能力通过系统性地组合低层级任务基元而形成。与以往扁平的任务分类体系相比，我们的层级结构明确建模了任务之间的依赖关系及其渐进式整合过程，从而阐明并拓展了此前被忽视的若干关键任务领域。

基于这一分类体系，我们构建了 MMTS-Bench，这是一个面向时间序列任务的新型多模态、多维度评估基准（见图 1），包含来自四个子集的 2,424 对时间序列问答对。其中一个子集基于合成时间序列数据构建：**（1）Base**，用于评估结构感知和特征分析能力。其余子集基于 LOTSA 数据集[40]中涵盖五个领域（例如交通领域；见附录 A.2）的真实世界时间序列构建：**（2）InWild**，侧重于特征分析和时间推理；**（3）匹配**，用于评估序列相似性匹配和形态对应；以及**（4）对齐**，用于衡量时间序列与自然语言之间的双向转换能力，以及高级跨模态语义理解能力。我们进一步使用 MMTS-Bench 对多种主流大语言模型开展了全面评估，提供细粒度的能力排名以及对当前优势和局限性的洞见，从而为未来时间序列基础模型和基准测试的开发提供参考。我们的主要贡献如下：

(1) **MMTS-Bench**。我们提出了一种以能力为导向的时间序列任务层级分类体系，并在 MMTS-Bench 中将其具体化。该多模态、多维度基准包含四个子集（Base、InWild、Match、Align）中的 2,424 个问答对，可对从特征分析到时间推理和跨模态对齐等能力进行细粒度评估。

(2) **渐进式真实世界时间序列问答生成**。我们提出了一个三阶段框架，用于从真实世界时间序列数据中生成高质量问答对，系统地提升生成的可靠性与可扩展性，以支持大规模基准构建。

(3) **可控合成数据流水线**。我们设计了一种基于模块化组合与模板化生成的可控合成数据生成流水线，可精确控制数据多样性和难度，从而有针对性地评估基础能力。

2 相关工作

2.1 时间序列适配大语言模型

多模态大语言模型（MLLMs）在自然语言处理和跨模态推理方面展现出强大的能力。对于时间序列大语言模型，近期已有多种实现范式得到探索。Time-MQA [20] 将时间序列序列化为文本输入，并报告了其在时间序列问答任务上的早期成效。ChatTime [35] 将连续值量化为有限的词元空间，从而能够在统一的大语言模型框架内进行持续预训练。Zhuang 等人 [51] 在绘制出的时间序列图上使用 GPT-4o [29]开展两阶段、由粗到细的异常检测流水线；而 InsightMiner [48] 和 FinVis-GPT [38] 则分别基于 LLaVA [23] 进行时间序列描述和 K线图分析。

然而，将密集时间序列表示为文本或图表会显著增加序列长度和词元成本，而性能提升往往有限。基于对齐的方法通过保留专用时间序列编码器，并学习一个映射到大语言模型嵌入空间的轻量投影器来解决这一问题，从而实现高效的时间序列与文本交互。基于这一范式，Chow等人 [5]和 ChatTS [42] 在分类、描述、问答和推理方面取得了有竞争力的结果。尽管如此，目前仍缺乏一个统一的基准，来系统评估时间序列大语言模型的多维能力。

2.2 时间序列问答数据集

尽管近期研究已将大语言模型与时间序列相结合并发布了数据集，但大多数工作仍局限于预测 22, 24, 36], [19], ，而公开可用的时间序列问答数据集十分稀缺。此外，现有的时间序列问答数据集 [4, 20, 35, 37, 42] 存在领域不一致、能力分类体系扁平、问题格式僵化等问题，导致不同研究彼此孤立，并妨碍横向比较。

在单变量方面，ChatTime-TSQA [35] 依赖固定模板，针对四种基础属性（趋势、波动性、季节性和异常值）。Time-MQA [20] 和 Chat-TS [32] 通过单轮提示从真实世界数据中生成问答对，但尽管经过人工筛选，其覆盖范围仍然有限。在多变量设置下，EngineMT-QA [37] 通过结合数据分析、大语言模型优化和专家验证，从航空发动机时间序列中提取问答对，但其领域狭窄，限制了其作为基准的通用性

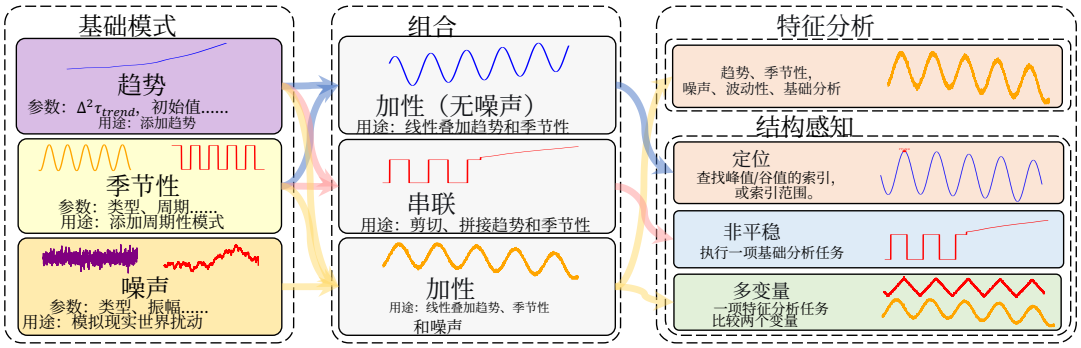


图2：基础构建流程。通过拼接并叠加趋势、季节性和噪声等基本成分，生成具有可控特征的合成时间序列。此图的绘图风格改编自 [4]。

。ChatTS [42] 提出了 TSEvol-Instruct，通过对多种时间序列进行迭代提示，但其扁平的分类体系和僵化的问题格式妨碍了对能力进行细粒度评估。

相比之下，我们提出的 MMTS-Bench 是一个用于时间序列问答的多模态、多维度基准，涵盖不同难度和领域，并包含单变量与多变量场景下的合成数据和真实数据。通过反复进行专家筛选和人工验证，该基准为评估模型性能提供了均衡且可靠的基础。

3 MMTS-BENCH

3.1 多维任务分类框架

为系统评估模型在时间序列分析中的理解与推理能力，我们提出了一个多维任务分类框架及相应的数据集构建方法。现有的时间序列问答数据集往往缺乏一致性和层次结构。为解决这类扁平分类方案的局限性，我们将模型的时间理解分解为五个核心维度（详情见附录A.3）：结构感知、特征分析、时间推理、序列匹配和跨模态理解。针对每个维度，我们定义了基础任务：结构感知包括平稳性与非平稳性识别、局部与全局模式识别，以及单变量与多变量分析；特征分析涵盖趋势、季节性、波动性、噪声和基本统计量；时间推理包括演绎、归纳、因果、类比和反事实推理；序列匹配侧重于同构、鲁棒性、定位和逆向对应；跨模态理解则涉及时间序列与语义表示之间的双向映射。通过组合这些任务，该框架生成了 286^3 细粒度的组合子任务。

基于这一框架，我们在 MMTS-BENCH 下设计了四个子集。**Base** 提供了一个受控的合成环境，重点考察结构感知和特征分析等基础能力，不涉及复杂推理。该子集包含多项选择题、二选一题和数值问题。为进行评估，我们将该子集划分为两个部分：Choice 子集和数值题子集。**InWild** 利用真实世界时间序列，进一步考察大语言模型在复杂且噪声较多的条件下进行特征分析和时间推理的能力。该子集包含多项选择题和二选一题。**Match**和**Align**聚焦于时间序列分析中研究较少的两个维度，即相似性匹配和跨模态理解，并由多项选择题构成。表1总结了五个核心维度下的子任务及其与各子集的对应关系；完整定义见表9（附录A.3）。这种多维度构建能够细粒度评估大语言模型在时间序列分析方面的能力，并有助于识别其在复杂分析推理任务中面临的瓶颈。

表1：MMTS-BENCH 中五个核心维度、任务及相关子集概览。

维度	Task	子集
结构感知	非平稳性、局部-全局、单变量-多变量	Base
特征分析	趋势、季节性、噪声、波动性、基础	基础与野外
时间推理	演绎、归纳、因果、类比、计数	InWild
序列匹配	同构、鲁棒、定位、反向	匹配
跨模态理解	时间序列→语义，语义→时间序列	对齐

3.2 合成数据集

Base旨在利用合成时间序列，在可控条件下进行基础评估，同时与以往的模块化基准测试明确区分。它采用 17 个由专家设计的模板，并通过明确且连续的参数，针对可解释序列生成问答对，符合既有的合成实践 [4, 7, 11, 49]。不同于早期工作采用相对固定的组合式流程、且主要依赖定性标注，我们的框架采用灵活的受 STL 启发的分解，支持多种组合模式（包括加法合成和拼接合成），使用比简单函数形式更丰富的趋势/噪声过程，并记录构造过程中使用的精确数值参数。这种参数级记录能够支持分级问题，并针对时间序列的组成部分开展精细的敏感性测试（例如，区分弱季节性与强季节性），

[3]结合特征分析与时间推理，可形成 35 个组合子任务。加入结构感知后，子任务数量增至 280 个。再加上序列匹配中的 4 个和跨模态理解中的 2 个，总计达到 286 个。



图3：InWild构建流程。该流程图展示了如何通过统计分析和面向大语言模型的多模态输入准备，将特定领域的时间序列转换为可用数据。图中重点呈现了大语言模型生成与专家验证之间的反馈回路，说明如何为来自多个领域（如经济学、交通、云运维等）的原始序列添加特征（如趋势强度、季节强度、熵等），并将其整理为自动问答生成所需的输入。

而拼接模式则通过受控的状态转移对非平稳性进行显式建模。这些设计选择共同支持在**Base** 中构建更丰富、更具诊断性的任务（详细的数学公式见附录A.4.1）。

3.3 真实世界数据集

InWild 旨在处理真实且复杂的时间序列场景，以评估模型在真实世界中的泛化能力及其分析与推理能力，作为对强调可控性和基本属性评估的**Base**的补充。我们采用一套原则明确的三阶段、面向推理的流程构建**InWild**，并使用大语言模型（图3；附录 A.4.2）。**(1) 多模态上下文整合与描述生成**。给定原始序列、可视化结果、用于锚定物理语义的领域元数据（例如，每小时交通流量、CPU/内存利用率）以及预计算统计特征，大语言模型会生成连贯的全局摘要和细粒度的局部描述。**(2) 面向推理的问答合成**。在预定义的初始化参数（任务类型、全局与局部范围、单变量或多变量设置以及目标索引范围）条件下，模型将上下文描述转换为“问题—答案—解释”（Q-A-E）三元组，并明确要求通过解释论证结论，而非依赖模板化的单轮生成。**(3) 多维验证与专家优化**。对每个实例进行逻辑有效性、数学正确性以及数值表示与视觉表示之间跨模态一致性的检查；随后对生成的问答对进行标准化，并由人工专家进一步审核，以确保其可靠性和领域保真度。

匹配 旨在评估模型在时间序列上进行相似度匹配的能力。该任务通过从真实世界时间序列中提取片段，并应用动态时间规整（DTW），搜索四个不同相似度级别的候选序列构建而成。随后，将每个片段及其候选序列组合成一道选择题，要求模型根据题目模板从中找出与该片段最相似的候选序列（图4）。此外，我们对片段样本进行平滑、扩展和反转等操作，构建了四种难度各异的任务范式（详情见附录 A.4.2）：（1）同构：匹配长度相同的序列；（2）鲁棒：匹配平滑后的序列；（3）局部定位：在较长序列中定位目标模式；（4）反向：匹配反转后的序列。

对齐评估模型在时间序列与自然语言描述之间的跨模态对齐能力。我们重复使用 **InWild** 第一阶段生成的描述，并将每条描述与其对应的时间序列配对，构成多项选择实例（详见附录 A.4.2）。由此得到两个对称任务：（1）时间序列 →语义（*TS→Sem*），模型需要从一组选项中为给定序列选择正确描述；（2）语义 → 时间序列（*Sem→TS*），模型需要从候选序列中找出与给定描述匹配的序列。为提高难度，干扰序列取自三个值域重叠的现实世界领域，使错误选项也具有迷惑性。

3.4 人在回路中的整理

为确保较高的数据集质量，我们在所有子集中采用系统化的人在回路策展 [41]，由一支固定的十名时间序列领域专家团队执行。审核标准因子集而异：对于**InWild**，专家评估问题的合理性与推理的有效性；对于**匹配**，专家通过目视确认DTW最近候选项在形状上也最为相似，从而使基于度量的检索与人类感知对齐；对于**对齐**，专家核验每条描述是否忠实地反映底层时间模式。我们进一步进行统计验证

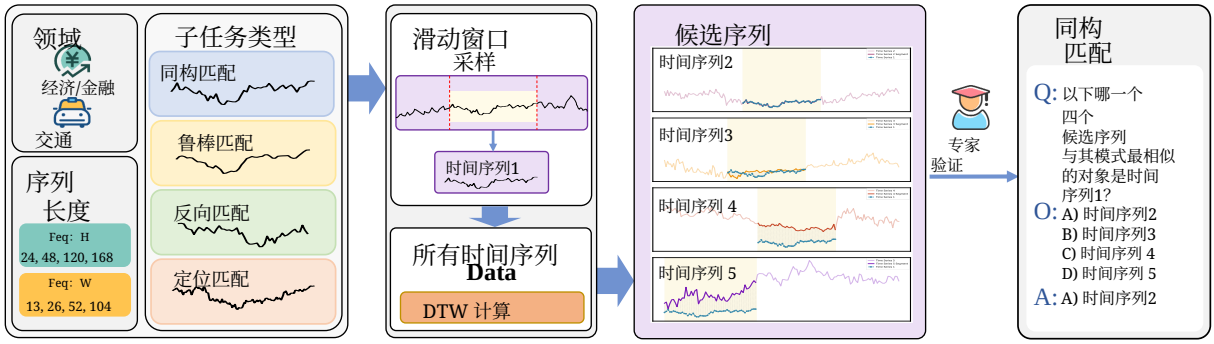


图4：匹配构建流程。从真实世界时间序列中提取片段，并使用 DTW 检索相似度各异的候选序列。随后使用固定模板构建问答对，同时通过平滑、扩展和反转等变换，形成难度各异的四种任务范式。

以评估标注可靠性：Fleiss' κ [10] 达到0.73，表明存在实质性的标注者间一致性 [21]。最终真实标签由多数投票确定，存在歧义的案例则升级至第二轮裁定。

为说明我们设计的**InWild** 三阶段流程的必要性，我们将其与单步提示进行比较，并考察随着逐步加入结构化约束和人工筛选，性能如何变化。在没有时间序列输入的情况下，单步提示的准确率约为 57%（考虑到选择题和二元问题的组合，随机猜测的预期准确率为 37.5%）。使用未经专家审核的三阶段流程后，准确率降至 44%；加入专家审核后，准确率进一步降至 35%。这一趋势表明，分阶段构建能更好地约束生成；结合专家筛选，还能减少虚假线索和潜在的答案泄漏，否则这些因素可能会抬高表面上的准确率。

3.5 统计可靠性与有效性分析

为补充人工筛选，我们开展统计分析以验证基准的可靠性与有效性（见附录D）。具体而言，我们进行了两项实验：**(1) 评估稳定性：**通过自助法置信区间和迭代子抽样，MMTS-Bench 得到的区间始终较窄，变异系数约为 10^{-3}，表明其对抽样方差具有很强的鲁棒性；**(2)针对捷径学习的有效性：**为确保性能反映的是内在时间序列推理，而非表面特征，我们分析了准确率与序列长度、维度等表面属性之间的关系，观察到相关性可忽略不计（例如，准确率与序列长度的相关性为 $|r| < 0.08$），且性能差距极小。这表明 MMTS-Bench 在很大程度上不受这些因素影响，并能抵御捷径学习。

4 评估结果

利用 MMTS-Bench数据集，我们对最新的开源和闭源大语言模型以及最先进的时间序列大语言模型进行了系统的基准测试与分析。我们报告了准确率、*Accuracy@N%*和相对准确率 （定义见附录 A.1），涵盖了跨越不同任务维度的多个子集。为确保统计稳健性和实验可靠性，所有实验均在温度 1.0 下独立进行五次，并对各次试验的结果取平均。

评估模型选择。我们使用三类具有代表性的大语言模型在 MMTS-Bench 上进行全面评估，以考察它们在时间序列问答任务上的表现。所选模型包括：(1) 闭源模型：Claude 3.7 Sonnet [1], Claude Sonnet 4 [2], Gemini 2.5 Flash/Pro [6], GPT-5 Minimal/High [31], GPT-4.1/4.1 mini [30], 和 GPT-4o [29]; (2) 开源模型：DeepSeek V3 [8], Kimi K2 [33], 和 Qwen 系列 [43, 44]（包括采用不同参数规模的 2.5 和 3 代模型）。值得注意的是，对于 Qwen3 系列，我们评估了思考模式和非思考模式。所有受评估的 Qwen2.5 模型均经过指令微调；为简洁起见，后文不再注明这一点；(3) 时间序列大语言模型：ChatTS [42], ITFormer [37], 和 ChatTime [35], ，它们专为时间序列数据分析而设计。对于多模态大语言模型（Qwen2.5-VL [3], Claude Sonnet 系列 [1, 2], Gemini 2.5 Pro [6],GPT-4 系列 [28]），我们分别采用纯文本、纯视觉以及视觉与文本组合输入进行评估。为确保评估公平，我们设计了标准化的时间序列输入格式（详见附录 B）；对于时间序列大语言模型，我们则谨慎地沿用其原始推理脚本。该多维评估框架旨在阐明不同模型架构和输入模态对时间序列理解与推理任务的影响。

4.1 大语言模型的纵向比较

时间序列大语言模型的泛化能力有限。从表 2 以及附录C中更全面的实验结果可以看出，在 MMTS-BENCH 的多个子任务中，包括 **InWild**、**匹配**、**对齐**以及 **Base** 中的选择题子集，闭源和开源通用大语言模型的表现始终优于时间序列大语言模型。时间序列大语言模型在分布外泛化方面表现出明显不足：ChatTS 的表现与其基座模型 Qwen2.5-14B 相当，或略逊一筹；ITFormer 应用于航空发动机领域之外时，表现远逊于 Qwen2.5-7B；而 ChatTime 则无法生成有效输出。这些发现表明，时间序列大语言模型虽然在狭窄领域内可能表现尚可，但其泛化能力仍然非常有限。

表 2：代表性大语言模型在 MMTS-BENCH 上的表现。TS 表示时间序列。最佳结果以粗体和下划线标出。

Cat.	模型	类型	平均值	MMTS-BENCH 子集			
				Base	InWild	匹配	对齐
闭源	GPT-5 High	Text	**0.74**	**0.51**	**0.72**	**0.82**	**0.99**
	GPT-4o	Text	0.61	0.42	0.62	0.50	0.97
	Claude Sonnet 4	Text	0.71	0.49	0.71	0.71	0.98
	Gemini 2.5 Pro	Text	0.70	0.48	0.68	0.79	0.98
Open	Kimi K2	Text	**0.63**	**0.45**	**0.63**	0.60	**0.95**
	DeepSeek V3	Text	0.62	0.41	0.61	**0.65**	**0.95**
	Qwen2.5-14b	Text	0.55	0.35	0.53	0.57	0.88
	Qwen2.5-7b	Text	0.45	0.33	0.44	0.40	0.69
TS	ChatTS	TS	**0.49**	**0.39**	**0.50**	**0.37**	**0.80**
	ITFormer	TS	0.31	0.31	0.33	0.24	0.29

现有时间序列大语言模型中的多模态对齐仍然效率低下。为探究影响时间序列大语言模型性能的关键因素，我们修改了当前最先进模型 ChatTS 的训练流水线，并围绕编码器架构、规模、位置编码、骨干大语言模型规模和提示前缀开展了一系列消融研究。结果表明，模型性能主要由骨干大语言模型规模决定，而对编码器结构、规模和位置编码基本不敏感，这说明编码器的贡献仍然有限，尚未得到充分开发。值得注意的是，在提示中加入简单的统计摘要可显著提升推理准确率，凸显了任务感知提示设计的重要性。详情见第5节。

4.2 大语言模型性能的跨维度比较

大语言模型在时间推理上的表现不如特征分析。如表15（附录C）所示，当前大语言模型在**野外**特征分析任务上的平均准确率达到62%，显著高于时间推理任务的55%。在统一输入模态下进行比较发现，Claude Sonnet 4 和Gemini 2.5 Pro 是表现最强的闭源大语言模型[4]，，而 DeepSeek V3 和 Kimi K2 则在开源大语言模型中领先（图5）。结果还表明，特征分析表现较弱通常与时间推理能力较差相伴。例如，闭源模型中的 GPT-4.1 Mini 和开源模型中的 Qwen2.5-7B 都符合这一趋势，说明特征分析能力不足会限制时间理解能力。

在特征分析维度中，季节性任务的准确率始终最低，**Base**和 **InWild**。这表明，大语言模型可能难以捕捉季节性模式，因此季节性是一项特别具有挑战性的任务。在时间推理维度上，结果进一步表明，与其他推理任务相比，因果推理和反事实推理尤为困难。如图 8（附录 A.4.2）所示，大语言模型在因果任务中经常陷入局部推理陷阱，无法从全局视角捕捉因果关系。同样，反事实推理需要在假设条件下重建依赖关系。当大语言模型缺乏充分的局部和全局意识时，往往会作出错误的条件假设，最终导致推理失败。

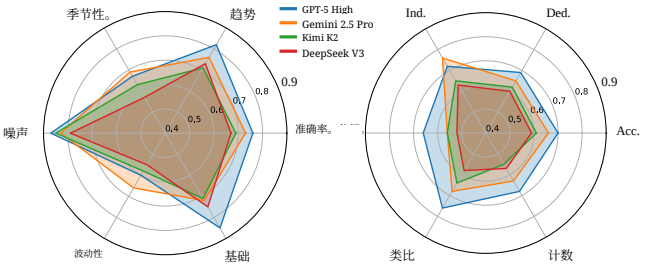


图5：排名靠前的大语言模型在 InWild 上的任务级准确率（左：特征分析；右：时间推理）。

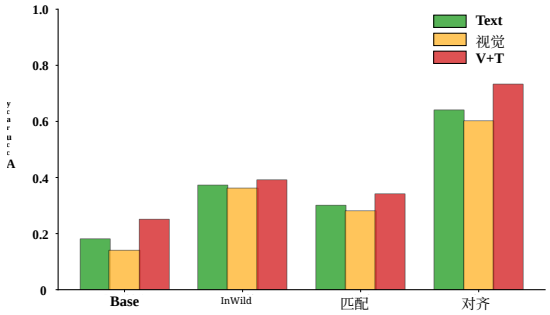
与全局任务相比，大语言模型在局部任务上表现较弱。如 **Base** 结果（表3）所示，大语言模型在结构感知维度的局部子任务上表现出最明显的弱点。在所有模型类别中，局部任务的准确率仍然较低，而全局任务的表现则明显更好。这表明，尽管大语言模型能够捕捉时间序列结构的总体规律，但它们通常缺乏精确定位能力，也难以在不同输入模态之间进行细粒度判别——这一问题与上文讨论的推理失败有关。相比之下，非平稳性以及单变量–多变量子任务上的表现差距相对较小。非平稳序列通常由具有不同统计性质的子序列拼接而成，从而产生可通过总体结构线索和分布线索检测到的全局偏移。

表 3：不同模型在基础子集数值题划分上的*Accuracy@10%* 指标。

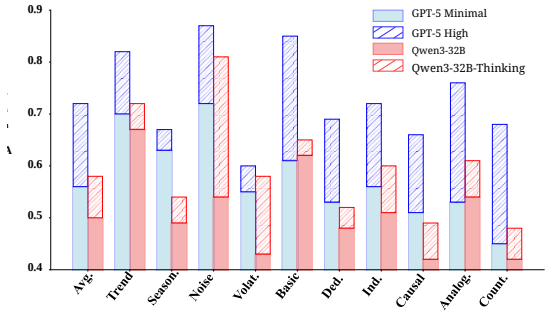
Cat.	模型	Type	结构感知					
			平稳	非平稳	局部	整体	单变量	多变量
Open	Kimi-K2	文本	0.56	0.56	0.19	0.45	0.56	0.54
	Qwen3-32b^{cot}	文本	0.40	0.40	0.20	0.32	0.40	0.33
	Qwen3-32b	文本	0.47	0.50	0.07	0.38	0.47	0.42
闭源	GPT-4o	文本	0.53	**0.58**	0.13	0.43	0.53	0.51
	GPT-4o	V	0.34	0.29	**0.31**	0.31	0.34	0.34
	GPT-4o	V+T	**0.59**	0.57	**0.31**	**0.56**	**0.59**	**0.56**
TS	ChatTS	TS	0.41	0.42	0.12	0.40	0.41	0.40

此外，在**匹配** 和**对齐** 子集的评估中（见附录C中的表16和表17），我们观察到两个关键发现：(1) 在四项序列匹配任务中，大语言模型在定位匹配和反向匹配上的表现显著逊于其余任务。这可能是由于注意力机制和自回归范式在处理时间方向变换时存在固有限制，但仍需进一步实验加以确认；(2) 在这两项任务中

[4]Claude 3.7 Sonnet 排名第二，但由于其深度参与了数据集构建，因此被排除。



（a）Qwen2.5-VL-7B 在四个子集上的表现



（b）思维链与非思维链设置下的准确率对比

图 6：（a）Qwen2.5-VL-7B 在四个子集上的评估结果。（b）GPT-5 和 Qwen3-32B 在 In-Wild 的不同子任务中，采用和不采用思维链时的准确率提升。

在跨模态维度下，大语言模型通常表现出色。这是因为大语言模型能够利用基础统计线索（例如最大值和最小值）并进行逻辑推理，从而将序列与文本描述对齐。

4.3 提升大语言模型在时间序列任务上性能的方法分析

多模态融合提升大语言模型进行时间序列分析的能力。如表15（附录C）所示，Gemini 2.5 Pro 在纯文本输入下的准确率为 68%，在纯视觉输入下为 72%，而结合多种模态后则达到 76%。同样，GPT-4.1 也通过多模态融合取得了类似的提升。基于这一观察，我们进一步在 MMTS-Bench 上评估了 Qwen2.5-VL-7B，也发现其表现稳定提升（图6a）。然而，这种提升并非普遍存在；例如，GPT-4.1 Mini 在 **InWild** 上的表现有所下降。我们推测，这一局限源于不同大语言模型整合多种模态的内在能力存在差异。因此，引入额外模态是缩小大语言模型在时间序列分析中信息差距的有效方法，但其效果主要取决于大语言模型的融合设计和训练方法。

思维链推理提升时间序列分析中的大语言模型能力，超越参数规模扩展。表15（附录C）中对Qwen2.5系列的实验展现了缩放定律效应 [19]：参数规模从7B扩展到14B可带来明显的性能提升，但继续增至32B后，提升幅度微乎其微，这表明在时间序列分析的时间推理任务中，收益会逐渐递减。相比之下，如图6b所示，启用思维链推理可显著提升Qwen3和GPT-5在所有**InWild** 子任务上的表现。在时间推理方面，这种提升尤为明显，平均提升幅度超过特征分析任务中的提升；对其他子集的进一步评估也证实了这一趋势。这些发现表明，缩放定律对基于参数规模的性能提升设定了内在限制，而启用思维链推理则能让模型更有效地捕捉时间依赖关系。因此，未来的工作应着重利用思维链推理提升大语言模型在时间序列分析中的表现，而非仅仅依赖扩大大语言模型的参数规模。

5 消融研究

在 MMTS-Bench 中，ChatTS [42] 是最强的开源时间序列大语言模型基线，展现出鲁棒的时间序列分析与推理能力。为理解其性能的驱动因素，并为未来的时间序列大语言模型设计提供指导，我们使用已发布的训练数据和推荐的训练策略，复现并修改了 ChatTS 官方训练流水线，针对时间序列编码器（架构、规模和位置编码）、大语言模型骨干规模以及统计提示前缀开展了受控消融实验。实验设置见附录B.2。

5.1 编码器架构的影响有限

我们在保持总体训练设置不变的情况下改变编码器架构，对比了多层感知机（17.1M）、卷积神经网络（50.4M）和 Transformer（6.3M）。如表4所示，不同架构之间的性能差异很小，这表明在当前对齐范式下，**编码器类型并非主要性能瓶颈**。相较于 Qwen2.5-3B 基线（将时间序列视为纯文本），添加编码器对**InWild** 和**Match**的影响很小，但提升了 Sem→TS 的表现，同时降低了 TS→Sem 的表现，表明编码器与训练数据交互可能引入了潜在的方向偏差。

表4：采用不同时间序列编码器架构的时间序列大语言模型的性能。基线模型为 Qwen2.5-3B-Instruct。

数据集	基线	多层感知机（17.1M）	卷积神经网络（50.4M）	Trans.（6.3M）
InWild	0.39	0.38	0.37	0.39
匹配	0.27	0.31	0.28	0.30
Sem→时间序列	0.49	0.59	0.61	0.61
时间序列→Sem	0.64	0.46	0.44	0.48

5.2 骨干网络规模占主导，编码器则不然

为进一步考察固定架构下的规模效应，我们评估了不同深度（1、3、5 和 7 层）的多层感知机编码器，这些编码器均基于同一 Qwen2.5-3B-Instruct 骨干网络构建，结果见表 5。此外，我们还将这些模型与采用更大骨干网络的基线进行比较，包括 Qwen2.5-14B-Instruct 和 ChatTS 模型。值得注意的是，ChatTS 采用 Qwen2.5-14B-Instruct 作为骨干网络，并使用 5 层多层感知机作为时间序列编码器。结果表明

表 5：时间序列编码器深度的消融实验。

Data	3B 1L（0.3M）	3L（8.7M）	5L（17.1M）	7L（25.5M）	14B ChatTS	
InWild	0.39	0.39	0.38	0.38	0.37	0.53	0.50
匹配	0.27	0.31	0.30	0.31	0.33	0.57	0.37
Sem→TS	0.49	0.61	0.52	0.59	0.61	0.87	0.91
时间序列→Sem	0.64	0.45	0.44	0.46	0.45	0.89	0.68

增加多层感知机层数并未带来单调提升，这表明单纯加深编码器并不能稳定地提升性能。对齐的骨干网络上的结果也进一步体现了先前的方向性趋势：Sem→TS 显著提升，而 TS→Sem 相较于指令微调基线则有所下降，这进一步印证了当前训练设置可能会促成非对称对齐行为这一观察。

5.3 位置编码的影响很小

随后，我们评估了官方代码中使用的位置编码策略，包括不使用位置编码、可学习的位置嵌入以及归一化索引值。表6显示，不同配置之间的差异很小，这表明与骨干网络扩展或其他因素相比，位置编码选择带来的提升有限。

表6：采用不同位置编码策略的模型性能。no_emb 表示不使用位置编码，pos_emb 表示可学习的位置嵌入，pos_idx 表示用作位置编码的归一化索引值。

数据集	no emb_	位置_嵌入	位置_索引
InWild	0.39	0.38	0.39
匹配	0.31	0.30	0.29
语义→时间序列	0.59	0.52	0.52
时间序列→语文	0.42	0.44	0.45

5.4 统计提示前缀是关键杠杆

最后，我们研究 ChatTS 中的统计提示前缀，它编码了输入序列的基础描述信息（例如偏移量、缩放因子、长度、最小值/最大值和边界值）。我们比较了使用前缀进行训练和评估（**开启**）与不使用前缀（**关闭**）的效果，还测试了不使用前缀训练、但使用前缀评估的模型（**关闭***）。表 7 显示，提示前缀带来了显著提升，尤其是在两个对齐任务上；在推理时提供该前缀，也基本恢复了模型性能。这表明，明确的统计信息可作为有效的辅助信号，提升可解释性，并降低推理过程中的推理门槛。

5.5 研究结果总结

总体而言，我们的消融实验得出三点结论。**(1) 编码器的影响有限**：在当前训练范式下，编码器架构、深度和位置编码仅带来细微变化，这表明更有效的时间序列—文本对齐机制仍有待探索。**(2) 骨干网络规模起主导作用**：扩大大语言模型骨干的规模，能持续提升各项

表 7：使用和未使用提示前缀训练的模型的性能。

数据集	ON	OFF	OFF*
InWild	0.38	0.37	0.37
匹配	0.31	0.26	0.33
Sem→TS	0.59	0.24	0.60
TS→Sem	0.46	0.21	0.46

任务上的表现，凸显了通用推理能力的核心作用。**(3) 提示设计很重要**：统计前缀带来了显著且实用的提升，这表明轻量级辅助信号和提示工程是增强时间序列大语言模型推理能力的有前景方向。

6 结论

我们介绍了 MMTS-Bench，这是一个全面的基准，包含 2,424 个时间序列问答对，分布在四个专门的子集中，用于评估多模态时间序列理解与推理能力。我们的广泛评估表明，通用大语言模型在跨领域泛化方面优于时间序列大语言模型，而当前的大语言模型在细粒度定位和复杂推理方面仍面临困难。我们还发现，单纯扩大模型规模的收益会逐渐递减；相比之下，通过多模态输入和思维链等显式推理策略，更能有效提升性能，而骨干大语言模型的能力是决定成功与否的主导因素。补充的消融研究表明，在当前训练范式下，常用的编码器设计带来的提升有限，这进一步凸显了深入研究时间序列大语言模型架构，以及探索超越直接扩大编码器规模的更有效时间序列—文本对齐机制的必要性。

局限性与未来工作。 MMTS-Bench 目前尚未涵盖若干传统时间序列评估任务，例如预测、插补、异常检测和表征学习，而这些也是时间序列能力的重要组成部分。该基准也高度依赖英文标注，因此可能无法充分反映多语言时序推理方面的见解。未来工作包括：(1) 将评估扩展至超出当前上下文限制的更长时间跨度序列；(2) 增加针对传统时间序列任务（预测、插补、异常检测和表征学习）的配套测试套件，以更全面地展现模型能力；(3) 开发超越当前编码器架构的更有效时间序列—文本对齐范式；(4) 探索能够更系统地利用统计特征的提示与工具使用策略。我们希望这一评估协议能够推动时间序列大语言模型的发展，使其具备鲁棒、可泛化的时间序列理解能力。

参考文献

[1] Anthropic。2025。Claude 3.7 Sonnet 和 Claude Code。urlhttps://www.anthropic.com/news/claude-3-7-sonnet。访问日期：2025-09-18。[2] Anthropic。2025。推出 Claude 4：Claude Opus 4 和 Claude Sonnet 4。https://www.anthropic.com/news/claude-4。访问日期：2025-09-18。[3] Shuai Bai、Keqin Chen、Xuejing Liu、Jialin Wang、Wenbin Ge、Sibo Song、Kai Dang、Peng Wang、Shijie Wang、Jun Tang、Humen Zhong、Yuanzhi Zhu、Mingkun Yang、Zhaohai Li、Jianqiang Wan、Pengfei Wang、Wei Ding、Zheren Fu、Yiheng Xu、Jiabo Ye、Xi Zhang、Tianbao Xie、Zesen Cheng、Hang Zhang、Zhibo Yang、

Haiyang Xu 和 Junyang Lin。2025。Qwen2.5-VL 技术报告。arXiv 预印本 arXiv:2502.13923（2025）。[4] Yifu Cai、Arjun Choudhry、Mononito Goswami 和 Artur Dubrawski。2024。Time-SeriesExam：一项时间序列理解考试。arXiv 预印本 arXiv:2410.14752（2024）。[5] Winnie Chow、Lauren Gardiner、Haraldur T Hallgrímsson、Maxwell A Xu 和 Shirley You Ren。2024。利用大语言模型进行时间序列推理。arXiv 预印本 arXiv:2409.11376（2024）。[6] Gheorghe Comanici、Eric Bieber、Mike Schaekermann、Ice Pasupat、Noveen Sachdeva、Inderjit Dhillon、Marcel Blistein、Ori Ram、Dan Zhang、Evan Rosen 等。2025。Gemini 2.5：借助先进推理、多模态、长上下文和新一代智能体能力，拓展前沿。arXiv 预印本 arXiv:2507.06261（2025）。[7] Abhimanyu Das、Weihao Kong、Rajat Sen 和 Yichen Zhou。2024。用于时间序列预测的仅解码器基础模型。arXiv:2310.10688 [cs.CL] https://arxiv.org/abs/ 2310 10688 。[8] 2024 DeepSeek-AI。。DeepSeek-V3 技术报告。arXiv:2412.19437 [cs.CL] https://arxiv.org/abs/ 2412 19437 。Zihan Dong、Xinyu Fan 和 Zhiyuan Peng。2024。[9] FNSPID：一个全面的时间序列金融新闻数据集。arXiv:2402.06698 [q-fin.ST] https://arxiv. org/abs/ 2402 06698 。Joseph L Fleiss。。衡量多名评定者之间的名义尺度一致性。[10] 1971 《心理学通报》76，5（1971），378。[11] Fanzhe Fu、Junru Chen、Jing Zhang、Carl Yang、Lvbin Ma 和 Yang Yang。2024。合成时间序列数据真的不如真实数据吗？arXiv: 2402 00607 [ ] 2402 00607 . cs.LG [12] https://arxiv.org/abs/ 。Robert Geirhos、Jörn-Henrik Jacobsen、Claudio Michaelis、Richard Zemel、Wieland Brendel、Matthias Bethge 和 Felix A Wichmann。2020。深度神经网络中的捷径学习。《自然·机器智能》2，11（2020），665–673。[13] Rakshitha Wathsadini Godahewa、Christoph Bergmeir、Geoffrey I. Webb、Rob Hyndman 和 Pablo Montero-Manso。2021。莫纳什时间序列预测档案库。载于第三十五届神经信息处理系统大会数据集与基准测试专题（第二轮）。https://openreview.net/forum?id= wEc1mgAjU-[14] Suchin Gururangan、Swabha Swayamdipta、Omer Levy、Roy Schwartz、Samuel Bowman 和 Noah A Smith。2018。自然语言推理数据中的标注伪影。载于《2018年计算语言学协会北美分会人类语言技术会议论文集》第2卷（短文），107–112。[15] Yuxiao Hu、Qian Li、Dongxiao Zhang、Jinyue Yan 和 Yuntian Chen。2025。上下文对齐：激活并增强大语言模型的时间序列能力。载于第十三届国际学习表征会议。https: // openreview.net/forum?id=syC2764fPc [16] Dieuwke Hupkes、Verna Dankers、Mathijs Mul 和 Elia Bruni。2020。组合性解析：神经网络如何实现泛化？《人工智能研究杂志》67（2020），757–795。[17] Ming Jin、Shiyu Wang、Lintao Ma、Zhixuan Chu、James Y Zhang、Xiaoming Shi、Pin-Yu Chen、Yuxuan Liang、Yuan-Fang Li、Shirui Pan 和 Qingsong Wen。2024。Time-LLM：通过重编程大语言模型进行时间序列预测。载于国际学习表征会议（ICLR）。[18] Ming Jin、Yifan Zhang、Wei Chen、Kexin Zhang、Yuxuan Liang、Bin Yang、Jindong Wang、Shirui Pan 和 Qingsong Wen。2024。立场论文：大语言模型能告诉我们关于时间序列分析的什么？载于国际机器学习会议（ICML 2024）。[19] Jared Kaplan、Sam McCandlish、Tom Henighan、Tom B. Brown、Benjamin Chess、Rewon Child、Scott Gray、Alec Radford、Jeffrey Wu 和 Dario Amodei。2020。神经语言模型的缩放定律。arXiv:2001.08361 [cs.LG] https: //arxiv.org/abs/2001.08361[20] Yaxuan Kong、Yiyuan Yang、Yoontae Hwang、Wenjie Du、Stefan Zohren、Z hangyang Wang、Ming Jin 和 Qingsong Wen。2025。Time-MQA：结合上下文增强的时间序列多任务问答。arXiv 预印本 arXiv:2503.01875（2025）。[21] J Richard Landis 和 Gary G Koch。1977。分类数据中观察者一致性的测量。《生物统计学》（1977），159–174。[22] Chenxi Liu、Qianxiong Xu、Hao Miao、Sun Yang、Lingzheng Zhang、Cheng Long、Ziyue Li 和 Rui Zhao。2025。TimeCMA：通过跨模态对齐实现大语言模型赋能的多变量时间序列预测。载于 AAAI。[23] Haotian Liu、Chunyuan Li、Qingyang Wu 和 Yong Jae Lee。2023。视觉指令微调。载于 NeurIPS。[24] Haoxin Liu、Shangqing Xu、Zhiyuan Zhao、Lingkai Kong、Harshavardhan Ka-marthi、Aditya B. Sasanur、Megha Sharma、Jiaming Cui、Qingsong Wen、Chao Zhang 和 B. Aditya Prakash。2024。Time-MMD：一个用于时间序列分析的新型多领域多模态数据集。arXiv:2406.08627[25] Yuan Liu、Haodong Duan、Yuanhan Zhang、Bo Li、Songyang Zhang、Wangbo Zhao、Yike Yuan、Jiaqi Wang、Conghui He、Ziwei Liu 等。2024。Mmbench：你的多模态模型是全能选手吗？载于欧洲计算机视觉会议。施普林格，216–233。[26] Tung Nguyen、Jason Jewik、Hritik Bansal、Prakhar Sharma 和 Aditya Grover。2023。ClimateLearn：面向天气与气候建模的机器学习基准测试。载于神经信息处理系统进展，A. Oh、T. Naumann、A. Globerson、K. Saenko、M. Hardt 和 S. Levine（编），第 36 卷。Curran Associates, Inc.，75009– 75025。https://proceedings.neurips.cc/paper_files/paper/2023/file/ ed73c36e771881b232ef35fa3a1dec14-Paper-Datasets_and_Benchmarks.pdf [27] Yuqi Nie、Nam H. Nguyen、Phanwadee Sinthong 和 Jayant Kalagnanam。2023。一个时间序列胜过 64 个词：使用 Transformer 进行长期预测。载于国际学习表征会议。[28] OpenAI。2023。GPT-4 研究。https://openai.com/index/gpt-4-research/。访问日期：2025-09-18。[29] OpenAI。2024。Hello GPT-4o。https://openai.com/index/hello-gpt-4o/。访问日期：2025-09-18。[30] OpenAI。2025。在 API 中推出 GPT-4.1。https://openai.com/index/gpt-4-1/。访问日期：2025-09-18。[31] OpenAI。2025。推出 GPT-5。https://openai.com/index/introducing-gpt-5/。访问日期：2025-09-18。[32] Paul Quinlan、Qingguo Li 和 Xiaodan Zhu。2025。Chat-TS：增强对时间序列和自然语言数据的多模态推理。*arXiv* 预印本 *arXiv:2503.10883* (2025)。[33] Kimi Team、Yifan Bai、Yiping Bao、Guanduo Chen、Jiahao Chen、Ningxin Chen、Ruijue Chen、Yanru Chen、Yuankun Chen、Yutian Chen、Zhuofu Chen、Jialei Cui、Hao Ding、Mengnan Dong、Angang Du、Chenzhuang Du、Dikang Du、Yulun Du、Yu Fan、Yichen Feng、Kelin Fu、Bofei Gao、Hongcheng Gao、Peizhong Gao、Tong Gao、Xinran Gu、Longyu Guan、Haiqing Guo、Jianhang Guo、Hao Hu、Xiaoru Hao、Tianhong He、Weiran He、Wenyang He、Chao Hong、Yangyang Hu、Zhenxing Hu、Weixiao Huang、Zhiqi Huang、Zihao Huang、Tao Jiang、Zhejun Jiang、Xinyi Jin、Yongsheng Kang、Guokun Lai、Cheng Li、Fang Li、Haoyang Li、Ming Li、Wentao Li、Yanhao Li、Yiwei Li、Zhaowei Li、Zheming Li、Hongzhan Lin、Xiaohan Lin、Zongyu Lin、Chengyin Liu、Chenyu Liu、Hongzhang Liu、Jingyuan Liu、Junqi Liu、Liang Liu、Shaowei Liu、T. Y. Liu、Tianwei Liu、Weizhou Liu、Yangyang Liu、Yibo Liu、Yiping Liu、Yue Liu、Zhengying Liu、Enzhe Lu、Lijun Lu、Shengling Ma、Xinyu Ma、Yingwei Ma、Shaoguang Mao、Jie Mei、Xin Men、Yibo Miao、Siyuan Pan、Yebo Peng、Ruoyu Qin、Bowen Qu、Zeyu Shang、Lidong Shi、Shengyuan Shi、Feifan Song、Jianlin Su、Zhengyuan Su、Xinjie Sun、Flood Sung、Heyi Tang、Jiawen Tao、Qifeng Teng、Chensi Wang、Dinglu Wang、Feng Wang、Haiming Wang、Jianzhou Wang、Jiaxing Wang、Jinhong Wang、Shengjie Wang、Shuyi Wang、Yao Wang、Yejie Wang、Yiqin Wang、Yuxin Wang、Yuzhi Wang、Zhaoji Wang、Zhengtao Wang、Zhexu Wang、Chu Wei、Qianqian Wei、Wenhao Wu、Xingzhe Wu、Yuxin Wu、Chenjun Xiao、Xiaotong Xie、Weimin Xiong、Boyu Xu、Jing Xu、Jinjing Xu、L. H. Xu、Lin Xu、Suting Xu、Weixin Xu、Xinran Xu、Yangchuan Xu、Ziyao Xu、Junjie Yan、Yuzi Yan、Xiaofei Yang、Ying Yang、Zhen Yang、Zhilin Yang、Zonghan Yang、Haotian Yao、Xingcheng Yao、Wenjie Ye、Zhuorui Ye、Bohong Yin、Longhui Yu、Enming Yuan、Hongbang Yuan、Mengjie Yuan、Haobing Zhan、Dehao Zhang、Hao Zhang、Wanlu Zhang、Xiaobin Zhang、Yangkun Zhang、Yizhi Zhang、Yongting Zhang、Yu Zhang、Yutao Zhang、Yutong Zhang、Zheng Zhang、Haotian Zhao、Yikai Zhao、Huabin Zheng、Shaojie Zheng、Jianren Zhou、Xinyu Zhou、Zaida Zhou、Zhen Zhu、Weiyu Zhuang 和 Xinxing Zu。2025。Kimi K2：开放式智能体智能。arXiv:2507.20534 [cs.LG] https://arxiv.org/abs/2507.20534[34] MiniCPM Team、Chaojun Xiao、Yuxuan Li、Xu Han、Yuzhuo Bai、Jie Cai、Haotian Chen、Wentong Chen、Xin Cong、Ganqu Cui 等。2025。MiniCPM4：面向终端设备的超高效大语言模型。*arXiv* 预印本 *arXiv:2506.07900* (2025)。[35] Chengsen Wang、Qi Qi、Jingyu Wang、Haifeng Sun、Zirui Zhuang、Jinming Wu、Lei Zhang 和 Jianxin Liao。2025。ChatTime：连接数值数据与文本数据的统一多模态时间序列基础模型。载于*AAAI* 人工智能会议论文集，第 39 卷。12694–12702。[36] Xinlei Wang、Maike Feng、Jing Qiu、Jinjin Gu 和 Junhua Zhao。2024。从新闻到预测：在基于大语言模型的时间序列预测中结合事件分析与反思。载于神经信息处理系统。[37] Yilin Wang、Peixuan Lei、Jie Song、Yuzhe Hao、Tao Chen、Yuxuan Zhang、Lei Jia、Yuanxiang Li 和 Zhongyu Wei。2025。ITFormer：借助大规模多任务数据集连接时间序列与自然语言，实现多模态问答。载于国际机器学习会议（*ICML*）。[38] Ziao Wang、Yuhang Li、Junda Wu、Jaehyeon Soon 和 Xiaofeng Zhang。2023。FinVis-GPT：用于金融图表分析的多模态大语言模型。*arXiv* 预印本 *arXiv:2308.01430* (2023)。[39] Gerald Woo、Chenghao Liu、Akshat Kumar 和 Doyen Sahoo。2024。突破云运维领域时间序列预测预训练的极限。https://openreview.net/forum?id=ZkEsEFFUyo [40] Gerald Woo、Chenghao Liu、Akshat Kumar、Caiming Xiong、Silvio Savarese 和 Doyen Sahoo。2024。通用时间序列预测 Transformer 的统一训练。载于第四十一届国际机器学习会议。[41] Xingjiao Wu、Luwei Xiao、Yixuan Sun、Junhang Zhang、Tianlong Ma 和 Liang He。2022。人在回路机器学习综述。未来一代计算机系统135 (2022)，364–381。[42] Zhe Xie、Zeyan Li、Xiao He、Longlong Xu、Xidao Wen、Tieying Zhang、Jianjun Chen、Rui Shi 和 Dan Pei。2024。ChatTS：将时间序列与大语言模型对齐

通过合成数据增强理解与推理。*arXiv* 预印本 *arXiv:2412.03104* (2024)。[43] An Yang、Anfeng Li、Baosong Yang、Beichen Zhang、Binyuan Hui、Bo Zheng、Bowen Yu、Chang Gao、Chengen Huang、Chenxu Lv、Chujie Zheng、Dayiheng Liu、Fan Zhou、Fei Huang、Feng Hu、Hao Ge、Haoran Wei、Huan Lin、Jialong Tang、Jian Yang、Jianhong Tu、Jianwei Zhang、Jianxin Yang、Jiaxi Yang、Jing Zhou、Jingren Zhou、Junyang Lin、Kai Dang、Keqin Bao、Kexin Yang、Le Yu、Lianghao Deng、Mei Li、Mingfeng Xue、Mingze Li、Pei Zhang、Peng Wang、Qin Zhu、Rui Men、Ruize Gao、Shixuan Liu、Shuang Luo、Tianhao Li、Tianyi Tang、Wenbiao Yin、Xingzhang Ren、Xinyu Wang、Xinyu Zhang、Xuancheng Ren、Yang Fan、Yang Su、Yichang Zhang、Yinger Zhang、Yu Wan、Yuqiong Liu、Zekun Wang、Zeyu Cui、Zhenru Zhang、Zhipeng Zhou 和 Zihan Qiu。2025。Qwen3 技术报告。*arXiv* 预印本 *arXiv:2505.09388* (2025)。[44] An Yang、Baosong Yang、Beichen Zhang、Binyuan Hui、Bo Zheng、Bowen Yu、Chengyuan Li、Dayiheng Liu、Fei Huang、Haoran Wei、Huan Lin、Jian Yang、Jianhong Tu、Jianwei Zhang、Jianxin Yang、Jiaxi Yang、Jingren Zhou、Junyang Lin、Kai Dang、Keming Lu、Keqin Bao、Kexin Yang、Le Yu、Mei Li、Mingfeng Xue、Pei Zhang、Qin Zhu、Rui Men、Runji Lin、Tianhao Li、Tingyu Xia、Xingzhang Ren、Xuancheng Ren、Yang Fan、Yang Su、Yichang Zhang、Yu Wan、Yuqiong Liu、Zeyu Cui、Zhenru Zhang 和 Zihan Qiu。2024。Qwen2.5 技术报告。*arXiv* 预印本 *arXiv:2412.15115* (2024)。[45] Han Yu、Peikun Guo 和 Akane Sano。2024。心电图语义整合器（ ESI ）：使用大语言模型增强的心脏病学文本进行预训练的基础心电图模型。*arXiv* 预印本 *arXiv:2405.19366* (2024)。[46] Zhen Zeng、Rachneet Kaur、Suchetha Siddagangappa、Saba Rahimi、Tucker Balch 和 Manuela Veloso。2023。使用卷积神经网络和 Transformer 进行金融时间序列预测。*arXiv* 预印本 *arXiv:2304.04912* (2023)。[47] Xuchao Zhang、Yifeng Gao、Jessica Lin 和 Chang-Tien Lu。2020。TapNet：使用注意力原型网络进行多变量时间序列分类。载于*AAAI* 人工智能会议论文集，第 34 卷。6845–6852。[48] Yunkai Zhang、Yawen Zhang、Ming Zheng、Kezhen Chen、Chongyang Gao、Ruian Ge、Siyuan Teng、Amine Jelloul、Jinmeng Rao、Xiaoyuan Guo 等。2023。洞察挖掘器：用于跨领域对齐自然语言的时间序列分析数据集。载于*NeurIPS 2023* 人工智能促进科学研讨会。[49] Zhenwei Zhang、Ruiqi Wang、Ran Ding 和 Yuantao Gu。2024。揭示异常：一种用于时间序列异常检测的端到端季节-趋势分解方法。载于*ICASSP 2024 - 2024 IEEE* 电气电子工程师学会国际声学、语音与信号处理会议（*ICASSP*）。5415–5419。doi:10.1109/ICASSP48485. 2024.10446482[50] Haoyi Zhou、Shanghang Zhang、Jieqi Peng、Shuai Zhang、Jianxin Li、Hui Xiong 和 Wancai Zhang。2021。Informer：超越高效 Transformer 的长序列时间序列预测。载于第三十五届人工智能促进协会会议，*AAAI 2021*，线上会议，第 35 卷。AAAI 出版社，11106–11115。[51] Jiaxin Zhuang、Leon Yan、Zhenwei Zhang、Ruiqi Wang、Jiawei Zhang 和 Yuantao Gu。2024。看见它，思考它，解决它：大型多模态模型是少样本时间序列异常分析器。*arXiv* 预印本 *arXiv:2411.02465* (2024)。

目录

- 附录 A：MMTS-Bench 详情。评估指标、真实世界时间序列数据来源、数据集分类和数据集构建方法的定义。
- 附录 B：评估与实验设置。标准化时间序列输入格式、评估提示、时间序列可视化，以及消融研究的实验设置。
- 附录C：完整结果。使用 MMTS-Bench 评估的大语言模型完整测试结果。
- 附录D：统计稳健性与伪影分析。通过统计实验展示 MMTS-BENCH 对数据集伪影的鲁棒性、评估稳定性和有效性。
- 附录 E：复现声明。

A MMTS-BENCH 详情

A.1 指标

我们采用分层评估方案，并针对答案空间 A 中的每种答案类型采用专门的指标，从而确保对不同问题格式进行全面、公平的评估。

类别评估（A_{mc} 和 A_{bf}）。对于多项选择题和二元选择任务，我们采用**准确率** 作为主要评估指标，通过指示函数衡量精确匹配表现：

$$\text{Accuracy} = \mathbb{I}(\hat{A} = A_{gt}) \tag{1}$$

其中，$\mathbb{I}(\hat{A} = A_{gt}) = 1$ 表示预测答案 $\hat{A}$ 与事实 A_{gt} 匹配时的取值，否则为 0。

数值评估（A_{num}）。对于数值任务，我们采用两个互补指标，分别衡量连续接近程度和基于阈值的精确度。相对准确率量化预测值与事实值之间的相对接近程度，并给出 $[0, 1]$ 范围内的归一化得分，其中 1 表示预测完全准确：

$$\text{Relative Accuracy} = \max\left(1.0 - \frac{|\hat{A} - A_{gt}|}{|A_{gt}|}, 0.0\right) \tag{2}$$

Accuracy@10% 提供了更严格的二元评估标准，用于判断预测是否处于 10% 的相对误差容限内：

$$\text{Accuracy@10\%} = \mathbb{I}\left(\frac{|\hat{A} - A_{gt}|}{|A_{gt}|} \leq 0.1\right) \tag{3}$$

我们在协议中引入的评分函数 $M(\hat{A},\ A_{gt})$ 使用这些类型特定指标进行实例化，确保根据答案类型进行适当评估，同时保持整个评估框架的一致性。

A.2 真实世界数据集来源

我们的基准中的真实世界部分基于 LOTSA [13, 26, 39, 40] 数据集集合构建。为确保广泛的领域覆盖，同时保持代表性和高质量，我们选取了五个主要领域：交通、云运营、气候、经济学和医疗保健。这些领域中的代表性数据集包括 Traffic Hourly、阿里巴巴集群跟踪数据集 2018、ERA5 2018、M4 Weekly、医院和新冠死亡人数，其统计参数汇总于表 8 中。具体详情如下。

交通（Traffic Hourly）该数据集来自加利福尼亚州交通运输部，记录了旧金山湾区多个传感器在 48 个月期间（2015–2016 年）采集的每小时高速公路占有率数据。数据集包含 862 条时间序列，每条序列有 17,376 个点，取值范围为 [0,1]。由于时间跨度较长且季节性模式明显，我们采用了最大长度为 672 个点的滑动窗口。为减少大语言模型的令牌用量并避免无关精度，数值均乘以 100 后四舍五入至小数点后两位。

云运营（阿里巴巴集群跟踪数据集 2018）该数据集记录了一个由约 4,000 台机器组成的集群在八天内（2018 年 1 月 2 日至 1 月 8 日）的中央处理器和内存利用率，采样间隔为五分钟。数据集包含 58,409 对时间序列。理论序列长度为 1,728 个点，但部分序列因存在缺失样本而较短（100–1,728 个点）。值的范围为 [0,100]。由于序列长度适中，因此未采用窗口化处理，而是随机抽取序列，并保留两位小数。

气候（ERA5 2018）该数据集来自欧洲中期天气预报中心，提供 2018 年的全球逐小时再分析数据，分辨率为 2.8125°（64×128 个网格点），涵盖七个气压层上的 45 50、250、500、600、700、850、925 8,736 个变量（以及 hPa）。每对时间序列包含 个点。为构建基准子集，我们从七个气压层中选取了相对湿度和温度，其值在 [0,100] 范围内。为体现空间多样性，我们在全球范围内随机抽取了 50 个地点，然后应用长度为 720 的滑动窗口。所有值均四舍五入至小数点后两位。

经济学（M4 Weekly）该数据集是 M4 竞赛（2018）的一个子集，其中包含 100,000 条不同频率的时间序列。每周子集包括 359 条经济及商业相关序列，例如销售额、需求和指数值。序列长度介于 80 到 2,597 个数据点之间。为保留潜在的季节性并平衡序列长度，我们使用了最大长度为 104 个数据点的滑动窗口，约为两年的长度。较短的序列则完整保留。所有值均四舍五入至小数点后两位。

医疗保健（医院与新冠死亡人数）医院数据集记录了 2000 年 1 月至 2006 年 12 月与医疗产品相关的月度患者人数。该数据集包含 767 条长度为 72 的序列。我们采用了长度为 36、60 和 72 个数据点的常见月度截取长度进行滑动窗口处理。所有值均四舍五入至小数点后两位。COVID死亡数据集来源于约翰·霍普金斯大学存储库。该数据集包含 2020 年 1 月 22 日至 8 月 20 日各国家和地区的每日累计死亡人数。数据集由 266 条每日序列组成，每条长度为 182 个数据点。我们采用最大长度为 180 个数据点的滑动窗口。所有值均四舍五入至小数点后两位。

A.3 数据集分类

为系统评估大语言模型在所提出的多维框架中的能力，我们构建了四个专门的子集，共同构成 MMTS-BENCH 的核心。其设计

表 8：LOTSA 子集的统计参数。

数据集	领域	频率	#时间序列	#观测值	#变量
Traffic Hourly	交通	H	862	14,978,112	1
阿里巴巴集群跟踪数据集 2018	云运维	5T	58,409	95,192,530	2
ERA5 2018	气候	H	245,760	2,146,959,000	45
M4 Weekly	经济学	W	359	366,912	1
医院	医疗保健	M	767	55,224	1
新冠死亡人数	医疗保健	D	266	48,412	1

基于一个核心假设：模型对时间序列的理解具有层次性和渐进性，基础分析能力的不足会导致高层次推理任务中出现系统性偏差。为将这一假设具体化，我们将时间理解分解为五个核心维度。

我们将时间序列理解分解为若干信息维度，每个维度都代表一种独特且不可替代的处理要求。由于单个任务通常仅依赖其中部分维度，我们据此组织并标注任务，以避免定义混杂或冗余。层级这一概念意味着，更高层次的能力是由多个维度组合而成，而非引入全新的维度。尽管基础子集和 InWild 子集中的维度彼此相关，我们仍有意赋予它们不同的作用。基础子集旨在通过受控设置隔离并"激活"各个维度（例如结构感知）。相比之下，源自真实场景的 InWild 任务需要同时整合多个维度。通过明确标注这些维度之间的依赖关系，我们能够进行细粒度的失败分析：InWild 任务表现不佳，可以追溯到特定维度（例如结构感知）的不足，而非归因于笼统的总体失败。

为此，我们采用双层评估架构：先**使用合成数据评估基础能力**，再**使用真实世界数据评估高级能力**。这种方法既能对核心能力进行精确、可控的评估，也能真实地评估实际表现。

Base。该子集使用精确控制的合成数据构建，消除了真实世界数据中存在的混杂变量，从而在受控环境中对模型的基础时间序列分析能力进行标准化、细粒度的评估。它包含 700 个问答对，涵盖结构感知（D_s）和特征分析（D_f）两个核心维度。

以下三个子集基于 LOTSA 基准中的真实世界数据构建，用于评估模型在复杂场景中的高级能力。

InWild。该子集涵盖五个专业领域（交通、云运维、气候、经济/金融、医疗保健），用于评估模型在高级时间序列理解与推理方面的能力。通过组合三个核心维度——结构感知（D_s）、特征分析（D_f）和时间推理（D_r）——该子集生成了 1,084 个问答对，覆盖 140 种子任务类型。

匹配。该子集由交通和经济/金融领域构成，用于评估模型在时间序列相似性匹配和形态对应方面的表现。我们通过改变序列匹配（D_m）维度，同时保持结构感知（D_s）的维度——平稳性、全局范围和单变量序列——不变。该子集包含400个问答对，每个类别各有100对。

对齐。该子集涵盖五个专业领域（交通、云运维、气候），旨在系统评估模型在数值时间序列与自然语言之间进行双向理解和跨模态转换的能力。从跨模态理解维度（D_c）出发，我们基于同源时间序列构建了 240 组双向问答对。

这一层次化数据集架构使 MMTS-Bench 能够全面评估从基础分析技能到高级分析能力的各个层面，并提供详细的诊断能力画像，识别模型在时间序列理解与推理任务中的具体优势和不足。

A.4 数据集构建方法

为全面评估多模态时间序列理解能力，我们开发了一种结合合成数据与真实世界数据源的双路径构建方法。合成路径采用模块化组件合成，并进行系统化参数控制，以便对基础时间序列属性开展可控评估。真实世界路径则运用渐进式对话框架、相似性匹配算法和跨模态转换技术，构建全面的评测基准。该方法生成了四个专业子集：**Base** 用于可控合成评估，**InWild** 用于多维推理，**Match** 用于相似性匹配，以及**对齐**用于基于真实 LOSTA 数据的跨模态理解。

A.4.1 合成数据集。 **Base**子集中的时间序列数据采用模块化合成方法生成。该流程首先创建三个基本原始组件：**(1) 趋势组件**，可配置特定的方向和幅度；**(2) 季节性组件**，支持多种周期性和波形，涵盖标准模式（如正弦波）以及代表真实世界场景的复杂复合波形；**(3) 噪声组件**，其中包含多种噪声类型。

最终的时间序列由这些组件叠加而成。这一原理通过一个加性模型形式化表示，其中生成的时间序列 y_t 是三个加权组件之和：

$$y^{(t)} = \tau^{(t)}_{\text{trend}} + \tau^{(t)}_{\text{seasonal}} + \tau^{(t)}_{\text{noise}}, \quad t = 1, 2, \ldots, T. \tag{4}$$

每个组件 $\tau^{(t)}$ 均通过将相应的**基信号** $S(t)$乘以随机采样的权重 w来合成，使得$\tau^{(t)}$趋势 $= w_{\text{trend}} \cdot S_{\text{trend}}(t)$，其他组件也以此类推。这些基信号的生成由一组可配置参数控制，以确保多样性，详情如下。

首先，通过从区间 $[128, 2048]$（对应于最小_长度和最大_长度）中采样，为所有基础信号确定一个全局**序列长度**（T）。然后，按如下方式构造每个基础信号：

- 趋势基准信号（S (t) 趋势）旨在捕捉长期非平稳行为。其建模为 ARIMA(0,2,0)

表9：MMTS-Bench 描述的时间序列任务的核心维度、涵盖的子任务类型及相关子集。

维度	子任务	定义	相关子集
结构感知	非平稳性 局部-全局 单变量-多变量	分析拼接子序列的统计性质。 定位并分析特定序列片段。 联合处理和分析多项时间序列数据。	Base
特征分析	趋势 季节性 噪声 波动性 基础	识别长期方向性模式和趋势强度。 捕捉季节性模式和季节性强度。 区分随机波动与信号成分。 量化时间变化性和不稳定性。 计算基本统计量（均值、方差、范围等）。	Base、InWild
时间推理	演绎 归纳 因果 类比 反事实	运用一般规则推断特定区间的属性。 从观测序列中归纳特征。 识别序列之间的因果关系或领先—滞后关系。 通过比较时间模式推断相似性。 预测假设性变化下的结果。	In Wild
序列匹配	同构 鲁棒 定位 反向	在等长约束下找到最相似的序列。 在预处理变换后仍能鲁棒地匹配模式。 在较长序列中定位目标模式。 识别时间反转后的相似性。	匹配
跨模态	时间序列→语义 语义→时间序列	将时间序列模式转换为文本描述。 将文本描述映射到相应的时间序列数据。	对齐

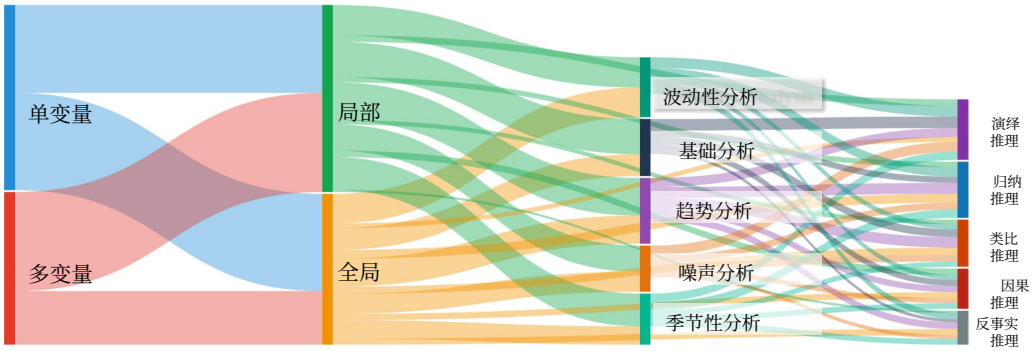


图 7：theInWild 子集子任务标签的桑基图。该图以清晰直观的方式展示了 InWild 子集中各子任务标签之间的关系和转换，并突显了它们之间的相互依赖性。

过程，代表二阶随机游走：

$$S_{\text{trend}}^{(t)} = \sum_{n=1}^{t}\sum_{m=1}^{n} X_m, \quad \text{where} \quad X_t \sim \mathcal{N}(0,\sigma^2). \tag{5}$$

在数据集生成过程中，该趋势的平滑度由二阶差分参数 δ_s (delta_s) 控制，该参数对应方差 σ^2，并从 $[0.01, 0.1]$ 中采样。基信号的总体振幅从 $[0.1, 1000]$ 中采样。

- 噪声基信号 ($S(t)$ 噪声) 会引入各种类型的随机波动。它可以从若干不同的统计分布中抽取：

$$S_{\text{noise}}^{(t)} = X_t, \tag{6}$$

其中 X_t 遵循以下其中一种：

$$X_t = \begin{cases} \mathcal{N}(\mu,\sigma^2), & \text{Gaussian} \\ U(a,b), & \text{Uniform} \\ \text{round}(V_t/q)\cdot q,\ V_t \sim U(a,b), & \text{Quantization} \\ X_{t-1} + V_t,\ V_t \sim U(a,b), & \text{Random walk} \end{cases} \tag{7}$$

每种噪声类型的具体参数（例如，μ，σ^2 表示高斯噪声，a，b 表示均匀分布噪声）均经过配置，以生成多种噪声轮廓。对于量化噪声，步长 q 被专门设定为信号振幅的十分之一。

- 分量权重(w) 在生成三个基础信号后，它们各自对最终时间

表10：本表横向比较了现有的时间序列理解与推理数据集。时间序列类型列表示时间序列数据的来源，其中“R”表示真实世界数据，“S”表示合成数据。领域列表示时间序列所属的应用领域。问题类型列指定数据集中包含的问题类型，例如选择题或数值题。分类体系列展示了各数据集如何对不同的能力维度进行分类。输入方法列描述了这些工作如何处理时间序列，包括文本式时间序列、时间序列编码和时间序列词元。跨变量列表示数据集是否包含跨变量分析任务。最后，生成列概述了不同数据集的构建方法。如需查看详细的比较表，请参阅 https: //anonymous.4open.science/r/MMTS-BENCH-BEF7/comparison.md

数据集	时间序列类型	领域	Size	问题类型	分类体系	推理	输入方法	跨变量	生成
EngineMT-QA	R	单变量	11K/–	选择题	Flat	✓	时间序列编码	×	模板 +润色
Time-MQA-TSQA	R	多项	200K/1.4K	选择题+编号。	Flat	✓	文本式时间序列	×	单轮提示
ChatTime-TSQA	S	–	48K/0	选择题	Flat	×	时间序列词元	×	模板
TimeSeriesExam	S	–	746/–	选择题	Flat	✓	–	✓	模板
ChatTS	R+S	多项	11K/525	选择题+数量。	Flat	✓	时间序列编码	✓	时间序列自进化
Chat-TS	R	多项	3741/100	选择题	Flat	✓	时间序列词元	×	单轮提示
MMTS-BENCH	R+S	多项	2524/1724	选择题+编号	多层级	✓	–	✓	三阶段提示 +模板

序列由一组权重决定。这些权重$\boldsymbol{w} = (w_{\text{trend}},\ w_{\text{seasonal}},\ w_{\text{noise}})$从对称狄利克雷分布$\boldsymbol{w} \sim \text{Dir}(\boldsymbol{\alpha})$中采样，其中$\boldsymbol{\alpha} = (1,\ 1,\ 1)$。这确保了无偏组合，其中权重均为正数且总和为一。

在构建子集期间，Feature Analysis（特征分析）D_f维度中四类子任务的事实标签，会根据底层基本组件的参数体系系统地生成。对于 Structural Awareness（结构感知）D_s维度中的子任务，则采用差异化的构建策略：

- **单变量 vs. 多变量：**生成多变量序列，以评估模型在分析单变量与多变量统计性质方面的差异化能力。
- **局部 vs. 全局：**通过在合成时间序列中定义子区间来创建描述类和定位类任务，从而测试模型的局部和全局感知能力。
- **平稳性与非平稳性：**复合时间序列由两个统计性质不同的子序列拼接而成。随后将这些序列与特征分析任务结合，用于评估模型识别非平稳性的能力。

问答对基于一组包含 17 种不同模板的模板系统化生成。这些模板涵盖了多种任务，从定性特征分析到定量数值计算。除了一部分需要人工标注的区间定位问题外，绝大多数问答对都是自动生成的。这种基于模板的自动化构成了我们流水线的执行层，其相较于以往工作的独特优势——尤其是在多样性和对齐精度方面——将在下文详述。

生成流水线的进展。 模块化合成是 TimeSeriesExam[4] 和 ChatTS [42], 等近期研究共同采用的范式，而我们的流水线则在多样性、精度和对齐方面引入了关键改进。

- **生成多样性：**与依赖简单基础模式（例如线性、指数）和受限子选项的 TimeSeriesExam 不同，我们采用受 STL 启发的分解方法，将时间序列拆分为趋势、季节性和噪声模块，并提供丰富的控制参数。例如，我们的趋势通过二阶随机游走（由 δ^2 和初始值控制）生成，而非简单函数；此外，我们还纳入了多种噪声类型（例如量化噪声、随机游走噪声）和灵活的组合策略（加性、串联），超越了 ChatTS 中采用的固定组合方案。
- **精度与对齐：**我们记录精确的定量参数（例如具体的波形配置），而非定性标签。这种细粒度记录使问答生成引擎能够利用精确的数值，从而构建细致的评估条目——例如，根据季节性_强度参数区分弱、中、强季节性——而不是将评估局限于对季节性是否存在的二元判断。

A.4.2 真实世界数据集。 为全面评估多模态时间序列理解在不同分析维度上的表现，我们从真实世界的 LOSTA 数据中构建了三个专门的子集，分别针对时间序列分析的不同方面：全面的基于问答的评估（**InWild**）、序列相似性匹配（**匹配**）以及跨模态语言理解（**对齐**）。每个子集都采用系统化的构建流程，结合特定领域的采样策略，并基于五个关键领域的真实时间数据，运用自动生成框架进行构建。

INWILD 子集

该子集通过一种创新的渐进式多轮对话方法，评估模型在时间序列分析中的综合理解与推理能力。该子集采用开放式问题模板，确保内容丰富多样，同时避免僵化的模式。生成过程协同整合了三个分析维度：结构感知、特征分析和时间推理，形成 140 种独特的维度组合，促使模型对潜在模式及其动态关系进行深入推断。

三阶段渐进式生成流水线我们的数据集构建遵循系统化的三阶段流程，其中每个

阶段在创建高质量问答对的过程中各司其职。

- **阶段 1：初始化与上下文生成** 该过程从随机采样开始，用于定义核心参数，包括问题类型（选择题或判断题）、任务类型（结合特征分析与时间推理能力）、变量数量（单变量或多变量）以及分析范围（局部子序列或全局序列）。系统为 Claude 3.7 Sonnet 提供全面的多模态上下文：时间序列可视化结果、原始数值序列、领域元数据，以及来自专用库的预计算统计特征。模型会生成与指定任务类型相匹配的结构化文本描述。

- **阶段 2：任务规范与推理构建** 在初始描述的基础上，本阶段根据预定义的变量和分析范围进一步明确任务。对于局部范围的问题，系统会提供关键时间点的索引。模型对结构化信息进行抽象，以构建具体问题、真实答案和详细的推理解释。

- **阶段 3：格式化与质量验证** 模型根据指定的问题类型，将输出整理为标准化的问答对，同时自动从三个维度进行一致性验证：逻辑与计算的数学正确性、文本内容与数据之间的描述准确性，以及推理连贯性的逻辑可解释性。

Match子集

匹配 子集专门用于评估模型在时间序列相似性匹配和形态对应分析方面的能力。该子集采用标准化的问答模板，并固定结构感知参数（平稳性、单变量和全局序列），同时系统地改变序列匹配维度，构建四种不同的匹配范式。

四种匹配任务：**(1) 同构匹配** 评估模型在相同时间尺度和数据分布下识别序列相似性的基本能力，主要考察其识别统计特征和动态模式的准确率；**(2) 鲁棒匹配** 评估模型在移动平均平滑等数据预处理变换下保持匹配准确率的能力，以检验其对数据质量变化的适应性；**(3) 定位匹配** 评估模型在较长时间窗口内定位目标时间模式的精确度，重点考察时间模式检索和空间定位表现；**(4) 反向匹配**评估模型对时间方向变换的适应能力，检验其在时间反转条件下识别序列对应关系的能力。

这四个渐进式难度梯度全面考察了模型在各种约束和挑战性场景下进行时间序列相似度测量、模式对齐和形态识别的综合能力。

数据集构建流程 构建过程首先依据序列的典型周期性模式对序列进行分段。采用滑动窗口方法，我们计算动态时间规整（DTW）距离，以识别四个片段：DTW 距离最小的片段、距离中位数对应的片段、距离最大的片段以及距离第二大的片段。随后，我们随机打乱这四个片段的子序列，将其作为问答对中的四个选项。我们使用标准化的问答模板构建最终数据集，涵盖两个领域的四种序列匹配任务。

ALIGN 子集

对齐子集评估模型在时间序列数据与自然语言之间进行双向转换的能力。该子集被有意设计为对齐校准子任务，旨在评估模型能否在语言歧义较低的条件下，将时间序列准确匹配到自然语言描述。因此，这些描述在趋势范围和幅度变化方面均有意设置得十分具体，从而确保评估重点在于精确的跨模态对齐。

双向跨模态任务（1）**时间序列到语义转换**要求模型识别出与给定时间序列的统计特征和动态模式最匹配的文本描述。（2）**语义到时间序列转换**要求模型从候选序列中选择与给定自然语言描述的趋势、波动和周期性最相符的时间序列。

数据集构建流程该子集采用对称式构建，以确保任务的一致性。数据来源于三个现实世界领域：Traffic、云运维和气候。为防止模型基于领域或数值范围进行捷径学习，我们实施了两项策略：**（1）数值缩放**。对云运维和气候序列进行缩放，使其范围相近，以减轻基于幅度的偏差。**（2）受控采样**。对于每个正确样本，我们选择三个干扰项，并确保至少有一个干扰项来自相同领域，且与正确答案具有相似的统计分布。这迫使模型依赖细粒度模式和数值细节，而非领域先验。

对于时间序列到语义的任务，Claude 3.7 Sonnet 会为四个时间样本生成特征描述，作为答案选项。对于语义到时间序列的任务，模型会生成正确样本的结构化描述，将其作为问题提示词，并以时间序列作为选项。

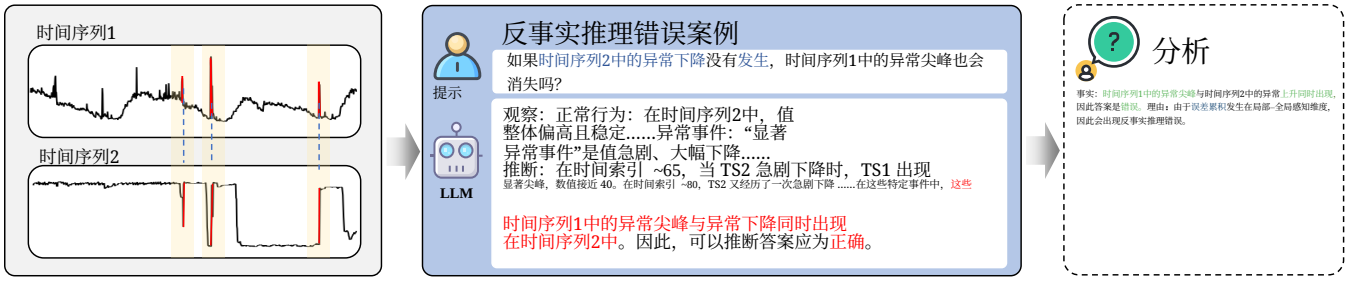
B 评估与实验设置

我们使用以下提示词来规范选择题问答和数值问答的评估。系统提示词规定答案格式和歧义处理策略；用户提示词则注入每个条目的内容。模型输出通过提取 <final_answer> 标签内的单个字母或数字进行评分。

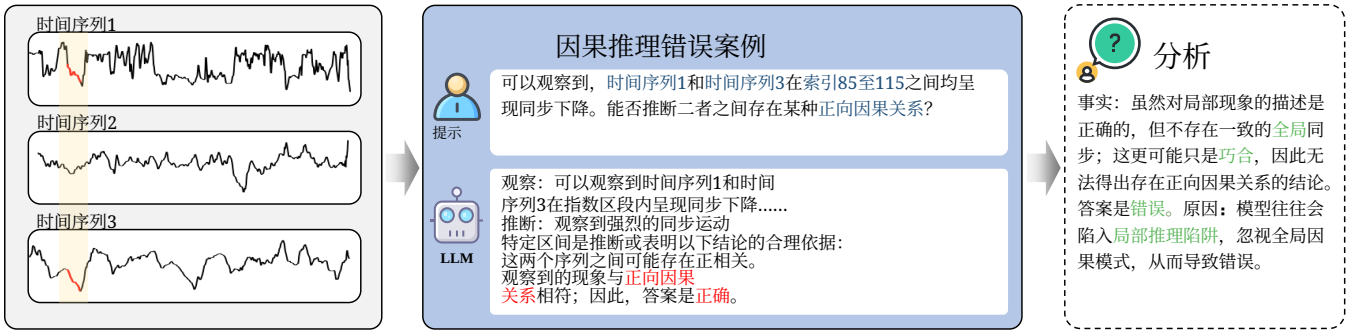
选择题问答的系统提示词：

```
你是一名专家级 AI 助手，专门以高准确率和一致性回答选择题。你的任务

重要指示：1. 只能从给定选项中选择答案 2. 最终答案必须是一个字母
```



(a) 反事实推理错误案例分析



(b) 因果推理错误案例分析

图 8：上方子图展示了一个反事实推理错误案例，下方子图展示了一个因果推理错误案例。在两个子图中，左侧面板将时间序列数据可视化，中间面板呈现问题及大语言模型回答，右侧面板给出真实答案并附有错误分析。红色文本突出显示错误的推理要点，绿色文本表示正确的推理要点，蓝色文本标示问题中的关键词。

合理的解释 4. 不要编造问题中未提供的信息 回复格式：你必须严格按照

数值问答系统提示词：

```
你是一名专业的 AI 助手，擅长以高度准确且一致的方式回答数值问题。你

重要指示：。
1

回答格式：你必须严格按照以下格式组织最终答案：<final_answer >将你
```

请记住：你的最终答案只能包含数值，不能包含其他任何内容。

用户提示词：

```
请根据给定信息回答以下多项选择题/数值问题：

问题：{question}{ 给定_值_str }

    请仔细分析这个问题，考虑给定的值和所有可用选项，然后按照系统指令中
```

B.1 时间序列可视化设置

我们遵循 Zhuang 等人 [51] 的绘图风格，并将其调整用于多通道时间序列。具体而言，我们保留单通道分辨率（1500×320 为 100 每英寸点数），并通过堆叠逐通道子图（每个通道的固定高度为 320 px，分辨率为 100 每英寸点数），使图像高度随通道数量线性缩放。这样既能确保各通道的时间轴保持一致，也能维持各通道间可比的垂直分辨率。

```
定义 绘制_时间_序列_为_图像(值_列表):如果长度(值_列表) > 8: 数
量_通道 = 1 # 单通道否则:
```

```
    num_通道 = 长度(值_列表)  # 多通道

宽度_英寸 = 15.0  # 1500 px / 100 每英寸点数
高度_每_通道 = 3.2  # 320 像素 / 100 每英寸点数
总计_高度 = 高度_每_通道 * 数量_通道
fig = 绘图图形(
    图形尺寸 =( 宽度_英寸, 总计_高度),  每英寸点数 =100
)

if num_通道 == 1:
    plt.plot(范围(len(值_list)),
             值_list , 'b-', linewidth =1.5)
    plt.title('时间 序列 ', 字体大小 =12)
    plt.xlabel('时间 索引 ', 字体大小 =10)
    plt.ylabel('值 ', 字体大小 =10)
    plt.grid(True , alpha =0.3)
    plt.xlim(0, len(值_list) - 1)
else:
    for i , ch 中 枚举(值_列表):
        plt.subplot(num_通道 , 1, i + 1)
        plt.plot(范围(长度(ch)),
                 ch , 'b-', linewidth =1.5)
        plt.title(f'时间 序列 {i+1}', 字号 =10)
        plt.xlabel('时间 索引 ', 字号 =8)
        plt.ylabel('值 ', 字体大小 =8)
        绘制网格(True , 透明度 =0.3)
        设置 x 轴范围(0, 长度(ch) - 1)
紧凑_布局 ()
```

B.2 消融研究的实验设置

在ChatTS的[5] 原始实现中，它采用Qwen2.5-14B-Instruct作为骨干大语言模型，并以一个5层多层感知机作为时间序列编码器。在训练过程中[6], , 文本嵌入与时间序列嵌入进行对齐，从而使模型具备时间序列推理能力。为考察编码器的作用，我们将多层感知机替换为其他架构，包括不同深度的卷积神经网络和Transformer编码器。我们进一步测试了向时间序列输入中引入可学习位置嵌入或基于索引的位置特征所产生的效果。

我们的实验使用 Qwen2.5-3B-Instruct 作为大语言模型骨干，并报告其在 MMTS-Bench 上的纯文本基线性能。为进行比较，我们还纳入了 Qwen2.5-14B-Instruct 的性能，以便单独考察大语言模型骨干规模的影响。最后，由于 ChatTS 使用了包含统计信息（例如偏移量、缩放因子、长度、最大值/最小值、左/右边界值）的提示前缀，我们测试了使用和不使用该前缀训练的模型，以衡量其贡献。

所有模型均在 **InWild**、**匹配** 和 **对齐** 上进行评估。

[5]https://huggingface.co/bytedance-research/ChatTS-14B [6] https://github.com/xiezhe-24/ChatTS-Training

C 完整结果

表11：不同模型在基础子集的 Choice 子集上的准确率指标。 ^{cot} 表示思考模式。 [1] 表示在没有任何时间序列输入的情况下评估的模型。 [2] 表示不含内置统计计算模块的 ChatTS。-VL =视觉-语言。"TS"表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。"–"表示模型未能正确作答。粗体加下划线的值表示各类别中每项指标的最佳表现，粗体值表示次佳表现。Stat. 和 Non-Stat. 列分别表示涉及平稳和非平稳时间序列的问题。

类别	模型名称	模态	总计	趋势	季节性	噪声	局部	总体
开源	DeepSeek-V3	Text	0.41	0.49	0.37	0.33	**0.53**	0.49
	Kimi-K2	Text	**0.45**	**0.50**	0.40	**0.39**	**0.55**	**0.50**
	Qwen3-32b^{cot}	Text	**0.42**	**0.53**	**0.45**	**0.37**	0.32	**0.53**
	Qwen3-32b	Text	0.40	0.46	0.37	0.35	0.47	0.46
	Qwen3-8b^{cot}	Text	0.35	0.38	0.40	0.33	0.26	0.38
	Qwen3-8b	Text	0.32	0.33	0.27	0.26	0.50	0.33
	Qwen2.5-32b	Text	0.35	0.39	0.28	0.30	0.48	0.39
	Qwen2.5-14b	Text	0.35	0.39	0.31	0.27	0.46	0.39
	Qwen2.5-7b	Text	0.33	0.44	0.22	0.28	0.46	0.44
	Qwen2.5-32b[1]	Text	0.30	0.39	0.23	0.19	0.44	0.39
	Qwen2.5-7b-VL	Text	0.29	0.35	0.23	0.24	0.36	0.35
	Qwen2.5-7b-VL	视觉	0.32	0.32	**0.41**	0.28	0.29	0.32
	Qwen2.5-7b-VL	V+T	0.34	0.41	0.34	0.27	0.34	0.41
闭源	Claude-3.7-Sonnet	Text	0.49	**0.59**	0.54	0.35	0.52	**0.59**
	Claude-Sonnet-4	Text	0.49	0.57	0.49	0.35	**0.63**	0.57
	Gemini-2.5-Pro	Text	0.48	0.50	0.51	0.39	0.59	0.50
	Gemini-2.5-Flash	Text	0.39	0.49	0.36	0.24	0.56	0.49
	GPT-5-Minimal	Text	0.45	0.45	0.40	**0.42**	0.56	0.45
	GPT-5-High	Text	**0.51**	0.53	0.51	**0.45**	**0.60**	0.53
	GPT-4o	Text	0.42	0.51	0.36	0.34	0.52	0.51
	GPT-4o	视觉	**0.55**	**0.60**	**0.64**	**0.42**	0.56	**0.60**
	GPT-4o	V+T	**0.51**	0.53	**0.60**	0.39	0.56	0.53
	GPT-4o-mini	Text	0.36	0.43	0.34	0.28	0.44	0.43
时间序列	ChatTS	TS	0.39	0.42	0.39	0.31	0.49	0.37
	ChatTS[2]	TS	0.39	0.37	0.39	0.34	0.53	0.36
	ITFormer	TS	0.31	0.30	0.29	0.28	0.42	0.29
	ChatTime	TS	–	–	–	–	–	–

表12：不同模型在基础子集数值划分上的*Accuracy@10%* 指标。 *cot* 表示思考 模式。 [2]表示不含内置统计计算模块的ChatTS。- VL = 视觉-语言。"TS"表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。"–"表示模型未能正确作答。加粗并带下划线的值表示各指标在每个类别中的最佳表现，加粗值表示次佳表现。平稳与非平稳列分别表示包含平稳时间序列和非平稳时间序列的问题。

类别	模型名称	模态	总计	趋势	季节性		基本统计量		非平稳	局部	总体	单变量		多变量
开源	DeepSeek-V3	Text	**0.41**	**0.08**	0.02	**0.57**	**0.57**	**0.52**	**0.21**	**0.47**	**0.57**	**0.56**		
	Kimi-K2	Text	**0.42**	0.04	**0.13**	**0.56**	**0.56**	**0.56**	0.19	**0.45**	**0.56**	**0.54**		
	Qwen3-32b*cot*	Text	0.31	0.00	**0.10**	0.40	0.40	0.40	0.20	0.32	0.40	0.33		
	Qwen3-32b	Text	0.35	0.02	0.03	0.47	0.47	0.50	0.07	0.38	0.47	0.42		
	Qwen3-8b*cot*	Text	0.27	0.00	0.00	0.34	0.34	0.38	0.13	0.27	0.34	0.29		
	Qwen3-8b	Text	0.26	0.02	0.03	0.33	0.33	0.39	0.07	0.27	0.33	0.26		
	Qwen2.5-32b	Text	0.34	0.03	0.00	0.45	0.45	0.50	0.11	0.36	0.45	0.44		
	Qwen2.5-14b	Text	0.25	0.05	0.00	0.29	0.29	0.41	0.04	0.24	0.29	0.34		
	Qwen2.5-7b	Text	0.18	**0.06**	0.00	0.19	0.19	0.31	0.01	0.16	0.19	0.31		
	Qwen2.5-7b-VL	Text	0.14	0.03	0.03	0.14	0.14	0.23	0.04	0.12	0.14	0.21		
	Qwen2.5-7b-VL	视觉	0.25	0.05	0.09	0.28	0.28	0.27	**0.27**	0.24	0.28	0.35		
	Qwen2.5-7b-VL	V+T	0.19	0.05	0.04	0.21	0.21	0.27	0.09	0.18	0.21	0.31		
闭源	Claude-3.7-Sonnet	Text	**0.54**	0.13	0.20	**0.66**	**0.66**	**0.65**	0.44	**0.55**	**0.66**	**0.66**		
	Claude-Sonnet-4	Text	0.53	0.04	0.33	0.62	0.62	0.61	**0.53**	0.50	0.62	0.66		
	Gemini-2.5-Pro	Text	**0.63**	0.14	**0.41**	**0.72**	**0.72**	**0.69**	**0.71**	**0.60**	**0.72**	**0.72**		
	Gemini-2.5-Flash	Text	0.43	0.04	0.20	0.56	0.56	0.55	0.20	0.45	0.56	0.40		
	GPT-5-Minimal	Text	0.42	0.10	0.09	0.58	0.58	0.53	0.16	0.48	0.58	0.57		
	GPT-4o	Text	0.41	0.06	0.09	0.53	0.53	0.58	0.13	0.43	0.53	0.51		
	GPT-4o	视觉	0.28	**0.17**	0.10	0.34	0.34	0.29	0.31	0.31	0.34	0.34		
	GPT-4o	V+T	0.51	**0.42**	**0.37**	0.59	0.59	0.57	0.31	0.56	0.59	0.56		
	GPT-4o-mini	Text	0.33	0.02	0.00	0.40	0.40	0.51	0.09	0.32	0.40	0.43		
时间序列	ChatTS	TS	0.37	0.36	0.43	0.41	0.41	0.42	0.12	0.40	0.41	0.40		
	ChatTS[2]	TS	0.01	0.00	0.45	0.01	0.01	0.01	0.11	0.01	0.01	0.02		
	ITFormer	TS	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.00	0.00		
	ChatTime	TS	–	–	–	–	–	–	–	–	–	–		

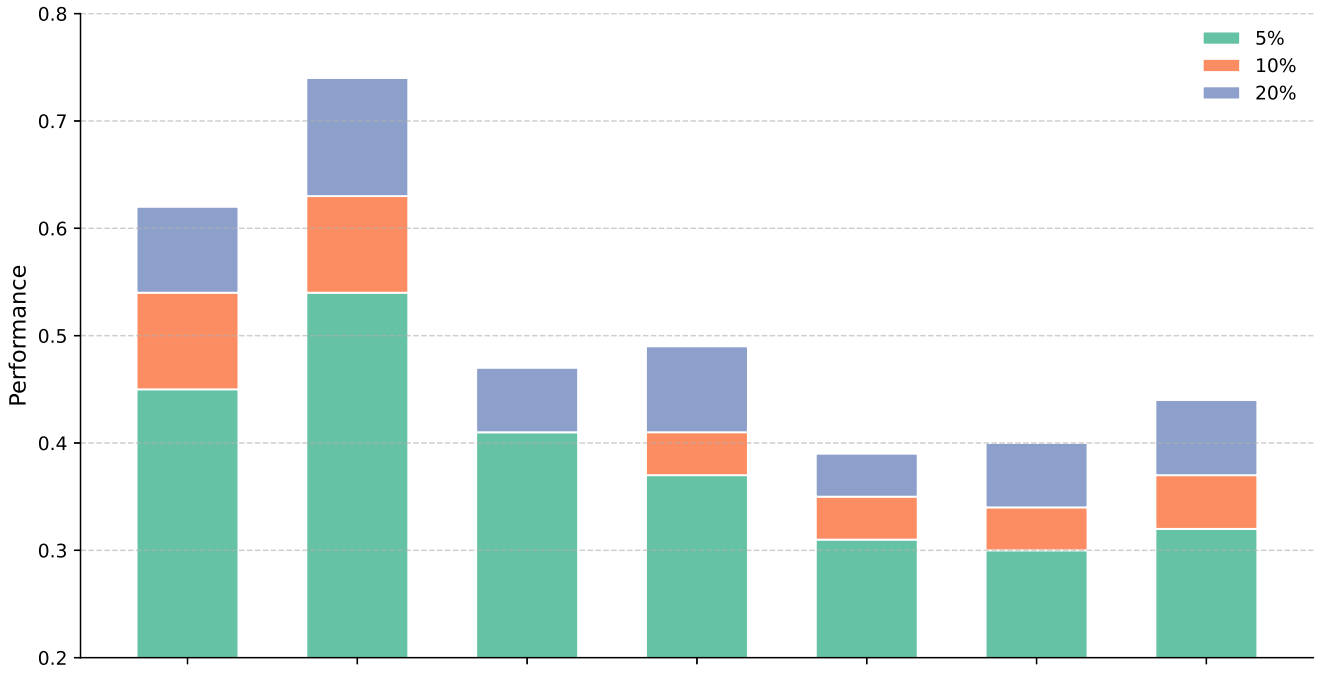


图9：该堆叠条形图展示了若干代表性模型（闭源、开源和时间序列大语言模型）在 Basesubset 的数值划分上的 *Accuracy@N%* 表现。为探究数值推理能力的层次化分布，我们报告了 N = 5、10、20 时的结果。

表13：不同模型在基础子集数值划分上的相对准确率指标。 *cot* **表示思考模式。** [2]**表示未内置统计计算模块的 ChatTS。-VL = 视觉-语言。'TS'表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。'–'表示模型未能正确作答。粗体下划线值表示各类别中每项指标的最佳表现，粗体值表示次佳表现。Stat. 和 Non-Stat. 列分别表示涉及平稳和非平稳时间序列的问题。**

类别	模型名称	模态	总计	趋势	季节性		基本统计量		非平稳	局部	总体	单变量		多变量
开源	DeepSeek-V3	Text	**0.59**	**0.27**	0.32	**0.73**	**0.73**	**0.66**	**0.41**	**0.63**	**0.73**	**0.76**		
	Kimi-K2	Text	**0.65**	**0.37**	**0.49**	**0.74**	**0.74**	**0.71**	**0.56**	**0.66**	**0.74**	**0.72**		
	Qwen3-32b^{cot}	Text	0.42	0.00	0.30	0.51	0.51	0.52	0.27	0.41	0.51	0.47		
	Qwen3-32b	Text	0.49	0.07	0.25	0.61	0.61	0.64	0.29	0.50	0.61	0.61		
	Qwen3-8b^{cot}	Text	0.37	0.00	0.19	0.47	0.47	0.49	0.19	0.37	0.47	0.43		
	Qwen3-8b	Text	0.40	0.13	0.19	0.49	0.49	0.54	0.16	0.42	0.49	0.46		
	Qwen2.5-32b	Text	0.51	0.16	0.13	0.66	0.66	0.63	0.30	0.56	0.66	0.64		
	Qwen2.5-14b	Text	0.44	0.14	0.10	0.51	0.51	0.59	0.29	0.43	0.51	0.55		
	Qwen2.5-7b	Text	0.35	0.20	0.12	0.40	0.40	0.49	0.14	0.36	0.40	0.50		
	Qwen2.5-7b-VL	Text	0.32	0.16	0.15	0.37	0.37	0.43	0.12	0.33	0.37	0.42		
	Qwen2.5-7b-VL	视觉	0.40	0.26	**0.36**	0.44	0.44	0.48	0.22	0.4	0.44	0.50		
	Qwen2.5-7b-VL	V+T	0.19	0.05	0.04	0.21	0.21	0.27	0.09	0.18	0.21	0.31		
闭源	Claude-3.7-Sonnet	Text	**0.74**	0.51	0.59	**0.80**	**0.80**	**0.79**	0.72	**0.74**	**0.80**	**0.80**		
	Claude-Sonnet-4	Text	0.73	0.36	0.61	0.78	0.78	0.79	**0.79**	0.69	0.78	**0.80**		
	Gemini-2.5-Pro	Text	**0.82**	**0.56**	**0.78**	**0.85**	**0.85**	**0.83**	**0.88**	**0.79**	**0.85**	**0.83**		
	Gemini-2.5-Flash	Text	0.57	0.10	0.55	0.69	0.69	0.66	0.34	0.57	0.69	0.53		
	GPT-5-Minimal	Text	0.61	0.24	0.33	0.77	0.77	0.68	0.45	0.66	0.77	0.78		
	GPT-4o	Text	0.60	0.32	0.35	0.73	0.73	0.7	0.38	0.64	0.73	0.67		
	GPT-4o	视觉	0.58	0.30	0.71	0.60	0.60	0.55	0.63	0.55	0.60	0.57		
	GPT-4o	V+T	0.70	**0.59**	**0.75**	0.76	0.76	0.70	0.60	0.73	0.76	0.74		
	GPT-4o-mini	Text	0.50	0.17	0.14	0.60	0.60	0.66	0.31	0.51	0.60	0.61		
时间序列	ChatTS	TS	0.55	0.46	0.79	0.59	0.59	0.56	0.34	0.56	0.59	0.56		
	ChatTS[2]	TS	0.19	0.06	0.76	0.10	0.10	0.10	0.36	0.09	0.10	0.10		
	ITFormer	TS	0.15	0.02	0.06	0.15	0.15	0.19	0.23	0.12	0.15	0.18		
	ChatTime	TS	–	–	–	–	–	–	–	–	–	–		

表14：不同模型在基础子集数值划分上的平均偏移量指标。 *cot* **表示思考模式。** [2]**表示未内置统计计算模块的 ChatTS。-VL = 视觉-语言。'TS'表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。'–'表示模型未能正确作答。粗体下划线值表示各类别中每项指标的最佳表现，粗体值表示次佳表现。Stat. 和 Non-Stat. 列分别表示涉及平稳和非平稳时间序列的问题。~~H 标记表示平均偏移量高于 1e5~~。**

类别	模型名称	模态	总计	趋势	季节性	基础	平稳	非平稳	局部	总体	单变量	多变量
开源	DeepSeek-V3	Text	**1.06**	**1.17**	0.68	**0.69**	**0.69**	1.64	**0.69**	**0.79**	**0.69**	**1.94**
	Kimi-K2	Text	**1.24**	**1.80**	**0.57**	**1.14**	**1.14**	1.68	**0.49**	**1.27**	**1.14**	**1.59**
	Qwen3-32b*cot*	Text	49.19	512.64	1.07	2.32	2.32	3.73	8.45	107.63	2.32	4.94
	Qwen3-32b	Text	4.57	34.94	0.75	1.43	1.43	1.48	2.72	8.34	1.43	3.17
	Qwen3-8b*cot*	Text	54.04	568.67	1.04	2.45	2.45	3.47	7.73	119.29	2.45	5.25
	Qwen3-8b	Text	6.01	39.83	0.81	3.05	3.05	1.76	4.96	10.64	3.05	4.15
	Qwen2.5-32b	Text	1.65	5.91	0.92	1.49	1.49	**1.05**	1.22	2.40	1.49	3.77
	Qwen2.5-14b	Text	16.56	170.23	0.90	2.06	2.06	**1.12**	1.33	36.76	2.06	5.37
	Qwen2.5-7b	Text	3.02	3.49	0.88	3.20	3.20	1.97	6.00	3.26	3.20	4.81
	Qwen2.5-7b-VL	Text	H	7.06	1.82	5.55	5.55	H	98.73	5.86	5.55	15.49
	Qwen2.5-7b-VL	视觉	H	13.24	**0.61**	31.51	31.51	1.32	1.17	28.08	31.51	H
	Qwen2.5-7b-VL	V+T	H	34.87	1.01	H	H	H	17.56	H	H	4.15
闭源	Claude-3.7-Sonnet	Text	**0.48**	**1.24**	0.46	**0.32**	**0.32**	**0.39**	0.62	**0.51**	**0.32**	0.50
	Claude-Sonnet-4	Text	**0.72**	**1.26**	0.56	0.62	0.62	0.78	**0.58**	0.75	0.62	**0.44**
	Gemini-2.5-Pro	Text	1.13	9.62	**0.27**	0.36	0.36	**0.30**	**0.35**	2.21	0.36	**0.27**
	Gemini-2.5-Flash	Text	34.04	350.9	0.62	2.49	2.49	1.48	8.87	74.38	2.49	5.57
	GPT-5-Minimal	Text	0.91	1.85	0.69	0.70	0.70	0.97	0.83	0.94	0.70	0.67
	GPT-4o	Text	2.27	16.22	0.71	0.45	0.45	1.18	1.40	3.70	0.45	2.57
	GPT-4o	视觉	1.47	2.70	0.30	2.46	2.46	1.10	0.63	2.50	2.46	4.18
	GPT-4o	V+T	**0.72**	1.66	**0.25**	**0.31**	**0.31**	0.73	1.35	**0.59**	**0.31**	0.48
	GPT-4o-mini	Text	1.24	2.58	0.86	0.94	0.94	1.30	1.21	1.28	0.94	1.63
时间序列	ChatTS	TS	1.79	6.75	0.23	1.68	1.68	1.34	0.90	2.72	1.68	2.26
	ChatTS[2]	TS	21.89	130.25	0.24	20.54	20.54	8.88	0.88	43.18	20.54	7.21
	ITFormer	TS	H	7.08	0.94	10.89	10.89	1.59	0.79	10.10	10.89	H
	ChatTime	TS	–	–	–	–	–	–	–	–	–	–

表15：不同模型在 InWild 子集上的性能。 *cot* **表示思考 模式。** [1] **表示在没有任何时间序列输入的情况下评估的模型。** [2] **表示不含内置统计计算模块的 ChatTS。-VL = 视觉-语言。"TS"表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。"–"表示该模型未能正确响应。在推理任务中，缩写为：演绎（演绎）、归纳（归纳）、类比（类比）和计数（反事实）。每个类别中的最佳结果和次佳结果分别用下划线和加粗标出。**

类别	模型名称	模态	平均值	特征分析						时间推理					
				准确率	趋势	季节性	噪声	波动性	基础	准确率	演绎	归纳	因果	类比	计数
开源	DeepSeek-V3	Text	**0.61**	**0.67**	**0.73**	**0.57**	0.79	**0.55**	**0.75**	**0.59**	**0.60**	**0.63**	**0.52**	0.58	**0.57**
	Kimi-K2	Text	**0.63**	**0.69**	0.71	**0.63**	**0.85**	**0.58**	**0.71**	**0.61**	**0.62**	**0.65**	**0.56**	**0.64**	**0.55**
	Qwen3-32b^{cot}	Text	0.58	0.65	**0.72**	0.54	**0.81**	**0.58**	0.65	0.55	0.52	0.60	0.49	**0.61**	0.48
	Qwen3-32b	Text	0.50	0.56	0.67	0.49	0.54	0.43	0.62	0.48	0.48	0.51	0.42	0.54	0.42
	Qwen3-8b^{cot}	Text	0.50	0.57	0.54	0.51	0.74	0.54	0.55	0.47	0.41	0.52	0.43	0.49	0.48
	Qwen3-8b	Text	0.45	0.48	0.45	0.43	0.65	0.34	0.58	0.44	0.48	0.46	0.42	0.45	0.37
	Qwen2.5-32b	Text	0.53	0.62	0.66	0.53	0.77	0.52	0.62	0.49	0.47	0.53	0.51	0.50	0.44
	Qwen2.5-14b	Text	0.53	0.61	0.60	0.55	0.73	**0.58**	0.63	0.49	0.48	0.53	0.48	0.53	0.41
	Qwen2.5-7b	Text	0.44	0.45	0.61	0.48	0.35	0.35	0.42	0.44	0.41	0.47	0.43	0.45	0.44
	Qwen2.5-7b-VL	Text	0.37	0.37	0.41	0.32	0.44	0.28	0.38	0.37	0.38	0.35	0.40	0.35	0.36
	Qwen2.5-7b-VL	视觉	0.36	0.37	0.39	0.44	0.37	0.30	0.33	0.35	0.38	0.36	0.36	0.33	0.33
	Qwen2.5-7b-VL	V+T	0.39	0.41	0.48	0.41	0.45	0.30	0.40	0.38	0.39	0.37	0.43	0.36	0.35
	Qwen2.5-32b[1]	Text	0.34	0.34	0.40	0.40	0.24	0.33	0.31	0.34	0.35	0.35	0.38	0.31	0.30
闭源	Claude-3.7-Sonnet	Text	0.69	0.78	0.81	**0.72**	0.85	0.73	0.81	0.65	0.66	0.67	0.58	0.71	0.60
	Claude-3.7-Sonnet	视觉	0.69	0.75	0.82	0.67	0.82	0.70	0.75	0.66	0.64	0.72	0.62	0.71	0.60
	Claude-3.7-Sonnet	V+T	**0.73**	0.79	**0.84**	**0.71**	0.87	0.70	0.85	**0.71**	0.66	**0.76**	**0.70**	0.74	0.66
	Claude-Sonnet-4	Text	0.71	0.79	0.78	0.67	**0.91**	0.70	**0.89**	0.68	0.67	0.75	0.60	0.69	0.63
	Claude-Sonnet-4	视觉	0.67	0.74	0.84	0.59	0.87	0.63	0.79	0.64	0.59	0.75	0.55	0.66	0.60
	Claude-Sonnet-4	V+T	0.71	0.78	0.81	0.64	**0.90**	0.72	0.86	0.68	0.65	**0.76**	0.62	0.72	0.66
	Gemini-2.5-Pro	Text	0.68	0.73	0.76	0.69	0.83	0.66	0.72	0.66	0.65	**0.76**	0.56	0.68	0.63
	Gemini-2.5-Pro	视觉	0.72	**0.80**	**0.84**	0.63	0.89	**0.77**	0.85	0.69	**0.69**	**0.76**	0.60	0.71	0.64
	Gemini-2.5-Pro	V+T	**0.76**	**0.83**	**0.86**	**0.71**	0.87	**0.76**	**0.92**	**0.73**	**0.75**	**0.78**	0.62	**0.77**	**0.69**
	Gemini-2.5-Flash	Text	0.50	0.50	0.53	0.40	0.59	0.45	0.55	0.51	0.47	0.57	0.45	0.56	0.45
	GPT-5-High	Text	0.72	0.76	0.82	0.67	0.87	0.60	0.85	0.70	0.69	0.72	**0.66**	**0.76**	**0.68**
	GPT-5-Minimal	Text	0.56	0.64	0.70	0.63	0.72	0.55	0.61	0.52	0.53	0.56	0.51	0.53	0.45
	GPT-4.1	Text	0.58	0.64	0.63	0.57	0.76	0.58	0.68	0.56	0.58	0.61	0.52	0.52	0.55
	GPT-4.1	视觉	0.58	0.61	0.70	0.46	0.70	0.51	0.67	0.57	0.57	0.61	0.57	0.55	0.56
	GPT-4.1	V+T	0.63	0.67	0.73	0.55	0.85	0.60	0.68	0.62	0.62	0.69	0.59	0.59	0.56
	GPT-4.1-Mini	Text	0.59	0.66	0.69	0.49	0.88	0.55	0.73	0.56	0.56	0.60	0.54	0.56	0.49
	GPT-4.1-Mini	视觉	0.49	0.51	0.56	0.37	0.54	0.51	0.55	0.49	0.51	0.45	0.52	0.50	0.48
	GPT-4.1-Mini	V+T	0.58	0.64	0.68	0.54	0.75	0.53	0.73	0.56	0.59	0.61	0.55	0.54	0.44
	GPT-4o	Text	0.62	0.70	0.74	0.61	0.79	0.61	0.75	0.58	0.59	0.62	0.50	0.66	0.50
	GPT-4o	视觉	0.59	0.67	0.76	0.56	0.74	0.64	0.66	0.56	0.55	0.62	0.51	0.57	0.50
	GPT-4o	V+T	0.63	0.71	0.75	0.59	0.85	0.61	0.77	0.59	0.59	0.65	0.55	0.63	0.53
时间序列	ChatTS	TS	0.50	0.55	0.50	0.61	0.53	0.54	0.58	0.48	0.51	0.50	0.49	0.44	0.45
	ChatTS[2]	TS	0.48	0.51	0.53	0.55	0.48	0.52	0.50	0.48	0.50	0.50	0.52	0.42	0.43
	ITFormer	TS	0.33	0.37	0.29	0.31	0.37	0.29	0.36	0.36	0.37	0.40	0.35	0.35	0.35
	ChatTime	TS	–	–	–	–	–	–	–	–	–	–	–	–	–
人类	专家	–	0.67	0.71	0.59	0.67	0.75	0.75	0.80	0.66	0.66	0.72	0.63	0.67	0.58

表16：不同模型在 Match子集上的性能。 *cot* **表示思考 模式。** [1] **表示在没有任何时间序列输入的情况下评估的模型。** [2] **表示不含内置统计计算模块的 ChatTS。-VL = 视觉-语言。"TS"表示时间序列模态，因为时间序列专用模型引入了时间序列编码器。"–"表示该模型未能正确响应。加粗并带下划线的值表示每个类别中各项指标的最佳性能，加粗的值表示次佳性能。**

类别	模型名称	模态	平均同构	鲁棒定位			反向
开源	DeepSeek-V3	Text	**0.65**	**0.90**	**0.79**	**0.57**	0.35
	Kimi-K2	Text	0.60	0.78	0.66	0.50	0.44
	Qwen3-32b^{cot}	Text	0.60	0.80	0.64	0.42	**0.52**
	Qwen3-32b	Text	0.50	0.72	0.58	0.36	0.35
	Qwen3-8b^{cot}	Text	0.50	0.62	0.56	0.36	**0.46**
	Qwen3-8b	Text	0.42	0.56	0.48	0.31	0.32
	Qwen2.5-32b	Text	**0.62**	**0.84**	**0.72**	**0.53**	0.41
	Qwen2.5-14b	Text	0.57	0.78	0.61	0.42	0.45
	Qwen2.5-7b	Text	0.40	0.45	0.49	0.35	0.29
	Qwen2.5-7b-VL	Text	0.30	0.31	0.36	0.27	0.28
	Qwen2.5-7b-VL	视觉	0.28	0.30	0.31	0.28	0.25
	Qwen2.5-7b-VL	V+T	0.34	0.40	0.41	0.27	0.30
	Qwen2.5-32b[1]	Text	0.25	0.25	0.25	0.25	0.25
闭源	Claude-3.7-Sonnet	Text	0.74	0.93	**0.81**	**0.67**	0.54
	Claude-Sonnet-4	Text	0.71	0.85	0.79	**0.61**	0.60
	Gemini-2.5-Pro	Text	**0.79**	**0.96**	0.80	0.59	**0.80**
	Gemini-2.5-Flash	Text	0.44	0.63	0.57	0.28	0.26
	GPT-5-High	Text	**0.81**	**0.98**	**0.81**	0.60	**0.86**
	GPT-5-Minimal	Text	0.57	0.84	0.67	0.53	0.26
	GPT-4.1	Text	0.67	0.89	**0.82**	0.55	0.40
	GPT-4.1-Mini	Text	0.63	0.90	0.78	0.44	0.40
	GPT-4o	Text	0.50	0.68	0.60	0.41	0.34
	GPT-4o	视觉	0.45	0.56	0.50	0.34	0.38
	GPT-4o	V+T	0.55	0.79	0.64	0.38	0.38
时间序列	ChatTS	TS	0.37	0.47	0.40	0.24	0.36
	ChatTS[2]	TS	0.32	0.46	0.41	0.22	0.20
	ITFormer	TS	0.24	0.16	0.25	0.25	0.29
	ChatTime	TS	–	–	–	–	–

表17：不同模型在对齐子集上的性能。 *cot* **表示思考 模式。** [1] **表示在没有任何时间序列输入的情况下评估的模型。** [2] **表示不含内置统计计算模块的 ChatTS。-VL = 视觉-语言。"TS"代表时间序列模态，因为时间序列专用模型引入了时间序列编码器。"–"表示模型未能正确作答。加粗且带下划线的值表示各类别在每项指标上的最佳性能，加粗值表示次佳性能。**

类别	模型名称	模态	平均值	TS→Sem	Sem→TS
开源	DeepSeek-V3	Text	**0.94**	**0.95**	**0.94**
	Kimi-K2	Text	**0.95**	**0.94**	**0.96**
	Qwen2.5-32b	Text	0.93	0.92	**0.94**
	Qwen2.5-14b	Text	0.88	0.87	0.88
	Qwen2.5-7b	Text	0.69	0.68	0.71
	Qwen3-32b^{cot}	Text	0.89	0.92	0.86
	Qwen3-32b	Text	0.87	0.86	0.88
	Qwen3-8b^{cot}	Text	0.86	0.88	0.83
	Qwen3-8b	Text	0.79	0.74	0.84
	Qwen2.5-7b-VL	Text	0.64	0.68	0.59
	Qwen2.5-7b-VL	视觉	0.60	0.61	0.60
	Qwen2.5-7b-VL	V+T	0.73	0.78	0.67
	Qwen2.5-32b[1]	Text	0.27	0.29	0.26
闭源	Claude-3.7-Sonnet	Text	0.97	0.97	**0.98**
	Claude-Sonnet-4	Text	**0.98**	**0.98**	**0.99**
	Gemini-2.5-Pro	Text	0.97	0.97	**0.99**
	Gemini-2.5-Flash	Text	0.94	0.94	0.95
	GPT-5-High	Text	**0.99**	**0.99**	**0.99**
	GPT-5-Minimal	Text	0.97	0.97	**0.98**
	GPT-4o	Text	0.96	0.96	0.97
	GPT-4o-Mini	Text	0.86	0.82	0.90
时间序列	ChatTS	TS	0.80	0.68	0.91
	ChatTS[2]	TS	0.45	0.49	0.42
	ITFormer	TS	0.29	0.32	0.26
	ChatTime	TS	–	–	–

表18：ChatTS模型在不同时间序列长度范围内的Match子集上的表现。总计**对应范围 [13,504], , "–"表示该任务在给定长度范围内没有问题。**

长度范围	同构	鲁棒	定位	反向
总计	0.47	0.40	0.24	0.36
[64, 1024]	**0.53**	**0.57**	0.23	**0.44**
[256, 512]	–	–	**0.36**	–

表19：在 InWild 子集（1,084 个样本）上进行单次评估的词元成本。

模型	输入/输出词元	价格成本
Qwen2.5-32B	≈6M / ≈20k	0
DeepSeek-V3	≈5M / ≈250k	≈$1.62
GPT-4o	≈5M / ≈400k	≈$16.50
Claude-Sonnet-4	≈5M / ≈600k	≈$24.00
Gemini-2.5-Pro	≈6M / ≈500k	≈$12.50

D 统计鲁棒性分析

为评估 MMTS-Bench 测试的稳定性、鲁棒性和有效性，我们开展了全面的统计评估和偏差分析：**Bootstrap 置信区间**、**迭代子采样分析**和**数据集伪影评估**。这些实验分别评估数据集测试的稳定性、其规模是否充分，以及其抵御伪相关的能力。

D.1 自助法置信区间

为评估模型性能的可靠性并量化其不确定性，我们采用了非参数自助法。具体实验设置如下：给定规模为 D 的测试集，我们通过有放回抽样生成了 $N = 1000$ 个同样规模为 D 的自助样本。

我们选取了若干代表性模型（涵盖闭源模型、开源大语言模型和时间序列大语言模型），分别在**InWild** 子集、**Base** 子集的选择题子集以及完整的 MMTS-Bench 数据集上进行评估。随后，我们计算了各模型的准确率分数，以获得该指标的经验分布。基于这一分布，我们报告平均准确率、标准差（std）和 95%置信区间（CI）。

表 20：不同模型在 MMTS-InWild 数据集上的自助法置信区间。

模型	Mean	Std	置信区间下限	置信区间上限
Gemini-1.5-Pro（文本）	0.6827	0.0142	0.6541	0.7103
Gemini-2.5-Pro（视觉）	0.7262	0.0138	0.6983	0.7537
GPT-4o（文本）	0.6128	0.0148	0.5830	0.6421
DeepSeekV3（文本）	0.6218	0.0150	0.5932	0.6504
Qwen2.5-32b（文本）	0.5379	0.0156	0.5083	0.5683

表21：不同模型在 MMTS-Base 的 Choice 子集上的自助法置信区间。

模型	Mean	Std	置信区间下限	置信区间上限
Gemini-1.5-Pro（文本）	0.6340	0.0205	0.5933	0.6726
Claude-3.7-Sonnet（文本）	0.5967	0.0207	0.5581	0.6391
GPT-4o（文本）	0.5609	0.0210	0.5211	0.6021
DeepSeekV3（文本）	0.5433	0.0205	0.5053	0.5845
Qwen2.5-32b（文本）	0.4727	0.0218	0.4331	0.5158

表 22：不同模型在完整MMTS-Bench数据集上的自助法置信区间。

模型	Mean	Std	置信区间下限	置信区间上限
Gemini-1.5-Pro（文本）	0.7235	0.0094	0.7059	0.7422
GPT-4o（文本）	0.6059	0.0103	0.5864	0.6265
DeepSeekV3（文本）	0.6366	0.0100	0.6165	0.6562
Qwen2.5-32b（文本）	0.5789	0.0104	0.5580	0.5986

实验结果（表 20、21、22 和图 10）表明，无论是在完整数据集还是各子集上，所有模型的性能评估结果均表现出较低的统计离散度。具体而言，自助法实验显示，标准差（std）仅介于 $0.01 \sim 0.02$ 之间，且 95%置信区间（CI）的宽度保持在较窄范围内（约为 $0.5 \sim 0.8$ 个百分点）。

这些严格的误差界限有力地证实了 MMTS-Bench 的统计鲁棒性。这表明该基准对数据抽样方差不敏感，能够为跨模型和跨能力比较提供稳定且可复现的评估结果。此外，为进一步抑制模型生成中的随机噪声，我们还在评估过程中引入了多次采样和多数投票机制，从而建立了更鲁棒的性能基线。

D.2 迭代子抽样分析

为探究评估稳定性与数据集规模之间的关系，并估算获得鲁棒结果所需的最小样本量，我们针对代表性模型 Gemini-2.5-Pro 的文本输入开展了迭代子抽样分析。

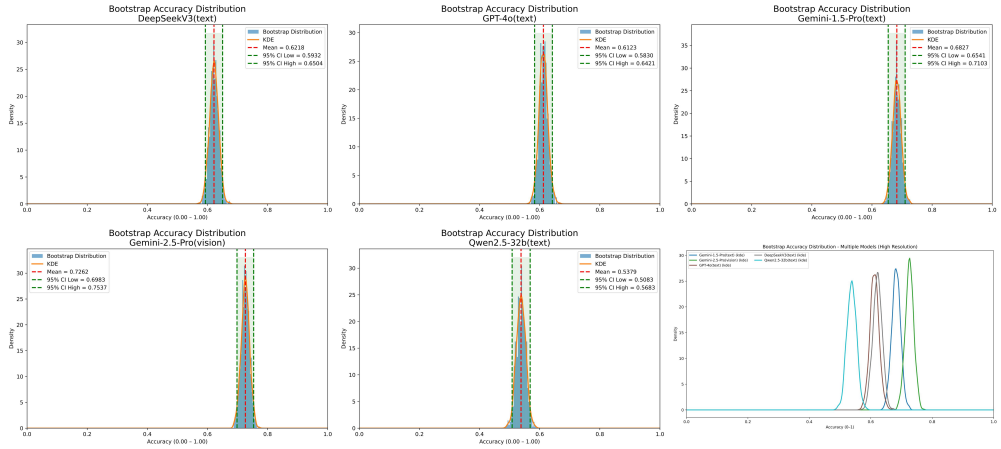
具体实验设置如下：对于给定的数据集规模 D，我们将子抽样规模 S 设为一个变量，使其从初始值逐步增加至 D，增量步长为 $T = 20$。在每个固定规模 S 下，我们进行 $N = 50$ 次独立抽样重复，并计算模型性能的均值、标准差（std）和变异系数（CV）。

我们使用变异系数（$CV = \sigma/\mu$）作为衡量评估稳定性的核心指标。当 CV 曲线随着 S增加而呈下降趋势，并低于预设的收敛阈值 $\tau = 0.02$时，认为数据集大小 S 具备充分的统计稳定性。[7] 本实验涵盖了整个 MMTS-Bench 及其四个子集。

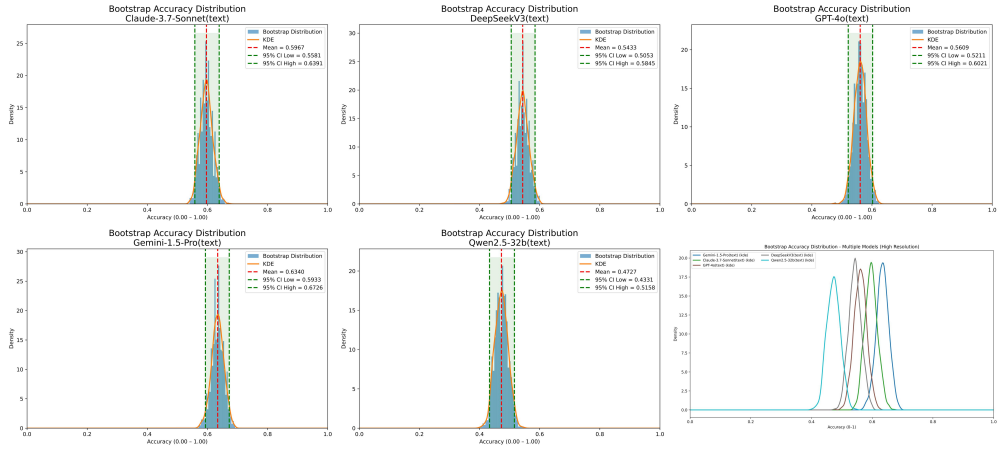
表 23：在子抽样分析实验中，稳定评估所需的最小样本量与实际样本量的比较，以及在四个数据子集和整个 MMTS-Bench 数据集上，模型在实际样本量下的均值、标准差和变异系数。

数据集	Full	Min	Mean	Std	CV
对齐	240	60	0.9813	0.0100	0.0041
基础（选择）	568	400	0.6357	0.0203	0.0038
匹配	400	260	0.7795	0.0199	0.0047
InWild	1084	600	0.6811	0.0130	0.0011
Full	2292	600	0.7199	0.0093	0.0010

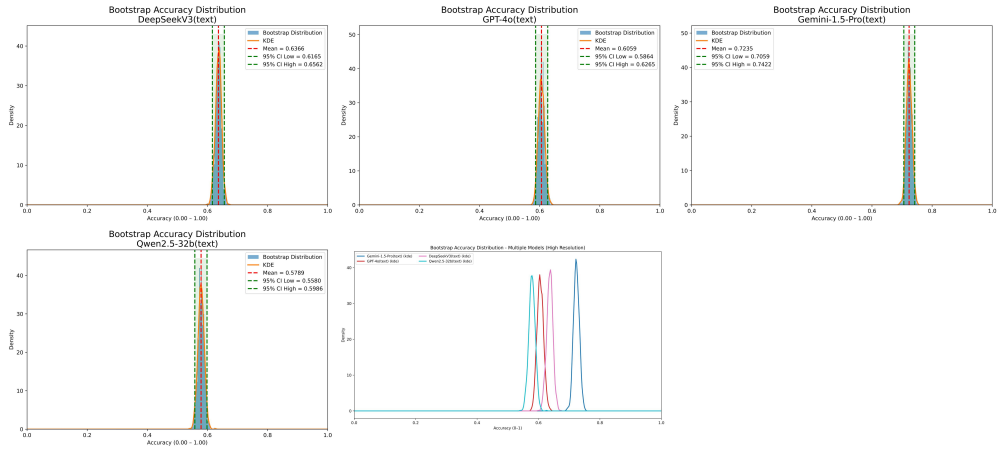
[7]我们通过实证将收敛阈值设为 $\tau = 0.02$，要求评估分数的标准差控制在均值的 2% 以内。对于典型的模型准确率范围（50% ~ 80%），这意味着测量误差的绝对范围限制在约1% ~ 1.6%。这一严格的稳定性约束对于抑制由抽样方差导致的“排名反转”至关重要，可确保该基准能够可靠地区分性能略有差异的模型。



(a) MMTS-InWild



(b) MMTS-Base（选择题子集）



(c) MMTS-Bench

图 10：准确率分布的可视化 **以及不同代表性模型的置信区间。**

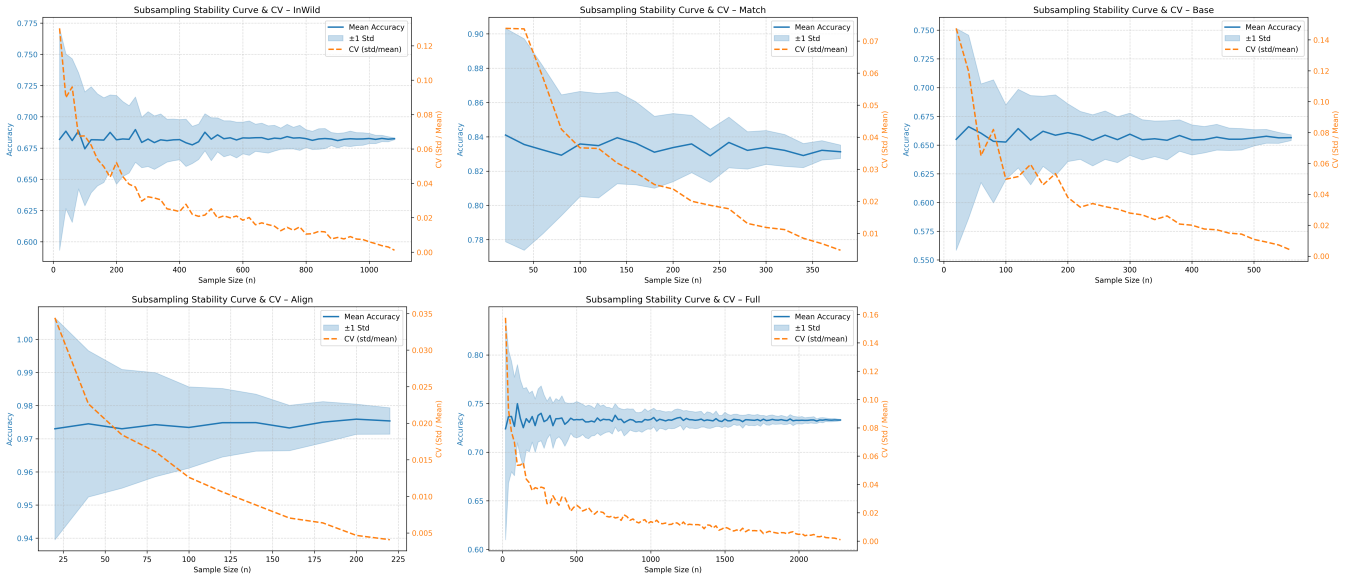


图 11：四个子集及整个 MMTS-BENCH 上模型准确率指标（均值、标准差和变异系数）随子抽样规模变化的趋势。

实验结果（表 23 和图 11）表明，当前 MMTS-BENCH 的数据规模为评估稳定性提供了充足的安全余量。具体而言，每个子集的实际样本量远超达到收敛阈值所需的最小样本数（约为所需最小样本数的 1.42 ~ 4 倍），且最低变异系数已降至 10^{-3} 量级。这一结果证实，我们不仅确保了当前规模下评估结果具有较高的置信度，也在统计稳健性与评估效率（计算和时间成本）之间实现了良好平衡。

D.3 数据集伪影与捷径学习评估

大量研究警示，基准表现可能由伪相关或显式特征驱动，而非真正的推理 [12, 14]。为确保 MMTS-Bench 评估的是鲁棒的时间序列推理，而不是依赖数据集伪影，我们分析了模型表现对显式表层属性的依赖性。具体而言，我们考察了 InWild 子集中时间序列问答准确率与三个显式因素之间的相关性：序列长度（L）、变量数量（V）和问题文本长度（T）。我们评估了三个代表性模型：GPT-4o、Qwen2.5-32B 和 ChatTS[42]。我们引入三项指标来量化这些依赖关系：**(1) 相关性**（r_L，r_T）。准确率与序列长度对数（r_L）或问题文本长度（r_T）之间的皮尔逊相关系数。数值接近 0 表示不存在相关性。**(2) 长度敏感性**（Δ_{long}）。最长四分位数（第 ≥ 75 百分位）与最短四分位数（第 ≤ 25 百分位）样本的平均准确率之差。**(3) 维度差距**（Δ_{dim}）。多变量样本与单变量样本的平均准确率之差。

如表 24 所示，结果表明模型表现对这些伪影的依赖极小。所有模型的表现与序列长度几乎没有相关性（$|r_L| < 0.08$），极端长度样本之间的准确率差距（Δ_{long}）也处于较窄范围内（约为 0.05 ~ 0.08），说明模型并未持续偏向较短或较长的序列。同样，单变量序列与多变量序列之间的表现差距很小（$|\Delta_{\text{dim}}| < 0.04$），问题长度的影响也较弱（$|r_T| \approx 0.15$）。这些发现证实，MMTS-Bench 的表现无法仅凭简单的元数据特征轻易预测。

表 24：InWild 子集上潜在数据集伪影与捷径学习分析。

模型名称	r_L	Δ_{long}	Δ_{dim}	r_T
GPT-4o	-0.0693	0.0824	-0.0101	-0.1451
Qwen2.5-32B	0.0547	-0.0525	0.0353	-0.0264
ChatTS	0.0727	-0.0502	-0.0163	-0.0862

D.4 结论

我们开展了三项系统分析——自助法置信区间、迭代子采样分析和数据集伪影评估。综合来看，这些分析表明，MMTS-Bench 具有较高的统计稳健性、适中的规模，并且能够有效检验模型是否存在捷径学习。结果证实，模型在 MMTS-Bench 上的表现取决于其真正的时间序列理解能力，而非显而易见的表层特征（例如序列长度或维度），因此评估结果稳定、可靠且值得信赖。

E 可复现性声明

本文及附录详细介绍了基准构建和实验设置的实现细节。为便于端到端复现，我们发布了一份匿名化的包含全部数据和代码的仓库位于 https://anonymous.4open.science/r/MMTS-BENCH-BEF7/。我们将在审稿期间维护该匿名仓库，并在论文被接收后迁移至公共代码仓库，同时归档一份快照以支持长期可用。