Simul-LLM: A Framework for Exploring High-Quality Simultaneous Translation with Large Language Models

完成人：邢清画 日期：2024/10/16

<table>
<tr><td rowspan="3">基本信息</td><td>发表刊物</td><td>ACL</td><td>发表年份</td><td>2024</td><td>第一完成单位（国内）</td><td>Oregon State University</td></tr>
<tr><td>作者</td><td colspan="5">Victor Agostinelli, Max Wild, Matthew Raffel, Kazi Ahmed, Asif Fuad, Lizhong Chen</td></tr>
<tr><td>关键词（中文）</td><td colspan="5">大型语言模型（LLM）、
神经机器翻译（NMT）、
同时翻译（SimulMT）、
Simul-LLM、微调、评估</td></tr>
<tr><td></td><td>关键词（英文）</td><td colspan="5">Large Language Models (LLMs),
Neural Machine Translation (NMT),
Simultaneous Translation (SimulMT),
Simul-LLM, Fine-tuning, Evaluation</td></tr>
<tr><td rowspan="3">论文内容</td><td>解决的问题（如有实际应用场景请说明）</td><td colspan="5">这篇论文解决的问题是如何将大型语言模型（LLMs）应用于同时翻译（SimulMT）这一具挑战性的任务。
论文通过开发一个名为 Simul-LLM 的开源框架，探索如何将已针对神经机器翻译（NMT）微调的 LLMs 适配到同时翻译任务中。该框架支持 LLMs 的微调和评估，提出了两种新的提示结构以改进翻译质量，并研究了在不同等待策略(wait-k)下的 LLMs 表现。
实际应用场景包括跨语言的实时翻译，如会议、直播等需要低延迟翻译的场合。</td></tr>
<tr><td>解决问题的方法（采用什么模型框架等）</td><td colspan="5">Simul-LLM 框架：开发了 Simul-LLM 开源框架，提供 LLMs 的微调和评估功能，支持如 Falcon、Llama 等流行的 LLM 模型。
提示结构优化：提出了两种新的提示结构（分割源-目标提示结构 Split Source-Target Prompt Structure；单输出词提示结构 Single Output Word Prompt Structure），尤其是单词输出提示结构，可以精确模拟同时翻译的推理行为，从而减少训练和推理时上下文不匹配的问题。
等待策略：使用了经典的等待-k（wait-k）策略，确保在翻译过程中处理不完整的源语言输入。论文还探索了基于贪心解码和推测波束搜索（SBS）等解码策略的适应性。
微调和参数高效微调（PEFT）：框架支持使用 LoRA（低秩适配）等技术在资源有限的硬件上进行参数高效微调，保持 LLMs 的整体能力，同时优化翻译性能。</td></tr>
<tr><td>仍旧存在的问题（注明论文中说明的问题或</td><td colspan="5">论文中说明的仍旧存在的问题：
上下文不匹配：尽管提出了优化提示结构的方法，LLMs 在微调阶段和推理阶段仍然可能存在上下文不匹配的问题。特别是在使用等待-k 策略时，源语言输入逐步增加，而 LLMs 通常依赖于完整的输入上下文来进行推理。</td></tr>
</table>

	自己认为存在的问题）	**解码策略复杂度**：推测波束搜索（SBS）等复杂解码策略尽管在某些情况下提升了翻译质量，但在不同语言对（如英语到西班牙语）上表现不一致，表明这些策略对具体实现和参数选择较为敏感。 **计算资源限制**：论文中提到，由于计算资源的限制，微调和实验的样本量受到限制，特别是对于大规模的 LLMs 来说，完整微调存在困难，可能影响模型的最终性能。 **自己认为存在的问题：** **推理时延**：LLMs 在同时翻译中会带来额外的计算时延，特别是在需要高实时性（如直播翻译）的场景中，推理时间可能会影响整体的用户体验。 **多语言支持的局限性**：虽然 LLMs 具备多语言能力，但在处理语法结构差异较大的语言对时，模型的性能可能不稳定，尤其是在同时翻译的低延迟场景中，高精度和低延迟难以同时满足。 **微调数据集的限制**：论文使用的 MuST-C 数据集主要用于语音到文本的翻译，尽管被改造为文本到文本任务，但并不完全符合文本同时翻译的实际需求。这可能限制了对模型在真实应用中的评估。
实验内容	实验采用的数据集	**MuST-C** 数据集（是一个广泛用于语音到文本同时翻译（SimulST）的数据集） 为了适应本文的文本到文本翻译需求，研究者对 MuST-C 数据集进行了预处理，过滤了某些不必要的声学标记（如代表停顿的“-”字符）。 实验中主要使用了**两对语言：英语到德语（en-de）** 和 **英语到西班牙语（en-es）**。 原始数据集包含约 27 万条训练集样本和约 2500-3000 条测试集样本。为了支持 SimulMT（同时翻译）的微调，研究者扩展了数据集，将其转换为大约 500 万条训练样本的版本。
	数据集内容是否和待解决问题模型对应	**MuST-C** 数据集最初是为**语音到文本**的同时翻译（SimulST）任务设计的，但论文将其修改为适用于**文本到文本**翻译任务。这种改造使得数据集在一定程度上与待解决的问题模型（即大型语言模型用于同时翻译）对应。 但在某些细节上可能仍与原本设计的语音翻译任务有一定的差异。这对待解决问题模型的评估产生了一定的局限性。
	实验是否涉及实际应用场景	没有直接验证纯文本场景中的广泛实际应用，但模型和框架的设计明确面向实际的实时翻译场景。 **实时翻译场景：** 论文解决的核心问题是如何将大型语言模型应用于同时翻译任务，如在**会议翻译、直播翻译、国际交流**等需要快速响应的场景中。通过等待-k 策略，模型可以在不完全获取输入的情况下开始翻译，以减少翻译的延迟，这与实际场景中的低延迟需求相符。 **跨语言翻译：** 实验使用的英语到德语、英语到西班牙语**语言对**，也是实际中常见的跨语言翻译场景。模型需要处理语法结构差异较大的语言。
	实验采用的对比方法	**与经典的非 LLM 模型对比：** 论文使用了传统的神经机器翻译（NMT）模型和单调 Transformer 模型作为基线。具体对比了**经典的 NMT Transformer**（非同时翻译）和

		使用等待策略的单调 Transformer（Wait-5）的翻译性能。这些经典模型的表现被用作评估 LLM 在同时翻译任务中的对照基线。 **不同解码策略的对比：** 对比了多种解码策略的表现，特别是针对 LLM 的推测波束搜索（SBS，Speculative Beam Search）解码策略。实验中评估了不同窗口大小和推测长度的 SBS 对翻译质量和延迟的影响。论文还采用了贪心解码（greedy decoding）作为另一种简单解码方式的对比。 **不同等待-k 值的对比：** 为了测试不同延迟和翻译质量的权衡，实验采用了**不同的等待-k 值**（如 Wait-3、Wait-7）进行微调和测试，验证了 SimulMT 系统的性能与等待-k 值之间的关系。实验结果表明，较高的等待-k 值在提高翻译质量的同时，也带来了较大的延迟。 **NMT LLM 与 SimulMT LLM 的对比：** 对比了**两种不同微调策略的 LLM 模型**，即已微调为传统神经机器翻译任务的 NMT LLM 和专门为同时翻译任务微调的 SimulMT LLM。通过对比这两类模型，实验评估了直接适配 NMT LLM 用于同时翻译与专门为 SimulMT 任务微调 LLM 的差异。 **不同模型架构的对比：** 对比了不同 LLM 模型的表现，包括 Falcon、Llama、Mistral 等不同的语言模型架构，来评估不同架构对翻译任务的影响。
	实验任务	评估和探索大型语言模型（LLMs）在同时翻译（SimulMT）任务中的表现，重点验证 Simul-LLM 框架的有效性。 **模型适配与微调；解码策略评估；提示结构验证；等待策略测试；模型性能对比：**
	实验衡量指标	**BLEU 分数**：评估生成的译文与参考译文之间的相似度 **延迟（Lagging）：平均滞后（AL）**衡量翻译系统在输入不完整的情况下开始生成翻译的滞后程；LAAL（（Length-Adaptive Average Lagging）反应不同长度输入下的滞后表现。 **推理时间**：模型生成译文的时间消耗。 **解码策略对比**：SBS 等解码策略对比了不同窗口大小和推测长度的影响；贪心解码和推测波束搜索。 **等待策略评估**：不同的等待-k 值（Wait-3、Wait-7）
	实验说明所提出方法的优点	**提升翻译质量**：SimulMT LLM 模型的 BLEU 分数显著高于基线模型，表明经过微调的 LLMs 能够生成高质量的翻译。 **适应性强**：支持多种解码策略（贪心解码和推测波束搜索），通过新的提示结构优化，增强了 LLMs 在不完整上下文情况下生成准确翻译的能力，有效解决了微调和推理时的上下文不匹配问题。 **等待策略的灵活性**：可支持不同的等待-k 策略（如 Wait-3 和 Wait-7），在一定延迟的基础上保持了良好的翻译质量。通过较高的等待-k 值，模型能够在同时翻译的低延迟要求下取得更好的通用性和翻译准确性。 **对现有模型的微调与扩展**：无缝适配和微调现有的 LLMs，如 Falcon、Llama、Mistral 等，并提供了参数高效微调（PEFT）的选项，降低了对硬件资源的要求，使得低性能设备也能够参与模型的微调和应用。

		框架的开放性和扩展性：开源框架，具备良好的扩展性。实验表明，研究者可以轻松地将该框架应用于不同的 LLM 模型和同时翻译任务。
考阅论后己考容（读文自思填充）	论文的主要优点是什么	创新的**框架**设计：第一个专门为大型语言模型（LLMs）设计的同时翻译（SimulMT）微调和评估开源框架 **Simul-LLM。** **翻译质量**的提升：在同时翻译任务中的翻译质量接近甚至超越了传统的 NMT 模型。 翻译**灵活性**提升：允许使用不同的提示结构和解码策略 硬件**资源**的高效利用：引入了参数高效微调（PEFT），允许在低性能硬件上进行微调，降低了资源需求。 广泛的**应用扩展性**：支持多种 LLM 模型（Falcon、Llama、Mistral 等）对同时翻译任务做出**实际**贡献：对低延迟实时翻译任务。
	论文仍然可以改进的地方是什么	**上下文不匹配问题**：在低等待-k 策略下，源语言输入不完整时会影响翻译准确性。 **解码策略的敏感性**：SBS 对窗口大小和推测长度非常敏感，解码策略的选择在不同语言对上表现不一致。 **计算资源的限制**：实验的训练和微调仅限于**部分数据集和模型。** **缺乏 SimulMT 数据集**：论文使用的 **MuST-C 数据集**虽然适合语音到文本翻译，但并不是为**文本到文本**的同时翻译专门设计的。 **实时翻译的计算延迟优化不足：** LLMs 的推理速度是一个瓶颈，模型在实时应用中的推理时间较长，而且在高复杂度的 SBS 下会增加额外的计算延迟。 **跨语言对的扩展性不足：**实验主要集中在英语到德语和英语到西班牙语的翻译
	选择读这篇论文的原因是什么	直接原因是信息检索系统作业的要求。 LLM 在当下受到广泛关注，我目前也在做相关方面的研究，并撰写论文。它与我**目前的工作以及研究方向**有交叉，并能够为我的研究提供启发： 1. 启发**新的评价指标**设计： 我们在之前的单模态论文影响力预测论文中，已经提出了一个新的评价指标 **TNCSISP**，用于改进影响力预测的准确性。Simul-LLM 中的研究中对不同等待-k 策略的权衡分析，让我思考多模态任务中设计**时间性与准确性兼顾的复合评价指标**，以增强模型。 2. **多模态方言翻译**项目的借鉴： 我目前正在进行的多模态方言翻译项目，涉及如何将不同模态的信息（文本、音频、图像等）整合在一起，进行有效的翻译任务。Simul-LLM 框架中提出的同时翻译方法，特别是**等待-k 机制**和**流式解码策略**，可以应用到我的方言翻译项目中。继续研究如何在方言翻译中平衡**低延迟与翻译准确性**之间的关系，尤其是在**多模态信息不完全**的情况下进行有效翻译。 3. **LLM 微调与高效资源利用：** Simul-LLM 论文中使用了参数高效微调 PEFT）的方法，能够在**计算资源有限**的情况下优化 LLM 的性能，这一点非常符合我在多模态方言翻译项目中的需求。**方言数据往往稀缺且多样**，PEFT 等方法为在有限数据和资源下提升模型表现提供新思路。

	以此论文为出发点，如果需要你做一篇和其相关的顶会论文，你需要的资源是什么？数据，硬件，技术支持等	1. **数据资源** **Simul-LLM 论文使用了 MuST-C 数据集（语音到文本标准数据集)**，需要该数据集或类似的语音/文本翻译数据集。 多种语言对：不仅限于英语到德语和西班牙语（本论文）的翻译，还应包括其他语言对（如中日、法德等） 其他更复杂或特定领域的数据集 Simul-LLM 主要处理**文本**翻译任务。如果扩展到多模态任务，如文本、语音和视频的结合，需要自定义的多模态数据集。 2. **硬件资源** 涉及对大型语言模型（如 Falcon、Llama）的微调和推理，需要强大的计算资源，比如 GPU 和 TPU。**多个高端 GPU（如 A100、V100 等)**，**云计算资源（**AWS、Google Cloud、Azure 等平台提供的 TPU 服务)，大量**存储和内存** 3. **技术支持** LoRA（低秩适配）和 PEFT（参数高效微调），推测波束搜索（SBS）等解码策略，同时翻译优化算法。 4. **软件支持** Hugging Face 等平台的开源预训练模型（如 Falcon、Llama）进行实验和对比研究，支持与这些库的集成。 5. **实验环境与基础设施** 大量的对比和参数调优，需要工具来自动化管理实验过程，如 **Weights & Biases** 或 **TensorBoard** 来记录实验结果和参数配置。
	所选这篇论文和目前自己在做的内容能够想到的相关点	Simul-LLM 论文在**多模态信息的优化处理、高效模型微调、实时任务的解码策略、以及评价指标的设计**等方面，为我的**基于 LLM 的论文影响力预测**和**多模态方言翻译**项目提供启发。虽然任务不同，但在 LLM 的应用、资源限制的应对、以及对复杂任务的处理策略上有很多可以借鉴的联系。 1. **LLM 在多任务场景中的应用：** Simul-LLM 使用 LLM 在同时翻译任务中探索了如何通过优化提示结构和等待-k 机制提高实时翻译质量，为我研究如何在多模态任务中改进模型预测性能提供思路。比如如何结合不同模态信息（如文字与图片）进行任务优化。 2. **多模态信息融合：** Simul-LLM 主要处理单一模态（文本）的翻译任务，但方法中的**提示结构优化**和**流式解码策略**为我的多模态信息融合提供了启示，比如可以参考 Simul-LLM 中提出的流式处理和分步解码的思路，来实现对图像和文字的同步处理。 3. **高效模型微调与资源限制：** 我在进行多模态方言翻译项目，低使用人数的**方言数据存在稀缺且训练资源有限**。Simul-LLM 中采用的参数高效微调 PEFT 技术，允许在低资源环境下优化 LLM 的性能，为我的多模态翻译任务提供了可能的解决方案。 4. **评价指标与优化策略：** 在之前关于论文影响力预测的论文中，我们提出了 **TNCSISP 评价指**

		标，Simul-LLM 则提出了优化后的提示结构和等待-k 策略以平衡翻译质量与延迟，为我们在影响力预测任务中平衡预测的准确性与资源消耗提供新的思路。 5. **流式处理与实时性：** 我目前正在进行的多模态方言翻译项目涉及**实时翻译场景**，而 Simul-LLM 中的等待-k 机制和流式解码策略是为了解决实时性问题的。可以借鉴到翻译项目中，帮助优化模型在接收不完整输入时如何实时生成高质量的翻译，尤其在多模态输入可能**不同时同步**的情况下。
	其他想要补充说明的内容	暂无