Day5

在今天的调研中，我将调研一下Omni领域里面目前的常见痛点、想要实现的功能、以及未来可能发展方向。由于今天事情很多，因此很多的调研总结是ChatGPT来帮忙进行的。同时，我将一些我认为重要的内容加粗。在调研的同时，我也发现了一些值得明天补上的文献泛读：

Omni-DuplexEval: Evaluating Real-time Duplex Omni-modal Interaction

MLLMsareDeeply Affected by Modality Bias

A Survey of Multimodal Hallucination Evaluation and Detection

Qwen3.5-Omni Technical Report

FutureOmni: Evaluating Future Forecasting from Omni-Modal Context for Multimodal LLMs

痛点

(Text-Centric) Alignment

AnyGPT中将所有模态都和语言进行对齐，这种做法被很多工作接受，因为在工程上有很多的有点，尤其是能够直接利用LLM基模的强大能力。但是从人类成长体验来说，我们对于视觉、听觉和文字的感知和理解是同时进行的，而不是先理解了text，然后再拥有视觉和听觉来向文本对齐。这样可能会让MLLM的推理过度依赖于text，产生幻觉；但是如果是放弃LLM基模pretrain from scratch，那么也有很大的工程成本，包含数据和训练两方面的困难。

Omni-MLLM survey 指出，目前 Omni-MLLM 虽然持续扩展模态数量，但大多数模型实际只能处理 2–3 类非语言模态，继续sequential式地扩展时会遇到训练成本、**灾难性遗忘**和低资源模态数据不足等问题。其中训练成本尚且不是问题，低资源模态也可以通过GPT构造数据等来解决(虽然可能引入bias)，但是灾难性遗忘是个很有意思的一点。LLM backbone一个个地扩展理解新模态，会遇到传统LLM中遇到的灾难性遗忘。**因此很多的工作中的训练，都是使用interleaved data来环节这个问题。**

Any-to-any仍有进步空间

早期 Omni 工作的重要目标是从多模态感知为主走向 “any-to-any input-output”。NExT-GPT 的动机正是：当时多数 MM-LLM 只擅长输入侧多模态理解，却不能输出多种模态；因此它把 LLM、multimodal adapters 和 diffusion decoders 连接起来，实现 text、image、video、audio 的任意输入输出组合。 AnyGPT 则采用离散 token 表示，把 speech、text、image、music 都转成统一的离散序列，使模型可以像学习新语言一样学习新模态，并构造了 108K any-to-any 多轮指令数据。

但这些方案也暴露出问题：NExT-GPT 依赖外部专用 encoder/decoder 和 diffusion 模型，统一性更多体现在系统层；AnyGPT 的统一性更强，但输出质量和跨模态细节保真度很大程度取决于 tokenizer/codec/generative decoder 的能力。**Omni 的难点不是简单加音频 tokenizer或接图片生成器之类的，而是如何让理解、推理、生成在一个统一训练目标下共同优化。**当然我能够理解现在绝大部分做法的妥协之处在于：如果所有的encoder and generator都是不用各自inductive bias下训好的现有组件，训练将会是难度极高的一件事情。

数据

高质量跨模态训练数据稀缺，而合成数据容易引入偏差。Omni 数据构造比图文 MLLM 更难，因为它不仅需要 image-text、audio-text、video-text，还需要 image-audio-text、video-audio-text、多轮交互、跨模态生成、实时中断、情绪语音等复杂格式。**并且为了我们希望Omni获得的新功能或新特征，我们还需要设计相对应的训练数据。**Omni综述总结目前常见数据构造方式包括模板构造、GPT 生成、Text-to-X 合成，例如把纯文本指令通过 DALL-E、MusicGen、TTS 等模型扩展成跨模态指令数据。

这种方式能快速扩大数据规模，但也会带来两个问题：第一，合成模态不等于真实模态，**低资源模态如果主要靠生成数据，模型可能学到生成器偏差而不是真实世界分布**；第二，很多 instruction 数据仍然是 text-centric 的，即以文字描述作为桥梁，并不一定训练出真正的音视频联合推理能力。OpenOmni指出，**标准 omnimodal learning 往往需要 image-text-speech 三元组**，但高质量三模态数据稀缺。

长视频、长音频、多流输入

上述的输入带来的问题是token数量的大幅上升，造成OOM。token compression 或 token sampling 可以缓解，但通常会损失跨模态性能。Video-MME 的设计也反映了这一痛点：它覆盖 11 秒到 1 小时的视频，并引入字幕、音频等输入，用来检验模型是否能处理长时序、多模态上下文。**同时对于视频和音频的temporal axis如何让MLLM能够准确get到，并且需要能够时间轴对齐**

时序对齐与实时交互

> Omni-DuplexEval: Evaluating Real-time Duplex Omni-modal Interaction

Omni 相比普通 MLLM 的关键难点之一是：现实交互不是离线 QA，而是连续的音视频流。模型要边看、边听、边说，并且在说话过程中还能被打断、修正、补充上下文。VITA 通过 state tokens 和 duplex deployment，让一个模型生成回答，另一个模型持续监听环境输入，从而实现 non-awakening interaction 和 audio interrupt interaction。而MiniCPM更偏向模型内部原生的 full-duplex cognition。

MiniCPM-o 4.5 则把问题进一步表述为 interaction paradigm 的瓶颈：现有模型往往把 perception 和 response 分成轮流发生的阶段，导致生成时无法及时吸收新输入，也难以主动响应环境变化；它提出 Omni-Flow，将输入输出沿共享时间轴对齐，支持 real-time full-duplex omni-modal interaction。

最近arxiv刚出了一篇Omni-DuplexEval benchmark，来自清华姚远团队，其 benchmark 评估实时描述和主动提醒，当前最优 duplex MLLM 总体只有 39.6%，在 Proactive Reminder 上只有 20.0%，**说明 real-time full-duplex omni-modal interaction问题还没有解决。**

模态偏置、幻觉、跨模态不一致

Omni 模型很容易出现 modality bias：训练数据中语言最密集、LLM backbone 最强，因此模型在冲突或复杂场景中可能过度相信文本，而忽略视觉/音频。Survey 明确指出，训练数据体量不平衡和不同模态 encoder 能力差异，会导致 Omni-MLLM 关注 dominant modality 而忽视其他模态。 更一般地，MLLM modality bias 研究也认为，语言数据更紧凑抽象、视觉数据冗余复杂，加上 LLM backbone 过强，**会使模型形成对语言的捷径依赖。**

幻觉问题在 Omni 中会更复杂：文本模型的幻觉主要是事实性问题；MLLM/Omni 的幻觉还包括视觉对象幻觉、音频事件幻觉、音画关系幻觉、时序因果幻觉、跨模态冲突时的错误归因等。多模态幻觉 survey 将 hallucination 概括为模型输出看似合理但与输入内容或世界知识相矛盾，并强调其 evaluation 和 detection 仍是开放问题。

Omni 模型想要实现的核心功能

Any-to-any

Omni和MLLM最大的区别就是Omni不仅仅是多模态的理解，更要有多模态的生成。

实时语音-视觉交互，而不是 ASR + LLM + TTS 拼接

GPT-4o 的一个关键叙事是：旧 Voice Mode 是 ASR → text LLM → TTS 的三模型 pipeline，主模型无法直接观察语气、多说话人、背景噪声，也不能自然输出笑声、歌唱和情绪；GPT-4o 则是 **end-to-end 跨 text、vision、audio 训练的单模型**，能够实时处理音频、视觉和文本。

Omni 的第二层目标是**端到端语音理解与语音生成**：模型不只转写语音内容，还要理解 prosody、emotion、speaker turn-taking、background sound、interruptions，并以自然、有情绪、有节奏的语音回应。Qwen2.5-Omni 的 Thinker-Talker 架构就是一个典型尝试.

Full-duplex：边看边听边说，可被打断、可实时修正

传统聊天是半双工：用户说完，模型再答；模型答完，用户再说。但真实人类对话是 full-duplex：对方说话时我们仍在听、看、判断，并且可以插话、停顿、改口。VITA 已经通过双模型部署实现 audio interruption；MiniCPM-o 4.5 则明确提出同时看、听、说，并通过共享时间轴进行 full-duplex 建模。

主动感知与主动提醒，而不是只响应显式问题

现在很多 MLLM 是 reactive system：用户问，模型答。但 Omni 的目标更接近环境智能体：**模型持续观察音视频场景，在必要时主动提醒、评论或询问。**MiniCPM-o 4.5 明确把 proactive behavior 作为目标，例如基于连续场景理解发出提醒或评论。

精细音视频 grounding

未来 Omni 模型不应该只做粗粒度视频问答，而要能回答“什么时候发生了什么”“谁说了什么”“声音来自哪里”“画面事件和音频事件如何对应”。Qwen3.5-Omni 报告中提到它支持 controllable audio-visual captioning，包括结构化描述、剧本级细粒度 caption、自动分段、timestamp annotation，以及人物与音频关系描述。**没有时间戳和事件边界，模型很难进行可靠的长视频总结、事故分析、实时监控、教学反馈或机器人规划。**

Omni-Agent

最新趋势是把 Omni 和 agent 结合。Qwen3.5-Omni 把 native omnimodal agentic behavior 作为关键能力，包括 autonomous WebSearch、complex FunctionCall，以及 Audio-Visual Vibe Coding，即从音视频指令直接生成可执行代码。Omni 的最终目标不是“多模态聊天机器人”，**而是能感知真实世界、理解用户意图、调用工具并完成任务的 general-purpose multimodal agent。**

未来可能的发展方向

text-centric alignment 走向 native omni pretraining

当前很多模型仍然依赖语言作为 pivot：图像对齐到文本、语音对齐到文本、视频对齐到文本。这个路线短期有效，但长期会限制模型对非语言信息的建模能力。未来更可能走向 **native omni pretraining**：在预训练阶段就联合学习 text、image、audio、video 的时序结构、跨模态对应关系和生成目标，而不是后期给 LLM “外挂”模态模块。

Qwen3.5-Omni 的路线已经体现这种趋势：它声称在 massive text、visual data 和超过 100 million hours audio-visual data 上进行 native omnimodal training，并设计 Thinker-Talker、Hybrid-Attention MoE、256k context、ARIA 等机制来支撑大规模多模态理解、推理、生成和行动。 MiniCPM-o 4.5 也强调端到端 token-level continuous connections，使 multimodal encoders、LLM backbone 和 speech decoder 能共同训练。

“简单拼接 token”走向“时空对齐的多尺度 token 系统”

文本 token 是离散且低频的；语音 codec token 高频且连续；视频 frame token 数量巨大；音频事件和视觉事件可能不同步。简单把 token 串接进 LLM 上下文会越来越吃力。因此更可能的发展方向是：多尺度 tokenization、事件级 memory、time-aware positional encoding、cross-modal temporal index、动态 token routing等范式。

Full-duplex 和 proactive interaction

是否能在真实时间中持续感知、持续更新、持续交互。这也会改变训练数据和评测方式。未来需要大量带时间戳的交互轨迹：什么时候用户插话，什么时候模型停顿，什么时候应该主动提醒，什么时候保持沉默。Omni-DuplexEval 的结果显示当前模型在“何时响应”和“响应什么”之间难以平衡。

从回顾性理解走向预测、规划和世界模型

现在多数 benchmark 关注 retrospective understanding：视频里发生了什么，音频里有什么声音，图像里有什么对象。但人类智能的重要部分是预测和规划：接下来可能发生什么，我应该做什么。FutureOmni 正是针对这个空白提出的 benchmark，要求模型从 audio-visual environments 中进行 future forecasting，需要跨模态因果推理、时序推理和内部知识调用；其评测显示当前系统在视听未来预测上仍然困难，尤其是 speech-heavy 场景。