【mPLUG-Owl2：利用模态协作机制的多模态大语言模型】

——【mPLUG-Owl2: Revolutionizing Multi-modal Large Language Model with Modality Collaboration】

1 相关资源

pdf: https://doi.org/10.1109/CVPR52733.2024.01239
ppt:
短视频:
数据集:
源码: https://github.com/X-PLUG/mPLUG-Owl/tree/main/mPLUG-Owl2
网站:
【除了网站，其他资源尽量下载】

2 论文属性

论文来源: CVPR 2024
【给出具体会议名称和年限，不要仅仅写 ACM，IEEE】
论文类别: Multi-modal Large Language Model
【论文的类别，比如移动计算、轨迹处理、深度学习等】
论文关键字: Periodic features, Traffic flow prediction, Graph structure learning, Graph convolutional network, Spatio-temporal model
推荐程度：3 （其他说明可标注）
（5 非常棒，建议认真研读、小组讨论和复现；4 好，建议细读，考虑复现；3 可以，部分内容值得注意；2 一般，简单浏览即可；1 没有意义，不建议阅读）

3 工作团队

作者: Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang
单位: Alibaba Group
团队情况描述:

4 论文介绍

(1) 研究目的
【研究背景是什么？本文工作有什么用?】
多模态大模型在各类开放式任务中展现出卓越的指令能力，现有的方法主要聚焦于增强多模态能力，

而本文提出了通用型多模态大模型 mPLUG-Owl2，通过有效利用模态协作机制同步提升文本与多模态任务表现。
大量实验表明，mPLUG-Owl2 能以单一通用模型实现文本任务与多模态任务的协同泛化，并取得最先进的性能。同时，该模型首次在纯文本和多模态场景中均验证了模态协作现象，为未来多模态基础模型的开发开辟了新范式。

(2) 研究现状
【当前的最好研究做到什么程度了？存在的问题是什么？这里采信论文的说法，可以给出自己的评点】
在多模态大模型方面，当前构建多模态大模型主要有三种方法，均在视觉语言领域展现出强大的零样本泛化能力：Flamingo 采用冻结视觉编码器与配备门控交叉注意力的大模型实现跨模态对齐；而 PaLM-E 则通过线性层将视觉特征直接注入 5200 亿参数预训练出了 PaLM 模型；但该方法会生成冗长视觉序列，为此 BLIP-2 借鉴 DETR 开发 Q-former 压缩视觉特征序列。这些方法将视觉特征与 LLMs 直接对齐，忽视了视觉与语言模态的粒度差异。
在利用指令调优进行预训练方面，随着多模态大模型的兴起，学界开始构建高质量、多样化的多模态数据集，如 MiniGPT-4 利用 GPT-3.5 重述预训练模型生成的描述；LLaVA、SVIT 和 LRV-Instruction 则借助物体边界框、图像描述和区域标注等图像注释，通过自指令方法促使 GPT-4 生成指令与响应；Peng 等与 Yang 等采用上下文学习改进生成数据质量；Li 等和 Marino 等则直接通过上下文学习优化 MLLMs；mPLUG-Owl 和 LLaVA-1.5 等模型通过联合训练纯文本与视觉-文本指令数据，减轻语言知识灾难性遗忘风险。而本文中的 mPLUG-Owl2 则更进一步：借助模态自适应模块，在联合训练纯文本与多模态指令数据时，不仅能避免灾难性遗忘，还能通过模态协作同时提升多模态与纯文本任务表现。

(3) 本文解决的问题
【一句话概括本文解决的核心问题】
本文提出了新型通用多模态基础模型 mPLUG-Owl2，有效利用模态协作机制同步提升了文本与多模态的任务表现。

(4) 创新与优势
【本文的创新之处是什么？新场景？新发现？新视角？新方法？请明确指出】
【本文工作的贡献或优点是什么?】

1. 提出 mPLUG-Owl2，采用模块化网络设计，通过语言解码器作为管理多模态信号的统一接口，兼顾模态协作与模态干扰；通过共享功能模块促进模态协作。
2. 提出包含视觉语言预训练和联合视觉语言指令调优的量阶段训练范式。

(5) 解决思路
【本文是怎样解决问题的？包括方法、技术、模型等，以自己理解的方式表述清楚】
模型结构
图 1 为 mPLUG-Owl2 的总体架构，该模型由视觉编码器、视觉抽象器、文本嵌入层和语言解码器构成。
总的来说，该模型在视觉方面：就是首先通过视觉编码器将输入的图像转换为视觉 tokens；然后再利用视觉抽象器，通过一组可学习的查询，从视觉 tokens 中提取更重要的部分，以减少视觉 tokens 的长度。在文本方面，将输入的文本信息通过 Embedding 转化为文本 tokens。最后，再利用语言解

码器，将文本将视觉和文本特征融合并生成最终的输出。

在语言解码器中集成了模态自适应模块，用于促进视觉和文本模态的协作。该模块首先将视觉信息和文本信息进行分离；然后进行归一化处理，确保它们在同一个量级；之后再通过重构的自注意力操作，使用不同的线性投影层对 Key 和 Value 进行投影，分别对视觉和文本信息进行处理，但同时通过 Query 操作保留它们之间的联系；最后利用共享前馈网络，把视觉信息和文本信息结合起来，生成最终的输出。

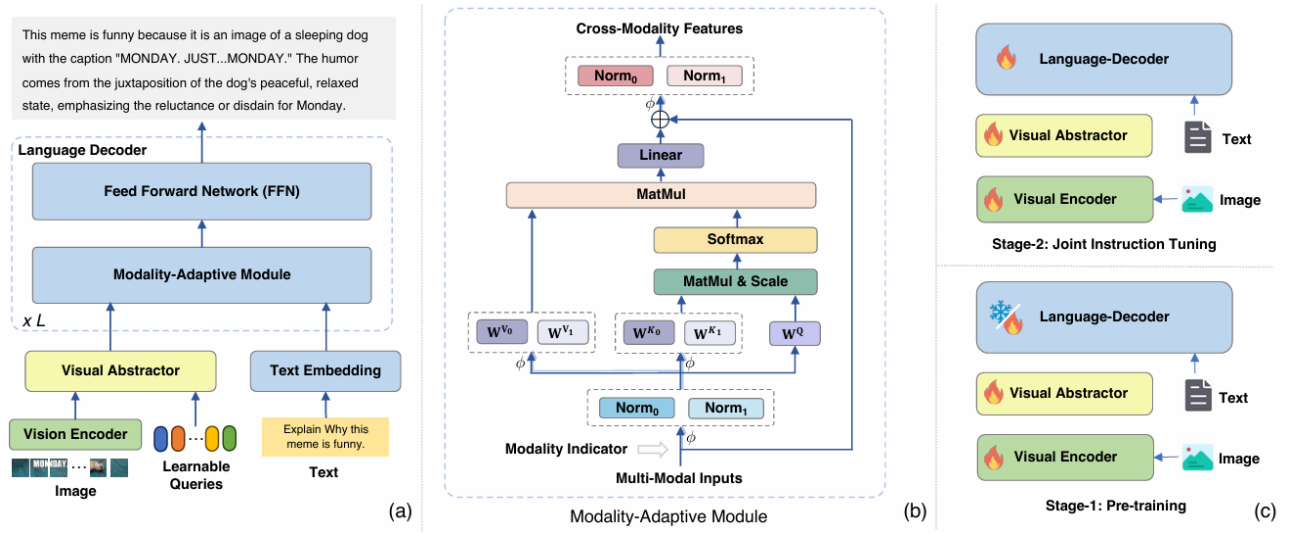

图 1: Illustration of the proposed mPLUG-Owl2 and its training paradigm.

具体来说，作者采用 ViT-L/14 作为视觉编码器，LLaMA-2-7B 作为语言解码器。

视觉编码器在处理 $H \times W$ 分辨率的输入图像，生成 $\frac{H}{14} \times \frac{W}{14}$ 个序列 tokens。这些视觉 tokens 会在随后与文本标记嵌入相结合，输入到作为通用接口的语言解码器中，而语言解码器的作用就是将各类视觉-语言任务转化为文本生成任务。

随着图像分辨率的提升，编码后的视觉 tokens 会呈指数级增长，同时会引入大量冗余信息，因此本文提出了配备固定可学习查询集的视觉抽象器，用于从图像中提取更高层次的语义特征，具体实现时，将提取的视觉标记序列 $\mathcal{I} = [I_1, I_2, \cdots, I_P] \in \mathbb{R}^{P \times d}$ 与固定数量的 K 个可学习查询 $\mathcal{Q} \in \mathbb{R}^{K \times d}$ 输入视觉抽象器。其中 $P = \frac{H}{14} \times \frac{W}{14}$ 表示视觉块数量，D 为隐藏维度。视觉抽象器由若干抽象层堆叠而成，在第 i 层中，压缩后的视觉表征 $\mathcal{V}^{i+1}$ 按如下方式计算:

$$\mathcal{C}^i = \text{Attn}\left(\mathcal{V}^i, [\mathcal{I}; \mathcal{V}^i], [\mathcal{I}; \mathcal{V}^i]\right), \tag{1}$$

$$\mathcal{V}^{i+1} = \text{SwiGLU}\left(\mathcal{C}^i W_1\right) W_2. \tag{2}$$

获得压缩视觉特征后，将其与文本标记嵌入拼接，最终通过语言解码器生成预测结果。

模态自适应模块

该模块通过将视觉特征和语言特征投影到共享语义空间来实现视觉-语言表征解耦，同时保留各模态的独有特性，具体做法如下:

- 模态分离：给定一个视觉-语言序列 $X \in \mathbb{R}^{(L_V + L_T) \times d}$ 和模态指示器 $M \in \{0, 1\}^{(L_v + L_T)}$，将模态分离操作 ϕ 定义如下。

$$\phi(X, M, m) = X \odot \mathbb{1}_{\{M=m\}}, \tag{3}$$

- 归一化操作：给定前一层输出向量 $H_{l-1}, l \in [1, L]$（L 为语言解码器层数），将不同模态归一化至相同量级，这里 LN_V 和 LN_T 分别是视觉特征与语言特征的层归一化。

$$\widetilde{H}_{l-1} = LN_V\left(\phi\left(H_{l-1}, M, 0\right)\right) + LN_T\left(\phi\left(H_{l-1}, M, 1\right)\right), \tag{4}$$

- 重构自注意力操作：通过为键投影矩阵和值投影矩阵配置独立线性投影层 (同时保持查询投影矩阵共享) 来重构自注意力操作。

$$H_l^Q = \widetilde{H}_{l-1} W_l^Q, \tag{5}$$

$$H_l^K = \phi\left(\widetilde{H}_{l-1}, M, 0\right) W_l^{K_0} + \phi\left(\widetilde{H}_{l-1}, M, 1\right) W_l^{K_1}, \tag{6}$$

$$H_l^V = \phi\left(\widetilde{H}_{l-1}, M, 0\right) W_l^{V_0} + \phi\left(\widetilde{H}_{l-1}, M, 1\right) W_l^{V_1}, \tag{7}$$

$$C_l = \text{Softmax}\left(\frac{H_l^Q {H_l^K}^\top}{\sqrt{d}}\right) H_l^V, \tag{8}$$

其中 $W_l^Q, W_l^{K_0}, W_l^{K_1}, W_l^{V_0}, W_l^{V_1} \in \mathbb{R}^{d \times d}$ 是可学习投影矩阵，$C_l \in \mathbb{R}^{(L_V + L_T) \times d}$ 表示第 l 层的上下文特征。

训练范式

如图 1(c) 所示，作者采用两个阶段训练 mPLUG-Owl2。

- 预训练阶段：旨在对齐视觉编码器与语言模型。在预训练阶段，保持语言模型被冻结，仅开放视觉编码器、视觉抽象器和部分模态自适应模块的参数更新。
- 指令微调阶段：旨在通过语言建模损失微调语言模型。作者采用联合训练策略，同步处理文本指令与多模态指令，这样可以通过多模态指令增强模型对文本内嵌视觉概念的理解，同时通过文本指令数据强化模型对复杂自然指令的解析能力，保证语言能力不退化。

(6) 可改进的地方

【本文工作的局限性是什么？你觉得可以从哪些方面改进工作?】

mPLUG-Owl2 主要是在图像和文本的对齐方面进行了训练，对于其他模态如视频、音频等的支持可能不太够；同时 mPLUG-Owl2 的训练和推理过程需要大量的计算资源，会限制它的使用。

(7) 可借鉴的地方

【你觉得本文哪些方面可以借鉴？比如思路、方法、技术等】

首先是文章的模型采用模块化网络设计，这样可以非常清晰地了解模型的整个框架组成，非常值得学习。

本文提出的模态自适应模块的设计可以解决模态间协作与干扰的问题，它先将文本和视觉分离，分别利用各自的规则提取 key 和 value，但是又通过 Query 将两者相互关联起来，能将视觉特征和语言特征投影到一个共享的语义空间中，同时保留各自模态的独特属性，我觉得是一个很好的思路。

文章在描述模型以及训练过程中，首先简要提出前人研究中的不足，然后再提出自己的做法以及优点，写文章时可以借鉴。

(8) 其他收获

【你有什么其他收获吗？比如了解了哪些团队和大牛在某领域做得很好，某类问题通常用什么技术解决，某些技术之间存在什么样的关联，某些会议和期刊在某领域很知名……】

mPLUG-Owl 系列模型是阿里实验室在多模态大模型中作出的研究，这个是第二代，我简要看了一下这几代的模型。

mPLUG-Owl 是第一个版本 (2023.04)，主要是使大语言模型实现了基础的多模态能力，同时引入了视觉对齐和语言模型微调的训练模式。
mPLUG-Owl2 是第二个版本 (2023.11)，即本文提出的模型，主要是通过引入模态自适应模块解决了模态协作问题，不仅在多模态任务中取得了优异的结果，同时在纯文本任务中也显著提高了性能。
mPLUG-Owl3 是第三个版本 (2024.08)，它主要专注于长序列处理能力，提出了 Hyper-Attention 模块，以将视觉和语言有效地整合到一个通用的语言指导的语义空间中，从而促进了扩展的多图像场景的处理；在保持图像细粒度视觉信息的同时，不引入大量参数并且高效处理任意的图像-文本输入，单模型实现通用的单图、多图、视频理解。

5 评阅人

姓名:
时间: