大模型原理与应用

第 9 课：多模态大模型 | 视觉、语言与跨模态理解

从看图到看懂图、会对话、会生成

这一课看什么

- 图像为什么也能像文本一样进 Transformer
- 图像和文本如何对齐到同一空间
- 多模态模型如何走向问答、对话和生成
- 大型多模态模型为什么会走向 instruction tuning 与统一接口

本课主线

1. 先把视觉变成 token 和表示
2. 再把视觉与语言放进同一空间
3. 然后看大型多模态模型怎样把“看图”和“对话”接起来
4. 最后理解多模态系统的能力边界

什么是多模态

- 多模态 AI 处理的不再只是文本
- 输入可以来自图像、视频、音频、传感器、文字等
- 目标是形成统一表示或统一理解

这意味着模型不只是“看文字”，而是同时理解多个世界接口。

为什么大模型必须走向多模态

因为真实世界本来就是多模态的：

- 人看图、听声音、读文字
- 监控系统有视频、音频、告警、工单、说明文档
- 机器人看环境、听指令、执行动作

所以大模型要真正进入现实应用，就必须从纯文本走向多模态。

先把多模态任务地图打开

- 图像分类
- 图文检索
- 图像描述
- 视觉问答 VQA
- 视觉推理
- 目标检测
- 语义分割
- 文生图
- 图像编辑

先把多模态任务地图打开

也可以概括成三大类：

1. 图像 - 文本对齐
2. 图像 + 文本理解
3. 文本到图像生成

先看视觉模型是怎么走到今天的

- 早期视觉任务与 benchmark
- 浅层特征工程时代
- CNN 在 ImageNet 上的突破
- GAN / VAE 带来的生成时代
- 2020 年代进入 Vision Transformer 时代

这条线和 NLP 很像：

- 先是专用模型
- 后来是统一结构
- 最后是与语言结合

图像怎么才能喂进 Transformer

文本里有 token，图像怎么办?

Vision Transformer 给出的答案是:

- 把图片切成固定大小 patch
- 每个 patch 当成一个 token
- 线性投影到向量空间
- 加上位置编码
- 再送入 Transformer encoder

这就是“图像 token 化”。

ViT 的核心思想

- Image GPT 最早尝试把图像当序列
- ViT 进一步把图像切成 patch 序列
- 用标准 Transformer encoder 处理整张图
- 可加 [CLS] token 做分类

这一步的意义非常大：

- 把视觉也纳入 Transformer 统一框架

Vision Transformer 的优缺点

优点：

- 架构统一
- 可扩展
- 在大数据上效果强

代价：

- 对数据量要求更高
- 缺少 CNN 的局部归纳偏置

DeiT 等方法进一步解决了“没有超大私有数据也要训好 ViT”的问题。

视觉预训练与自监督

材料里提到的视觉路线包括：

- Image GPT
- ViT
- DeiT
- DINO
- MAE

说明一个重要趋势：

- 图像也像文本一样进入了“预训练 + 迁移”时代

这为后面的多模态融合打下了视觉基础。

多模态的第一步，为什么一定是对齐

如果图像编码器和文本编码器各学各的，会有什么问题?

- 图像向量和文本向量不在同一空间
- “一只狗的图片”和“a photo of a dog” 无法天然接近

所以关键挑战是：

> 把语义等价的图像和文本映射到共同空间

CLIP：图文对齐的代表模型

- 图像编码器编码图片
- 文本编码器编码文字
- 用对比学习拉近匹配对，拉远不匹配对

训练目标：

- 给定一个 batch 的 N 组图文对
- 在 N x N 个可能配对里学会找真配对

为什么 CLIP 很重要

它带来的不是“又一个图像分类器”，而是：

- 零样本图像分类
- 图像检索
- 图文检索
- 更可迁移的视觉表示

也就是：

- 不用为每个视觉任务单独重训一个头
- 用语言 prompt 就能把分类任务改写成匹配任务

CLIP 的优势与局限

优势：

- web-scale 图文数据
- 强 zero-shot transfer
- 任务迁移能力强
- 对自然分布偏移更鲁棒

局限：

- 不会生成图像
- 对 prompt 工程较敏感
- 对计数、纹理、复杂视觉关系仍有限制

所以 CLIP 更像“强表示模型”，不是完整多模态助手。

CLIP 之后，图文对齐还怎么做

- OSCAR
- VinVL
- ALIGN
- BLIP
- CoCa
- SigLIP

这些模型的差别主要在于：

- 用区域特征还是 patch
- 只做对比学习，还是兼顾生成
- 是否借助对象标签

区域级视觉语言模型

在 ViT 之前，很多模型先检测对象区域，再和文本融合。

代表包括：

- LXMERT
- ViLBERT
- VisualBERT
- UNITER
- OSCAR

它们的直觉是：

- 图像理解离不开对象及其关系
- 先把对象提出来，再做跨模态融合

OSCAR 的思路

- 先用目标检测器提区域特征
- 再把对象标签当“锚点语义”
- 让图像区域、对象标签和文本 token 更容易对齐

优点：

- 学对齐更快
- 更参数高效

局限：

- 强依赖目标检测器
- 若文本里不提显著对象，收益会下降

Patch 路线：更统一、更可扩展

随着 ViT 普及，另一条路线变强：

- 直接使用图像 patch
- 不依赖外部目标检测器
- 训练与推理更统一

材料里提到的例子：

- ViLT
- ALBEF
- BLIP
- BEIT3

这条路线更接近“视觉版 BERT / GPT”。

多模态预训练通常在学什么

- ITM：Image-Text Matching
- MLM：Masked Language Modeling
- WPA：Word-Patch / Region Alignment
- ITC：Image-Text Contrastive Learning
- MIM：Masked Image Modeling
- ITG：Image-to-Text Generation

这些目标的组合不同，就形成了不同的模型风格。

ALBEF / BLIP：对齐 + 匹配 + 生成

这类模型通常把几个目标一起优化：

- 图文对比
- 图文匹配
- 掩码语言建模或生成

好处是：

- 既学共享语义空间
- 又保留生成能力
- 对下游 VQA、captioning、retrieval 都更有帮助

这也说明多模态预训练开始从“判别”走向“判别 + 生成”。

CoCa：表示学习与生成统一

它的关键思想是：

- 同时优化对比学习目标
- 也优化 caption generation 目标

这使模型同时具备：

- 强判别表示
- 更细粒度的图像描述能力

它体现了一个很重要的趋势：

> 不是只做对齐，而是让模型既会认，也会说

多模态表示是否真的在收敛

- 不同模态、不同目标训练出来的表示，可能在趋向同一个共享统计结构

也就是所谓“Platonic Representation Hypothesis”。

这对多模态模型的启发是：

- 视觉和语言并非完全割裂
- 在足够大规模下，它们的高层语义空间可能越来越接近

这从理论上支持了图文对齐与跨模态迁移的可行性。

视觉问答：从看图到理解图

材料多次提到 VQA 和视觉推理：

- 模型不只是识别物体
- 而是要理解场景、关系和问题
- 然后生成或选择答案

这要求模型同时具备：

- 视觉感知
- 语言理解
- 跨模态对齐
- 推理能力

从图文模型走向大型多模态模型

- 把强视觉编码器和强语言模型接起来
- 先做 feature alignment
- 再做 visual instruction tuning

这类模型通常叫：

- Large Multimodal Models
- Multimodal LLMs

Flamingo 给了什么新启发

Flamingo 很重要，不只是因为它“效果强”，而是因为它提出了一个很清楚的工程目标：

- 尽量复用强语言模型
- 再想办法把图像 / 视频输入接进去

它想解决的问题是：

- 文本大模型已经很强
- 但它原本不会处理图像
- 我们不想为了多模态就从零重训所有模块

这和后面很多多模态大模型的思路是一致的。

Interleaved Inputs 为什么重要

Flamingo 一类模型强调的一点是：

- 输入不一定是“先一张图，再一句话”这么简单

更真实的形式是：

- 图像
- 文本
- 图像
- 文本
- 继续提问

也就是 interleaved inputs。

这很重要，因为真实世界的人机交互本来就是交错进行的：

- 看图
- 提问
- 再补图

多模态模型为什么会走向 visual instruction tuning

只做图文对齐还不够，因为用户真正需要的是：

- 能问问题
- 能得到结构化回答
- 能多轮追问
- 能按要求解释图像细节

所以后来很多系统都会再加一层：

- visual instruction tuning

也就是：

- 用“图像 + 指令 + 回答”的数据
- 把模型调成一个真正能看图对话的助手

LLaVA：多模态聊天模型的典型代表

材料里给出的 LLaVA 配方很典型：

- Vision Encoder：CLIP ViT-L/14
- Projection：线性层
- Language Model：Vicuna / LLaMA 等

训练一般分两步：

1. 先把视觉特征对齐到语言空间
2. 再用视觉指令数据做 end-to-end tuning

为什么 LLaVA 有代表性

因为它揭示了很多多模态大模型的工程套路：

- 不从零训练全部模块
- 复用现成强视觉模型
- 复用现成强语言模型
- 中间靠 projector / adapter 连接
- 再用指令数据把模型调成“能看图聊天”

这和文本大模型时代的“预训练 + 指令微调”高度一致。

InstructBLIP / PaLI / InternVL 在说明什么

不同大模型路线虽然细节不同，但都在回答同一个问题：

- 怎样把强视觉模块和强语言模块接得更自然

它们常见的设计差异包括：

- 视觉特征先怎么压缩
- 中间是否加入 query / projector / adapter
- 是偏理解，还是兼顾生成
- 是否一开始就支持多图、多轮或更复杂输入

所以这批模型的价值，不只是“谁分数高”，而是它们逐渐把多模态系统变成统一接口。

多模态模型开始像什么

从 LLaVA、Flamingo、InstructBLIP 往后看，多模态模型越来越像：

- 一个统一的感知入口
- 一个统一的语言输出接口
- 中间靠共享语义空间和指令数据维持一致性

这意味着：

- 图像输入不再是独立任务
- 它开始像文本一样，被接入同一个对话系统

多模态 Prompt 为什么比想象中更重要

材料里其实反复提醒了一件事：

- 多模态模型也很依赖 prompt

因为你问得不同，模型会被引向不同层次：

- 物体识别
- 场景理解
- 关系判断
- 幽默解释
- OCR 读图

也就是说：

- 同一张图，不同 prompt 会调出不同能力

多模态系统最常见的边界在哪里

即使模型已经能“看图说话”，也不代表它什么都稳定。

最常见的问题包括：

- 计数不稳
- 空间关系容易错
- OCR 读图有时不稳定
- 细粒度属性绑定容易漂移
- 看起来说得很像，但证据不足

所以多模态系统的关键，不只是“能生成一句描述”，而是：

- 能不能把视觉证据和语言回答真正对齐

从“看图”到“看懂图”差在哪里

“看图”常常只意味着：

- 识别几个显著对象

“看懂图”则要求：

- 理解场景
- 读懂关系
- 找到问题对应的视觉证据
- 在回答中保持一致

这也是为什么课堂练习不能只做：

- 请描述这张图

还应该继续做：

- 你的判断依据是什么
- 图里哪几个细节支持这个结论

多模态模型正在连接语言之外的世界

`Connecting Language to the World` 这组材料给出的更大视角是：

- vision 只是第一步

后面还会继续走向：

- speech
- audio
- video
- code
- action

所以多模态这一章的重要性，不只是“多学了一种输入”，而是：

- 大模型开始真正接入现实世界

数据怎么来

材料里还提到一个关键现实问题：

- 多模态指令数据很贵

常见解决办法：

- 用 GPT-4 等更强模型生成视觉指令数据
- 形成 conversation、detailed description、complex reasoning 等不同子集

也就是说，多模态 instruction data 也进入了“AI 生成监督信号”的阶段。

生成式多模态模型

除了“看图说话”和“图文检索”，另一条主线是：

- 文本生成图像
- 图像变体生成
- 图像编辑

unCLIP / DALL-E 2 路线

unCLIP 的两阶段思路很典型：

1. 先从文本生成 CLIP 图像 embedding
2. 再从 embedding 生成图像

也就是说：

- CLIP 不再只是用于检索
- 它的潜在空间还能作为图像生成的中间语义接口

unCLIP 的优点与局限

优点：

- 图像变体、插值、风格变化更自然
- CLIP latent space 可控性较好
- 在 photorealism / diversity 上表现强

局限：

- 属性绑定仍不稳定
- 复杂场景细节不够
- 文本渲染容易失败

这也说明多模态生成并不只是“会画图”，还涉及语义绑定与结构一致性。

多模态模型正在连接更多世界接口

- vision
- language
- other modalities
- code
- action

这意味着多模态大模型未来不只停在图像和文本，而会继续走向：

- 语音
- 音频
- 视频
- 机器人动作
- 世界交互

多模态模型的统一趋势

把这一课所有材料串起来，可以看到三次统一：

1. 结构统一
    - Transformer 进入视觉
2. 表示统一
    - 图像与文本共享语义空间
3. 能力统一
    - 同一个模型开始同时做理解、问答、生成、交互

这就是“多模态大模型”的真正含义。

为什么这节课和真实世界直接相关

对你的后续课程和工程应用来说，关键启发是：

- 图像不是孤立信号，要和语言一起理解
- 表示对齐通常比单任务监督更可迁移
- 生成能力和理解能力越来越融合
- 多模态系统的价值在于统一感知、推理和输出

例如：

- 监控场景：视频 + 音频 + 告警 + 文档
- 教育场景：图片 + 题目 + 文字反馈
- 工业场景：图像 + 传感器 + 工单 + 手册

这节课你该带走什么

1. 图像进入 Transformer 时代后，视觉与语言开始统一
2. ViT 解决图像如何 token 化
3. CLIP 解决图文如何对齐
4. Flamingo、LLaVA、InstructBLIP 说明多模态模型正在走向统一对话接口
5. BLIP / CoCa / unCLIP 把对齐、理解、生成接起来
6. 多模态 Prompt 会显著影响模型调出的能力层次
7. 多模态模型正在从看图走向理解世界，但仍有明显边界

第 4 课收束

多模态的关键，是把不同世界接口接到同一语义空间

- 先把不同模态编码出来
- 再把它们对齐到共同语义空间
- 最后在统一表示上做理解、生成与行动