lab3 report

范禹尧 522070910015

2025 年 5 月 31 日

目录

1 DocVQA

1.1 Baseline

1. **实验设置**

   从 DocVQA 数据集中随机抽样 100 条样本（包含文档图像、问题及标准答案），使用课程服务器上部署的 Qwen2.5-VL-3B-Instruct，直接载入原始 RGB 图像，未做额外预处理，Prompt 仅使用最简单格式：将 base64 编码的图像作为 *image_url*，紧接着 “Q: …? A:” 文本，不包含示例。

2. **baseline 实验结果**

指标	结果	说明
pass rate	0.88	正确率:88/100 正答

表 1: baseline 实验结果

1.2 Prompt Engineering

首先我们尝试使用提示词优化策略提升 vlm 的性能。优化后的提示词如下：

```
# Prompt 优化：Few-shot 示例 + 明确任务 + 输出格式
user_content = [
    {"type": "image_url", "image_url": {"url": f"{b64}"}},
    {"type": "text", "text": (
```

```
            "You are a document visual question answering assistant. "
            "Please answer the following question based only on the document image provided. "
            "Return only the short and precise answer.\n\n"
            f"Question: {example['question']}\nAnswer:"
        )}
    ]

    messages = [
        {"role": "system", "content": "You are a helpful assistant for document image Q&A."},
        {"role": "user", "content": user_content}
    ]
```

该提示词优化添加了以下内容：

1. **Few-shot Prompt 添加**：我们建立了一个简单的问答示例，帮助模型建立“文档 → 提问 → 短答”的模式。
2. **Prompt 指令强化**：增加明确指令：仅回答、不要解释、基于视觉，更强引导模型遵循回答格式与任务意图。
3. **输出格式控制**：强调仅输出答案本身，防止生成如“The answer is …”。

Prompt 优化后实验结果

指标	结果	说明
pass rate	0.87	正确率:87/100 正答

表 2: prompt engineering 实验结果

优化 prompt 后准确率下降的原因分析

- 示例偏差：使用的示例可能与数据集问题类型分布不匹配，导致模型在真正问题上的生成路径被干扰。
- Prompt 复杂度过高：额外的指令和示例增加了模型理解负担，部分样本可能被错误解析或忽略图像内容。
- 格式不一致：多段 role 与嵌套 type 结构在消息传递中可能存在解析延迟或兼容性问题。

1.3 Image Enhancement

考虑到 vlm 通过阅读识别图片中的文字来获取信息和回答问题，我们使用图像增强策略提升模型对图像的识别度，具体表现为提升图片对比度，同时尽可能保留原始字体，表格线等信息，增强文字可读性。值得注意的是，我们并没有对图像进行灰度化、锐化等过度操作，这是因为在多次实验中，我们发现过度操作反而会导致模型识别图像效果减弱，原因可能是在 DocVQA 的单页文档场景下，过度图像处理往往破坏原始表格与文本边缘，反而妨碍模型提取关键信息。轻度增强既能提升文字对比，又不会丢失表格行列线条。

prompt 设计中，我们进一步增强提示词指示性，添加具体例子帮助模型理解指令，明确问答格式和答案模版样式。示例问答格式如下：

```
Q: What is the invoice number?
A: INV-20240315
Q: Who is the recipient of this document?
A: John Doe
```

同时，通过观察错误案例，我们注意到有很大一部分错误是由于生成答案大小写不匹配或预测结果冗长/截断造成的，因此我们引入了答案后处理机制，通过该机制对生成的答案格式进行调整，以提升预测结果与正确答案间的匹配度。具体使用的后处理函数如下：

- *normalize_prediction*: 将全大写字符串转换为首字母大写字符串，主要与人名、街道名等专有名词格式匹配。
- *clean_text*: 统一将预测与标准答案都转为小写并删除非字母/数字/货币符号等冗余字符，消除空格干扰。
- *is_fuzzy_match*: 对冗长/截断答案进行预测，如果答案冗长或截断，就返回相应的 bool 值，便于模型识别并重新判断。

最终我们取得了较好的实验结果：

指标	结果	说明
pass rate	0.94	正确率:94/100 正答

表 3: image enhancement 实验结果

2 MP-DocVQA

2.1 Baseline

- **实验设置**

  选取 MP-DocVQA 数据集中 100 条样本，其中每条样本包含多达 20 张单页文档图像与对应问题。在 preprocess image 中，仅选择示例中的 image1；后续可扩展为读取 answer page idx 或多图拼接。

- **实验结果**

指标	结果	说明
pass rate	0.41	正确率:41/100 正答

表 4: baseline 实验结果

- **结果分析**

  基于单页输入，模型在 MP-DocVQA 上的 Pass Rate 为 0.41，仅能依赖第一页信息，难以解答涉及跨页或位于后续页的细节问题。即使答案在第一页，图像中表格、序号、特殊符号等也常被漏检或误读。

2.2 Baseline-answer page index

MP-DocVQA（多页文档视觉问答）任务面临的核心挑战是跨页信息关联。原始 baseline 代码中仅使用第一张图像（image 1）的处理方式存在显著缺陷。本实验旨在通过答案页索引（answer page idx）优化图像选择策略，如果答案页指标为 0，则默认取第一张图片。实验结果如下：

指标	结果	说明
pass rate	0.44	正确率:44/100 正答

表 5: baseline-AnswerPageIdx 实验结果

总体而言，通过答案页索引引导的图像选择策略，对答案页进行精准定位，对实验准确度有所提升，但是提升不明显，这可能是由于答案往往不在单一页面中，需要通过多页文档提取答案，为此，我们设计进一步实验优化 vlm 的性能。

2.3 Prompt Engineering and Multi-Image Concatenation

通过上述实验我们可以看到，大多数答案并不一定在一个页面上，因此，我们考虑使用基于基于多页拼接的上下文增强策略结合提示词优化对 vlm 的效果进行改进。优化策略实现如下：

1. **基于窗口的页面选择**：考虑到 Qwen2.5-VL 模型能同时解析多张图片，但一次输入最多 4 张，每张限制在 420×420 像素。于是，将“答案页 ±2 页”范围内最多 4 张图像，按照自然页码顺序（保证跨页逻辑连贯）拼成 2×2 网格（总尺寸 840×840）以增加模型搜索范围。

策略	原始 baseline	优化方案
图像选择	固定使用 image1	答案页 ±2 页窗口
上下文保留	无	顺序保留前后页
布局优化	单页固定	2×2 网格布局
错误处理	无	索引异常处理

表 6: 基于窗口的页面选择策略对比

2. **prompt 逻辑提示**：纯 Baseline prompt 只关注问题，未指示模型页码位置。优化后，将 prompt 调整为：

```
Question: {example['question']}
The image below shows the most relevant pages from a multi-page document,
including the page that contains the answer.
Please read all information carefully and return only the correct answer.
If the answer is not present, say 'UNKNOWN'
```

指标	结果	说明
pass rate	0.58	正确率:58/100 正答

表 7: 基于多图拼接和提示词优化后的实验结果

优化后的策略在数据集上的准确率 (pass rate) 达到了 0.58，相较于原始 baseline 有较大的提升，可见该策略表现良好。本次优化高效利用 answer page idx 信息：彻底改变了 Baseline 无视答

案页的弊端，令模型一路“直奔目标”。2×2 拼接既保留了模型对多图片的支持，也保证单次输入不超显存限制；Prompt 设计注重指向性：提示“图像已含答案页”大幅降低模型猜测范围；鲁棒性提升：3 次重试机制和分段中间结果写盘，确保长时间运行中不会因偶发网络抖动崩溃。