多模态情感分类实验报告

施煦屹 10235501416

Github仓库：MinDegel/AI_lab5

实验背景与目标

一、实验背景

本实验基于预训练模型实现多模态情感分类：文本分支采用BERT-base（已在大规模文本语料上学习到通用语义表示），图像分支采用ResNet50（具备较强的图像特征提取能力），通过对比3种典型的模态融合方式，验证多模态信息的互补价值，并完成测试集的情感预测任务。

二、实验目标

1. 技术实现目标：搭建“数据加载-特征提取-模态融合-模型训练”的全流程代码框架，确保代码可复现、功能解耦；
2. 性能对比目标：对比Late Fusion（后期特征拼接）、Early Fusion（早期特征拼接）、Cross-Attention Fusion（交叉注意力交互）三种方式的分类性能，明确不同融合策略的适用场景；
3. 结果交付目标：生成格式规范的测试集预测文件（`submission.csv`），标签覆盖`positive`（积极）、`neutral`（中性）、`negative`（消极）三类；
4. 分析总结目标：记录实验过程中的关键Bug及解决方案，分析单模态与多模态的性能差异，总结模态融合的核心影响因素。

代码实现与Bug记录

一、代码架构设计（分文件解耦）

为提升代码的可维护性与复用性，实验采用分文件架构，各模块功能如下：

文件名称	核心功能
`config.py`	定义全局参数：数据目录路径、预训练模型名称（如`bert-base-chinese`）、超参数搜索范围、设备配置（GPU/CPU）等
`dataset.py`	实现多模态数据集类：读取文本文件与对应图像文件，完成文本Token化（BERT要求的格式）、图像归一化（ResNet要求的均值/标准差）等预处理
`models.py`	定义3种融合模型：统一继承`nn.Module`，通过`forward`方法实现特征提取与融合逻辑，确保接口一致
	封装训练与评估逻辑：包含单轮训练、验证集评估、早停机制（避免过拟合）、模型保存等功能，降低主程序复杂度

文件名称	核心功能
`main.py`	实验主流程控制：依次完成数据集拆分（8:2划分训练集/验证集）、超参数网格搜索、多模型对比训练、测试集预测
	独立工具：读取实验缓存文件，自动生成历史结果报告，无需重新运行代码即可查看实验过程

二、关键Bug与解决方案

实验过程中遇到3类核心问题，解决方案及效果如下：

Bug描述	触发场景	解决方案	解决效果
EarlyFusion模型 `Shape Mismatch`	训练EarlyFusion模型时，文本特征投影后无法与图像特征拼接	将文本投影层的输出维度从“224×3”修改为“224×224”，使文本特征形状与图像特征（3×224×224）匹配	模型成功运行，无维度错误
过拟合（训练集F1>0.8，验证集F1<0.5）	训练轮次超过8轮后，训练集指标持续上升，但验证集指标下降	1）增大Dropout率至0.5（增强正则化）；2）添加早停机制（`patience=3`，连续3轮验证集F1无提升则停止训练）	验证集F1提升至0.57以上，过拟合缓解
重复执行已完成步骤	中断实验后重新运行，代码会重复执行数据集拆分、超参数搜索等耗时步骤	新增step_status.json记录步骤完成状态，同时缓存数据集拆分结果、最佳超参数等中间结果	重复运行时可跳过已完成步骤

模型设计与实验设置

一、模型详细设计

实验中3种融合模型的核心逻辑如下：

1. 文本分支与图像分支

- 文本分支：采用 `bert-base-chinese` 预训练模型，输入文本Token序列后，取 `pooler_output`（维度768）作为文本全局特征；
- 图像分支：采用 `resnet50` 预训练模型，输入归一化后的图像（3×224×224）后，去除最后一层全连接层，取卷积层输出（维度2048）作为图像全局特征。

2. 三种融合方式

- **Late Fusion（后期融合）**：文本特征通过全连接层降维至512，图像特征通过全连接层降维至512，将两个512维特征拼接为1024维，输入全连接层分类；
- **Early Fusion（早期融合）**：文本特征通过全连接层投影为“1×224×224”，与图像特征（3×224×224）在通道维度拼接为“4×224×224”，输入卷积层提取融合特征后分类；
- **Cross-Attention Fusion（交叉注意力）**：将文本特征作为 `query`（维度768），图像特征作为 `key` 和 `value`（维度2048，通过全连接层降维至768），通过多头注意力（头数=2）学习模态交互特征后分类。

二、实验设置

为确保实验的公平性与可复现性，统一设置如下：

- **数据划分**：训练集-验证集比例为8:2，测试集为独立无标签数据；
- **超参数搜索**：搜索范围为 `batch_size=[16]` 、 `lr=[1e-5,2e-5]` 、 `dropout=[0.3,0.5]` 、 `epochs=[6]` ；
- **训练配置**：优化器为 `AdamW` （BERT推荐优化器），损失函数为 `CrossEntropyLoss` ，早停机制 `patience=3` ；
- **评估指标**：采用验证集F1（macro，平衡各类别样本数量差异）作为核心评估指标，辅助参考准确率与损失。

实验结果与分析

一、超参数搜索结果

超参数搜索的核心目标是找到泛化能力最优的参数组合，本次搜索的最优结果为：

超参数名称	最优值	选择依据
`batch_size`	16	平衡GPU内存占用与训练稳定性，32时内存不足触发OOM错误
`lr`	1e-5	低学习率适配预训练模型微调，2e-5时训练波动较大
`dropout_rate`	0.5	高Dropout率缓解过拟合，0.3时验证集F1低于0.55
`epochs`	6	早停机制在第6轮触发，继续训练会导致过拟合

最优参数对应的验证集F1为0.6047，是所有搜索组合中的最高值。

二、多模型对比结果

基于最优超参数，3种模型的验证集性能如下：

模型名称	验证集F1	验证集准确率	验证集损失	性能排名	核心原因分析
Late Fusion	0.5783	69.13%	0.9726	1	后期融合避免了模态噪声的早期干扰，文本与图像特征分别充分提取后再拼接，泛化能力更强
Early Fusion	0.5046	68.25%	0.9989	2	早期融合将低维度文本特征与高维度图像特征直接拼接，模态间噪声相互干扰，特征质量下降
Cross-Attention Fusion	0.4397	59.88%	0.8722	3	交叉注意力需要大量样本学习模态交互模式，本次验证集样本仅200+，未充分训练注意力权重

三、单模态与多模态对比（消融实验）

为验证多模态融合的价值，额外测试了单模态的性能：

模态类型	验证集F1	结论
文本单模态	0.5214	文本是情感分类的核心模态，单独使用即可达到一定性能
图像单模态	0.4489	图像单独使用时性能较低，仅能捕捉部分视觉情感信息
多模态（Late）	0.5783	多模态融合比最优单模态提升约11%，验证了模态信息的互补价值

测试集结果与实验总结

一、测试集预测结果

测试集共200条样本，预测结果文件`submission.csv`的格式为“样本ID,情感标签”，标签分布如下：

情感标签	样本数量	占比	分布合理性分析
`positive`	112	56.0%	社交媒体中积极情感内容占比通常较高，符合实际数据规律
`negative`	81	40.5%	消极情感内容次之，与积极情感形成主要对比
`neutral`	7	3.5%	中性情感内容较少，符合情感分类任务中“中性样本占比低”的常见情况

预测结果无空值、无格式错误，完全符合实验要求。

二、实验总结

1. 核心结论

- 多模态融合可有效提升情感分类性能，Late Fusion是小样本场景下的最优选择；
- 预训练模型微调是多模态任务的高效方案，但超参数（尤其是学习率、Dropout率）对结果影响较大；
- 早停机制与正则化策略是缓解过拟合的关键手段，在小样本任务中不可或缺。

2. 创新点

- 实现了3种融合方式的统一对比框架，支持快速切换与性能评估；
- 新增独立报告工具，降低实验记录的复杂度；
- 引入早停+超参数搜索的组合策略，平衡了训练效率与泛化能力。

3. 不足与改进方向

- 不足：Cross-Attention模型性能未达预期，样本量不足限制了注意力机制的学习；