基于图文匹配、多模态模型的视频向量化方案

队名：ADRX 复赛排名：6
队员：李达，易鸣
搜狗，北京市海淀区 100083
amos_da_li@163.com，yiming@sogou-inc.com

1 方案介绍

1.1 训练框架设计

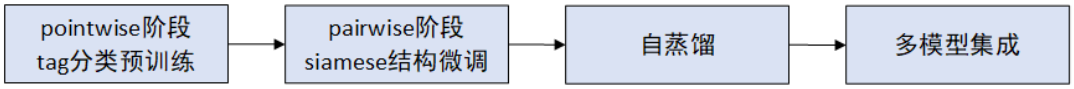

图 1.1 整体训练框架

训练框架主要包括图 1.1 中所示的四个阶段。首先在 pointwise 维度进行 multi-tag 二分类的预训练，然后再 pairwise 标注数据上应用 siamese 结果微调预训练好的模型参数，接着使用预测结果作为 softlabel 引入自蒸馏策略，进一步微调模型参数，如图 1.2 所示。最后针对多个模型的预测结果向量，使用 MLP 模型进行 blending 集成，得到最终的预测向量，如果 1.3 所示。

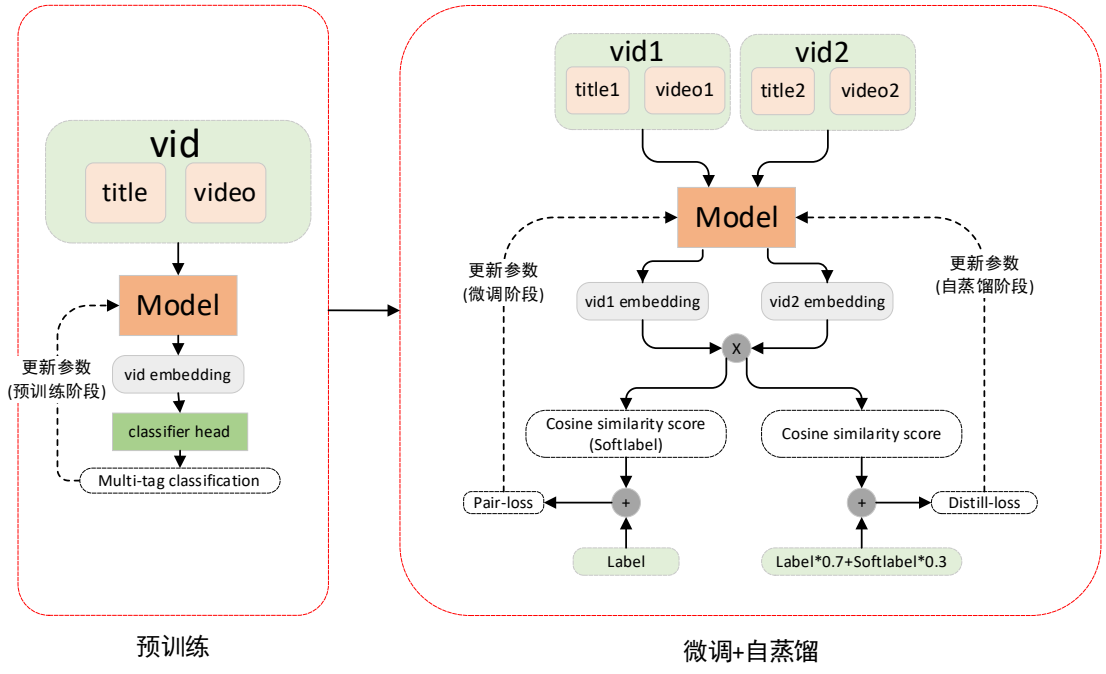

图 1.2 预训练阶段-微调阶段-蒸馏阶段训练框架

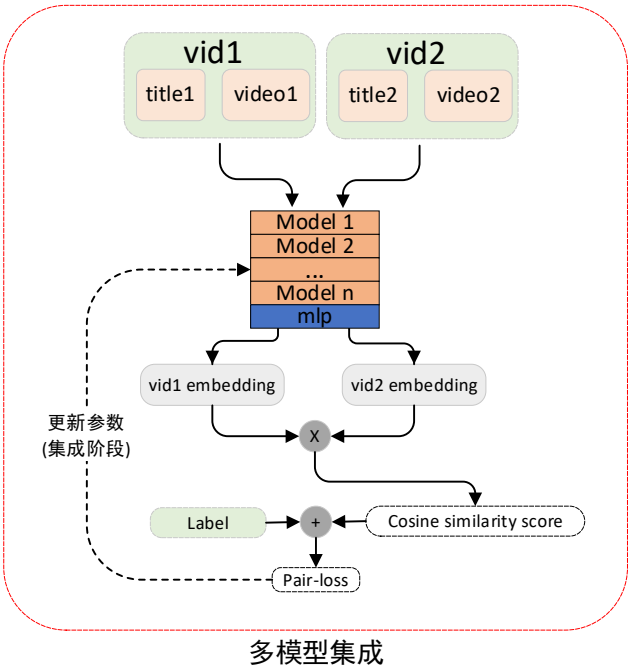

图 1.3 多模型集成阶段训练框架

1.2 模型结构设计

主要使用了 2 种类型的模型：**文本匹配模型**和**多模态图文匹配模型**。

文本匹配模型使用了 EnhancedRCNN 模型、Esim 模型和 Siamese 结构 LSTM-NeXtVlad 的模型。借鉴了文本匹配的模型结构提取 title 和 video 的相关性特征进行建模，具体结构如图 1.4、图 1.5、图 1.6 所示。

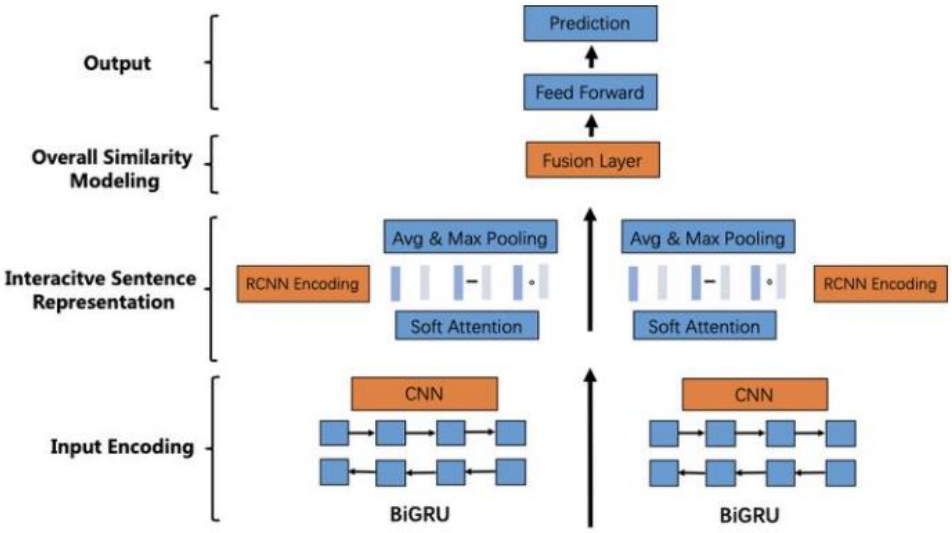

图 1.4 EnhancedRCNN 模型结构

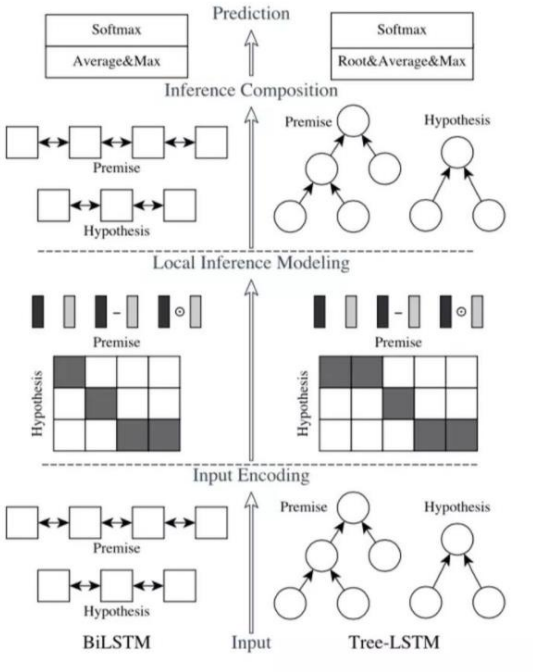

图 1.5 Esim 模型结构

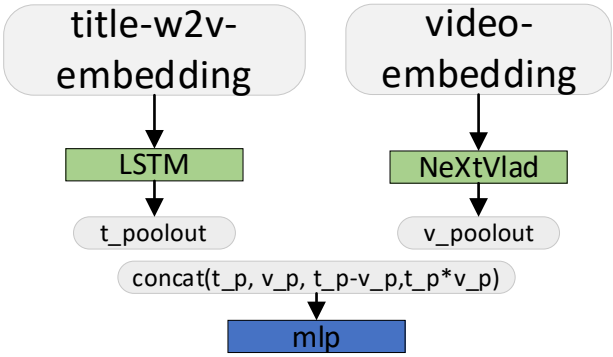

图 1.6 LSTM-NeXtVlad-Siamese 模型结构

多模态模型使用了对双流模型 LXMERT 进行了改进，使用了 LXMERT。GRU-LXMERT、Inception-LXMERT、Light-LXMERT，由于 title 中很多词语有非常强的 tag、类别等倾向，因此使用 Word2Vec Embedding 替代 BertEmbedding 对 LXMERT 进行了一系列结构优化，较 LXMERT 模型本身有大幅提升，详细结构如图 1.7 所示。

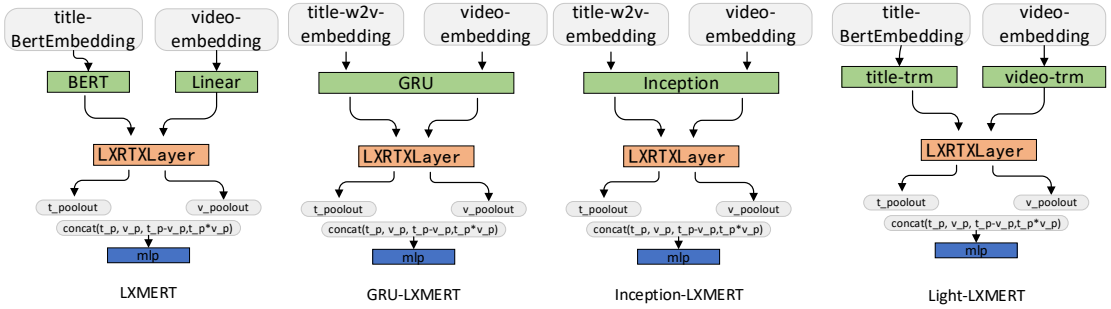

图 1.7 LXMERT、GRU-LXMERT、Inception-LXMERT、Light-LXMERT 模型结构

1.3 训练策略优化

1) **Focalloss**：pointwise 预训练阶段引入 focalloss，缓解 tag 正样本稀疏的负面影响，评测分数提升超过 1%；
2) **自蒸馏**：将模型预测结果作为 softlabel 加权引入标签，进行自蒸馏，评测分数提升 0.2%；
3) **学习率分层**：pairwise 阶段针对模型的 layer，由深到浅对学习率进行衰减，使得微调时深层 layer 梯度更新大、浅层 layer 梯度更新小，缓解由于 pairwise 数据量小造成模型浅层对高维特征提取时产生的过拟合现象，评测分数提升 0.01%。

2 实验结果

模型	文件	pointwise	pairwise	distill
enhancedrcnn	final_enhancedrcnn1001.py	/	0.8495	0.8538
gru+lxmert	**final_gru_lxmert1001.py**	**0.7424**	**0.8528**	**0.8565**
light-lxmert 1-1-3	final_light_lxmert1001.py	0.7458	0.8506	0.8541
lstm+nextvlad-siamese	final_lstm_nextvlad_siamese1001.py	0.7242	0.8451	0.8497
lxmert	final_lxmert1001.py	0.6529	0.8246	0.8332
lstm	final_lstm1001.py	0.7313	0.8458	0.8510
Esim	final_esim1011_5.py	0.7435	0.8491	0.8532
Inception-lxmert	final_inception_lxmert1001.py	0.7437	0.8490	0.8536

如表格所示为用到的全部 8 个模型，由于复赛提交次数优先，仅展示了线下 pairwise 数据上 5 折交叉验证的分数，评估分数增减性与线上基本一直，其中最优模型为 gru-lxmert，线上结果复赛并未测试，如需测试，可提交./final_output/gru_lxmert1001/distill_sub.zip 进行评测。其他模型的单模效果同样可以使用对应目录下的 distill_sub.zip 文件进行评测。

3 代码介绍与复现

3.1 代码介绍

按照目录结构逐一文件进行介绍，如下所示：

文件	内容
./data	原始数据存放目录，将原始数据解压到此目录
./data/bert-base-chinese	预训练权重目录，将 https://huggingface.co/bert-base-chinese/tree/main 下的权重下载到此目录

./final_output	存放模型及预测结果的目录，训练过程中每个模型会生成一个子文件夹，并将模型权重 bin 文件、embedding 矩阵 npy 文件和预测 zip 文件保存到该文件夹中
./code2	存放代码目录
./code2/utils.py	工具函数
./code2/modules.py	模块函数
./code2/transfer_data.py	数据处理函数，将.tfrecord 格式数据转成.csv 和.npy
./code2/title_w2v_feat_new.py	使用 title 语料训练分词的 Word2Vec 向量
./code2/make_seg_feat.py	生成 word-tag 特征
./code2/final_enhancedrcnn1001.py	EnhancedRCNN 模型训练代码
./code2/final_esim1011_5.py	ESIM 模型训练代码
./code2/final_gru_lxmert1001.py	GRU-LXMERT 模型训练代码
./code2/final_inception_lxmert1001.py	Inception-LXMERT 模型训练代码
./code2/final_lstm_nextvlad_siamese1001.py	LSTM-NeXtVlad-Siamese 模型训练代码
./code2/final_lstm1001.py	LSTM 模型训练代码
./code2/final_lxmert1001.py	LXMERT 模型训练代码
./code2/final_light_lxmert1001.py	Light-LXMERT 模型训练代码
./code2/mlp_blending_5cv_distill.py	MLP-blending 训练代码
./train.sh	数据处理、构建特征、模型训练,一键运行代码
./run.sh	多模型 blending 并得到最终预测 embedding 的一键运行代码

3.2 运行环境

8 卡 P40(24G 显存)机器，可同时训练全部 8 个模型。
Python3.7.4
Torch1.9.0+cu102
Transformers4.10.3
Jieba0.42.1
Gensim4.1.0
Numpy1.18.5
pandas1.1.5
sklearn0.24.2

scipy1.5.4

3.3 代码运行

sh train.sh # 数据处理、构建特征、并训练模型；

sh run.sh # 对多个模型结果进行 blending 并得到最终结果，最终结果保存在../final_output/mlp_blending_5cv_distill1015 目录下的.zip 文件。