代码文件说明

|——src1 方案一HAN分类文件

　　|——HAN_infer.py 推理使用代码

　　|——model_han_infer 推理使用model

　　|——HAN_model_fin.py 训练使用代码

　　|——model_han_fin 训练存储model地址

　　|——result HAN模型推理及训练保存文件

　　　　|——result_han_fin.csv 训练预测结果文件

　　　　|——result_han_infer.csv 推理结果文件

|——src2：方案二prompt分类 与 摘要文件

　　|——classify　prompt分类文件

　　　　|——infer_prompt.py 推理使用代码

　　　　|——train_prompt.py 训练 prompt分类使用代码

　　　　|——model_seed{...}.bin 五个prompt分类推理使用模型文件

　　|——extract 摘要文件

　　　　|——dataprepare.py 数据预处理

　　　　|——extract_vectorize.py 句子向量提取

　　　　|——extract_model.py 摘要抽取模型训练代码

　　　　|——infer.py　摘要抽取模型推理代码

　　|——result

　　　　|——final_fuxian.xls prompt分类结果文件

　　　　|——final_rule.xls 抽取结果文件

|——FinBERT_L-12_H-768_A-12_pytorch 预训练模型

|——FinBERT_L-12_H-768_A-12-tf　预训练模型

|——Dockerfile

|——ensemble.py 两个分类方案的融合代码

|——requriment.txt

|——result.xls　两个分类方案的融合结果，摘要结果，可直接提交文件

|——run.sh　推理复现代码,可直接运行

复现说明

上传文件包含 **代码文件**

- 该docker环境由服务器conda导出的配置文件`environment.yml`，conda 环境名为 `paddle`
- 创建docker镜像命令`docker build -t tianma .`
- 进入docker镜像后输入 `/bin/bash`进入conda环境paddle
- 该项目的其中一个依赖包无法pip 安装，**需要复现人员手动安装**，已提供离线安装文件 `OpenPrompt-main` 安装命令为:

```
cd OpenPrompt-main/
cd OpenPrompt-main/
python setup.py install
```

- 由于本机资源限制、无GPU环境最终测试，除bert相关代码外，其余代码均在cpu上通过运行测试。

- 推理复现：`bash run.sh`
- 推理必要条件： 进入paddle环境、完成OpenPrompt安装
- 训练复现：`bash train.sh`(代码包含torch和tensorflow、由于tensorflow框架特性以及采样随机性，训练过程可能无法完全复现，效果可能有微小波动，具体训练过程由于GPU原因未在docker测试，在build过程中未加入train.sh，若虚拟机没有此文件，需要手动添加)

文本摘要部分

摘要数据分析

摘要训练数据的统计信息：

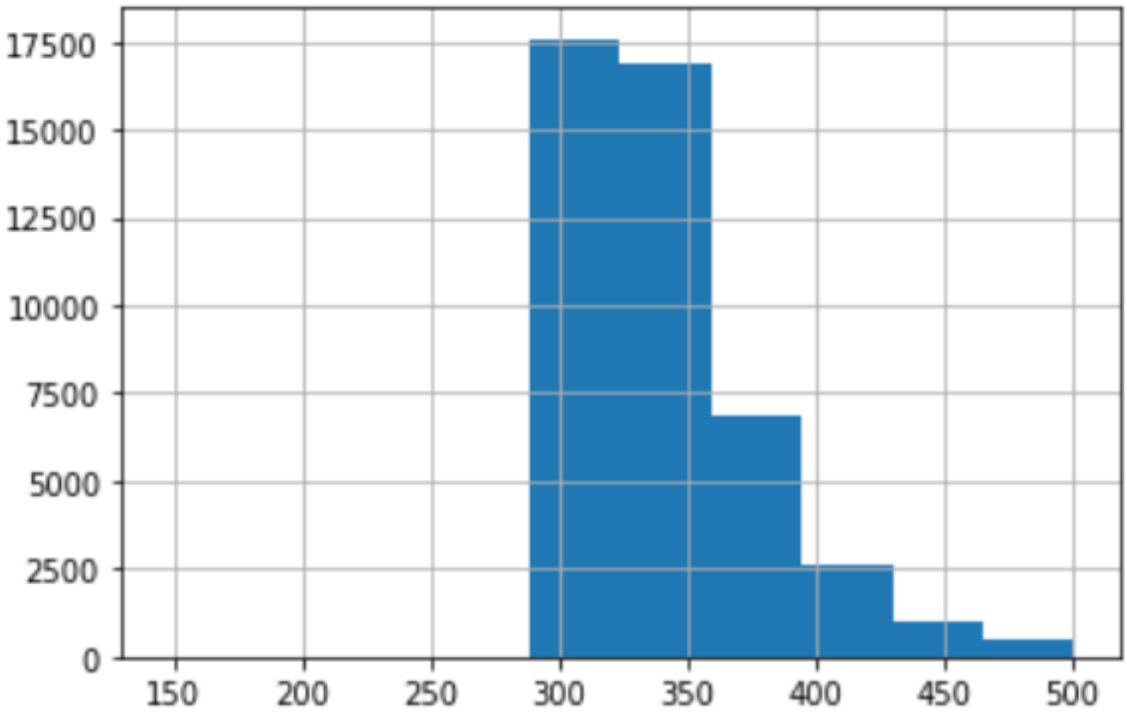
1.摘要训练集样本45440，重复样本为2369条

2.摘要最多字数为501，平均字数为341，摘要字数统计如下图

3.句号作为句子分割，最大包含35句话，平均包含5句话，最小为1句话

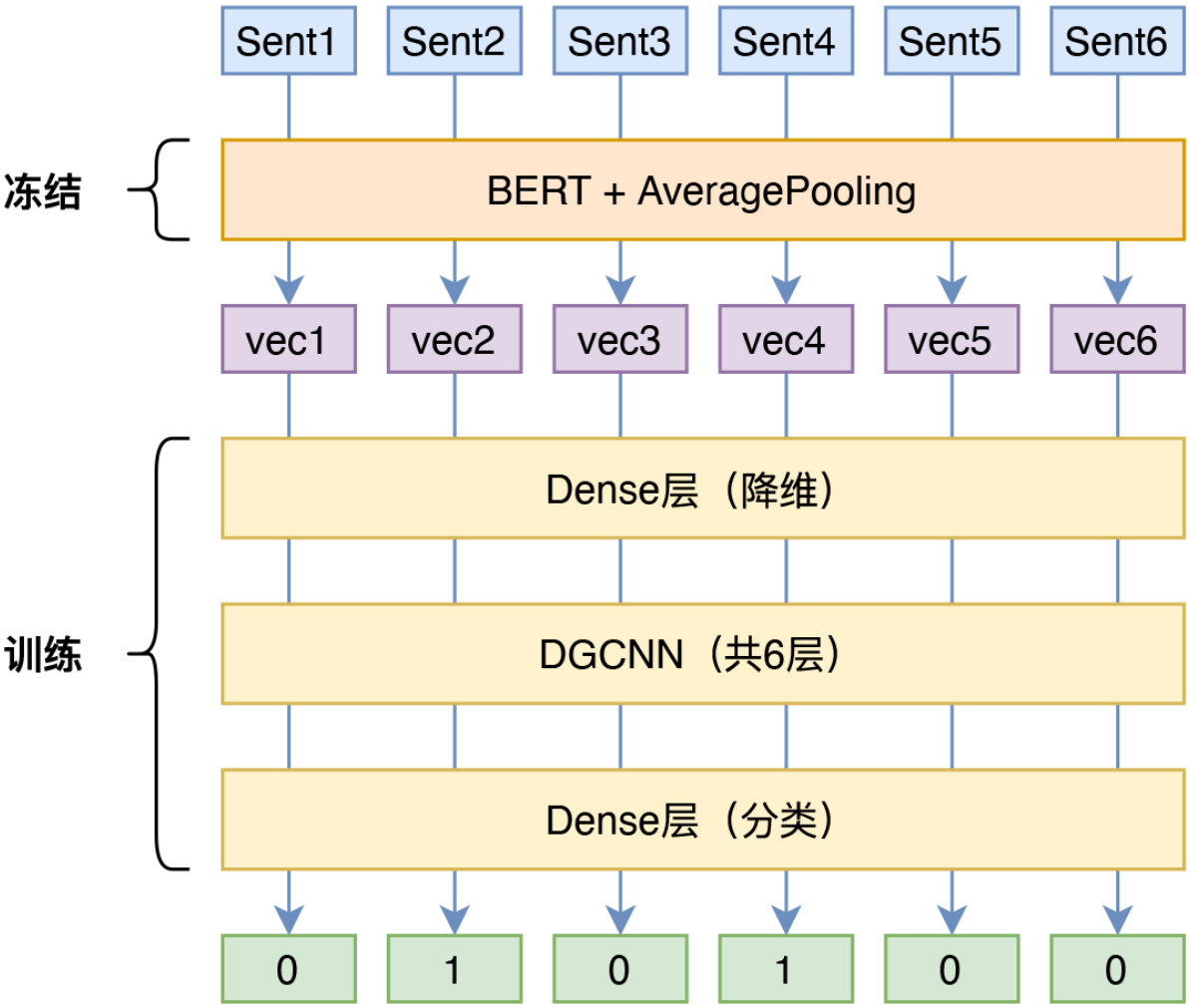
4.计算摘要在新闻文本中的覆盖率为100%，

故本次摘要全部来自新闻文本，无生成的字段，故在模型构建以摘要抽取模型为主，无需考虑摘要生成模型。



方案简介

在本赛提得文本摘要部分中，使用金融领域特定的预训练模型FinBert
结合AveragePooling的方式提取句子向量。将句子向量的序列输入DGCNN模型对每个句子进行摘要分类，判断每个句子是否为摘要，模型结构如下：



操作步骤

1. 数据预处理:对文本以及摘要基于句号的分句，提取文本前20句，根据摘要进行句子级别的二分类标注转换。
2. 句子向量提取：加载Finbert权重，前向传播得到句子向量。
3. 训练句子级别分类器DGCNN，作为抽取模型。

文本分类部分

文本分类数据分析

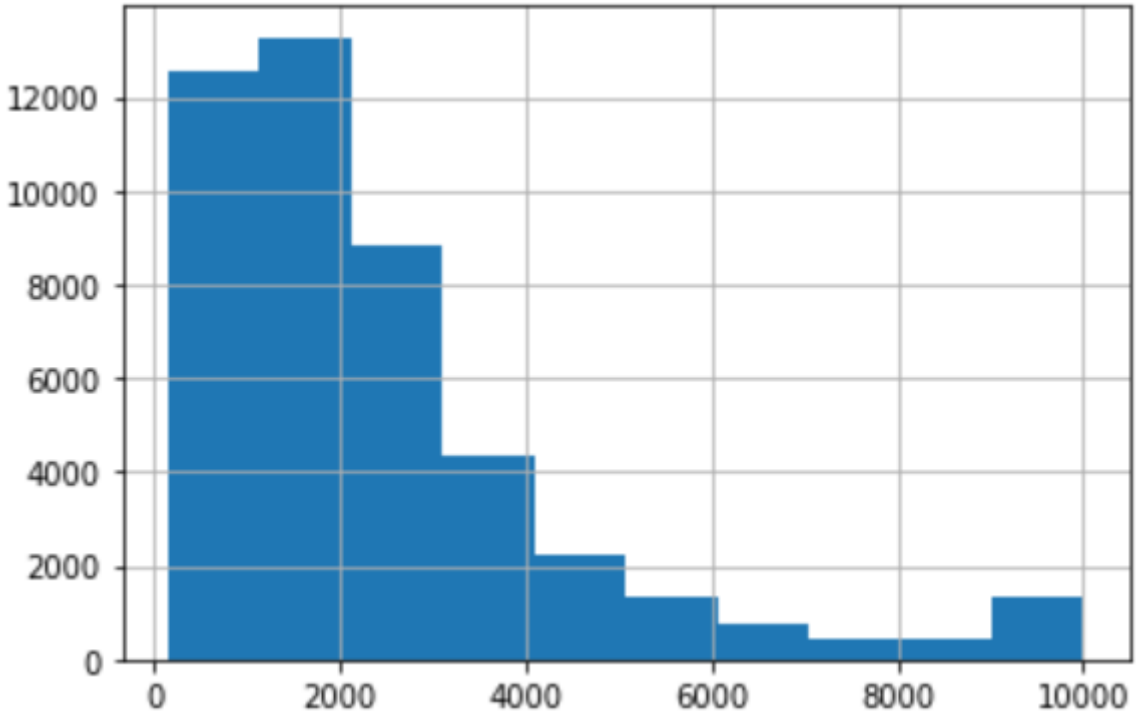
新闻文本训练数据的统计信息：

1.新闻文本训练集样本45440，完全重复样本为218条。

2.新闻文本最多字数为10000，平均字数为2421，新闻文本字数统计如下图。

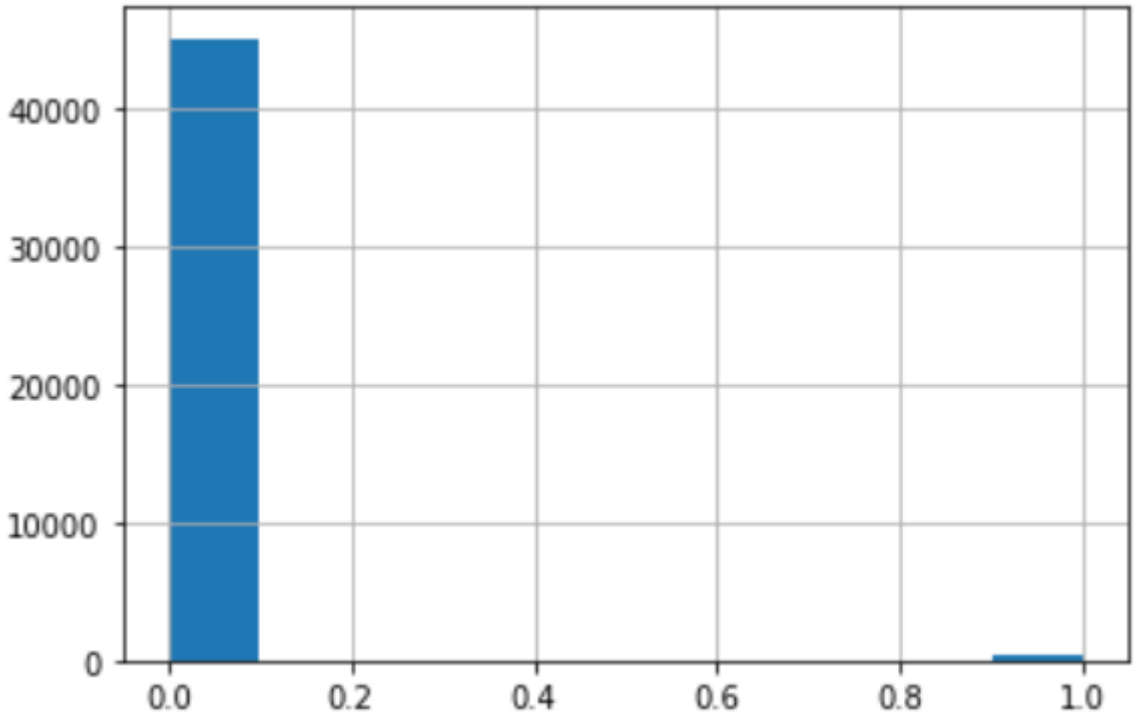
3.文本句号作为句子分割，最大包含637句话，平均包含37句话。

分析新闻本文数据可知直接使用新闻文本会产生过多冗余的信息，而摘要是对新闻文本的概括，在对文本分类时，两个模型都是按照512截断进行数据处理。



文本分类难点分析:

1.样本不平衡，训练集有45440条样本，400多条正样本，正样本占比仅为0.0099，导致模型在训练的时候容易过度拟合负样本，如下图所示



2.跨领域问题，本次数据训练集为金融科技三个子领域的数据，测试集为另外两子领域的数据，本次分类最大的难点为跨领域0样本学习。

方案简介

本赛题文本分类部分属于长文档分类任务，难点是跨领域的0样本学习。为解决难点，本团队使用Finbert句子向量计算文档相似度，从源域筛选出目标域相似样本；使用疑似金融实体的词典进行实体统一操作；

- 使用二种分类方案的融合：

1. 针对zero-shot的自监督Prompt-learning:文本的前512字符截断，将任务构造为完形填空形式，使用Finbert继续进行MLM任务的训练。该思路源于文献[1]。
2. 长文本分类利器HAN模型。

方案一：prompt分类模型

数据处理方式

1.训练文本属于金融科技领域的三个子领域，测试集属于金融科技其他两个领域，为拉近领域之间的距离。我们利用自己构建的实体统一库。将文本数据中所属实体统一为【金融科技】，然后再用lac分词配置【金融科技】。

2.比赛文本较长，全部数据用来训练,信息过多。本次文本都与新闻相关，大部分信息都被摘要所包含，分析摘要的平均长度为341，故截取新闻文本的512长度

操作步骤

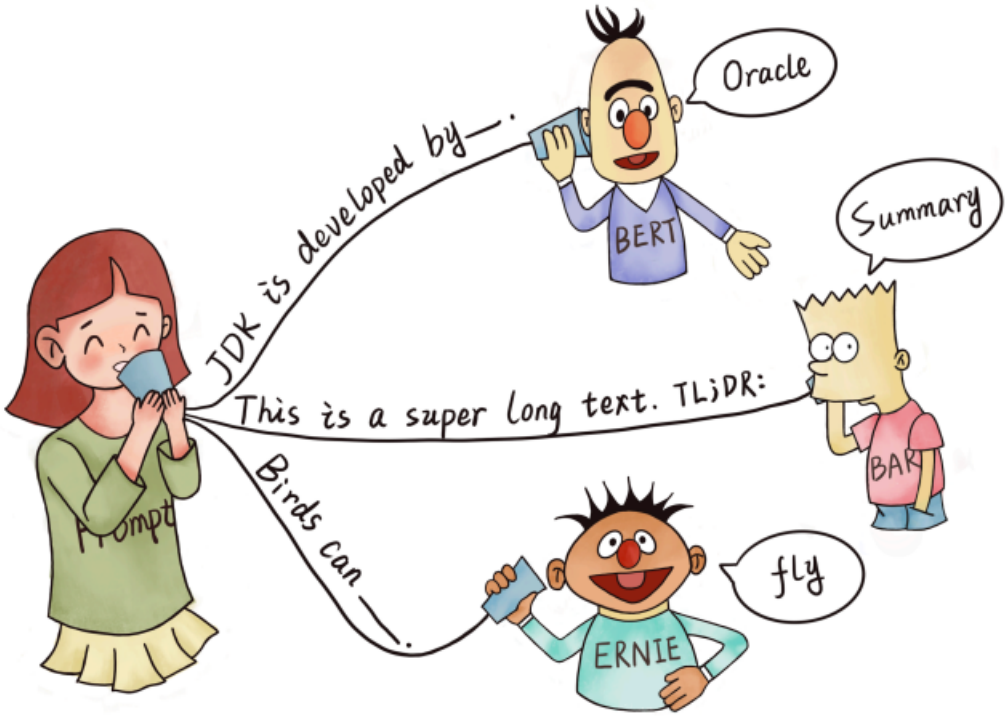
1.使用命名实体识别模型与词性标注模型，从数据中提取机构实体片段，结合先验知识构建实体统一库。

2.使用摘要部分的句子向量，计算训练集和测试集样本的余弦相似度矩阵，初步筛选与测试集分布相近的训练集。

3.利用bagging思想，每次选取不同的不样本，训练模型进行融合‘

prompt模型

融入了Prompt的新模式大致可以归纳成”pre-train, prompt, and predict“，在该模式中，下游任务被重新调整成类似预训练任务的形式。例如，通常的预训练任务有Masked Language Model， 在文本情感分类任务中，对于 **"I love this movie."** 这句输入，可以在后面加上prompt **"The movie is __"** 这样的形式，然后让PLM用表示情感的答案填空如 "great"、"fantastic" 等等，最后再将该答案转化成情感分类的标签，这样以来，通过选取合适的prompt，我们可以控制模型预测输出，从而一个完全无监督训练的PLM可以被用来解决各种各样的下游任务。



prompt 方法在少样本学习中非常有用，当没有足够的训练示例来完全指定期望的行为，使用 prompt 将模型推向正确的方向特别有效。因此prompt适合本次的跨领域分类任务。

方案二：长文本分类利器HAN模型

数据处理方式

1.训练文本属于金融科技领域的三个子领域，测试集属于金融科技其他两个领域，为拉近领域之间的距离。我们利用自己构建的实体统一库。将文本数据中所属实体统一为【金融科技】，然后再用lac分词配置【金融科技】。

2.文本数据一共45440条，并且都是属于金融科技相关新闻领域的文本。由于文本所属领域比较固定，我们采用了重新在比赛文本上训练词向量的方式。

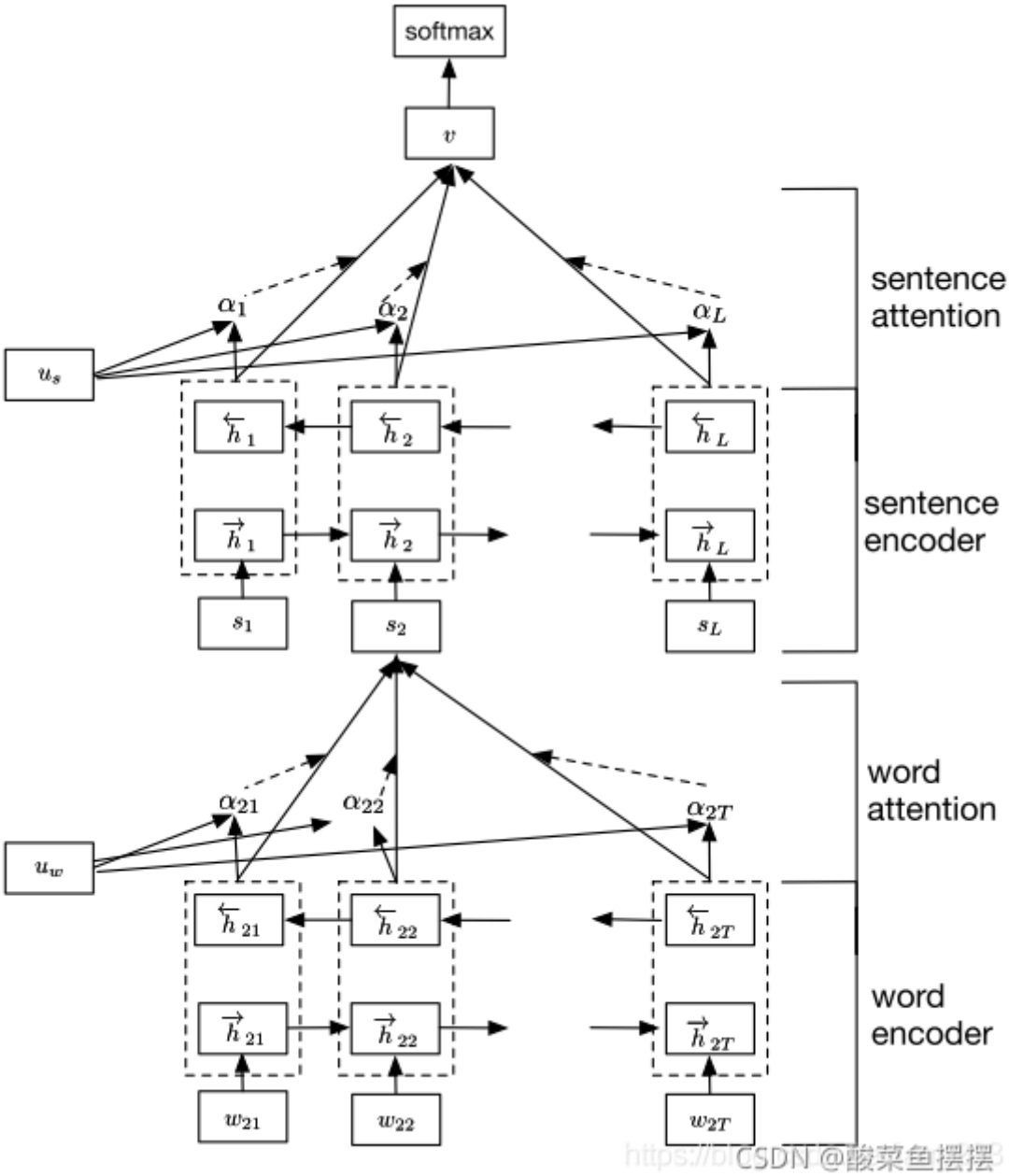
3.比赛文本较长，全部数据用来训练,信息过多。本次文本都与新闻相关，大部分信息都被摘要所包含，分析摘要的平均长度为341，故截取新闻文本的512长度。

4.训练集有45440条样本，其中正样本占比仅为0.0099。只有400多条正样本，导致模型在训练的时候容易过度拟合负样本。我们使用下采样的方式，采用的方式

是保留所有正样本，抽取2000条负样本进行训练。

建模方案

1.HAN模型结构[2]



2.HAN模型应用优势：
HAN模型的整体思路是句子由单词组成，文档由句子组成，据此可以构建一个自下而上的层次结构。由于HAN是一种基于层次化attention的文本分类模型，可以利用attention机制识别出一句话中比较重要的词语，利用重要的词语形成句子的表示，同样识别出重要的句子，利用重要句子表示来形成整篇文本的表示。在经过我们的【科技实体】统一后，HAN模型比较适合用于本次跨领域的分类任务中。

3.训练策略：
采用分层学习率，设置embed层学习率为其他网络层的10%，通过早停策略来使模型更加稳定防止过拟合（线下5折验证，实际训练过程中容易发生过拟合，导致acc上升的情况，线上f1_score反而减小）。

融合策略

1.摘要提取使用单模的15折做概率投票融合

2.两个分类模型使用概率投票融合

[1] Liu P, Yuan W, Fu J, et al. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing[J]. arXiv preprint arXiv:2107.13586, 2021.
[2]Hierarchical Attention Networks for Document Classification