项目报告

选题：新闻文本分类系统　　作者：蓝俊强

一、环境与依赖

1.1 开发环境

- **操作系统：Windows**
- **编程语言：Python 3.11.10**
- **IDE：PyCharm**

1.2 库

- **pandas==2.2.3**

  处理数据集，生成适合模型输入的格式

- **regex==2024.11.6**

  数据清洗

- **jieba==0.42.1**

  对中文文本进行分词处理

- **sklearn==1.6.1**

  数据集划分及模型评估

- **fasttext==0.9.2**

  开源模型

1.3 环境配置

- **虚拟环境：PyTorch 2.5.1　CUDA版本：12.4**

  pip安装：pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu124

二、技术算法及实现思路

2.1 数据集

(1)对于给定可下载的数据集，其中包含382688个样本，共15个分类，部分样本没有关键词

(2)样本格式：新闻ID，分类code，分类名称，新闻字符串（仅含标题），新闻关键词

样本示例：6552295326462509575_!_106_!_news_house_!_2018年的房价回是什么趋势？_!_房地产泡沫,海南岛,房价,大洗牌,房地产开发商

(3)由于选题规定了十个分类，所以将数据集中的6个分类划分为"其他"分类，分别是：'民生','文化','旅游','国际','证券','农业'

2.2 数据预处理

(1)使用pandas处理数据集，生成DataFrame格式，类似Excel文档。能够使用列索引名称来访问数据，数据分类会更清晰

(2)将新闻标题和关键词合并作为输入

(3)使用regex去除无用字符（如标点符号、特殊符号等），保留中文、英文、数字

(4)使用jieba对输入文本进行分词处理。虽然fasttext内置分词，但它会将每一个中文字符视为一个词，忽略了中文组词的意思

(5)使用五折交叉验证分层划分数据，首先对各个分类的新闻进行分层抽样，然后将数据分为五折，依次使用其中一折作为验证集，其余作为训练集，循环进行五次单独的模型训练

【①去除停用词后模型训练时，准确率为0.90，但预测效果很差，基本预测不对；②在未去除停用词时，训练集中的词数为1w3左右，去除后只剩6k，可能是数据集中已是标题+关键词了，其中的停用词多半在标题中，也起到了重要作用；③如果模型训练不去除停用词，而预测时去除停用词，则可能是因为数据处理方式不一致，效果比不去除停用词还要差一些；④虽然原预测代码不去除停用词，但jiaba.analyse实际会用到TF-IDF来计算词的评分，因此也是可以过滤掉一些常见的，无关紧要的词】

2.3 模型训练（这个部分我只是简单阅读了一下fasttext的代码，然后根据提问ai来写的）

(1)调用fasttext.train_supervised监督学习模型，传入适当超参数，这些超参数会被相应的函数映射到内部枚举值。之后会构造C++层的fasttext::args对象，而Python则只是作为一个接口。

(2)FastText使用 C++层的Matrix和Dictionary类来加载和存储数据，并会将每行文本分割为单词。随后计算每个单词在训练集中的出现次数。再根据minCount 参数丢弃频率低于阈值的单词。

n-gram字词是FastText的核心创新之一，即将每个词会被拆分为多个连续的子词片段（如 bigram 和 trigram）。例如：对于词 "fast"，其子词特征包括：<fa, ast, fas, st>。
< 和 > 是边界符号，用于区分词的开头和结尾。
这些子词特征与完整的词本身共同构成该词的表示空间。

并且会初始化输入矩阵和输出矩阵

(3)训练过程的核心是一个循环，执行以下步骤：

随机采样：
从训练集中随机抽取一批样本。
前向传播：
将输入文本转换为单词和子词的索引列表。
查找输入矩阵中对应的嵌入向量。
计算预测值（对于 supervised 模型，预测的是标签的概率分布）。
损失计算：
根据指定的损失函数（如 hs 或 softmax），计算预测值与真实标签之间的误差。
反向传播：
计算梯度并更新输入矩阵和输出矩阵的权重。
学习率调整：
根据 lrUpdateRate 参数动态调整学习率。
多轮训练：
上述步骤会在 epoch 指定的轮数内重复执行。每轮训练结束后，模型会重新打乱训练数据以提高泛化能力。

2.4 模型评估

(1)评估指标选择

准确率：准确率是分类正确的样本数占总样本数的比例，能够直观地反映模型的整体分类效果

f1分数：f1分数是精确率和召回率的调和平均值，用于综合评估模型在正类上的表现

(2)评估方法

采用了StratifiedKFold进行能够分层抽样的五折交叉验证来评估模型性能。具体步骤如下：

- 数据划分：
  StratifiedKFold能够读取数据标签，将数据集分为五折，其中分层抽样能够使得每个折中的类别分布与整体数据集的类别分布尽可能一致，每次使用其中一折作为验证集，其余四折作为训练集。整个过程重复五

次，确保每折数据都被用作验证集一次。

- 模型训练与预测：
  在每次循环中，使用 fasttext.train_supervised 方法训练模型，并对验证集进行预测。
- 结果记录：
  对于每次验证，分别计算准确率和宏平均 F1 分数，并将结果存储。
- 结果汇总：
  最终计算五次验证的平均准确率和平均 F1 分数，作为模型的整体评估指标。

折数	准确率	f1分数
1	0.9101	0.9079
2	0.9115	0.9094
3	0.9110	0.9087
4	0.9106	0.9081
5	0.9104	0.9086
平均	0.9107	0.9085

2.5 预测数据

使用pd.read_csv(), pd.read_excel()读取csv或xlsx文件，使用jieba.analyse提取新闻内容中的10个关键词，之后进行文本清洗和分词处理。

加载并调用model.predict方法，生成前3个最可能的分类标签及其对应的置信度。

这一部分主要考虑了各种错误异常的处理，例如：1.用于预测的文件有缺失值 2.新闻内容为空，只有新闻标题 3.读取的数据格式 4.使用exception进行异常捕获