哈尔滨工业大学（深圳）

Harbin Institute of Technology, Shenzhen

实验报告

Experimental report

学　院：　计算机科学与技术

题　目:　基于朴素贝叶斯的文本分类

姓　名:　邵彦铭、陆柏同

学　号:　200110430、200110410

专　业:　计算机类

一、实验目的和背景

实验旨在深入理解文本分类任务，学习如何运用朴素贝叶斯算法进行文本分类。文本分类作为自然语言处理领域的一个关键任务，其目标在于根据文本的内容或主题对其进行分类。

本次实验以新闻标题文本分类为案例，采用了 THUCNews 数据集，该数据集包含了近 10 年新浪新闻的标题以及相应的类别标签，共涵盖 14 个不同类别。数据集已经经过预处理，并被划分为训练集、验证集和测试集。

在本实验中，我们将运用朴素贝叶斯算法对数据集进行训练，该算法基于贝叶斯定理来实现文本分类。

二、数据集及处理方式

1. 数据集

本实验使用的数据集是 THUCNews 文本分类数据集。该数据集包含近 10 年新浪新闻标题和对应的分类标签，共有 14 个分类类别。

数据集预先处理，分为训练集、验证集和测试集三部分。训练集包含 678998 条数据，验证集 75636 条，测试集 77837 条。每条数据格式为“新闻标题文本 类别”。

2. 处理方式

实验先对原始数据集进行预处理：

(1) 分词:利用结巴分词对新闻标题文本进行分词处理,提取关键词。

(2) 去停用词:根据停用词表过滤掉常见停用词,如“的”“是”等没有区分力的词。

(3) 文本向量表示:利用 CountVectorizer 进行词袋表示,将分词文本转换为词频向量。

预处理后,训练模型时使用训练集训练、验证集验证,测试时使用测试集。CountVectorlizer 会根据训练集构建词汇表,用于将其他集文本也转换为相同格式的词频向量。数据集预处理保证了文本数据格式的规范性和模型训练的效率。

分词和去停用词提取了文本的关键信息。CountVectorizer 实现了从稀疏文本到稠密数字特征的转换,为后续算法训练和预测提供了便利。

三、实验步骤与关键过程

1. 数据加载与预处理

将数据集加载到内存中,对新闻标题文本进行分词处理,利用 jieba 分词工具提取词汇,再对词汇表进行去停用词过滤。其主要代码如下所示：

```
#Import experimental data
train_data = load_dataset('data/train.txt')
test_data = load_dataset('data/test.txt')

# Participle
train_data = [(tokenize(text), label) for text, label in train_data]
test_data = [(tokenize(text), label) for text, label in test_data]
```

其中使用到的自定义方法代码如下所示：

```
# Load the dataset
def load_dataset(file_path):
    dataset = []
    with open(file_path, 'r', encoding='utf-8') as file:
        for line in file:
            text, label = line.strip().split('\t')
            dataset.append((text, label))
    return dataset

# Text data segmentation
def tokenize(text):
    return ' '.join(jieba.cut(text))
```

2. 文本语料向量化

利用 sklearn 中的 CountVectorizer 对预处理后的标题文本进行数字化处理，将文本转换为词频矩阵，为朴素贝叶斯算法提供数字表示的特征。其主要代码如下所示：

```
# Text vectorization
vectorizer = CountVectorizer()
X_train = vectorizer.fit_transform([text for text, _ in train_data])
y_train = [label for _, label in train_data]
X_test = vectorizer.transform([text for text, _ in test_data])
y_test = [label for _, label in test_data]
```

3. 朴素贝叶斯模型训练

实例化 MultinomialNB 分类器对象，利用 TRAIN 的特征向量 X_train 和标签 y_train 进行模型训练，即统计每个类别下特征条件概率。其代码如下所示：

```
# Build and train a Naive Bayes classifier
clf = MultinomialNB()
clf.fit(X_train, y_train)
```

4. 模型评估与结果输出

将测试集特征向量 X_test 传入训练好的分类器，获得预测类别 y_test_pred。然后计算测试精度并打印分类报告，评估模型在新数据上的表现。输出测试集准确率和分类报告，以数值和报告的形式展示朴素贝叶斯模型在这个新闻分类任务中的效果。

其主要代码如下所示：

```
# Perform model performance testing
y_test_pred = clf.predict(X_test)
accuracy = accuracy_score(y_test, y_test_pred)
report = classification_report(y_test, y_test_pred)
print()
print("Test Accuracy:", accuracy)
print("Classification Report:\n", report)
```

四、实验结果

从实验结果可以看出，在测试集上的分类准确率为 89.82746%，表明模型在大多数情况下能正确预测新闻标题的类别。运行结果截图如下：

```
Test Accuracy: 0.8982745994835362
Classification Report:
              precision    recall  f1-score   support

          体育       0.95      0.97      0.96     12125
          娱乐       0.86      0.95      0.90      8514
          家居       0.91      0.82      0.86      3017
          彩票       0.98      0.74      0.84       720
          房产       0.95      0.77      0.85      1859
          教育       0.93      0.94      0.93      3965
          时尚       0.94      0.71      0.81      1272
          时政       0.87      0.89      0.88      5923
          星座       0.96      0.62      0.76       340
          游戏       0.93      0.75      0.83      2281
          社会       0.84      0.89      0.87      4783
          科技       0.90      0.90      0.90     15200
          股票       0.87      0.93      0.90     14334
          财经       0.93      0.74      0.82      3504

    accuracy                           0.90     77837
   macro avg       0.92      0.83      0.87     77837
weighted avg       0.90      0.90      0.90     77837
```

详细查看分类报告，可以看到不同类别的指标：

- 体育、教育、科技等类别的 precision 和 recall 均大于 0.9，F1 值也较高，说明模型在这些类别上的表现优异。
- 彩票、房产类别 recall 值较低，但 precision 值高，可能是因为这两个类别样本量较少，训练不足导致。
- 时尚、游戏类别 F1 值中等，显示模型识别能力需要进一步提升。
- 星座类别 precision 值高但 recall 和 F1 值较低，表明需要更多星座类文本来训练该类别的识别能力。

综上，朴素贝叶斯模型在大多数高样本量类别上的性能很好，但部分低样本类别如星座

识别能力需要优化。同时,由于新闻标题信息量小,特征间依赖性弱,符合朴素假设,因而算法效果显著。

为了进一步优化模型表现，可以从以下方面进行改进：

1. 对低样本类别进行过采样或数据增强,提升识别能力。
2. 增加文本特征如词序信息,使模型学习到上下文。

五、团队分工与实验心得

实验中邵彦铭同学负责实验代码的编写，陆柏同负责实验报告的写作。

通过本次实验，小组同学进一步理解了文本分类任务的流程,包括数据预处理、特征工程、算法模型建立与训练等各个环节，并且练习了朴素贝叶斯算法在文本分类中的应用。此外，小组同学也对如何合理评估分类模型有了更进一步的理解，选择错误率与分类报告两个方面进行量化检查。

授课教师评审意见：

□报告合格　□报告不合格

授课教师签字:

签字日期：