中国科学院大学

试 题 专 用 纸

课程编号：B2512009H

课程名称：自然语言处理（研讨课）

任课教师：赵阳

注意事项：

1. 考核方式：_____

期末大作业报告

1　项目背景

自然语言处理（Natural Language Processing, NLP）是人工智能领域的重要分支，旨在让计算机能够理解、解释和生成人类语言。随着深度学习技术的发展，特别是 Transformer 架构的提出，NLP 领域取得了突破性进展。本项目聚焦于文本分类任务，探索如何利用预训练语言模型提升分类性能。

文本分类是 NLP 中的基础任务之一，广泛应用于情感分析、主题分类、垃圾邮件检测等场景。传统方法依赖于手工特征工程和浅层机器学习模型，而现代方法则利用深度神经网络自动学习文本表示。

1.1　研究动机

尽管预训练语言模型在多个 NLP 任务上取得了优异表现，但在特定领域的应用中仍面临挑战：

1. 领域适应性问题：通用预训练模型在专业领域的表现往往不如预期
2. 计算资源限制：大规模模型的训练和推理需要大量计算资源
3. 数据标注成本：高质量标注数据的获取成本高昂
4. 模型可解释性：深度模型的决策过程难以解释

因此，本研究旨在探索轻量级且高效的文本分类方法，在保证性能的同时降低计算成本。

1.2　相关工作

近年来，文本分类领域涌现出多种有效方法。Kim (2014) 提出的 TextCNN 利用卷积神经网络捕获局部特征，在多个数据集上取得良好效果。随后，循环神经网络（RNN）及其变体 LSTM、GRU 被广泛应用于序列建模任务。

2017 年，Vaswani 等人提出的 Transformer 架构彻底改变了 NLP 领域。基于 Transformer 的预训练模型如 BERT、GPT 系列在各类任务上刷新了性能记录。这些模型通过在大规模语料上进行预训练，学习到丰富的语言知识，然后在下游任务上进行微调。

然而，大规模预训练模型的参数量巨大，部署成本高。为此，研究者提出了多种模型压缩技术，包括知识蒸馏、剪枝、量化等，旨在在保持性能的同时减小模型规模。

2　方法

本研究采用基于预训练语言模型的文本分类方法，主要包括以下几个步骤：

2.1　数据预处理

数据预处理是文本分类的重要环节，主要包括：

- 文本清洗：去除 HTML 标签、特殊字符等噪声
- 分词处理：使用 WordPiece 或 BPE 算法进行子词切分
- 序列截断：将文本长度统一到固定长度（如 512 个 token）
- 数据增强：通过同义词替换、回译等方法扩充训练数据

2.2　模型架构

我们采用 BERT 作为基础编码器，在其之上添加分类层。具体架构如下：

1. 输入层：将文本转换为 token 序列，添加特殊标记 [CLS] 和 [SEP]
2. 编码层：使用 12 层 Transformer 编码器提取文本表示
3. 池化层：提取 [CLS] 位置的隐藏状态作为句子表示
4. 分类层：通过全连接层和 Softmax 函数输出类别概率

模型的损失函数采用交叉熵损失：

$$\mathcal{L} = -\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log(\hat{y}_{ic})$$

其中 N 是样本数量，C 是类别数量，y_{ic} 是真实标签，$\hat{y}_{ic}$ 是预测概率。

2.3　训练策略

为了提高模型性能和训练效率，我们采用以下训练策略：

- 学习率预热：前 10% 的训练步骤线性增加学习率
- 学习率衰减：使用余弦退火策略逐步降低学习率
- 梯度裁剪：限制梯度范数不超过 1.0，防止梯度爆炸
- 早停机制：当验证集性能连续 5 个 epoch 未提升时停止训练
- 对抗训练：在 embedding 层添加扰动，提高模型鲁棒性

3　实验设置

3.1　数据集

我们在三个公开数据集上进行实验：

表 1: 数据集统计信息

数据集	训练集	验证集	测试集
IMDB	20,000	5,000	25,000
AG News	96,000	24,000	7,600
DBpedia	448,000	112,000	70,000

3.2　实验参数

主要超参数设置如下：

- 批次大小：32
- 学习率：2e-5
- 训练轮数：10
- 最大序列长度：512
- 优化器：AdamW
- 权重衰减：0.01

4　实验结果

我们将提出的方法与多个基线模型进行对比，包括传统机器学习方法（SVM、朴素贝叶斯）和深度学习方法（TextCNN、LSTM、BERT）。

实验结果表明，基于 BERT 的方法在所有数据集上均取得最佳性能。在 IMDB 数据集上，准确率达到 94.2%，相比 TextCNN 提升了 2.5 个百分点。在 AG News 数据集上，准确率为 95.1%，超过 LSTM 模型 3.8 个百分点。

此外，我们还进行了消融实验，验证各个组件的有效性。结果显示，对抗训练可以提升 0.8% 的准确率，数据增强带来 1.2% 的提升，学习率预热策略贡献 0.5% 的性能增益。

5　结论与展望

本研究探索了基于预训练语言模型的文本分类方法，通过合理的模型设计和训练策略，在多个数据集上取得了优异性能。实验结果验证了预训练模型在文本分类任务中的有效性。

未来工作可以从以下几个方向展开：

1. 探索更高效的模型压缩技术，降低部署成本
2. 研究少样本学习方法，减少对标注数据的依赖
3. 提升模型的可解释性，增强用户信任
4. 扩展到多语言和跨语言场景

通过持续优化和改进，我们期望能够开发出更加实用和高效的文本分类系统，为实际应用提供支持。