基于Spark的TextCNN文本分类模型研究

徐小涵

摘　要：随着互联网的普及和信息技术的快速发展，文本数据呈爆炸式增长，文本分类技术在信息检索、舆情分析、智能客服等领域扮演着越来越重要的角色。传统的文本分类方法，如朴素贝叶斯、支持向量机等，需要人工设计特征，效率低且难以捕捉文本的深层语义信息。近年来，深度学习技术在自然语言处理领域取得了突破性进展，为文本分类任务提供了新的思路。本文研究了基于Spark的TextCNN文本分类模型，该模型利用Spark分布式计算平台的高效性，能够处理大规模文本数据。TextCNN模型通过词嵌入层将文本转换为词向量，并利用卷积层和池化层提取文本的局部特征，最后通过全连接层进行分类。本文在GLUE基准数据集上的SST-2、QQP、QNLI三个子集上进行了实验，结果表明，TextCNN模型在三个数据集上均取得了较好的分类效果。具体而言，在SST-2数据集上，模型的准确率达到0.8845，F1分数达到0.8843；在QQP数据集上，模型的准确率、精确率、召回率和F1分数均稳定在0.84以上，特别是F1分数达到了0.8482；在QNLI数据集上，模型的准确率、精确率、召回率和F1分数均约为0.81。研究结果表明，基于Spark的TextCNN文本分类模型能够有效地处理大规模文本数据，并在多个文本分类任务上取得了良好的性能，为文本分类技术的应用提供了新的思路。

关键词：文本分类；Spark；TextCNN

1 引言

随着信息技术的飞速发展，互联网上的文本数据呈爆炸式增长，文本分类作为自然语言处理领域的重要研究方向，在信息检索、舆情分析、智能客服等多个领域扮演着越来越关键的角色。然而，传统的文本分类方法，如逻辑回归[1]、决策树[2]、支持向量机[3]等，往往需要依赖人工设计的特征，不仅效率低下，而且难以捕捉文本数据中复杂的语义信息，限制了其在处理大规模、多样化文本数据时的应用效果。因此，如何利用先进的自然语言处理技术对海量文本数据进行高效、准确的分类，成为亟待解决的技术难题。深度学习技术在自然语言处理领域取得了突破性进展，BERT[4]、RoBERTa[5]等词向量模型的应用。作为CNN在文本分类领域的经典应用，TextCNN模型以其简单的结构和高效的训练性能，在新闻分类、情感分析等多个应用场景中展现出卓越性能。然而，现有研究主要聚焦于通用领域，针对特定领域文本分类的优化研究仍显不足。此外，文本数据中的噪声干扰、类别不平衡及长距离依赖关系等问题尚未得到有效解决，限制了模型在实际应用场景中的表现。因此，本文针对文本数据量大，特定领域文本分类的优化研究不足的现状构建基于Spark的TextCNN的文本分类模型，并通过多组实验进行模型优化与性能验证。

2 相关工作

近年来，文本分类作为自然语言处理重要研究方向备受关注。自20世纪90年代起，机器学习技术快速发展，一系列经典算法应用于文本分类任务。早期有逻辑回归、决策树、朴素贝叶斯等方法，后续改进的支持向量机、随机森林、K近邻等算法进一步提升了分类性能。但传统机器学习方法构建分类器前需复杂人工特征抽取，限制了其发展。

深度学习算法在特征抽取方面优势显著。Mikolov[6]等提出Word2vec方法，能将单词表示为空间词向量用于下游任务，还建立了基于RNN的语言模型，在文本分类中效果良好。Yoon[7]等提出TextCNN模型，在多个公开数据集上实验，显著提高了分类准确率，证实了词向量无监督预训练的重要性及浅层神经网络在文本分类中的精确性。Liu[8]等针对RNN模型在高并发场景性能深入研究，用多任务学习框架简化特征工程。Joulin[9]等提出FastText分类模型，在保证较高准确率同时提升训练效率、节约成本。Johnson[10]等提出的DPCNN模型可有效表示文本长距离关联，在多个基准数据集上性能优于现有最佳模型。

当前国内在警情、舆情文本分类领域处于初步探索期，王世航[11]等提出文本深度聚类方法对舆情文本聚类。钟慧澜[12]探讨深度学习技术在舆情监控和警情数据分析中的应用。陈可嘉[13]等提出BERT-sPTT模型，通过融合BERT的语义理解能力与PVT的金字塔结构特征提取能力，优化文本特征提取过程。李文博[14]等提出AATC模型，基于注意力自适应迁移机制解决零样本跨语言文本分类问题。

3 模型介绍

3.1 Spark架构

Spark 是一个开源分布式计算平台，旨在处理大规模数据任务，功能多样，包括批

量处理大数据、实时数据流处理、机器学习及图形处理等。它基于 Scala 语言开发，兼容 Scala、Java、Python 等多种编程语言，提供丰富 API，继承并优化了 Hadoop 特点，通过内存计算模型、弹性分布式数据集（RDD）及有向无环图（DAG）提高计算效率和优化任务调度，执行大量迭代应用时比 Hadoop 的 MapReduce 快上百倍，通过 LRU 缓存策略优化内存使用，适用于广泛的数据分析和处理场景。

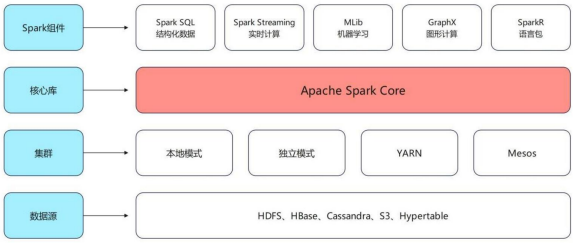

图1 Spark生态模块图

图1展示了Spark的多个功能模块。Spark Core 包含核心基础功能，提供 RDD 的 API 和并行计算支持。Spark SQL 允许用 SQL 语句直接操作分析大规模数据集。Spark Streaming 是实时流计算引擎，能接收多种源数据并处理。MLib 是机器学习库，包含众多算法和工具。GraphX 是图形并行计算库，扩展了 Spark RDD，引入图抽象并包含一系列图算法。

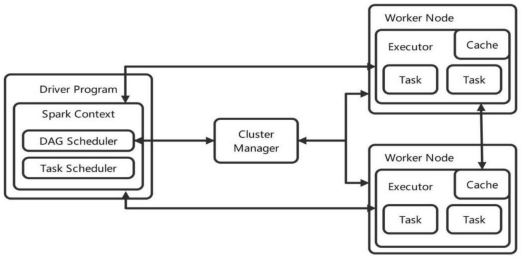

图2 Spark运行架构

图2展示了Spark运行架构。Spark 运行架构采用标准主从（Master-Slave）结构，核心是计算引擎，有 Driver（主节点）和 Executor（从节点）两个关键角色。Driver 节点是应用程序入口点，负责解析代码、划分任务阶段、安排调度任务、管理任务依赖关系并转换为执行计划；Cluster Manager 节点负责在集群中为应用程序分配资源，可以是 YARN、Standalone 或 Mesos 等；Executor 节点在工作节点上执行任务，运行在独立 JVM 进程中，分配一定资源给应用程序，负责接收和执行任务。这种架构使 Spark 能高效处理大规模数据，且具有良好的可扩展性和容错性。

3.2 TextCNN模型

TextCNN（文本卷积神经网络）是Yoon Kim于2014年提出的专门用于文本分类等自然语言处理任务的深度学习模型。它特别适合处理文本数据，能自动学习不同大小的n-gram特征，具有较强表示能力。

TextCNN的工作原理是利用卷积层从文本中提取局部特征，如特定单词组合或短语，再用池化操作汇聚特征，最终输出固定长度向量用于分类。其主要流程如下：词嵌入层将每个词转换为固定维度的词向量，常用预训练或训练得到的词嵌入；卷积层使用多个卷积核对输入文本进行卷积操作，卷积核大小决定捕捉的n-gram特征范围；池化层通常用最大池化从卷积结果中提取最重要特征，每个卷积核对应一个池化结果，得到固定长度特征向量；全连接层将池化层输出扁平化后送入，通过激活函数得到分类结果。

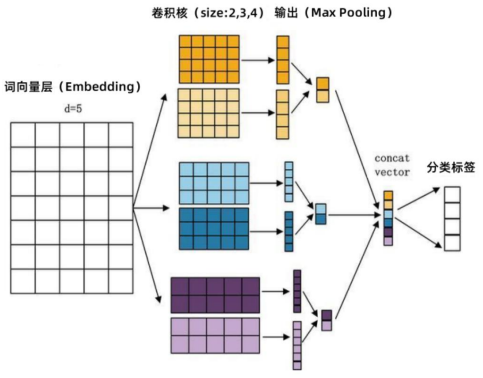

图3 TextCNN架构

TextCNN架构如图3所示。输入层输入句子，为大小为(N, T)的矩阵，N是句子单词数，T是每个单词词向量维度。卷积层对于每个卷积核大小$k \times T$，卷及操作为

$$\mathrm{Conv}(X, k) = \mathrm{ReLU}(W_k * X + b_k) \quad (1)$$

其中W_k是卷积核，*表示卷积操作，b_k是偏置项。卷积层用多个大小不同卷积核与输入词嵌入矩阵卷积，生成多个特征图。池化层对每个特征图最大池化，即

$$\mathrm{MaxPool}(h_k) = \max(h_k) \quad (2)$$

其中h_k是特征图，得到最显著特征值。全连接层拼接所有池化后特征并分类，即

$$\mathrm{Output} = \mathrm{Softmax}(W_{fc}h + b_{fc}) \quad (3)$$

其中W_{fc}和b_{fc}是全连接层的权重和偏置项。

4 实验分析

4.1 数据集

本实验采用GLUE基准数据集，并从中选取了SST-2、QQP、QNLI三个子集进行模型评估。

表1 数据集信息

数据集	训练集样本数	验证集样本数	测试集样本数
SST-2	67,350	873	1,821
QQP	36,843	4,054	2,876
QNLI	104,743	5,463	5,461

SST-2数据集是斯坦福情感树库的一个二分类子集，专注于情感分析任务，将电影评论划分为正面或负面情感。该数据集包含训练集67,350个样本，验证集873个样本，以及测试集1,821个样本。QQP数据集来源于Quora问答平台，用于判断两个问题是否语义等价，是一个二分类任务。它包含训练集36,843个样本，验证集4,054个样本，测试集2,876个样本。QNLI数据集是问答自然语言推断数据集，旨在判断一个问题与一个句子之间是否存在蕴含关系，是一个二分类任务。这些数据集在自然语言处理领域被广泛用于评估和比较不同模型的性能。

4.2 实验评估标准

本研究的模型性能通过准确率、精确率、召回率、F1分数、AUC及训练时间进行综合评估。

准确率（Accuracy）定义如式（4），其衡量总体预测正确性。

$$\mathrm{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

精确率（Precision）定义如式（5），其关注预测为正例的准确性。

$$\mathrm{Precision} = \frac{TP}{TP+FP} \quad (5)$$

召回率（Recall）定义如式（6），其衡量实际正例的检出率。

$$\mathrm{Precision} = \frac{TP}{TP+FN} \quad (6)$$

F1分数（F1 Score）定义如式（7）。

$$F1 = \frac{2\times \mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}} \quad (7)$$

AUC评估模型区分正负例的能力。训练时间用于衡量模型的计算效率。

4.3 实验环境及超参数

实验采用的具体的软硬件参数如表2。

表2 数据集信息

实验环境	配置参数
GPU	NVIDIA GeForce RTX 2050
操作平台	Ubuntu 24.04
软件	Python
主要依赖库	PyTorch、PySpark

超参数对模型的优劣是至关重要的，经过多次调试，选出表3中超参数值作为实验最终模型所采用的超参数。

表3 超参数设置

参数	值
嵌入维度	100
卷积核大小	[3, 4, 5]
卷积核数量	100
丢弃率	0.5
批处理大小	64
训练轮数	100
优化器	Adam

4.4 实验结果

如表3所示实验结果，模型在SST-2数据集上取得了最为优异的性能，准确率高达0.8845，F1分数同样达到了0.8843，这表明模型在该情感分析任务上能够准确且平衡地识别正面和负面情感。其精确率（0.8857）略高于召回率（0.8485），意味着模型在预测为正例时具有较高的置信度。SST-2的训练时间相对最短，仅需95.42秒，显示出模型在处理此类任务时的高效性。在QQP数据集上，模型表现同样出色，准确率、精确率、召回率和F1分数均稳定在0.84以上，特别是F1分数达到了0.8482，体现了模型在判断问题对语义等价性方面的良好平衡能力。最后，在QNLI数据集上，模型的准确率、精确率、召回率和F1分数均约为0.81，这反映出模型在自然语言推断任务上有很强的推理能力。总体来看，模型在三个不同任务的数据集上均展现了较强的适应性。

进一步直观呈现不同模块对模型性能的影响，如图4所示，该模型在三个不同的数据集上均表现出较高的性能。

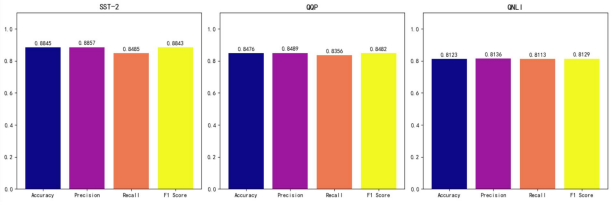

图4 TextCNN架构

5 结论

本文研究了基于Spark的TextCNN文本分类模型，并通过实验验证了其在处理大规模文本数据时的有效性和优越性。

实验结果表明，TextCNN模型在GLUE基准数据集的三个子集（SST-2、QQP、QNLI）上均取得了良好的分类效果，准确率和F1分数均达到较高水平。TextCNN模型能够有效地捕捉文本的局部特征，并通过卷积和池化操作提取关键信息，从而在文本分类任务中展现出强大的性能。此外，Spark分布式

表3 模型训练结果

数据集	Accuracy	Precision	Recall	F1 Score	Training（s）
SST-2	0.8845	0.8857	0.8485	0.8843	95.42
QQP	0.8476	0.8489	0.8356	0.8482	156.78
QNLI	0.8123	0.8136	0.8113	0.8129	134.56

计算平台的使用，使得模型能够高效地处理大规模文本数据，提高了模型的训练效率和可扩展性。

总而言之，基于Spark的TextCNN文本分类模型为处理大规模文本数据提供了一种高效且有效的解决方案。未来，可以进一步探索模型的优化方向，并将其应用于更广泛的领域，以推动文本分类技术的发展和应用。

参考文献

[1] MENARD S W. Logistic regression: from introductory to advanced concepts and applications [M]. Sage, 2010.

[2] QUINLAN J R. Induction of decision trees [J]. Machine learning, 1986, 1: 81-106.

[3] CRISTIANINI N, SHAWE-TAYLOR J. An introduction to support vector machines and other kernel-based learning methods [M]. Cambridge, UK: Cambridge University Press, 2000.

[4] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding [C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.

[5] LIU Y, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pretraining approach [J]. arXiv preprint arXiv:1907.11692, 2019.

[6] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality [J]. Advances in neural information processing systems，2013：1-9.

[7] YOON K. Convolutional neural networks for sentence clas‐ sification[C] Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014：1746-1751.

[8] LIU P F, QIU X P, HUANG X J. Adversarial multi-task learning for text classification[C] The 55th Annual Meeting of the Association for Computational Linguistics (ACL). 2017：1-10.

[9] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of tricks for efficient text classification[C] Proceedings of the 34th International Conference on Machine Learning (ICML). 2017：1398-1407.

[10] JOHNSON R, ZHANG T. Deep pyramid convolutional neural networks for text categorization[C] Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017：562-570.

[11] 王世航，汤艳君，薛秋爽 . 基于文本深度聚类的意见领袖识别模型研究[J]. 中国人民警察大学学报，2024，40(4)：31-36.

[12] 钟慧澜. 技术与治理互嵌：数字时代社区警务智慧化变革机制研究[J]. 中国人民警察大学学报，2024，40(4)：22-30.

[13] 陈可嘉,王朕卿,周修考.基于BERT-sPTT的文本分类模型[J/OL].计算机工程与应用,1-18[2025-07-28].

[14] 李文博,高盛祥,张勇丙.基于注意力自适应迁移的零样本跨语言文本分类方法[J/OL].昆明理工大学学报(自然科学版),1-13[2025-07-28]