文章编号: 1006-2475(2017) 04-0078-04

基于卷积神经网络的互联网短文本分类方法

郭东亮,刘小明,郑秋生

(中原工学院计算机学院,河南 郑州 450007)

摘要: 互联网短文本的分类是自然语言处理的一个研究热点。本文提出一种基于卷积神经网络(Convolutional Neural Networks,CNNs) 互联网短文本分类方法。首先通过 Word2vec 的 Skip-gram 模型获得短文特征,接着送入 CNNs 中进一步提取高层次特征,最后通过 K-max 池化操作后放入 Softmax 分类器得出分类模型。在实验中,该方法和机器学习方法以及 DBN 方法相比,结果表明本文方法不仅解决了文本向量的维数灾难和局部最优解问题,而且有效地提高了互联网短文本两级分类准确率,证实了基于 CNNs 的互联网短文本分类的有效性。

关键词: 卷积神经网络; 短文本分类; 深度学习; 机器学习

中图分类号: TP391 **文献标识码:** A **doi:** 10.3969/j.issn.1006-2475.2017.04.016

Internet Short-text Classification Method Based on CNNs

GUO Dong-liang, LIU Xiao-ming, ZHENG Qiu-sheng

(School of Computer Science, Zhongyuan University of Technology, Zhengzhou 450007, China)

Abstract: The Internet short-text classification is a hot research topic in natural language processing. This paper presents a short text classification method based on deep learning's convolutional neural networks. First short-text features are achieved by the Skip-gram model of Word2vec, then it is sent into the CNNs to extract high-level features, after the K-max pooling, it is put into the Softmax classifier to get a classification model. In the Internet short-text classification experiments, compared to machine learning and DBN's method, the results show that the proposed method not only solves the problems of the curse of dimensionality of text vector and the local optimal solution, but also effectively improves the accuracy of Internet short-text classification, and confirms the validity of the Internet short-text classification method based on CNNs.

Key words: CNNs; short-text classification; deep learning; machine learning

0 引 言

随着 Internet 的大规模普及和用户数量的进一步增加,互联网上的各种短文本正在成爆炸式地增长。互联网短文本指那些长度较短的文本形式,一般不超过 300 字,例如 BBS、博客、新闻评论等[1]。并且这种半结构或无结构化互联网文本信息具有稀疏性、实时性、不规范性、流行语不断出现等特征。互联网短文本分类作为信息处理关键技术之一,已经广泛应用于信息检索、知识挖掘和信息监管等领域[2]。为了实现大数量互联网短文本的自动快速分类,众多研究者对该问题进行了相关研究,主要包括: 支持向量机(Support Vector Machine,SVM)[3]、朴素贝叶斯分类法(Naïve Bayesian Classifier,NBC)[4]、K-最近邻法(K-Nearest Neighbor,KNN)[5]、决策树法(Decision Tree,DT)[6]等。

上述方法都是浅层的机器学习方法,在文本处理过程中没有考虑词与词之间的关系,本文提出一种基于卷积神经网络的互联网文本分类的方法,通过卷积神经网络层处理互联网的文本数据,加强文本数据中词与词之间、文本和文本之间关系,与传统的互联网短文本分类相比,显著提高了文本分类准确率。

1 相关研究

近年来,针对互联网短文本领域分类分析的研究已在国内外广泛展开。在国外,由于科技条件的进步

收稿日期: 2016-08-23

基金项目: 河南省科技攻关项目(132102310284) ; 河南省教育厅科学技术研究重点项目(14A520015)

作者简介: 郭东亮(1991-) ,男,河南林州人,中原工学院计算机学院硕士研究生,研究方向: 自然语言处理; 刘小明(1979-) ,男,河南许昌人,讲师,博士,研究方向: 机器学习,自然语言处理; 郑秋生(1965-) ,男,河南郑州人,教授,硕士,研究方向: 信息安全,数据资源管理。

以及浅层的机器学习技术比较成熟,因此最近深度学习的方法较多被用来进行自然语言处理。Mikolov[7]介绍了一种连续的 Skip-gram 模型,它是一种高效的学习方法,可以从大量的文本数据中训练得到高质量的词组向量。这些向量在词语的语义和句义方面更具代表性,在词语相似度方面具有非常好的效果。Yoon Kim[8]使用卷积神经网络对英文句子进行分类,使用微调的 CNNs 网络参数和静态的文本向量,取得了比较好的结果。特别是 Kalchbrenner[9]等人在 2014 年介绍了一种可以进行英文句子建模的动态卷积神经网络方法。通过 K-max 池化操作来获取全文的特征向量,从而摆脱决策树方法的依赖。在英文的问题分类的评测中取得了比较好的文本分类效果。

在国内,研究者大多通过改变文本输入的特征和结合其他算法改进已有的分类方法用来提高文本分类准确率,或者改进文本处理的时间,降低文本分类时间,为大数据处理提高效率。2013 年崔建明[10]等人采用 SVM 文本分类技术,把优化的粒子群算法引入 SVM 分类算法中,进行优化文本分类器的参数,将分类器的准确率作为粒子群算法适应度函数通过粒子移动操作找出最佳参数并用 SVM 算法进行分类,提高文本分类准确率。李玉鑑[11]等人提取每一类样本向量组的特征子空间并通过映射将子空间变换为高维空间中的点,然后把最近邻子空间搜索转化为最近邻搜索完成分类过程,有效提高文本分类的性能,具有较高的准确率。同济大学的陈翠平[12]利用深度置信网络(Deep Belief Network,DBN)从互联网文本高维的原始特征中抽取低维度高度可区分的低维特征,不仅能够考虑到英文文档足够的信息量,而且能够快速地训练,并且实验结果也表明利用深度信念网络,实现文本分类的性能很好。

尽管上述方法一定程度提高了文本分类准确率,但浅层学习方法存在以下问题: 1) 传统的方法容易出现局部最优问题,忽略了词语与词语之间关系,句子与句子之间关系; 2) 采用 CNNs 进行分类,大都是基于英文句子和问题层面,与中文短文本抽象的特征相比相对较小; 3) 采用深度学习的 DBN 文本分类方法,使用 TF-IDF 方法抽取文本特征向量,不能获得较高抽象的词组向量。

为解决以上问题,本文提出一种基于 CNNs 的互联网短文本分类方法。

2　基于 CNNs 的互联网短文本分类模型

基于 CNNs 的互联网文本分类模型主要分为 4 大层面,包括: 数据预处理层、卷积神经网络层、K-max 池化层、Softmax 分类层。它们之间是底层为上层提供服务的关系。其流程模型如图 1 所示。

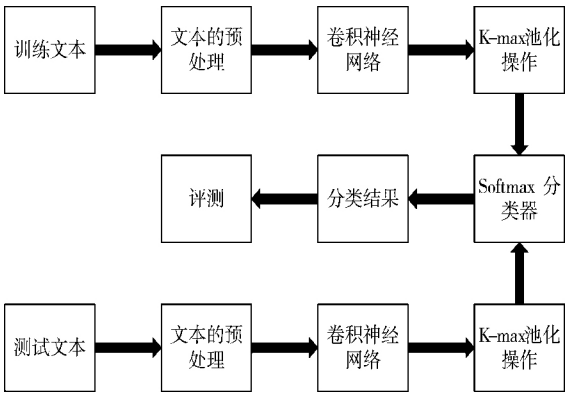

图 1　流程模型图

2.1　数据的预处理层

从互联网取得的数据是不能直接放进 CNNs 进行处理的,需要对其进行预处理。处理的过程主要分为 4 部分:

1) 网络文本数据不能直接使用,需要进行分词处理,本文使用中国科学院的开源代码汉语语法分析系统 2016 年的 ICTCLAS(Institute of Computing Technology,Chinese Lexical Analysis System)进行中文的分词处理[13]。

2) 去除停用词可以减少文本冗余使文本分类更加准确[14]。本文使用的停用词是 ICTCLAS 的 Data 目录下的 stopwords.txt,并结合网上发布的一些常用停用词对其进行了扩充。常见的停用词有“打开天窗说亮话”、“《”、“1”等。

3) 文本的向量化处理。去除停用词后的文本,使用 Word2vec 工具,利用 Skip-gram 模型给出每个词语有 100 维度的特征向量。

4) 文本向量的矩阵化处理。CNNs 处理的文本数据形式一般为矩阵形式,本文使用 ND4J 的向量模块进行文本向量矩阵的构造。为了方便程序的处理,以文本中最大的词文本构造定长的特征矩阵,其他向量不够进行补零操作。

2.2　卷积神经网络层

文本向量的卷积是文本特征向量进行高层次特征提取的过程,实验效果与该层的卷积窗口大小、学习率、卷积步长以及正则化系数有关。由于短文本通过数据预处理层以后,大概每 5 或 6 个词语构成一个句子,因而将卷积窗口设置成 5,卷积的步长设置为 1。经过反复的实验,窗口参数是 5 或 6 的时候,文本处理的效果比较好。学习速率设置的大小,需要根据具体文本进行调整。具体的文本卷积过程公式如下:

$$W=\begin{bmatrix} X_{11} & X_{12} & X_{13} & \cdots & X_{1N} \\ X_{21} & X_{22} & X_{23} & \cdots & X_{2N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ X_{K1} & X_{K2} & X_{K3} & \cdots & X_{KN} \end{bmatrix} \tag{1}$$

$$G=\begin{bmatrix} A_1 & 0 & 0 & & 0 \\ A_2 & A_1 & 0 & & 0 \\ \vdots & A_2 & A_1 & & 0 \\ A_{win} & \vdots & A_2 & \ddots & 0 \\ 0 & A_{win} & \vdots & \ddots & A_1 \\ \vdots & \vdots & 0 & & A_2 \\ 0 & 0 & \vdots & & \vdots \\ 0 & 0 & 0 & & A_{win} \end{bmatrix} \tag{2}$$

$$H=W^{*}\ G=\begin{bmatrix} H_{11} & H_{12} & H_{13} & & H_{1Q} \\ H_{21} & H_{22} & H_{23} & & H_{2Q} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ H_{K1} & H_{K2} & H_{K3} & & H_{KQ} \end{bmatrix} \tag{3}$$

矩阵 W 为输入的文本向量，其中 X_{ij} 代表第 i 个文本的第 j 个词语的特征向量。矩阵 G 为卷积核函数矩阵，该矩阵大小与输入的文本数 K 和卷积的窗口 win 有关。矩阵 H 为输入矩阵 W 与卷积核函数矩阵 G 卷积的结果。其中，H_{ij} 代表第 i 个文本通过 j 次卷积得到的向量。

2.3 K-max 池化层

最大池化操作为一个非线性的子抽样函数，进一步减少了文本的向量维度，用来生成一个最大值的子抽样的矩阵。本文通过调整 K-max 池化窗口参数，生成的矩阵的每一行是每篇文章进行一次最大池化抽样的结果。相比平均池化操作，K-max 池化操作减少文本矩阵补零操作噪声影响，抽样输出的结果更能够代表文本，实验得出的效果更好。

2.4 Softmax 分类器层

最后一层采用全连接的方式，通过 K-max 池化处理后的文本特征向量，送入 Softmax 分类器，用来预测类别概率。其过程如下：

池化层得到的 m 个训练集数据，其形式如下 $\{(x^{(1)},y^{(1)})(x^{(2)},y^{(2)})\cdots(x^{(m)},y^{(m)})\}$，其中输入特征 $x^{(i)}$ 代表文本特征向量，其维度与池化层输出节点有关，$y^{(i)}$ 代表文本类别。对于给定测试集文本向量 x，可以通过假设函数 $h_{\emptyset}$ 给出其属于哪一个类别概率 p。由于本文进行文本二分类，假设函数公式如下：

$$h_{\emptyset}=\frac{1}{1+\exp(-\emptyset^{T}x)} \tag{4}$$

其中，$\emptyset$ 代表模型参数，通过对 $\emptyset$ 的训练可以找到最小代价函数 $J(\emptyset)$，其公式表达式如下：

$$J(\emptyset)^{1}=\sum_{i=1}^{m}y^{(i)}\log h_{\emptyset}(x^{(i)}) \tag{5}$$

$$J(\emptyset)^{2}=\sum_{i=1}^{m}(1-y^{(i)})\log(1-h_{\emptyset}(x^{(i)})) \tag{6}$$

$$J(\emptyset)=-\frac{1}{m}[J(\emptyset)^{1}+J(\emptyset)^{2}] \tag{7}$$

3 实验设计

为了验证基于 CNNs 的互联网文本分类模型的效果，本文在互联网文本的领域分类任务中进行实验，以 SVM、KNN 以及 DBN 作为本文的基线方法，与其对比，其两级分类的类别目录划归标准可参考文献[1]。

3.1 数据来源与实验环境

实验语料使用谭松波中文文本分类的语料，从中选取符合短文本分类需求的游戏、电影、足球、篮球、理财、证券、健康、教育等 8 个类别的 6853 篇文本。本实验平台基于 Windows 7 下的 Eclipse，JDK 使用 1.8 版本，加入了 ND4J 和 DL4J 等类库实现 CNNs 模型。

3.2 实验对比及评测标准

本实验使用 SVM 和 KNN 文本分类方法作为本文方法的对比。SVM 的文本分类方法使用台湾大学林智仁教授开发的 LIBSVM 工具包，该工具包含多种核函数，满足本文实验的要求。SVM 分类使用的传统 TF-IDF 值[15]为文本的特征向量，构造对应词库标签，核函数选择线性核函数。KNN 文本分类方面，为了使测试数据实验的准确率较高，本文设置必须有 3 个以上的相同的词的训练文本，才进行测试文本的相似性比较，找出 8 个相似距离最近训练文本，查看它们的类别哪个最多，判定测试文本为哪个类别[16]。在深度学习 DBN 文本分类方法[12]中，根据实际中文语料文本情况以及领域词的规模，确定第一次输入节点数量，使用 7 层 BP 神经网络，每层节点数量折半，迭代 1500 次。

互联网文本分类的评测标准采用文本分类比较常用的准确率为指标，对上述实验的分类结果进行评测[17]，其公式如下：

准确率 = 分类正确的文档/文档的总数

3.3 实验结果与分析

表 1 一级目录分类准确率(%)

类别	SVM	KNN	DBN	CNNs
娱乐	96.7	77.6	97.5	99
体育	96.1	97.1	95.8	98.6
财经	98.3	87.5	98	97.14
生活	97.3	80.4	97.5	98.5

由表 1 可知采用不同的分类方法对于互联网文本的领域一级分类效果影响非常明显，使用基于 CNNs 的分类方法准确率最高达到 99%，而采用 SVM

和 KNN 的分类准确率最高仅为 98.3%，深度学习的 DBN 分类方法准确率最高为 98%，并且可以从表中明显看出，CNNs 进行分类的效果非常明显，达到了预期的效果。

表 2　二级目录分类准确率(%)

二级类别		SVM	KNN	DBN	CNNs
娱乐	游戏	69.1	52.2	75.4	85.2
	电影	87	98.7	86.2	85.3
体育	足球	84.2	62.32	87.4	100
	篮球	90	93.1	93	95.7
财经	理财	64.3	50	68	83.6
	证券	97.5	96.2	78	90.3
生活	健康	96	93.6	90	88.6
	教育	93.3	73.73	95	98

通过表 2 很容易看出，基于深度学习的 DBN 分类算法和传统的 SVM 和 KNN 算法相比，深度学习的 DBN 分类方法准确率比较稳定，准确率相对较高。而 KNN 和 SVM 二级目录文本分类方法则存在分类准确率一高一低情况，文本类别错判情况较高，特别是在体育和财经等一级目录类别下二级分类，其 KNN 最低文本分类的准确率为 50%，SVM 最低文本分类准确率为 64.3%。本文的 CNNs 文本分类方法和 DBN 的文本分类方法相比，其最低的文本分类准确率为 83.6%，而 DBN 最低的文本分类准确率为 68%。CNNs 的二级分类的平均准确率比 DBN 的二级分类平均准确率效率要高。

取得以上效果的原因，可以归纳为以下 3 点: 1) 使用 Word2vec 工具，生成的向量比 TF-IDF 生成的向量能得到更高质量的词组特征; 2) 通过 CNNs 处理后的文本特征能更好表示文本的特征; 3) KNN 只是单纯从文本之间相似度的距离出发，没有考虑词与词之间的关系。

通过对表 1 和表 2 结果分析，在互联网短文本领域分类，基于 CNNs 文本分类方法相比 SVM、KNN、以及 DBN 的互联网短文本分类方法，其文本分类效果更加显著。

4　结束语

本文利用深度学习 CNNs 方法可以解决文本向量的维数灾难、局部最优解以及过学习问题，通过组合低层特征形成更加抽象的高层表示。将 SVM、KNN 以及 DBN 的互联网短文本分类方法与深度学习神经网络的 CNNs 文本分类方法进行对比，弥补上述文本分类方法的不足，提高了文本的准确率。

在今后研究中，由于本文的方法还是在单机上运行，对于文本处理的时间还比较长，对于大规模数据的分类实用性比较低。因此，在分布式平台上进行 CNNs 的互联网文本分类将是笔者研究的重点，用以提高文本分类的时间效率。

参考文献:

[1] 江斌. 微博自动分类方法研究及应用 [D]. 哈尔滨: 哈尔滨工业大学, 2012.

[2] 张志飞，苗夺谦，高灿. 基于 LDA 主题模型的短文本分类方法 [J]. 计算机应用, 2013,33(6):1587-1590.

[3] 张爱丽，刘广利，刘长宇. 基于 SVM 的多类文本分类研究 [J]. 情报杂志, 2004,23(9):6-7.

[4] 郭泗辉，樊兴华. 一种改进的贝叶斯网络短文本分类算法 [J]. 广西师范大学学报(自然科学版), 2010,28(3):140-143.

[5] 张宁，贾自艳，史忠植. 使用 KNN 算法的文本分类 [J]. 计算机工程, 2005,31(8):171-172.

[6] 黄华. 基于决策树与 SVM 融合学习的科技文献分类方法研究 [D]. 郑州: 河南工业大学, 2011.

[7] Mikolov T, Chen Kai, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space [EB/OL]. https://arxiv.org/abs/1301.3781, 2013-01-16.

[8] Kim Y. Convolutional neural networks for sentence classification [C] // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014: 1746-1751.

[9] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences [EB/OL]. https://arxiv.org/abs/1404.2188, 2014-04-08.

[10] 崔建明，刘建明，廖周宇. 基于 SVM 算法的文本分类技术研究 [J]. 计算机仿真, 2013,30(2):299-302

[11] 李玉鑑，王影，冷强奎. 基于最近邻子空间搜索的两类文本分类方法 [J]. 计算机工程与科学, 2015,37(1):168-172.

[12] 陈翠平. 基于深度信念网络的文本分类算法 [J]. 计算机系统应用, 2015,24(2):121-126.

[13] 冯永，李华，钟将，等. 基于自适应中文分词和近似 SVM 的文本分类算法 [J]. 计算机科学, 2010,37(1):251-254.

[14] 顾益军，樊孝忠，王建华，等. 中文停用词表的自动选取 [J]. 北京理工大学学报, 2005,25(4):337-340.

[15] 施聪莺，徐朝军，杨晓江. TFIDF 算法研究综述 [J]. 计算机应用, 2009,29(S1):167-170.

[16] 耿丽娟，李星毅. 用于大数据分类的 KNN 算法研究 [J] 计算机应用研究, 2014,31(5):1343-1344.

[17] 宗成庆. 统计自然语言处理 [M]. 北京: 清华大学出版社, 2008:352-353.