2017 年 8 月 第 8 期
Vol. 38 No. 8 2017

基于吸引子传播聚类的改进双通道 CNN 短文本分类算法

王 儒[1,2],刘培玉[1,2],王培培[1,2]

[1](山东师范大学 信息科学与工程学院,济南 250358)
[2](山东省分布式计算机软件新技术重点实验室,济南 250358)
E-mail: 673202668@qq.com

摘 要: 传统的文本分类方法在处理短文本分类任务时遇到了很大的困难,针对短文本分类任务上的数据稀疏等难点,本文尝试在短文本特征输入和卷积神经网络结构上进行改进. 在特征表示 Word embedding 训练时采取 non-static 和 static 两种方式,将训练好的 Word embedding 进行聚类处理,聚类得到的 Word embedding 库作为模型输入的词典库;提出一种改进的双通道卷积神经网络结构,网络通过双通道获取更多的局部敏感信息增加特征数目,然后经过连续的池化实现特征抽取. 经实验验证,提出的语义聚类处理和改进的网络模型与传统的机器学习方法相比,在短文本分类任务的准确率上有显著的提升.

关 键 词: 词向量聚类;短文本;CNN;分类
中图分类号: TP311 **文献标识码:** A **文 章 编 号:** 1000-1220(2017)08-1730-05

Improved Two Channel CNN Short Text Classification Algorithm Based on Affinity Propagation Clustering

WANG Ru[1,2], LIU Pei-yu[1,2], WANG Pei-pei[1,2]
[1](School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China)
[2](Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Jinan 250358, China)

Abstract: In view of the difficulty of short text classification task, tried to improve on short text feature representation and convolution neural network structure. Above all, Word embedding training is taken in two ways: non-static and static. and the Word embedding is used as a model to input Word embedding clustering. Then an improved CNN structure, the network obtains more local sensitive information through two channels to improve the number of features. Experiments show that the improved semantic clustering approach and improved CNN model have a significant improvement on the accuracy of short text classification tasks compared with traditional machine learning methods.

Key words: word embedding clustering; CNN; short text; classification

1 引 言

随着互联网的迅猛发展以及网络用户数量的飞速增长,社交平台和电商网站等网络应用日渐成熟化,因此海量的数据在源源不断的产生,在海量数据中,数据的种类是多样的,但是文本一直是作为主要交互手段之一. 文本可以按照文本的长度粗略的分为长文本和短文本,根据其的特点短文本可定义为:长度不超过 140 字符的、表述能力丰富、组合相对灵活、文本长度不固定、数据总规模较大的文本类别. 目前短文本的表现形式主要是微博、电商评论、Q&A(Question and Answer)等,长度主要集中在 30-60 字符. 短文本是目前在新生成的文本类别中总量最大、增长速度最快的文本之一. 因此 NLP 任务中的对短文本的研究是一件非常有意义的事情,针对短文本的任务主要是短文本理解、短文本分类等,本文的主要工作是对短文本分类的研究.

传统的基于统计和基于机器学习的文本分类方法在处理长文本分类任务时有着非常好效果,但这些算法在处理短文本任务时效果却非常差,出现这种反差的原因是短文本的特点所导致的. 在处理长文本分类任务时较为经典分类方法有:支持向量机(*SVM*)、朴素贝叶斯(*Naive Bayes*)、逻辑回归、随机深林等方法,这些传统的、经典的方法在处理长文本时采用的文本特征表示方法一般是向量空间模型(*Vector Space Model*),该模型是一个非常成熟且经典的文本表示模型,该模型将文本映射到向量空间中,通过对空间向量的相似度计算来计算文本之间的相似度,这种模型在处理长文本时效果较好,但在处理短文本时效果就非常差. 原因是 *VSM* 的向量相似度计算是基于 *TF-IDF* 即词频和逆向文档频率的计算,可以想象在长文本分类任务中,词频和逆向文档频率能够突显出不同类别文档的差距,但相比之下短文本这类数据总量特别大、每个个体短的数据时缺点显而易见,词频已经区分不开差距,逆向文档频率也毫无区分度,这也使得传统的文本分类方法在处理短文本时有非常大的挑战. 本文通过查阅相关

收稿日期:2016-12-19 收修改稿日期:2017-01-04 基金项目:国家自然科学基金项目(61373148、61502151)资助;山东省自然基金项目(ZR2014FL010)资助;山东省社科规划项目(2012BXWJ01、15CXWJ13、16CFXJ05)资助. 作者简介:王 儒,男,1990 年生,硕士研究生,CCF 会员,研究方向为自然语言处理;刘培玉,男,1960 年生,硕士,教授,博士生导师,CCF 高级会员,研究方向为网络信息安全、自然语言处理;王培培,女,1992 年生,硕士研究生,CCF 会员,研究方向为推荐系统、自然语言处理.

资料总结了主要导致短文本处理困难的特点:

1) 长度短不超过 140 字符,信息单元少;

2) 词语较为开放,词语总量大,重复率低;

3) 词语更新快,新词、怪词出现频繁等.

尽管学者们已经在短文本分类任务上取得了一定的研究成果,但短文本分类任务的难点并没有得到完全解决. 为了提高分类的准确率、降低词向量的稀疏性对模型训练和测试的影响,本文从词向量聚类出发(将相似编码的相近语义词进行聚类,聚类的目的是为了提高模型的训练效率,减少同义词及低频词对训练的影响),将处理过的数据作为模型的输入,然后利用改进的 CNN 结构进行短文本分类.

2　相关工作

近年来 Deep learning 在各个领取上取得了辉煌成果,当深度学习的浪潮推向 NLP 时,Hinton [1] 在论文中创新思维的给出的 *Word embedding* 的词向量方法,该方法的核心思想是将词语采用 *Distributed representation* 将每个 Word 从高维的词语空间映射到低维实数向量(一般在训练中指定 K 维即模型中的超参数),该思想颠覆了传统的向量空间模型,利用低维的向量就将词语的大量的潜在信息表示出来. 随后 Tomas Mikolov [2] 等人采用 *CBOW*(*Continuous Bag-Of-Words*) 和 *Skip-Gram* 两种模型分别完成了对文本集高效的训练成 *Word embedding* 的任务,并公开了开源工具 *word2vec*. 在 *Word embedding* 的模型里将具有相近意义的词用相似的词向量表示:

例如 vec“清华”≈vec“北大”(相似度 =0.75724)

vec“中国”+“首都”=vec“北京”(相似度 =0.77138)

Word embedding 在一定程度上克服了短文本的缺点,由于 *Word embedding* 的日渐成熟,使得其被大量的应用到 NLP 任务中,并帮助 NLP 任务攻坚克难取得了相当优异的科研成果. *Word embedding* 的最初的核心思想 *Distributed representation* 的最早是在 1980 年出现 *Hinton* 提出,经典的原模型是由 Bengio [3] 等人建立,后续语言模型在 Mikolov [4] 等人的工作上完善,其中 *Word2vec* 是 *Google* 在 2013 年开源的高效工具,*Word2vec* 中有两种 *CBOW* 和 *Skip-Gram*.

Mikolov 在 *Word2vec* 说明中对两模型做出了总结: 在处理 *Word vector* 训练任务时,相比 *CBOW* 模型,*Skip-gram* 模型的速度相对比较慢一些,但该模型对低频词语效果更好,相反的 *CBOW* 模型在训练的速度上比较快,并且对高频词的效果会好一些. 本文针对的短文本分类任务数据几乎都是低频词,因此本文在词向量训练模型时选择了 *Skip-gram* 模型. google 代码链接[1].

卷积神经网络(CNN)是当前应用最流行的深度学习结构之一,这于其在图像处理任务中取得了可以称得上辉煌的成果密切相关,CNN 多次取得了国际图像任务挑战赛冠军,甚至在个别任务中刷新了该指标的最好结果. CNN 之所以取得优异的成绩应归因于其的显著优点: 卷积层特征提取,卷积 + 池化可以获取局部的敏感信息; 独特的网络结构使得降维(特征提取)速度极快,结合权值共享使得训练的参数相对较少; 网络结构高效简单适应性强.

CNN 在图像领域的优异成果,使得研究者将其应用到其各个领域,近几年 CNN 应用到 NLP 也如雨后春笋般显露出来,其中不乏一些非常高质量的研究成果. Kalchbrenner [5] 提出了一种新的称为 *DCNN*(*Dynamic Convolutional Neural Network*) 的网络模型,其工作主要的贡献是提出了一种动态池化的思想即在不同的网络层采取不同的 K 值进行 *pooling*,而不是单单只使用固定的 K 值 *pooling*,该方式 *pooling* 的结果就是得到 K 组最大值,相当于更加细致的获取到一个特征序列,通过更多的特征序列来获取更多的局部敏感信息. Kim [6] 提出的模型较 Kal 的比较简单但有着非常好的效果,该结构通过使用双 channel 即输入采用两种输入对比实验即 *static Word vector* 和 *non-static Word vector*,有效的解决了相同语境下反义词编码相似的缺点,在输入层采用双通道的结构计算得到,然后对 进行得到最后特征序列,然后经全连接到 决策层. Kim 的模型简单高效,并在个别分类任务上刷新了最好的记录. Ye Zhang [7] 等人总结卷积神经网络在句子建模上的应用,并提出了新的网络结构,对每个尺度都采用双滤波器进行卷积,该方式多样化的获取了局部信息,增加了特征数目,作者在对前人的工作上进行了大量的对比实验,从各个角度都 CNN 的结构效果进行验证,在实验中论证了该网络结构的有效性.

Peng Wang [8] 等人在前人的基础上对词向量进行语义单元聚类,将聚类完成的语义单元作为网络的输入,网络结构改变为先对原本的句子矩阵进行语义单元的获取,然后对获取到的语义单元特征矩阵进行卷积、池化已及全连接决策层. Wang 等人的工作在针对短文本在将编码相近的语义单元聚类,达到了合并同义词的效果,类似与对训练数据和测试数据进行了预处理,这样处理后的语义单元在输入时更加规则,在一定程度上增加了训练的速度和准确率,作者采用的聚类算法是 Alex Rodriguez [9] 等人 2014 年发表在 *science* 的快速聚类算法,该算法高效适用性强,但在参数设定上比较困难,如局部密度及截断密度的设定,本文的工作和 Peng Wang 等人的工作最为相近. 本文的主要创新: 一是使用两种词编码方式即 *static* 和 *non-static*,然后采取更加准确的聚类方法 AP 聚类算法分别对两类词向量进行聚类; 二是针对短文本特点提出一种新的双通道卷积结构用以获取更加详尽的局部信息. 通过大量的对比实验证明本文提出的算法思想较传统的分类算法在分类效果上有个大幅度的提高,相比前人提出的卷积结构对比分类效果也有了明显的提高.

3　Embedding Vector 聚类

尽管 word embedding 在编码时将词语语境相近的词语利用相近的编码表示,但在处理短文本时,如果单纯使用 word embedding 表示词语并不能够解决短文本稀疏的问题,因此本文利用聚类算法将 word embedding 进行聚类,利用聚类的特性聚拢近义词,表示同一类别的词语被使用相同的标记,同一类别的不同词语使用相似度表示词语直接的连接强度. 通过对 word embedding 聚类处理,降低词语库中同义多词对模型训练的影响.

[1] http://word2vec.googlecode.com/svn/trunk/

利用 Word2vec 工具训练得到的 *Word embedding*. 在训练时,针对短文本低频词较多的语料,在模型选择上选取了 *Skip-Gram* 模型,理由是该模型在训练数据的特点针对低频词语效果更好. *Word embedding* 的维度设定 $k=300$. 在训练过程中采用 *non-static* 和 *static* 两种方式进行词向量的训练,*non-static* 方式是在训练词向量时加入了分类任务的训练语料.

实验数据为复旦大学自然语言处理中文分类的公开数据集和搜狐新闻数据(SogouCS),在训练语料时将数据集文本加入训练语料,通过加入分类任务的语料可以丰富词向量训练词语的语境,训练出适合处理该样本分类任务高质量的 *Word embedding* 词向量. *static* 方式是不加分类语料只对语料进行训练,*static* 方式可以体现词语在原始语料中的信息. 因此同时采用两种方式的编码来获取更加详细的词语信息,训练的语料库选择的是语料 575479 共包含 57 万词汇(非重复的词典单词个数 57 万)共超 6 亿量. 语料连接[2].

在 Word2vec 模型中已经对生成的 vector 进行了相似度计算,计算距离使用的是余弦相似度,模型已经计算出了各 vector 的相似度,因此可以将每个 vector 看着是数据点,各个数据点之间的距离定义为相似程度,那么对训练得到的 vector 进行聚类处理得到语义簇,本文对 Word embedding Vector 聚类使用的算法是:吸引子传播 Affinity Propagation (AP)[10] 聚类算法,是 2007 年 Frey 和 Dueck 发表在 science 的聚类算法,其较少的参数设置、聚类准确等优点似的其广泛的应用在各个领域上.

AP 算法[11] 定义:

定义 1.

$S(n\times n)$ 是维度为 $n\times n$ 的相似度矩阵,每个元素表示数据点之间的相似度.

定义 2.

吸引度(*Responsibility*):表示点 k 适合作为数据点的聚类中心的程度.

$$R(i,k)=S(i,k)-\max\{A(i,j)+S(i,j)\}\quad(j\in\{1,2,\cdots,n,\text{但}j\neq k\}) \tag{1}$$

定义 3.

隶属度(*Availability*):表示点选择点 k 作为其聚类中心的适合程度.

$$A(i,k)=\min\{0,R(k,k)+\{\max(0,R(j,k))\}\}\quad(j\in\{1,2,\cdots,N,\text{但}j\neq i\text{ 且 }j\neq k\}) \tag{2}$$

$$R(k,k)=P(k)-\max\{A(k,j)+S(k,j)\}\quad(j\in\{1,2,\cdots,N,\text{但}j\neq k\}) \tag{3}$$

当 $P(k)$ 较大使得 $R(k,k)$ 较大时,$A(k,k)$ 也较大,从而类代表 k 作为最终聚类中心的可能性较大;同样,当越多的 $P(i)$ 较大时 越多的类代表倾向于成为最终的聚类中心.

算法步骤:

Step 1. 初始化 S(n,n),将相似度矩阵转化到相似度二元组的超集集合:

$$D=\{s(v_1,v_1),s(v_1,v_2),s(v_1,v_3),\cdots,s(v_n,v_n)\};$$

Step 2. 导入 word2vec 中各 word 之间相似性;

Step 3. 初始化消息定义 2、定义 3 消息矩阵:

$$R(i,k)=S(i,k)-\max\{A(i,j)+S(i,j)\}$$
$$a(i,j)=0\quad(k\neq j);$$

Step 4. 根据公式(2)、(3)计算消息矩阵;

Step 5. 根据公式(1)更新消息矩阵;

Step 6. 消息矩阵依次叠加 $R+A$,计算出每个样本 i 的候选聚类中心.

Step 7. 重复 Step 4-6 直到每个样本都找到聚类中心至算法收敛.

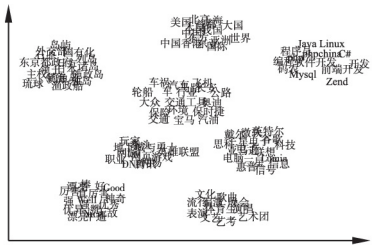

图 1 Word embedding 部分聚类

Fig. 1 Word embedding clustering

聚类结果另存到新的 *Word embedding* 库,同一语义簇的 *Word embedding* 有同一簇的标记和各 *Word embedding* 直接的距离(相似度). *Word embedding* 的部分聚类效果图如图 1 所示. 聚类之后的结果作为卷积神经网络模型的输入.

4 Double-CNN 分类模型

4.1 模型介绍

为了能充分提取局部敏感信息,本文提出了一种新的双通道卷积神经网络结构,网络结构如下页图 2 所示. 首先对短文本进行建模:定义每个短文本表示为:

$T(n\times k)=w_1\otimes w_2\otimes w_3\otimes\cdots\otimes w_n$,其中 $T(n\times k)$ 表示短文本的矩阵表示,n 是文本长度,k 是 *Word embedding* 的维度,w_1 表示文本中第词语,$\otimes$表示词语直接的语义连接符,因此短文本可以表示为 $n\times k$ 的矩阵表示.

模型网络结构如下页图 2 所示.

第一层是 INPUT 层,输入层的输入是本文采取两种方式训练的词向量,在文本矩阵表示时采用 *zero-padding* 扩展边界.

第二层是卷积层,卷积层分别对两种通道的输入采用三种大小(2,3,4)的滤波器进行卷积操作得到 *feature-map*,其特征计算公式为公式(4)所示

$$c_i=\sum_{1:n-h+1}(w\cdot T_{i:i+h-1}+b) \tag{4}$$

其 w 中为卷积核矩阵,$T_{i:i+h-1}$ 表示文本矩阵的第至 $i+h-1$ 行,输出的是特征矩阵. 如公式(5).

$$C=f((n-h+1)\times k+b) \tag{5}$$

其中 f 为激活函数,$f(x)=\max(0,x)$,b 为偏置项.

第三层是池化层(*pooling*),输入是 i-h+1 的特征矩阵,池化层对矩阵的每一行进行 k-max 操作,输出 $1\times(i-h+1)$ 是特征向量.

第四层是池化层,输入是对双 channel 形成的 6 个特征向量,池化层对特征向量进行 k-max 操作,输出是提取的 6 个(组)特征值.

[2] http://www.sogou.com/labs/resource/list-yuliao.php

第五层是全连接层,输入是最终的特征值,将输入按照滤波器的顺序进行组合层特征向量,输出全连接至决策层.

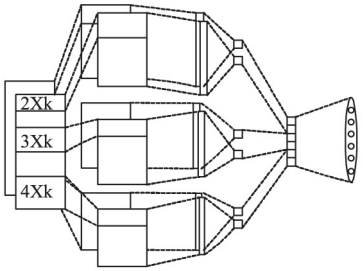

图 2　Double-CNN 网络

Fig. 2　Double-CNN NETWORK

第六层是 softmax 层,输入是全连接的特征向量,输出是判断的类别.

4.2　模型分析

针对短文本特征稀疏的特点,单通道获取到的语义信息比较少,因此并不能完全的表示文本信息,为了获取到更多的语义特征信息,采取了改进的双通道 *CNN*. 改进的双通道 *CNN* 模型通过不同的输入(*non-static*、*static*) 获取到双倍的局部特征,对比单通道 *CNN* 可以获得更多的局部语义信息. 采用两种方式的编码是为了防止两个通道获取到相同局部特征,因此对每个通道采取不同的编码输入(*static*、*non-static*) . 通过实验结果可以直观的看出双通道 *CNN* 对比单通道 *CNN* 在准确率上的提升; 不同编码输入对比单一编码方式在准确率上的提升.

4.3　模型的训练

本文提出的双通道卷积神经网络结构,可以处理长短不同的文本. 卷积核的矩阵采用的大小为(2、3、4) 的三种卷积核通过双通道卷积得到 6 张 $C = f((n-h+1)\times k + b)$ 的 feature map. 为了加快训练速度,在梯度下降学习时选择 mini-batch,其参数学习是在载入指定窗口大小后更新一次权值参数. 本文数据集一有 20000 条样本,因此经过反复实验,mini-batch 设置为 50 时训练时间和效果比较理想. 激活函数选择激活函数: $f(x) = \max(0, x)$. 在防止过拟合策略上选择 l_2 正则化对结构参数约束,l_2 norm constraint = 3. 全连接层采用了 Dropout 技术即在每一次训练学习过程中屏蔽一部分网络, Dropout = 0.5. 在 *Word embedding* 训练时,设置词向量维度 k = 300,静态词向量只对语料进行训练,non-static 词向量在训练时加入本文数据集中的 20000 条样本内容.

5　实验结果与分析

5.1　实验环境

实验环境如表 1 所示.

5.2　数据集

数据集的是文本分类语料库(复旦) 测试语料和搜狐新闻数据(SogouCS) ,数据集共有 20 类别/18 类别,本文选择其中 4 个类别(艺术、财经、医疗、航空) 共含 20000 条数据为实验数据. 在实验中,本文将 20000 条数据作为词向量训练语料加入到 non-static 词向量训练中,在模型训练和测试时使用的都是每个条目的标题条目,因此本文选择的训练和测试语料都是含有少量词汇的短文本,符合本文的研究目标: 短文本的分类,在训练数据时将数据类别顺序打乱输入,选择 4000 条每个数据作为测试数据(每个类别 1000 条) ,其余 16000 条作为训练数据. 数据集链接[3].

表 1　实验环境

Table 1　lab Environment

实 验 环 境	环 境 配 置
操作系统	14.04
CPU	Intel Core i5-3220 3.3GHz
内存	4G
编程语言	Python2.7
分词工具	jieba
词向量训练工具	Word2vec
深度学习系统	tensorflow

5.3　实验设计

本文在词向量表示的基础上提出了 *Word embedding* 聚类语义簇,针对卷积神经网络模型提出了改进的双通道 *CNN* 模型. 为了证明本文创新的有效性,设计了多个对比模型,具体实验设计如下:

Word embedding 聚类语义簇的有效性对比试验. 在保证相同的分类模型下,分别采用原始的 *Word embedding* 输入对比语义聚类之后的 *Word embedding* 的输入来证明语义聚类的有效性. 为了证明 *Word embedding* 对短文本的有效性,加入对传统的文本特征表示的输入采用不同模型对数据集进行实验.

表 2　数据类别数目

Table 2　Number of data categories

数据类别	艺术	经济	医疗	航空
数据条数	5000	5000	5000	5000

双通道 CNN 模型有效性对比试验. 通过保证相同输入,分别使用不同的分类模型,本文采取的对比模型: 传统的机器学习模型中的 Linear SVM; 单通道 CNN 模型(文献 [3] 中 Single-CNN 模型) : 本文提出的双通道 CNN 模型(Double-CNN) .

不同编码输入对实验结果影响的对比实验,对本文提出的 *Double-CNN* 模型分别采用单一输入即 static、non-static 和使用两张输入 *non-static* + *static* 的对比实验.

评价指标: 本文采用准确率来衡量模型分类的能力.

5.4　实验结果

实验结果如下页表 3、表 4、表 5 所示,实验结果均经过十折交叉验证.

5.5　实验结果分析

1) 通过表 3 可以看出 *Word embedding* 相比传统的向量空间模型(VSM) 在短文本分类任务上的对比: 通过实验数据可以明显的看出,传统的特征表示方法有效性较差,但采取了深度神经网络编码的 *Word embedding* 能够有效解决特征表示对模型的影响. 通过表 4 可以看出 *Word embedding* 语义聚

[3] http: //www. nlpir. org/download/tc - corpus - answer. rar
http: //www. sogou. com/labs/resource/cs. php

类有效性分析通过三组实验结果对比: 相同的模型下,采用语义聚类处理的输入在分类结果上都有了一定的提高,原因分

表 3 传统特征表示在短文本分类任务上的实验结果

Table 3 Traditional features represent experimental results

模型特征输入	模 型	准确率
传统的 VSM 特征表示	Linear SVM	60.53%
传统的 VSM 特征表示	Single-CNN	57.62%
传统的 VSM 特征表示	Double-CNN	59.35%

析: 在经过 *Word embedding* 语义聚类后,类似与近义词聚拢,这样经过处理后的数据对分类模型而言相当于降低了噪声数据对模型训练和测试的影响. 影响分类准确因素一般分为两种: 一是内在的数据特征,良好的特征表示方式能够增强类别之间的区分度,增加分类效果; 二是外在的影响(噪声影响) ,噪声的影响程度很大力度上能影响一个模型的训练和测试的准确度. 本文采取了 *Word embedding* 词向量方式来解决内在问题,采取词向量语义聚类的方式来降低噪声的影响,在对比实验中也证明了语义聚类的有效性.

表 4 语义聚类和改进模型的实验结果

Table 4 Semantic clustering and improved model results

模型特征输入	模 型	准确率
No Clustering Word	Linear SVM	85.43%
No Clustering Word	Single-CNN	89.21%
No Clustering Word	Double-CNN	91.14%
Clustering Word	Linear SVM	86.54%
Clustering Word	Single-CNN	91.52%
Clustering Word	Double-CNN	92.37%

2) 改进的双通道 CNN 模型有效性分析. 在与传统模型对比上,选择了分类功能高效并且现在广泛应用的 Linear SVM 模型进行了对比实验,在采取了语义聚类后的 Word embedding 作为输入时 SVM 同样表现了很好的分类效果. 在与相同基准模型卷积神经网络(CNN) 对比是选择了文献 [8] 中的模型,论文是 2015 年 ACL 上的一篇 short paper,该模型成熟高效具有对比意义. 在结构上本文采取了改进的双通道结构,通过两种不同方式的特征输入,本文模型在特征提取上针对局部敏感信息提取的更加全面,因此在分类结果上有了很大的提高.

表 5 不同编码输入对比单一编码输入的实验结果

Table 5 Different encoding inputs versus the single-code inputs

文本编码方式	模 型	准确率
Static	Double-CNN	90.87%
Non-static	Double-CNN	91.70%
Static + non-static	Double-CNN	92.37%

3) 表 5 可以看出双通道采用采用不同的编码输入对比采用单一的编码输入在准确性上的提升约有 1% 左右,原因式单一编码输入导致模型两个通道获取的相同或近似的语义信息发挥不出双通道的获取丰富语义信息的特点. 采取两种不同的编码输入则可以是双通道更有效的发挥效果.

4) 数据分析. 数据集是文本分类语料库(复旦) 测试语料和搜狐新闻数据(SogouCS) ,分类语料在一定层度上已经经过处理,因此针对数据集能取得比较好的结果,如果短文本数据选择开放性极强的短文本,那么分类效果会受到影响. 在 Word embedding 训练时,non-static 方式在原本的语料集(大规模文本语料) 加入了测试语料作为训练语料,因此在 Word embedding 方面对词向量的训练质量有一定的提高,为提高模型分类标准提供了基本条件.

6 结 语

文本分类是数据挖掘和自然语言处理任务中的重要任务之一,短文本分类是应对近几年高速涌现的海量短文本. 本文从短文本分类中的难点出发,在模型输入数据处理和分类模型改进两个方向进行了创新. 首先针对模型输入前的数据处理,本文选择了 AP 聚类算法对 *Word embedding* 语义聚类,通过对语义聚类使得近义 *Word embedding* 聚集在同一个簇,降低了 *Word embedding* 噪声对模型的影响,再者针对卷积神经网络(CNN) 模型提出了一种新的双通道卷积神经网络,通过两个通道不同的输入更为全面的获取局部敏感信息,最后经过网络结构中的连续池化实现了对特征的降维表示,模型改进特点在针对短文本在分类任务的特性上提出,有效的解决了短文本的分类问题. 通过实验验证,本文提出方法与传统的分类模型和文献 [8] 中 CNN 结构相比在分类准确度上有了很大的提高.

下一步工作展望: 由于在对短文本的表示时过度依赖词向量的质量,如何能训练出适用广泛的文本表示是进一步解决自然语言任务的基本条件,目前的文本分类仍是基于特征提取,下一步将研究如何建立有效文本的表示模型和对文本理解的方法.

References:

[1] Hinton G E. Learning distributed representations of concepts [C] . Proc of the 8th Annual Conference of the Cognitive Science Society,1986.

[2] Tomas Mikolov,Kai Chen,Greg Corrado,et al. Efficient estimation of word representtations in vector space [J] . arXiv preprint arXiv: 1301,3781,2013.

[3] Bengio Y,Ducharme R,Vincent P,et al. A neural probabilistic language model [J] . Journal of Machine Learning Research,2003,3 (2) : 1137-1155.

[4] 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: 3111-3119.

[5] Nal Kalchbrenner,Edward Grefenstette,Phil Blunsom. A convolutional neural network for modelling sentences [J] . arXiv preprint arXiv: 1404,2188,2014.

[6] Yoon Kim. Convolutional neural networks for sentence classification [J] . arXiv preprint arXiv: 1408,5882,2014.

[7] Zhang Y,Wallace B. A sensitivity analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification,2015.

[8] Wang Peng,Xu Jia-ming,Xu Bo,et al. 2015. Semantic clustering and convolutional neural network for short text categorization [J] . In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume2: Short Papers) , pages 352-357,Beijing,China,July. Association for Computational Linguistics,2015.

[9] Alex Rodriguez and Alessandro Laio. Clustering by fast search and find of density peaks [J] . Science,2014,344(6191) : 1492-1496.

[10] Guan Ren-chu,Pei Zhi-li,Shi Xiao-hu,et al. Weight affinity propagation and its application to text clustering [J] . Journal of Computer Research and Development,2010,10: 1733-1740.

[11] Frey B J,Dueck D. Clustering by passing messages between data points [J] . Science,2007,315(5814) : 972-976.

附中文参考文献:

[10] 管仁初,裴志利,时小虎,等. 权吸引子传播算法及其在文本聚类中的应用 [J] . 计算机研究与发展,2010,10: 1733-1740.