一种融合语义特征的图卷积文本分类方法

黎文杰 洪嘉伟 魏艳辉 左亚尧*
(广东工业大学计算机学院 广东 广州 510006)

摘 要 随着文本分类领域相关研究的推进,基于深度学习的文本分类方法成为了该领域的重要研究方向之一。深度学习模型因其强大的特征提取能力,在文本分类任务上有着颇为优越的表现。但由于文本数据的高维性和自然语言的语义复杂性,现有的深度学习模型在复合语义信息的提取上仍有待进一步优化,其表现对文本分类效果产生不可忽视的影响。为此,该文提出一种基于 LDA 和 GCN 的文本分类模型 LGCN。该模型利用 LDA 模型学习文档、单词和主题的关联信息,借助滑动窗口、PMI 值计算等方式获取字符间的联系,采用 TF-IDF 得到单词和文档的联系,通过融合这些丰富的语义信息得到以节点形式构建的图,使用 GCN 模型学习图中语义信息并对图中文档节点进行分类从而完成文本分类任务。实验结果表明,在相同的数据集上,LGCN 模型的文本分类效果优于 LSTM 等参照模型。

关键词 图卷积神经网络 隐狄利克雷分布 文本分类

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2024.05.038

A GRAPH CONVOLUTIONAL TEXT CLASSIFICATION METHOD WITH SEMANTIC FEATURES

Li Wenjie Hong Jiawei Wei Yanhui Zuo Yayao*
(*Faculty of Computer*, *Guangdong University of Technology*, *Guangzhou 510006*, *Guangdong*, *China*)

Abstract With the advancement of related research in the field of text classification, text classification methods based on deep learning have become one of the important research directions in this field. Due to its powerful feature extraction capabilities, deep learning models have quite superior performance on text classification tasks. However, due to the high dimensionality of text data and the semantic complexity of natural language, the existing deep learning models still need to be further optimized in the extraction of composite semantic information, and their performance has a non-negligible impact on the text classification effect. Therefore, this paper proposes a text classification model LGCN based on LDA and GCN. The model used the LDA model to learn the associated information of documents, words and topics, and used sliding windows and PMI value calculations to obtain the relationship between characters. TF-IDF was used to obtain the connection between words and documents, and a graph constructed in the form of nodes was obtained by fusing rich semantic information. The GCN model was used to learn the semantic information in the graph and classify the document nodes in the graph to complete the text classification task. The experimental results show that on the same data set, the text classification effect of LGCN model is better than that of reference models such as LSTM.

Keywords Graph convolutional network Latent Dirichlet allocation Text classification

0 引 言

随着互联网技术的迅速发展,网络逐渐成为信息传递的重要渠道,社交软件与新闻媒体中蕴藏着数以亿计的文本信息,而如何准确高效地对大量文本信息进行分类[1],便于人们对数据进行管理与分析,是文本分类的一个热点和难点问题。文本分类可被应用于信

收稿日期:2021-01-15。广东省自然科学基金项目(2018A030313934)。**黎文杰**,硕士生,主研领域:自然语言处理,深度学习。**洪嘉伟**,本科生。**魏艳辉**,本科生。**左亚尧**,副教授。

息检索、情感分析及垃圾邮件识别等领域,成为自然语言处理领域中最核心的一项任务。

在文本分类任务的处理上,一些学者采用了基于统计学和机器学习的方法,如 K 近邻算法[2]、GDBT、基于字符串匹配、TF-IDF[3]、朴素贝叶斯[4]和支持向量机[5]等。然而传统的方法在文本表示时是高维稀疏的,其特征表达能力不强,需要预先进行文本特征提取,而人工特征提取的质量会影响最终文本分类的精度。近年来,随着深度学习的迅速发展,学者们倾向于将深度学习应用于文本分类领域,其主要方法是使用循环神经网络(RNN)[6]、卷积神经网络(CNN)[7]以及长短期记忆网络(LSTM)[6]等模型完成文本分类任务。例如,Tang 等[9]用 RNN 善于提取上下文信息的特点,提出了门控循环神经网络,对句子间的关系进行编码并应用于情感分类领域。Wang 等[10]使用非连续循环神经网络进行文本分类任务,但在复杂文本信息中会出现梯度爆炸及梯度消失等问题。Li 等[11]利用基于 CNN 的相似性矩阵实现短文本的分类。Kim 等[12]最先利用 CNN 提取文本信息中局部特征的优势,提出 Text-CNN 模型,使用数个级别不一的卷积核进行特征提取,然后采用 MAX POOLING 提取出关键信息进行分类。Zhang 等[13]在文本分类任务中提出一种 Char-CNN 模型,其输入的是字符而不是单词,使模型适用性更高。但由于 CNN 采用滑动窗口提取特征,会丢失上下文信息及全局特征。因此 Rao 等[14]通过使用 LSTM 模型获取上下文的依赖关系和特征信息,LSTM 可以很好地获取上下文信息,同时解决传统 CNN 的梯度消失与梯度爆炸问题,但由于 LSTM 输入的是整个文本,对于长距离信息的获取效果不够优秀,训练时间较长。

鉴于单一的神经网络模型在文本分类任务的不足,有学者提出将多个模型结合起来,利用每个模型的优点进行互补,并将注意力机制[15]应用于自然语言处理任务类。Yin 等[16]提出 ABCNN 模型,将 Attention 机制和 CNN 通道结合起来。王海涛等[17]提出了一种 MLCNN 模型,通过三层 CNN 层获取局部特征,结合 LSTM 获取上下文信息。吴汉瑜等[18]提出 CNN-BiLSTM-Attention 混合模型,可以同时获取局部与全局结构信息,并最终融合得到文本表示。最近,GCN 因其处理非拓扑结构数据的优越性,引起了众多学者的关注,Kipf 等[19]提出的 GCN 可以很好地处理非欧式结构的数据,同时可以有效地提取图的全局信息。

但 GCN 图节点编码的形式使其难以获取到上下文间的局部语义信息,而 LDA 模型能够有效地学习文本语义信息。因此,本文结合 LDA 模型与 GCN 的优势,提出一种新的基于 GCN 的文本分类模型(Latent Dirichlet Allocation integrate Graph Convolutional Network,LGCN),该模型将文本数据处理成一张包含单词、文档和主题为节点的图,使用 GCN 学习复合语义的图信息,兼顾文档的全局语义信息与词语间的局部语义信息。其中,生成图的单词节点之间的边由 PMI 计算后生成,单词与文档间的边由 TF-IDF 生成,主题和单词、主题和文档之间的边由 LDA 模型生成。利用两层的 GCN 将图中其他节点的信息聚合到文档节点中并对文档节点进行分类,从而将文本分类任务转变为文档节点的分类任务。此方法通过 GCN 获取全局语义信息,利用 LDA 模型获取词语间的局部语义信息,两者的结合使得模型学习到更为丰富的语义特征,有利于进一步提高模型的文本分类精度。实验结果表明 LGCN 模型在多个公开数据集上的准确率高于参照模型。

1 相关技术

1.1 LDA

Jedrzejowicz 等[20]提出了一种混合方法,该方法使用了隐狄利克雷分布算法,并通过代表单词的词嵌入获得的知识加以扩展。而隐狄利克雷分布(Latent Dirichlet Allocation,LDA)主题模型最早是由 Blei 等[21]出。这是一种用于离散数据集合的概率生成模型的三层贝叶斯主题模型,每篇文档代表了一些主题所构成的概率分布,而每一个主题则代表了许多单词所构成的一个概率分布。其将文档集中的每篇文档的主题以概率分布的形式给出,通过无监督的方法获取文本中隐藏的主题信息。同时,它是一种典型的词袋模型,即一篇文档由一组单词构成,单词之间没有先后顺序的关系。LDA 的模型如图 1 所示。

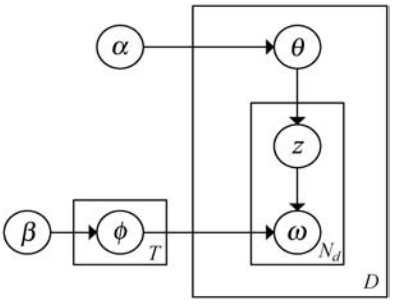

图 1 LDA 模型图

图 1 中,α 代表了 θ 的超参数,β 代表了 ϕ 的超参数,θ 表示文档-主题的概率分布,ϕ 表示单词-主题的概率分布,z 代表了词的主题分布,ω 代表了单词,D 代表了文档数,N 代表了单词数,T 代表了主题数。

对于已有文档集合,需要反求其文档-主题-单词的概率分布,即估计其未知参数。ϕ、θ 两个未知参数可以利用 Gibbs Sampling 采样[22]进行估算。参数计算

如式(1)和式(2)所示。

$$\theta_{mk} = \frac{n_{m,-i}^{(k)} + \alpha_k}{\sum_{k=1}^{K}(n_{m,-i}^{(k)} + \alpha_k)} \tag{1}$$

$$\varphi_{kt} = \frac{n_{k,-i}^{(t)} + \beta_t}{\sum_{t=1}^{V}(n_{k,-i}^{(t)} + \beta_t)} \tag{2}$$

式中:α 和 β 为给定的先验参数;$n_{m,-i}^{(k)}$ 表示主题 k 分配给文档 m 的次数;$n_{k,-i}^{(t)}$ 代表了单词 t 被观察到分配给主题 k 的次数。

1.2　图卷积神经网络

图卷积神经网络(Graph Convolutional Network,GCN)是一种基于图结构数据的多层神经网络,将深度学习中常用于图像识别的卷积神经网络应用于图数据上[23-24],解决了传统卷积神经网络在非欧氏几何数据上无法保持平移不变性的缺点。而谱域就是 GCN 的理论基础,由于 Laplacian 矩阵是特征矩阵,可以进行特征分解,有 n 个线性无关的特征向量且相互正交。故通过借助于图的拉普拉斯矩阵的特征值和特征向量实现拓扑图上的卷积操作。其特征分解公式如式(3)所示,其中 $\boldsymbol{U}=(\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_n)$ 是以列向量为单位特征的矩阵,$(\lambda_1,\lambda_2,\cdots,\lambda_n)$ 是 n 个特征值构成的对角阵:

$$\boldsymbol{L}=\boldsymbol{U}\begin{pmatrix}\lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n\end{pmatrix}\boldsymbol{U}^{\mathrm{T}} \tag{3}$$

传统傅里叶变换的基为拉普拉斯矩阵的一组特征向量,故图的傅里叶变换矩阵形式为 $gF\{x\}=\boldsymbol{U}^{\mathrm{T}}x$。而 GCN 类比频域卷积定理,得出卷积公式如式(4)所示。

$$(\boldsymbol{f}*\boldsymbol{h})_G=\boldsymbol{U}((\boldsymbol{U}^{\mathrm{T}}\boldsymbol{h})\odot(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{f})) \tag{4}$$

定义图 $G=(V,E)$,图中每个节点 i 的特征为,所有节点的特征构成 $N\times D$ 的特征矩阵 $\boldsymbol{X}$,其中 N 为节点数,D 为的特征向量维度。并将 $\boldsymbol{X}$ 作为 GCN 的输入,得到 $N\times F$ 的隐藏层特征矩阵 $\boldsymbol{H}$,其中 F 是每个节点的隐藏层的特征维度。一层 GCN 只能聚合节点一阶邻域内的信息,而多层 GCN 的传播学习能让模型聚合更广域的节点间信息,GCN 第 l 层信息的传播公式如式所示:

$$\boldsymbol{H}^{(l+1)}=f(\boldsymbol{A}\boldsymbol{H}^{(l)}\boldsymbol{W}) \tag{5}$$

式中:$\boldsymbol{H}^{(0)}=\boldsymbol{X}$,$\boldsymbol{A}$ 为邻接矩阵,$f(\cdot)$ 为激活函数,l 为 GCN 的层数。

2　LGCN 模型

2.1　模型概述

LGCN 模型结合 LDA 模型与 GCN 的特点,通过 PMI 获取单词间的关联信息,利用 TF-IDF 获取文档与单词之间的关联信息,采用 LDA 模型来学习到主题-文档分布和主题-单词分布的语义信息,藉由 GCN 模型将单词、主题的信息聚合到文档节点中,从而得到文档节点的特征向量以进行文本分类。下文将分别围绕语义学习、关系学习、文档语义学习和算法描述等部分展开介绍。

2.2　LGCN 模型构建

LGCN 模型首先利用 LDA 模型发掘语料中文档的主题信息,采用共现矩阵发掘文档中单词对后通过计算单词对之间的 PMI 值计算其关联程度并依此建立单词节点之间的信息边;再通过计算单词和文档之间的 TF-IDF 值来得到单词和文档的从属关系并建成单词和文档的联系边;之后,聚合文档、主题和单词的联系构建其联系图;最后,利用 GCN 融合单词特征和主题特征得到文档的特征向量并依此完成文档分类任务。LGCN 模型框架如图 2 所示。

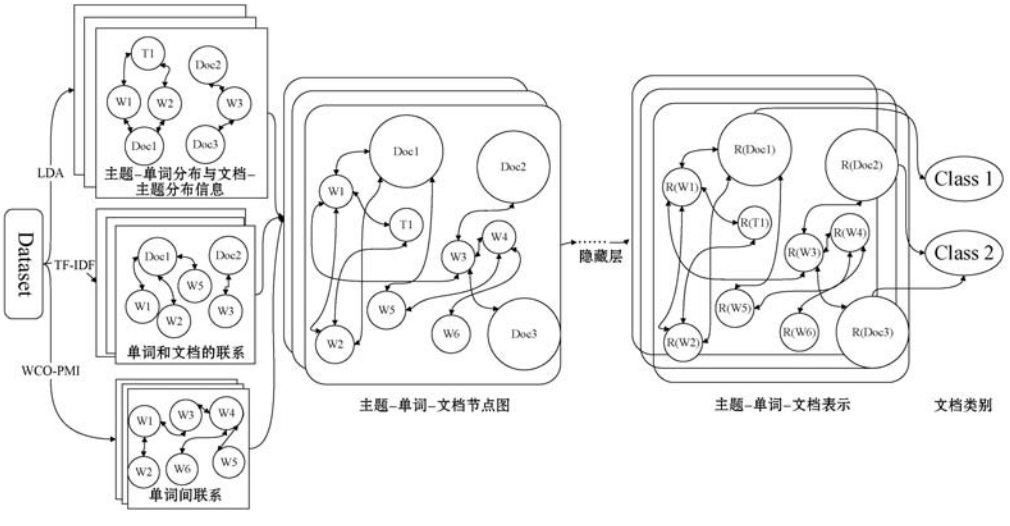

图 2　LGCN 模型框架

2.2.1 语义学习

对于图中的文档集合 Dataset 可记为 $D=[d_1,d_2,\cdots,d_n]$，其中 x_i 为第 i 篇文档，n 为数据集的个数，而对每个文档 x_i，通过预切词处理组成单词集合 $W=[w_1,w_2,\cdots,w_n]$，在 LGCN 模型中预设定 Topic 数量为 k，并采用 Gibbs 采样算法得到所有主题的集合 $Z=(Z_1,Z_2,\cdots,Z_n)$，利用 Dirichlet-multi 这组共轭分布，计算第 i 个文档的主题的条件分布 $p(\boldsymbol{z}_i|\boldsymbol{\alpha})$，其公式如式(6)所示：

$$p(\boldsymbol{z}_i|\boldsymbol{\alpha})=\frac{1}{\Delta(\boldsymbol{\alpha})}\int\prod_{k=1}^{K}p_k^{n_i^{(k)}+\alpha_{k-1}}\mathrm{d}\boldsymbol{\theta}_i \tag{6}$$

式中：第 i 个文档中的第 k 个主题的词的数量记为，其对应的多项分布为 $\boldsymbol{n}_i=(n_i^{(1)},n_i^{(2)},\cdots,n_i^{(K)})$；$\Delta(\boldsymbol{\alpha})$ 代表了归一化参数。

在得到所有文档的主题条件分布 θ_d 后，可由式(7)得词的联合分布 $p(\boldsymbol{w},\boldsymbol{z}|\boldsymbol{\alpha},\boldsymbol{\eta})$。

$$p(\boldsymbol{w},\boldsymbol{z}|\boldsymbol{\alpha},\boldsymbol{\eta})=\prod_{d=1}^{M}\frac{\Delta(\boldsymbol{n}_d+\boldsymbol{\alpha})}{\Delta(\boldsymbol{\alpha})}\prod_{k=1}^{K}\frac{\Delta(\boldsymbol{n}_k+\boldsymbol{\eta})}{\Delta(\boldsymbol{\eta})} \tag{7}$$

式中：$\boldsymbol{w}$ 为文档中词向量，$\boldsymbol{z}$ 为语料库中主题的分布，$\boldsymbol{\alpha}$ 与 $\boldsymbol{\eta}$ 为 LDA 模型中的超参数，K 为预先设定的主题数量。获取文档主题与主题单词的分布后，将文档主题间的概率转换成图中文档节点与主题节点之间边的权重，将主题单词间的概率转换成图中主题节点与单词节点间边的权重。

2.2.2 关系学习

LGCN 模型采用滑动窗口处理文档 X_i，得到其对应的候选词语集合 $S=(s_1,s_2,\cdots,s_m)$，S_i 为二元的词语，m 为候选词语的个数。并统计词语 S_i 及前后两个单词的出现频次，计算其点式互信息值 PMI；若 PMI 值大于 0，则认定为有关联，否则不存在联系。具体计算方式如式(8)、式(9)和式(10)所示。

$$PMI(i,j)=\log\frac{p(i,j)}{p(i)\cdot p(j)} \tag{8}$$

$$p(i,j)=\frac{S(i,j)}{|S|} \tag{9}$$

$$p(i)=\frac{S(i)}{|S|} \tag{10}$$

式中：$S(i)$ 是滑动窗口的候选词语集 S 中包含了第 i 个词的词语频率；$S(i,j)$ 是滑动窗口中候选词语集合 S 中词语为字符 i,j 的词语频率，$|S|$ 是候选词语集 S 的总数量。

为了挖掘单词跟文档的关系，LGCN 模型采用 TFIDF 的方法量化单词和文档的联系程度。对于给定文档集 X，首先计算每个文档所有单词的词频，再统计其单词出现在文档集 X 的次数得到逆文本频率，最后将单词的词频与其对应文档的 TFIDF 值相乘。具体计算公式如式(11)－式(13)所示。

$$TFIDF(i,j)=TF(i,j)\cdot IDF(i,j) \tag{11}$$

$$TF(i,j)=\frac{freq(i)}{|x_j|} \tag{12}$$

$$IDF(i,j)=\log\left(\frac{|X|}{|\{o:i\in X_a\}|+1}\right) \tag{13}$$

式中：i,j 代表第 j 个文档第 i 个单词，$freq(i)$ 表示第 i 个单词的频率，$|x_j|$ 为第 j 个文档的长度，$|\{o:i\in X_a\}|$ 表示包含单词 i 的文档的数目。

经过上述步骤的处理，将获取到单词、文档、主题三者间的联系合并得到邻接矩阵 $\boldsymbol{A}\in\mathbf{R}^{d\times d}$，计算公式如式(14)所示。

$$A_{i,j}=\begin{cases}PMI(i,j) & PMI(i,j)>0\text{ 且 }i,j\text{ 为单词}\\ TFIDF(i,j) & i\text{ 为单词},j\text{ 为文档}\\ WT(i,j) & i\text{ 为单词},j\text{ 为主题}\\ TD(i,j) & i\text{ 为主题},j\text{ 为文档}\\ 1 & i=j\\ 0 & \text{其他}\end{cases} \tag{14}$$

式中：d 大小为单词数量、主题数量和文档数量之和。

2.2.3 文档语义学习

LGCN 模型将语义学习与关系学习得到的邻接矩阵构成一幅图 $G=(V,E)$，其中节点集合为 $V=(w_1,w_2,\cdots,w_q,t_1,t_2,\cdots,t_k,d_1,d_2,\cdots,d_v)$，$q$、$k$ 和 v 分别为单词、主题和文档的数量，而边集合为 $E=\{(w_1,w_1,weight_1),\cdots,(w_i,t_j,weight_i),\cdots(t_m,d_n,weight_m)\}$，$weight_i$ 为两个节点之间的权重系数。LGCN 模型中采用两层的 GCN 用以提取语义特征，其传播过程如式(15)所示。

$$\boldsymbol{Z}=\tilde{\boldsymbol{A}}\,\mathrm{ReLU}(\tilde{\boldsymbol{A}}\boldsymbol{X}\boldsymbol{W}_0)\boldsymbol{W}_1 \tag{15}$$

式中：$\boldsymbol{W}_0$ 和 $\boldsymbol{W}_1$ 为 LGCN 模型的参数矩阵，$\mathrm{ReLU}(\cdot)$ 是激活函数，$\boldsymbol{X}\in\mathbf{R}^{|V|\times M}$ 为单词、主题和文档的特征向量矩阵，M 为特征向量的维度。$\tilde{\boldsymbol{A}}$ 为归一化对称邻接矩阵，计算如式(16)所示。

$$\tilde{\boldsymbol{A}}=\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}} \tag{16}$$

式中：$\boldsymbol{D}$ 为图 G 的度矩阵。最终将其输入到 Softmax 层得到文档分类的结果，计算式如式(17)和式(18)所示。

$$y_{\mathrm{p}}=\mathrm{Softmax}(\boldsymbol{Z}) \tag{17}$$

$$\mathrm{Softmax}(z_i)=\frac{\exp(z_i)}{\sum_{c=1}^{C}\exp(z_c)} \tag{18}$$

式中：y_{p} 代表预测的标签值；z_i 表示第 i 个节点的输出

值,C 为要分类的文本标签数量。

为了优化 LGCN 模型的训练参数,本文采用了如式(19)所示的交叉熵作为损失函数,训练模型时采用反向传播算法更新模型得到优化的参数。

$$L = -\frac{1}{N}\sum_{i}^{N}\sum_{c=1}^{C} y_{ic}\log(p_{ic}) \tag{19}$$

式中:y_{ic}是指示变量,如果该类别和文本 i 类别相同则是 1,否则为 0;p_{ic}是文本 i 属于类别 c 的概率值。

2.3　算法描述

综合上述构建过程,LGCN 模型对应的算法描述算法 1 所示。图中对于输入的文本集 X,先由 LDA 模型训练得到主题-单词分布 *topic_word* 和文档-主题分布 *doc_topic*,通过滑动窗口处理文本集得到候选词语集 *coword*,由 PMI 算得词语之间的关系 *word_w*、TFIDF 获得单词和文档的关系 *word_doc*,然后将其合并构建出邻接矩阵 ***adj*** 和特征向量 ***feature***;最后将 GCN 提取到的文档节点特征输入到 Softmax 层中得到文档节点属于某类标签的概率值,通过交叉熵计算损失值并反向传播更新参数。

算法 1　LGCN 文本分类算法

输入:文本集 X,正确的标签 y。

输出:预测文本 y'。

Initialize: topic_num, window

Begin:

```
topic_word, doc_topic = LDA( topic_num, X)
coword = wordCooc( win = window)
word_w = PMI( coword)
word_doc = TFIDF( X)
adj, feature = get_graph( topic_word, doc_topic, word_w, word_doc, X)
for i←0 to iter do
    adj, feature input to GCN
    gcnemb←GCN( adj, featture)
    y_p = Softmax( gcnemb)
    loss = CrossEntropy( y_p, y)
    update GCN by loss
End for
End
```

3　实验与结果分析

为了验证 GCN 对于文本分类任务的性能,本文在本地服务器搭建实验环境,构建了基于 GCN 和 LDA 的文本分类模型,在公开数据集上进行多轮的实验测试,下面分别从实验语料、实验设置与环境和实验结果等方面展开描述。

3.1　实验语料

实验环节主要采用 R8 数据集、20ng 数据集和 Ohsumed 数据集作为文本分类实验的语料集,它包含了文章序号、训练集、测试集和类别等。数据分布情况如表 1 所示。

(1) R8 数据集是属于 Reuters-21578 数据集的子集,包含 8 种类别,训练集数量为 5 485 条,测试集数量为 2 189 条。

(2) 20ng 数据集包含 20 个不同主题的新闻组文章,共 18 846 个文档,训练集数量为 11 314 条,测试集为 7 532 个。

(3) Ohsumed 数据集来源于医药信息数据库 MEDLINE10,包含了 348 566 个文档,训练集数量为 3 357 篇,测试集数量为 4 043 条。

表 1　实验数据分布表

数据集	训练集数量	测试集数量
R8	5 485	2 189
20ng	11 314	7 532
Ohsumed	3 357	4 043

3.2　实验设置与训练

本文模型使用 300 维随机初始化的节点向量,并使用 Dropout 率为 0.5 的 Dropout 层来避免模型出现过拟合现象。本文模型设置 200 轮的训练批次,若训练过程中 10 轮内精度没有提升,则结束模型的训练。通过在多个训练集上进行多组实验来选择最优参数。在 R8 数据集上,准确率随 LDA 主题数量和节点向量维度的变化趋势如图 3 和图 4 所示。可以看出,当主题数量和节点向量维度均为 300 时,本文模型取得最优准确率,当主题数量和节点向量维度增加或者减少时,准确率呈现降低趋势,因此本模型将主题数量和节点向量维度分别设置为 300 和 300。同理,利用以上实验方法可以获得其他参数的最优值,具体如表 2 所示。

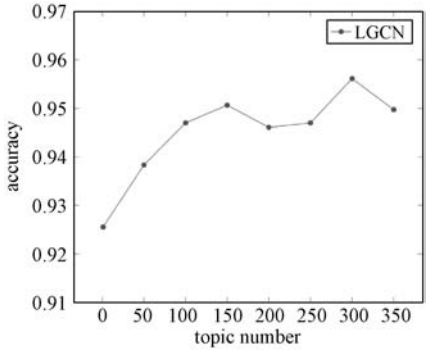

图 3　R8 数据集主题数量与测试精度图

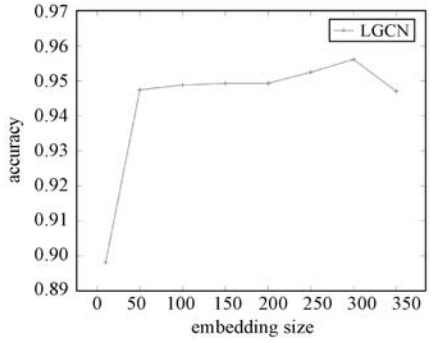

图 4 R8 节点向量维度与测试精度图

表 2 模型参数设置

参数名	值
隐藏层维度	200
节点向量维度	300
主题数量	300
GCN 层数	2
Dropout 参数	0.5
训练批次大小	200
学习率	0.02

LGCN 采用 Adam 的梯度下降方法进行模型的训练学习,有效解决了学习率消失、收敛过慢和高方差的参数更新导致的损失函数波动较大等问题,并采用 L2-regularization 来约束模型,防止参数出现过拟合问题。

从图 5 – 图 7 中可得出,随着迭代次数的增加,LGCN 模型训练的损失值不断下降,并在 100 轮次时收敛,其证明了本文模型能够有效优化参数并达到稳定。

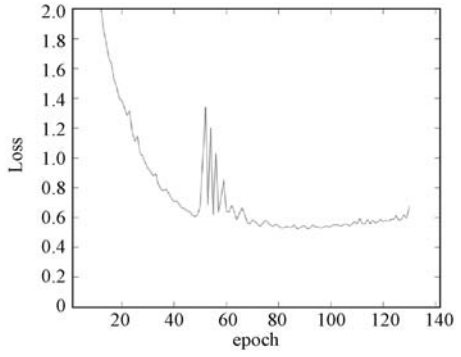

图 5 20ng 损失值图

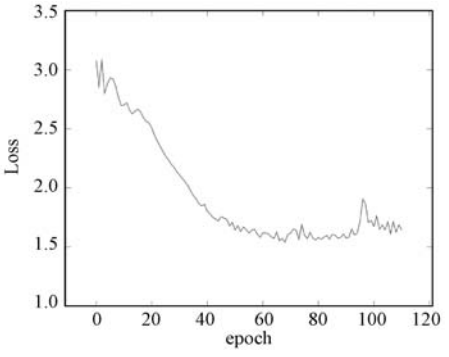

图 6 Ohsumed 损失值图

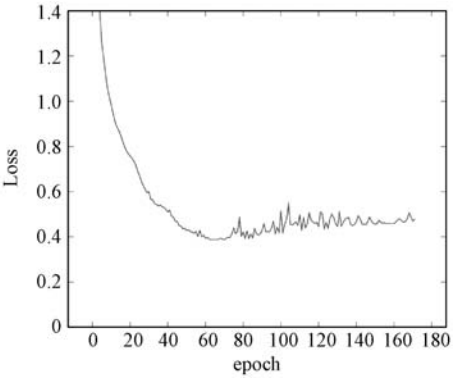

图 7 R8 损失值图

从图 3 可以得知不增加主题节点时,本文模型的测试精度为最低,随着主题节点数量的增加,本文模型的测试精度与主题数量基本呈正相关的关系,证明了 LDA 模型能够有效学习全文语义信息进而提高分类结果。但当主题数量超过 300 时其精度反而下降,原因可能是主题数量太多,LDA 模型较难达到收敛,有冗余的主题导致语义信息部分缺失。

从图 4 可以得知低维度的向量嵌入不能有效地表征字符的信息,而高维度的向量嵌入不会提高分类的结果,反而可能会增加了训练的时间成本。当节点向量维度为 300 时本文模型在 R8 数据集取得最高测试精度 95.61%。

3.3 实验配置及结果分析

本文的实验环境及其配置如表 3 所示。

表 3 软硬件环境

项目	实验环境
系统	Windows10
GPU	NVIDIA 3080
硬盘	1 TB
内存	16 GB
Python 版本	Python 3.6
PyTorch 版本	PyTorch1.1

实验采用准确率作为评价标准对 LGCN 模型的文本分类性能进行评测,准确率(Accuracy)评估的是实体标签的准确率,其具体计算式为:

$$P = \frac{c_{\text{orrect}}}{c_{\text{orrect}} + m_{\text{issing}}} \tag{20}$$

式中:c_{orrect}为正确分类的文本数;m_{issing}为分类错误的文本数。

本文以 CNN-rand、LSTM、PTE、PV-DBOW 和 fastText 作为参考基准模型,对 LGCN 模型的文本分类性能进行了对比分析,重点考察了文本分类的准确率,其性能对比结果如表 4 所示,占比越高代表文本分类的

准确率越高,则模型的性能越高。

表 4　实验准确率对比

模型	R8	20ng	Ohsumed
CNN-rand[25]	0.940 2	0.769 3	0.438 7
LSTM[26]	0.936 8	0.657 1	0.411 3
PV-DBOW[27]	0.858 7	0.743 6	0.466 5
FastText[28]	0.961 3	0.793 8	0.577 0
PTE[29]	**0.966 9**	0.767 4	0.535 8
LGCN(本文模型)	0.956 1	**0.800 5**	**0.612 2**

在数据集 R8、20ng 和 Ohsumed 上,Kim 等[25]使用随机初始化单词编码的 CNN-rand 模型,其测试精度为 0.940 2、0.769 3 和 0.438 7;Liu 等[26]使用 LSTM 的最后一个隐藏状态作为文本特征表征来分类,其测试精度为 0.936 8、0.657 1 和 0.411 3;Le 等[27]提出段落矢量模型,其测试精度为 0.858 7、0.743 6 和 0.466 5;Joulin 等[28]采用 word/n-gram 作为文档嵌入,提出了 FastText 模型,其测试精度为 0.961 3、0.793 8 和 0.577 0;Tang 等[29]采用基于包含词、文档的异构文本网络学习词嵌入和标签作为节点,然后平均单词嵌入为用于文本分类的文档嵌入,其取得精度为 0.966 9、0.767 4 和 0.535 8。

可见 LGCN 模型在 R8 数据集上的准确率较最低值提升 12.6%,在 20ng 数据集上的准确率相较对比模型中最优值提升 0.84%,比最低值高 21.82%,在 Ohsumed 数据集上比最高值提升 6.1%,比最低值高 48.85%。结果表明,相对于参考模型,LGCN 模型能够有效地提取文本的复合语义特征,挖掘出更丰富的语义信息,有利于进一步提升文本分类效果。

4　结　语

本文结合 LDA 模型和 GCN 的优点,在文本分类任务中提出了一种 LGCN 神经网络模型。LGCN 模型通过融合 PMI 计算得到的单词间关联信息、LDA 模型学习到的主题-文档分布和主题-单词分布的语义信息、TF-IDF 获得的单词和文档关联信息,得到文档语义特征并输入到 Softmax 以完成文本分类任务。实验结果表明,LGCN 模型在 20ng 和 Ohsumed 数据集上的性能均优于参考模型,在 R8 数据集上也取得了较为不错的结果。后续实验可考虑将图注意力机制引用到本文模型中,以进一步提升其分类性能。

参　考　文　献

[1] 熊回香,杨梦婷,李玉媛. 基于深度学习的信息组织与检索研究综述[J]. 情报科学,2020,38(3):3 – 10.

[2] Maillo J, Ramirez S, Triguero I, et al. kNN-IS: An iterative Spark-based design of the K-nearest neighbors classifier for big data[J]. Knowledge-Based Systems,2016,117:3 – 15.

[3] Pimpalkar A P, Raj R J. Influence of pre-processing strategies on the performance of ML Classifiers exploiting TF-IDF and BOW features[J]. Advances in Distributed Computing and Artificial Intelligence Journal,2020,9(2):49 – 68.

[4] Jiang L X, Li C Q, Wang S, et al. Deep feature weighting for naive Bayes and its application to text classification[J]. Engineering Applications of Artificial Intelligence,2016,52:26 – 39.

[5] Liu Z Y, Kan H P, Zhang T, et al. DUKMSVM: A framework of deep uniform kernel mapping support vector machine for short text classification[J]. Applied Sciences,2020,10(7):2348.

[6] Lan Y, Hao Y Z, Xia K, et al. Stacked residual recurrent neural networks with cross-layer attention for text classification[J]. IEEE Access,2020,8:70401 – 70410.

[7] Helmy A, Omar Y M K, Hodhod R. An innovative word encoding method for text classification using convolutional neural network[EB]. arXiv:1903.04146,2019.

[8] Huan H, Yan J Y, Xie Y Q, et al. Feature-enhanced non-equilibrium bidirectional long short-term memory model for Chinese text classification[J]. IEEE Access, 2020, 8: 199629 – 199637.

[9] Tang D Y, Qin B, Liu T. Document modeling with gated recurrent neural network for sentiment classification[C]//Conference on Empirical Methods in Natural Language Processing,2015:1422 – 1432.

[10] Wang B X. Disconnected recurrent neural networks for text categorization[C]//56th Annual Meeting of the Association for Computational Linguistics,2018:2311 – 2320.

[11] Li M C, Clinton G, Miao Y J, et al. Short text classification via knowledge powered attention with similarity matrix based CNN[EB]. arXiv:2002.03350,2020.

[12] Kim Y. Convolutional neural networks for sentence classification[EB]. arXiv:1408.5882,2014.

[13] Zhang X, Zhao J B, LeCun Y. Character-level convolutional net-works for text classification[C]//28th International Conference on Neural Information Processing Systems,2015:649 – 657.

[14] Rao A, Spasojevic N. Actionable and political text classification using word embeddings and LSTM[EB]. arXiv:1607.02501,2016.

[15] Xie J B, Hou Y J, Wang Y J, et al. Chinese text classification based on attention mechanism and feature-enhanced fusion neural network[J]. Computing,2020,102(3):683 – 700.

(下转第 285 页)

计算机应用与软件,2019,36(11):107 - 111.
[2] 雍菊亚,周忠眉. 基于互信息的多级特征选择算法[J]. 计算机应用,2020,40(12):3478 - 3484.
[3] 裴作飞,李兆玉,王云锋,等. 基于自适应遗传算法的混合特征选择方法[J]. 计算机应用与软件,2020,37(8):256 - 259,306.
[4] 李占山,刘兆赓,俞寅,等. 量子化信息素蚁群优化特征选择算法[J]. 东北大学学报(自然科学版),2020,41(1):17 - 22.
[5] 唐晓娜,张和生. 一种混合粒子群优化遗传算法的高分影像特征优化方法[J]. 遥感信息,2019,34(6):113 - 118.
[6] Chen K, Zhou F Y, Yuan X F. Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection [J]. Expert Systems with Applications, 2019, 128(7): 140 - 156.
[7] Cheraghchi F, Abualhaol I, Falcon R, et al. Modeling the speed-based vessel schedule recovery problem using evolutionary multi-objective optimization [J]. Information Sciences, 2018, 448(6)53 - 74.
[8] Faris H, Mafarja M, Heidari A, et al. An efficient binary Salp swarm algorithm with crossover scheme for feature selection problems [J]. Knowledge-Based Systems, 2018, 154(6): 43 - 67.
[9] 王萌,丁志军. 一种新的设备指纹特征选择及模型构建方法[J]. 计算机科学,2020,47(7):257 - 262.
[10] 李金霞,赵志刚,李强,等. 改进的局部和相似性保持特征选择算法[J]. 计算机科学,2020,47(S1):480 - 484.
[11] Hancer E, Xue B, Zhang M J. Differential evolution for filter feature selection based on information theory and feature ranking[J]. Knowledge-Based Systems, 2018, 140(6): 103 - 119.
[12] 黄学雨,徐浩特,陶剑文. 具有特征选择的多源自适应分类框架[J]. 计算机应用,2020,40(9):2499 - 2506.
[13] 钟昌康. 基于 K 近邻和粒子群优化的特征选择算法[J]. 现代计算机,2020(9):21 - 24,40.
[14] Ma X L, Zhang Q F, Tian G D, et al. On Thebycheff decomposition approaches for multi-objective evolutionary optimization[J]. IEEE Transactions on Evolutionary Computation, 2018, 22(2): 226 - 244.
[15] Mafarja M, Aljarah I, Heidari A, et al. Evolutionary population dynamics and grasshopper optimization approaches for feature selection problems [J]. Knowledge-Based Systems, 2018, 145(6): 25 - 45.
[16] Pan A Q, Wang L, Guo W A, et al. A diversity enhanced multi-objective particle swarm optimization[J]. Information Sciences, 2018, 436(6): 441 - 465.
[17] Taradeh M, Mafarja M, Heidari A, et al. An evolutionary gravitational search-based feature selection[J]. Information Sciences, 2019, 497(5): 219 - 239.

(上接第 253 页)

[16] Yin W P, Schütze H, Xiang B, et al. ABCNN: Attention-based convolutional neural network for modeling sentence pairs[J]. Transactions of the Association for Computational Linguistics, 2016, 4: 259 - 272.
[17] 王海涛,宋文,王辉. 一种基于 LSTM 和 CNN 混合模型的文本分类方法[J]. 小型微型计算机系统,2020,41(6):1163 - 1168.
[18] 吴汉瑜,严江,黄少滨,等. 用于文本分类的 CNNBiLSTMAttention 混合模型[J]. 计算机科学,2020,47(S2):24 - 27,34.
[19] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks [EB]. arXiv: 1609. 02907, 2016.
[20] Jedrzejowicz J, Zakrzewska M. Text classification using LDA W2V hybrid algorithm[M]//Intelligent Decision Technologies. Singapore: Springer, 2020: 227 - 237.
[21] Blei D M, Ng A Y, Jordan M I. Latent Dirichlet allocation [J]. Journal of machine Learning research, 2003, 3: 993 - 1022.
[22] Maier D, Waldherr A, Miltner P, et al. Applying LDA topic modeling in communication research: Toward a valid and reliable methodology[J]. Communication Methods and Measures, 2018, 12(2 - 3): 93 - 118.
[23] Yao L, Mao C S, Luo Y. Graph convolutional networks for text classification[C]//33rd AAAI Conference on Artificial Intelligence, 2019: 7370 - 7377.
[24] Li M S, Chen S, Chen X, et al. Actional-structural graph convolutional networks for skeleton-based action recognition [C]//IEEE Conference on Computer Vision and Pattern Recognition, 2019: 3595 - 3603.
[25] Liu P F, Qiu X P, Huang X J. Recurrent neural network for text classification with multi-task learning[EB]. arXiv: 1605. 05101, 2016.
[26] Le Q, Mikolov T. Distributed representations of sentences and documents[C]//31st International Conference on Machine Learning, 2014: 1188 - 1196.
[27] Joulin A, Grave E, Bojanowski P, et al. Fasttext. zip: Compressing text classification models[EB]. arXiv: 1612. 03651, 2016.
[28] Tang J, Qu M, Mei Q Z. PTE: Predictive text embedding through large-scale heterogeneous text networks[C]//21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015: 1165 - 1174.