第37卷 第3期 应 用 科 学 学 报 Vol. 37 No. 3
2019年5月 JOURNAL OF APPLIED SCIENCES — Electronics and Information Engineering May 2019

DOI: 10.3969/j.issn.0255-8297.2019.03.008

融合TF-IDF和LDA的中文FastText短文本分类方法

冯 勇[1], 屈渤浩[1], 徐红艳[1], 王嵘冰[1], 张永刚[2]

1. 辽宁大学 信息学院, 沈阳 110036

2. 吉林大学 符号计算与知识工程教育部重点实验室, 长春 130012

摘 要: FastText文本分类模型具有快速高效的优势, 但直接将其用于中文短文本分类则存在精确率不高的问题. 为此提出一种融合词频-逆文本频率(term frequency-inverse document frequency, TF-IDF)和隐含狄利克雷分布(latent Dirichlet allocation, LDA)的中文FastText短文本分类方法. 该方法在FastText文本分类模型的输入阶段对n元语法模型处理后的词典进行TF-IDF筛选, 使用LDA模型进行语料库主题分析, 依据所得结果对特征词典进行补充, 从而在计算输入词序列向量均值时偏向高区分度的词条, 使其更适用于中文短文本分类环境. 对比实验结果可知, 所提方法在中文短文本分类方面具有更高的精确率.

关键词: 中文短文本分类; FastText; 词频-逆文本频率; 词向量; 隐含狄利克雷分布

中图分类号: TP311 **文章编号**: 0255-8297(2019)03-0378-11

Chinese FastText Short Text Classification Method Integrating TF-IDF and LDA

FENG Yong[1], QU Bohao[1], XU Hongyan[1], WANG Rongbing[1], ZHANG Yonggang[2]

1. College of Information, Liaoning University, Shenyang 110036, China

2. Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China

Abstract: FastText text classification model has the advantages of high speed and high efficiency, but its application in Chinese short text classification has the problem of low precision. To solve this problem, a Chinese FastText short text classification method integrating TF-IDF and LDA is proposed. In the input phase of FastText text classification model, the dictionaries generated after n-gram processing are filtered by TF-IDF, and corpus thematic analysis is conducted by LDA model, then the feature dictionary is supplemented according to the obtained results. Thus, the highly differentiated entries are

收稿日期: 2018-09-28; **修订日期**: 2018-10-29

基金项目: 国家自然科学基金(No. 71771110); 中国博士后科学基金(No. 2018M631814); 辽宁省社会科学规划基金(No. L18AGL007); 符号计算与知识工程教育部重点实验室项目基金(No. 93K172018K01)资助

通信作者: 王嵘冰, 副教授, 研究方向: 数据挖掘、大数据技术, E-mail: wrb@lnu.edu.cn

biased in the process of computing the mean value of input word sequence vectors, making them more suitable for Chinese short text classification environment. The experimental results show that the proposed method has higher precision in Chinese short text classification.

Keywords: Chinese short text classification, FastText, term frequency-inverse document frequency (TF-IDF), word vector, latent Dirichlet allocation (LDA)

随着Web2.0的兴起以及移动智能设备的发展，用户由传统的信息消费者转变为信息供给者. 以微博、微博评论、电商评论等为代表的中文短文本已成为网络用户自由表达主观兴趣、观点和情感的主要方式. 开展中文短文本分类方法研究对用户兴趣挖掘、热点话题发现、电商商品质量检测以及个性化推荐等领域具有重要价值.

目前，针对短文本分类方法的研究大多基于向量空间模型（vector space model, VSM）[1]、连续词袋模型（continuous bag-of-words, CBoW）[2]、语义信息等. 如：文献[3]在使用Word2Vec的基础上引入TF-IDF（term frequency-inverse document frequency）对词向量进行加权，以实现加权的分类模型；文献[4-5]借助维基百科知识库对短文本的特征进行扩展，以辅助短文本分类；文献[6]采用正则化权值的方式对KNN（K-nearest neighbor）进行改进，并结合粒子群优化（particle swarm optimization, PSO）算法提高了计算效率和文本分类的效率. 文献[7]提出的深度学习框架可用于提取结构化文本的高级特征，将该特征与隐含狄利克雷分布（latent Dirichlet allocation, LDA）获取的主题特征相结合能产生更好的分类效果；文献[8]基于决策树算法对文本进行分类，并对传统机器学习方法进行了改进以提升文本分类的精确率和效率.

现有的文本分类方法取得了较为显著的应用效果，但应用于短文本分类环境时仍存在不足. 短文本相比于长文本具有稀疏性、不规范等特点，因而使用传统向量空间模型时，短文本的稀疏性使得分类不够准确. 为了弥补上述不足，通常借助外部知识库对短文本进行特征补充. 但因外部知识库语料庞大，且包含的领域以及主题分布广泛，导致计算开销大，进而影响算法的性能. 另外，外部知识库词汇更新速度慢，不适合互联网环境下对高速变化的中文短文本进行特征补充.

本文提出一种融合TF-IDF和LDA的中文FastText短文本分类方法（简称TL-FastText）. 该方法首先在FastText的基础上对n元语法模型(n-gram)处理后的输入词序列进行TF-IDF计算，并筛选高区分度的词条构建保留词典[9]；然后使用LDA模型对语料库进行主题分析，将词典中的词与LDA结果进行对比，如果有相同的词，则将LDA结果中含有该词的主题词序列加入保留词典，使模型在进行输入词序列向量平均值计算时向区分度高的词条偏移，从而使其更适用于中文短文本分类环境，实现快速准确地对中文短文本进行分类的目标.

1　相关工作

目前，针对中文短文本的分类大多采用基于深度学习的方法，但深度学习的模型训练时间过长，导致算法无法实现高速迭代. FastText分类模型虽具有训练速度快、分类精度高的优势，但FastText分类模型主要是根据英文短文本的特点设计实现的. 本文将结合中文短文本的特点对FastText分类模型进行改进，使之能适用于中文短文本分类环境.

1.1　TF-IDF

TF-IDF[10]是用于评估词条对于文本文件重要程度的经典统计方法，倾向于过滤去除区分度低的高频词，保留重要区分度高的低频词. 词频TF表示词条在文档中出现的频率，

用 $f_{i,j}$ 表示，其计算公式为

$$f_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}} \tag{1}$$

式中，$n_{i,j}$ 为词条 t_i 在文本 d_j 中出现的次数，分母表示文本 d_j 中所有词条出现的次数总和.

IDF 是词条普遍性的度量，表示词条的类别区分能力，用 q_i 表示，其计算公式为

$$q_i=\lg\frac{|D|}{|\{j:t_i\in d_j\}|} \tag{2}$$

式中，$|D|$ 为文本总数，分母为包含词条 t_i 的文本数量.

TF-IDF 值由 $f_{i,j}$ 值和 q_i 值相乘得到，用 $s_{i,j}$ 表示，其计算公式为

$$s_{i,j}=f_{i,j}q_i \tag{3}$$

如果某一文本内的高频词条在文本集合中呈现低频率，该词条便可在 TF-IDF 值上产生高权重，从而将其挑选出来作为区分度较高的词条.

1.2 LDA 模型

LDA 主题概率生成模型[11]是一种非监督学习的机器学习方法. 该模型分为词条、主题和文档3层结构，用以获取大规模文档中潜在的主题分布信息. 对于一篇文档，LDA 采用词袋模型将文本表示为词频向量，模型中词条出现的位置及先后顺序与最终得出的主题分布无关. LDA 是在 PLSA 模型的基础上加贝叶斯框架得到的，并且加入Dirichlet先验分布影响. LDA 模型结构如图 1 所示.

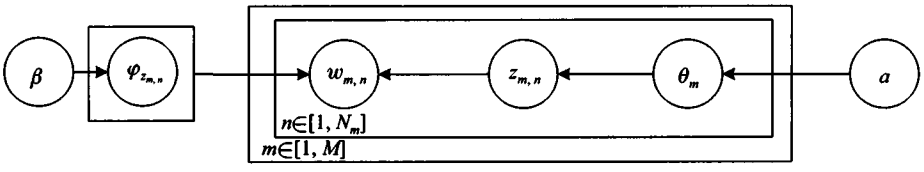

图 1 LDA 模型结构

Figure 1 LDA model structure

图 1 中，θ_m 表示文档 m 的主题分布，a 表示 θ_m 的先验分布，$z_{m,n}$ 表示从 θ_m 中取样生成文档 m 的第 n 个词的主题，$\varphi_{z_{m,n}}$ 表示词分布，β表示词分布的先验分布，$w_{m,n}$ 表示最终生成的第 m 篇文章的第 n 个词语，N_m 表示文档 m 中的词条总数，共有 M 篇文档.

LDA 中所有变量的联合分布计算公式为

$$p(w_m,z_m|a,\beta)=\sum_{n=1}^{N_m}p(w_{m,n}|z_{m,n},\beta)p(z_{m,n}|a) \tag{4}$$

式中，$p(w_{m,n}|z_{m,n})$ 为在主题下采样词条的概率. 在第 m 篇文档中各词条的概率分布公式为

$$p(w_{m,n})=\sum_{n=1}^{N}p(w_{m,n}|z_{m,n})p(z_{m,n}) \tag{5}$$

本文 LDA 的推导采用了 Gibbs 采样法，Gibbs 采样是马尔科夫蒙特卡罗理论中用来获取一系列近似等于指定多维概率分布观察样本的算法[11-12].

1.3 FastText

FastText[13-14]是2016年由Facebook公司推出的一款基于Word2Vec的快速而高效的文本分类器. 该分类器分为输入层、隐藏层和输出层：在输入层将输入的文本词语转换为向量作为输入，并加入了n-gram特征，使得语义信息更加完整；在输出层采用树形的层次Softmax（Hierarchical Softmax）[15]来代替扁平化的标准Softmax，大大提高了计算效率. 该模型结构如图2所示.

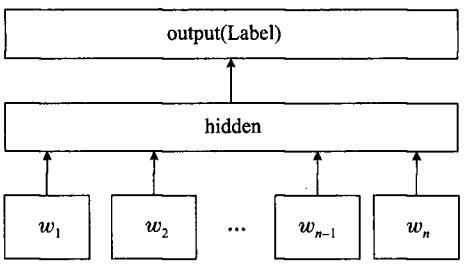

图 2 FastText模型框架

Figure 2 Frame of FastText model

图2中，FastText模型的输入$(w_1, w_2, \cdots, w_n)$是一个词序列或一段文本，隐藏层采用单层神经网络学习，输出的是这个词序列或文本属于不同类别标签的概率.

2 中文FastText分类方法

2.1 TL-FastText基本思想

FastText分类器中的n-gram处理对英文文本具有较好的学习效果，例如从“tabletennis”中可以学习到“tennis”是运动类文本，从“friendly”中学习到“friend”. 同时英文单词中包含着大量的“un”、“in”等英文前缀，这些前缀对英文文本的意思表达非常重要，且使用n-gram处理时可以学习到这些特征. 若将FastText分类器应用到中文文本中，则不但学习效果不如英文环境，而且会产生大量的冗余词条. 在处理英文文本后虽然也会出现大量的冗余词条，但英文环境可以学习到更多的特征. 中文仅在特定的一些词语中可以学习到特征，因而需要对n-gram处理后的词序列向量进行筛选，在构建保留词典时将无意义的词条和高频低区分度的词条过滤掉.

TF-IDF算法适用于过滤去除高频区分度低的词，保留重要的且区分度高的低频词. 所以本文使用TF-IDF算法来计算文本中各词条的TF-IDF值，将各词条按照TF-IDF值进行排序. 通过多次实验得到合适的阈值，文中阈值取值为0.3，选取TF-IDF值大于阈值的词条构建保留词典.

LDA主题模型可以挖掘出文档的主题，本文使用LDA主题模型对中文短文本构成的语料库整体进行主题挖掘，得出的主题词分布都是与主题有关的词条，具有较高的区分度. 因此，使用LDA主题挖掘的结果对保留词典进行特征补充可以取得更好的分类效果.

2.2 TL-FastText框架

经典的FastText由输入层、隐藏层和输出层组成. 在本文TL-FastText模型的输入层中添加计算模型，即先使用TF-IDF算法对n-gram处理后的输入词序列进行权重计算，得出保留词典；再由LDA主题模型对语料库整体进行主题分析进而得出主题词序列；最后将词典中的词与LDA结果进行对比，如果有相同的词，则将LDA结果中含有该词的主题词序列加入保留词典中，完成保留词典的重构，并将其送入隐藏层计算.

由于n-gram会产生大量的冗余词条，因此需要将无意义的词条和高频低区分度的词条过滤掉. 同时为了得到更好的学习效果，对n-gram过滤后的保留词序列进行特征补充. 补充的方式如下：使用LDA对语料库生成的文档进行主题分析得出主题词序列，然后从中选择符合匹配条件的主题词序列加入保留词典. TL-FastText模型结构如图3所示：

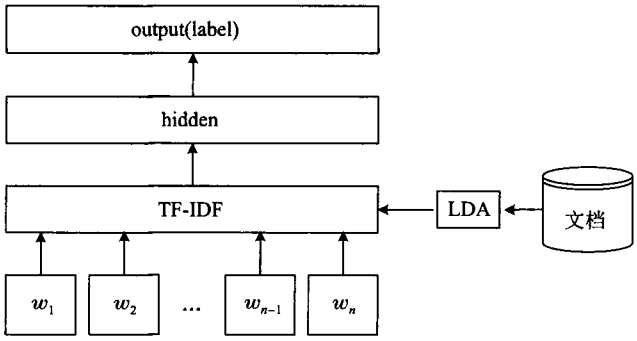

图 3 TL-FastText分类方法框架

Figure 3 Frame of TL-FastText text classification method

图3中，$(w_1, w_2, \cdots, w_n)$是一个词序列或一段文本，对其进行n-gram处理获得长度为N的字节片段序列. 对序列进行TF-IDF计算，筛选低频高区分度的词构成保留词典. 将词典中包含的词条与LDA结果进行对比，如果匹配到相同的词条，则将LDA结果中含有该词条的主题词序列加入到保留词典中，作为FastText模型的输入.

FastText模型使用输入的词序列来预测类别标签概率，将计算所得的词序列向量平均值作为文本表征向量. 因此，加入LDA主题词序列后的词序列均值将会向主题偏移，从而获得更好的分类效果. 具体公式为

$$p(L_{\mathrm{c}}|w_{\mathrm{o}1}, \cdots, w_{\mathrm{o}2m}) = \frac{\exp[U_{\mathrm{c}}^{\mathrm{T}}(V_{\mathrm{o}1} + \cdots + V_{\mathrm{o}2m})/2m]}{\sum\limits_{i \in V} \exp[U_i^{\mathrm{T}}(V_{\mathrm{o}1} + \cdots + V_{\mathrm{o}2m})/2m]} \tag{6}$$

式中，$w_{\mathrm{o}1}, w_{\mathrm{o}2}, \cdots, w_{\mathrm{o}2m}$表示输入的词序列，词典中的索引共有$2m$个词条，$L_{\mathrm{c}}$表示待预测的标签，$V_{\mathrm{o}1}, \cdots, V_{\mathrm{o}2m}$表示输入词的词语向量，$(V_{\mathrm{o}1} + \cdots + V_{\mathrm{o}2m})/2m$为输入词向量的平均值，$U_{\mathrm{c}}$为带预测标签的向量，$V$为文档词条总数.

模型层次之间的映射关系描述如下：将输入层中的词序列构成特征向量，再将特征向量通过线性变换映射到隐藏层. 该隐藏层通过求解最大似然函数后进行层次Softmax计算，即根据每个类别的权重和模型参数构建并输出哈夫曼树. 在构建哈夫曼树的过程中，每个叶子节点代表一个类别标签，在每一个非叶子节点处都需要做出走向左右分支的选择，走向节点左、右孩子的概率用逻辑回归公式表示，正类别概率σ_1如式(7)所示，负类别概率σ_2如式(8)所示：

$$\sigma_1 = \frac{1}{1 + \mathrm{e}^{-X_i\theta}} \tag{7}$$

$$\sigma_2 = 1 - \sigma_1 \tag{8}$$

式中，θ为哈夫曼树中间节点的参数. 每个类别标签在构造的哈夫曼树中都会有一条路径，对于训练样本的特征向量X_i和对应的类别标签Y_i在哈夫曼树中会有对应的路径. 而预测样

本 X_i 所属的类别，即为计算样本 X_i 属于所对应的类别标签 Y_i 的概率的计算公式为

$$P(Y_i|X_i)=\prod_{j=2}^{l}P(d_j|X_i,\theta_{j-1}) \tag{9}$$

将式(7)和(8)代入式(9)得

$$P(d_j|X_i,\theta_{j-1})=\begin{cases}\sigma_1, & d_j=1\\ \sigma_2, & d_j=0\end{cases} \tag{10}$$

2.3 TL-FastText 分类方法描述

TL-FastText 分类方法的具体步骤如下：

步骤1　按照时间窗口大小选取窗口内对应的数据集 d，对数据集 d 进行分词以及去除停用词等预处理操作，将LDA主题分析应用到预处理后的数据集上，从而得到该数据集的主题词分布 T. 按照FastText模型训练格式准备训练数据，在文本的结尾加标签 _label_ .

步骤2　对数据集 d 中每条短文本进行n-gram处理生成数据集 g；逐条计算 g 中每个数据项的TF-IDF值，筛选高区分度的词条构建每条短文本的保留词典 D.

步骤3　对保留词典 D 中的词与LDA主题分析的结果 T 进行对比，如果有相同的词，则将结果中含有该词的主题词序列加入保留词典 D 中，将保留词典 D 推入隐含层进行计算.

步骤4　本文处理的是文本分类问题，模型输出模式选择supervised方式，选择模型优化函数loss和梯度下降学习率 λ，采用随机梯度下降算法得到损失函数的梯度grad.

步骤5　根据步骤4设置的参数，输入的数据对为 (D, label, λ)，基于梯度下降算法对逐条数据进行相关梯度和权值的更新，直至所有数据训练完毕.

步骤6　模型预测时，测试集的各条文本根据重构后保留词典的词序列向量得到该条文本属于类别 i 的概率 P_i.

构建保留词典算法描述如下：

算法1 构建保留词典算法

Input short text data d
Output input dictionary D
Initialize threshold i
$T \leftarrow$ LDA analyzes the topic of d
for each text in d
　$g \leftarrow$ n-gram d
　for each term in g
　　$T_i \leftarrow$ Calculate term TF-IDF value
　　if ($T_i > i$) then
　　　add term to D
　end for
end for
if(Match D and T) then
　　add T to D
return D

在TL-FastText方法中，保留词典属于程序外接部分，而且词典中保留的均为文本数据，所占空间很小. 经测试，本文构建含有约10万条词汇的保留词典，所占空间为2.3 MB，一般应用环境中保留词典所占空间大小不会超过10 MB. 此外，保留词典在计算过程加入部分数据的同时，也删除了很多n-gram处理后没有意义的词. 因此，保留词典的引入对算法空间复杂度的影响是微小的.

3 实验分析

3.1 实验环境

实验环境为Intel Core i5-4460处理器、主频3.2GHz、内存8GB、1TB硬盘的PC机. 操作系统为Win7，编程语言使用Python，编译环境JetBrains PyCharm 2017.

3.2 实验数据及数据预处理

本文实验采集了75 740条新浪微博数据，在数据预处理阶段首先剔除图片、网址引用等空文本微博，对纯文本微博数据使用中科院分词工具ICTCLSA进行分词处理，对分词后的数据进行数据清洗，去除数据中无意义的语气词及表情符号等，将清洗后数据整合成文档. 数据集中的58 056条用于训练数据集，15 148条用于测试数据集. 包括明星、政治、体育、旅游、数码、汽车、游戏、宠物、时尚、财经共10个主题，如表1所示.

表1 各类别微博数量

Table 1 Number of microblogs in each category

主题	数量	主题	数量
明星	7 305	政治	6 427
体育	8 692	旅游	7 305
数码	7 850	汽车	7 563
游戏	7 365	宠物	6 387
时尚	8 529	财经	8 047

3.3 评价指标

本文实验采用精确率、召回率和F-measure 3个文本分类领域常用的指标来评价实验结果.

精确率是针对预测结果而言的，它表示预测为正的样本中有多少是真正的正样本. 预测为正有两种可能，一种是把正类预测为正类，此类样本数为T_{P}；另一种是把负类预测为正类，此类样本数为F_{P}，精确率P的计算公式为

$$P=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{P}}} \tag{11}$$

召回率R是针对原来的样本而言的，它表示样本中的正例有多少被正确预测. 有两种可能，一种是把原来的正类预测成正类，此类样本数为T_{P}；另一种就是把原来的正类预测为负类，此类样本数为F_{N}，召回率的计算公式为

$$R=\frac{T_{\mathrm{P}}}{T_{\mathrm{P}}+F_{\mathrm{N}}} \tag{12}$$

F-measure又称为F值，是用来评价分文分类效果的一种综合指标，计算方式为召回率和精确率的平均值，F值的计算公式为

$$F=\frac{2RP}{R+P} \tag{13}$$

3.4　实验结果与分析

本文用Python语言实现了FastText分类模型和TL-FastText模型. 在实验过程中，为了与FastText进行有效的对比，优先选用FastText在supervised模式下的默认参数：参数学习速率r_1=0.1，词向量维度d=200，上下文窗口大小w_s=5，迭代次数E_{epoch}=5，词语最小出现次数min_count=1. 损失函数loss的参数选用层次softmax，与原默认参数softmax相比，层次softmax作为损失函数的参数可以大幅度提高训练速度，同时可以提高精确率[13]. 因为中文的多数词语由两个字组成，因此词语级别n-gram最大长度word_ngrams=2，以适应中文短文本的特点.

本文实验对TL-FastText分类模型、经典FastText分类模型、基于TF-IDF特征的文本分类模型以及基于Word2Vec的文本分类模型在不同分类数量的分类效果进行对比. 本文在10分类、5分类和2分类上分别进行对比，4种算法在3种不同分类数量下的精确率、召回率和F值3项指标上的对比结果如图4~6所示.

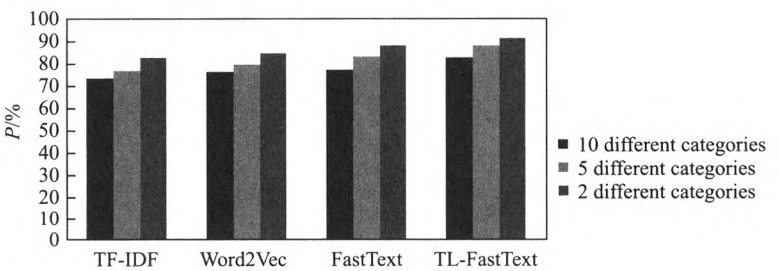

图 4　多分类精确率对比图

Figure 4　Precision comparison chart of multi-classification

图4中的数据表明，本文所提方法与其他3种对比方法（FastText、TF-IDF、Word2Vec）的精确率在2分类分别提高了3.32%、8.52%、6.73%；在5分类分别提高了4.52%、10.62%、8.11%；在10分类分别提高了5.21%、9.02%、6.29%.

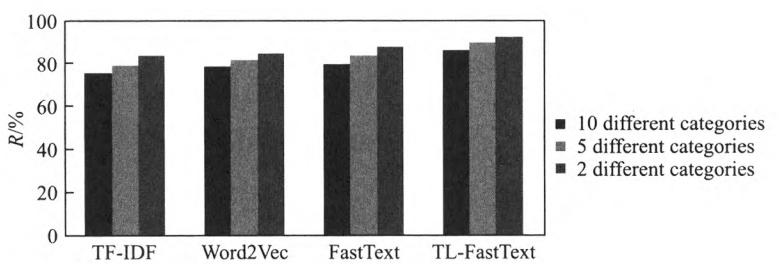

图 5　多分类召回率对比图

Figure 5　Recall ratio comparison chart of multi-classification

图5中的数据表明，本文所提方法与其他3种对比方法（FastText、TF-IDF、Word2Vec）

的召回率在2分类分别提高了4.95%、8.92%、7.83%；在5分类分别提高了5.92%、10.42%、8.04%；在10分类分别提高了6.45%、10.8%、7.65%.

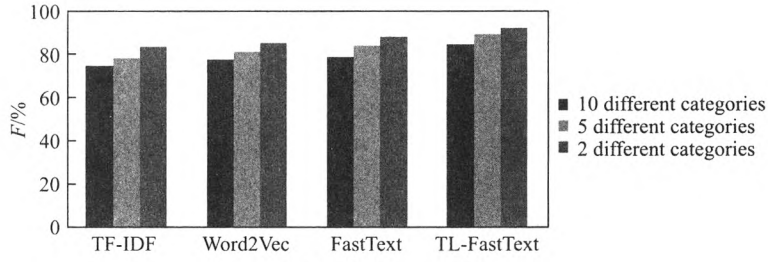

图6 多分类F值对比图

Figure 6 F value comparison chart of multi-classification

图6中的数据表明，本文所提方法与其他3种对比方法（FastText、TF-IDF、Word2Vec）的多分类F值在2分类分别提高了4.12%、8.71%、7.22%；在5分类分别提高了5.08%、10.63%、8.08%；在10分类分别提高了5.56%、9.9%、6.97%.

实验证明，TL-FastText在模型输入时对n-gram产生的词典进行了TF-IDF筛选和LDA主题词补充，使得在计算输入词序列向量均值偏向高区分度的词条，更适用于解决中文短文本分类问题，因此分类效果更佳.

为了验证所提算法的有效性，本文将4种算法应用在不同规模的训练集中进行实验，对4种算法的模型训练时间及模型预测时间进行对比，实验结果如图7和8所示.

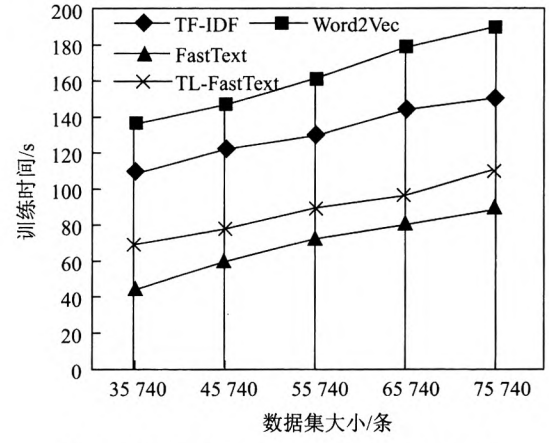

图7 不同数据集规模的模型训练时间对比

Figure 7 Comparison of model training time on different dataset scale

从图7可以看出，本文所提的TL-FastText分类方法与经典的FastText分类模型相比在模型训练时间上有一定的劣势，这是因为保留词典重构需要进行以下处理：1）在输入层对输入数据进行n-gram处理；2）对词典进行TF-IDF值计算，并且筛选后构建保留词典；3）使用TIF-LDA模型对数据集进行主题提取；4）将所得结果与保留词典进行对比，根据对比的结果对保留词典进行内容补充. 上述过程中的计算需要耗费一定的时间，因而导致本文方法的模型训练时间有所延长，但与基于TF-IDF特征的文本分类方法以及基于Word2Vec的文本分

类方法相比，本文方法在分类模型的训练时间上依然有很大的优势.

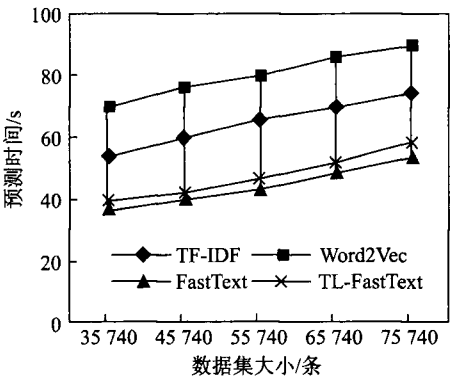

图 8　不同数据集规模的模型预测时间对比

Figure 8　Comparison of model prediction time on different dataset scale

从图8中可以看出，在相同规模数据集上的模型预测时间方面，当进行短文本分类时，本文的TL-FastText方法与经典FastText模型所用时间几乎没有差别，而且本文方法大幅度优于基于TF-IDF特征以及基于Word2Vec的文本分类方法.

4　结　语

本文对FastText模型进行了适合中文短文本环境的改进. 在改进过程中，使用TF-IDF对n-gram处理后的数据进行精简，去除高频低区分度的词条. 对词典进行了重构，同时使用LDA算法对所有短文本组成的文档进行LDA处理，得出主题词序列，将主题词添加到重构后的词典中，使得在计算输入词序列向量均值偏向高区分度的词条. 实验表明，本文提出的TL-FastText方法在中文短文本环境下的分类精确率有所提升.

参考文献:

[1] 段旭磊，张仰森，孙祎卓. 微博文本的句向量表示及相似度计算方法研究[J]. 计算机工程，2017, 43(5): 143-148.
DUAN X L, ZHANG Y S, SUN Y Z. Research on sentence vector representation and similarity calculation method about microblog texts [J]. Computer Engineering, 2017, 43(5): 143-148. (in Chinese)

[2] SPINELLIS D, RAPTIS K. Component mining: a process and its pattern language [J]. Information and Software Technology, 2000, 42(9): 609-617.

[3] 张谦，高章敏，刘嘉勇. 基于 Word2Vec 的微博短文本分类研究[J]. 信息网络安全，2017, 17(1): 57-62.
ZHANG Q, GAO Z M, LIU J Y. Research of weibo short text classification based on Word2Vec [J]. Netinfo Security, 2017, 17(1): 57-62. (in Chinese)

[4] 赵辉，刘怀亮. 一种基于维基百科的中文短文本分类算法[J]. 图书情报工作，2013, 57(11): 120-124.
ZHAO H, LIU H L. Classification algorithm of Chinese short texts based on Wikipedia [J]. Library and Information Service, 2013, 57(11): 120-124. (in Chinese)

[5] 范云杰，刘怀亮. 基于维基百科的中文短文本分类研究[J]. 现代图书情报技术，2012, 28(3): 47-52.
FAN Y J, LIU H L. Research on Chinese short text classification based on Wikipedia [J]. New Technology of Library and Information Service, 2012, 28(3): 47-52. (in Chinese)

[6] WU F L, ZHENG Y F. Adaptive normalized weighted KNN text classification based on PSO [J]. Scientific Bulletin of National Mining University, 2016, (1): 109-115.

[7] Liu J, Xu Y, Deng J, Wang L, Zhang L. Ld-CNNs: a deep learning system for structured text categorization based on LDA in content security [C]// International Conference on Network and System Security. Taiwan, 2016: 113-125.

[8] Bahassine S, Madani A, Kissi M. An improved Chi-square feature selection for Arabic text classification using decision tree [C]// International Conference on Intelligent Systems: Theories and Applications. Mohamrnedia, Morocco, IEEE, 2016: 2378-2536.

[9] 阳爱民，林江豪，周咏梅. 中文文本情感词典构建方法 [J]. 计算机科学与探索，2013, 7(11): 1033-1039.

Yang A M, Lin J H, Zhou Y M. Method on building Chinese text sentiment lexicon [J]. Journal of Frontiers of Computer Science and Technology, 2013, 7(11): 1033-1039. (in Chinese)

[10] 陈科文，张祖平，龙军. 文本分类中基于熵的词权重计算方法研究 [J]. 计算机科学与探索，2016, 10(9): 1299-1309.

Chen K W, Zhang Z P, Long J. Research on entropy-based term weighting methods in text categorization [J]. Journal of Frontiers of Computer Science and Technology, 2016, 10(9): 1299-1309. (in Chinese)

[11] Blei D M, Ng Y A, Jordan I M. Latent Dirichlet allocation [J]. Journal of Machine Learning Research, 2003, 3(1): 993-1022.

[12] Griffiths T L, Steyvers M. Finding scientific topics [C]// Proceedings of the National Academy of Sciences of the United States of America, 2004, 101(1): 5228-5235.

[13] Joulin A, Grave E, Bojanowski P, Mikolov T. Bag of tricks for efficient text classification [C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Spain, 2017: 427-431.

[14] Bojanowski P, Grave E, Joulin A, Mikolov T. Enriching word vectors with subword information [C]// Association for Computational Linguistics. Massachusetts, 2017: 135-146.

[15] Hinton G E, Salakhutdinov R. Replicated softmax: an undirected topic model [C]// International Conference on Neural Information Processing Systems. Canada, 2009: 1607-1614.

(编辑：管玉娟)