计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(5):209−215 [doi: 10.15888/j.cnki.csa.006325]
©中国科学院软件研究所版权所有.
E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 Word2Vec 的中文短文本分类问题研究①

汪　静, 罗　浪, 王德强

(中南民族大学 计算机科学学院, 武汉 430074)
通讯作者: 汪　静, E-mail: wjchrista@163.com

摘　要: 针对短文本中固有的特征稀疏以及传统分类模型存在的“词汇鸿沟”等问题, 我们利用 Word2Vec 模型可以有效缓解短文本中数据特征稀疏的问题, 并且引入传统文本分类模型中不具有的语义关系. 但进一步发现单纯利用 Word2Vec 模型忽略了不同词性的词语对短文本的影响力, 因此引入词性改进特征权重计算方法, 将词性对文本分类的贡献度嵌入到传统的 TF-IDF 算法中计算短文本中词的权重, 并结合 Word2Vec 词向量生成短文本向量, 最后利用 SVM 实现短文本分类. 在复旦大学中文文本分类语料库上的实验结果验证了该方法的有效性.

关键词: Word2Vec; TF-IDF; 文本表示; 短文本分类

引用格式: 汪静,罗浪,王德强.基于 Word2Vec 的中文短文本分类问题研究.计算机系统应用,2018,27(5):209–215. http://www.c-s-a.org.cn/1003-3254/6325.html

Research on Chinese Short Text Classification Based on Word2Vec

WANG Jing, LUO Lang, WANG De-Qiang

(School of Computer Science, South-Central University for Nationalities, Wuhan 430074, China)

Abstract: To address the problems such as the inherent sparsity in the short text and the “lexical gap” of traditional classification model, using Word2Vec model to map words to a spatial vector of low-dimensional real number according to context semantic relations can effectively ease the sparse feature issue of short text. However, further study found that only using Word2Vec will ignore the influence of different parts of speech on the short text. Therefore, we introduce part of speech to improve the feature weighting approach, in which the contribution of speech is embedded into the traditional TF-IDF algorithm to calculate the weight of the words in the short text, and the vector of short text is generated by combining the word vector of Word2Vec. Finally, we use the SVM to achieve short text classification. Experimental results on Fudan University Chinese text classification corpus validate the effectiveness of the proposed method.

Key words: Word2Vec; TF-IDF; text representation; short text classification

1　引言

移动终端的智能化和互联网技术的高速发展促使人们在移动互联网上交流的越来越频繁, 由此产生了大量的信息数据[1], 这些数据多以短文本的形式作为信息传递的载体, 例如微博和即时推送新闻等, 其内容通常都是简洁精炼并且含义概括, 具有很高的研究价值. 因此, 如何通过机器对这些短文本内容进行自动分类以及对短文本所具有的丰富含义进行有效的理解鉴别已经成为自然语言处理和机器学习领域研究的热点和难点[2].

短文本自动分类首先需要将文本转化为计算机能理解处理的形式, 即文本数据的表示, 其对文本分类至关重要, 可直接影响分类效果. 传统的文本表示方法主要基于空间向量模型 (Vector Space Model, VSM), 俗称

① 基金项目: 赛尔网络下一代互联网技术创新项目 (NGII20150106)
收稿时间: 2017-08-18; 修改时间: 2017-09-05; 采用时间: 2017-09-18; csa 在线出版时间: 2018-03-12

© 中国科学院软件研究所 http://www.c-s-a.org.cn

词袋模型[3], 该方法认为文档是无序词项的集合, 丢弃了词序、语法等文档结构信息, 忽略了词语间的语义关系, 存在数据高维稀疏问题, 对分类效果的提升存在瓶颈. 于是一些学者引入外部的知识库 (如搜索引擎、维基百科等) 对文本进行特征扩展, 丰富词语间语义关系[4,5], 但其严重依赖外部知识库的质量, 对于知识库中未收录的概念无能为力且计算量大、耗时长. 另有部分学者挖掘文本潜在的语义结构[6], 生成主题模型如 LSA, pLSI 和 LDA[7–9], 但模型构建属于“文本”层面, 缺少细节性研究. 因此短文本的表示方法还有待研究.

Bengio 在 2003 年首次提出了神经网络语言模型 (Neural Network Language Model, NNLM), 但由于其神经网络结构相对较复杂, 许多学者在其基础上进行改进优化, 最具代表性之一的当属 T. Mikolov 等人在 2013 年基于神经网络提出的 Word2Vec 模型[10]. Word2Vec 模型通过对词语的上下文及词语与上下文的语义关系进行建模, 将词语映射到一个抽象的低维实数空间, 生成对应的词向量. Word2vec 词向量的维度通常在 100~300 维之间, 每一维都代表了词的浅层语义特征[11], 通过向量之间的距离反映词语之间的相似度, 这使得 Word2Vec 模型生成的词向量广泛应用于自然语言处理 (Natural Language Processing, NLP) 的各类任务中, 如中文分词[12], POS 标注[13], 文本分类[14,15], 语法依赖关系分析[16]等.与传统的空间向量文本表示模型相比, 使用词向量表示文本, 既能解决传统向量空间模型的特征高维稀疏问题, 还能引入传统模型不具有的语义特征解决“词汇鸿沟”问题, 有助于短文本分类[17]. 但如何利用词向量有效表示短文本是当前的一个难点, 目前在这方面的研究进展缓慢, 常见的方法有对短文本所包含的所有词向量求平均值[18], 但却忽略了单个词向量对文本表示的重要程度不同, 对短文本向量的表示并不准确. Quoc Le 和 Tomas Mikolov[19]在 2014 年提出的 Doc2Vec 方法在句子训练过程中加入段落 ID, 在句子的训练过程中共享同一个段落向量, 但其效果与 Word2Vec 模型的效果相当, 甚至有时训练效果不如 Word2Vec 模型. 唐明等人[20]注重单个单词对整篇文档的影响力, 利用 TF-IDF 算法计算文档中词的权重, 结合 Word2Vec 词向量生成文档向量, 但其单纯以词频作为权重考虑因素太单一, 生成文本向量精确度不够, 未考虑文本中所含有的利于文本分类的因素的重要性, 比如名词、动词等不同词性的词对于文本内容的反映程度是不同的, 词性对于特征词语的权重应该也是有影响的. 在上述研究的基础上, 考虑到不同词性的词语对短文本分类的贡献度不同, 引入基于词性的贡献因子与 TF-IDF 算法结合作为词向量的权重对短文本中的词向量进行加权求和, 并在复旦大学中文文本分类语料库上进行测试, 测试结果验证了改进方法的有效性.

2 相关工作

短文本自动分类是在预定义的分类体系下, 让计算机根据短文本的特征 (词条或短语) 确定与它关联的类别, 是一个有监督的学习过程. 在自动文本分类领域常用的技术有朴素贝叶斯分类器 (Navie Bayes Classifier)、支持向量机 (Support Vector Machine, SVM)、K 进邻算法 (KNN) 等. 本文提出的短文本分类算法结合 Word2Vec 和改进的 TF-IDF 两种模型.

2.1 Word2Vec 模型

Word2Vec 模型包含了 Continuous Bag of Word (CBOW) 和 Skip-gram 两种训练模型, 这两种模型类似于 NNLM, 区别在于 NNLM 是为了训练语言模型, 词向量只是作为一个副产品同时得到, 而 CBOW 和 Skip-gram 模型的直接目的就是得到高质量的词向量, 且简化训练步骤优化合成方式, 直接降低了运算复杂度. 两种模型都包括输入层、投影层、输出层, 其中 CBOW 模型利用词 w_t 的上下文 w_{ct} 去预测给定词 w_t, 而 Skip-gram 模型是在已知给定词 w_t 的前提下预测该词的上下文 w_{ct}. 上下文 w_{ct} 的定义如公式 (1) 所示:

$$w_t = w_{t-c}, \cdots, w_{t-1}, w_{t+1}, \cdots, w_{t+c} \tag{1}$$

其中 c 是给定词 w_t 的前后词语数目. CBOW 模型和 Skip-gram 模型的优化目标函数分别为公式 (2) 和公式 (3) 的对数似然函数:

$$L_{\text{CBOW}} = \sum_{wt \in C} \log p(w_t|w_{ct}) \tag{2}$$

$$L_{\text{Skip-gram}} = \sum_{wt \in C} \sum_{-k \le c \le k} \log p(w_{t+c}|w_t) \tag{3}$$

其中 C 代表包含所有词语的语料库, k 代表当前词 w_t 的窗口大小, 即取当前词的前后各 k 个词语. 针对 NNLM 输出层采用 Softmax 函数进行归一化处理计算复杂度较大的问题, Word2Vec 模型结合赫夫曼编码的 Hierachical Softmax 算法和负采样 (Negative Sampling)

© 中国科学院软件研究所 http://www.c-s-a.org.cn

技术对式中的条件概率函数$p(w_t|w_{ct})$及$p(w_{t+c}|w_t)$的构造进行优化, 处理如公式 (4) 所示, v_w 和 v_w'分别代表词 w 的输入输出词向量, W 代表词典大小. 之后采用随机梯度下降算法对模型的最优参数进行求解.

$$p(w_O|w_I)=\frac{\exp(v'^{\mathrm{T}}_{wO}v_{wI})}{\sum_{w\in W}\exp(v'^{\mathrm{T}}_{wO}v_{wI})} \tag{4}$$

当模型训练完成时即可得到所有词语对应的词向量, 发现词向量间往往存在类似的规律: $v_{\mathrm{king}}-v_{\mathrm{man}}+v_{\mathrm{woman}}=v_{\mathrm{queen}}$, 由此可以看出 Word2Vec 模型对语义特征的有效表达.

2.2 TF-IDF 模型

词频与逆文档频率 (Term Frequency-inverse Document Frequency, TF-IDF) 是一种统计方法, 用以评估词语对于一份文件或者一个文件集的重要程度. 字词的重要性随着它在文件中出现的次数成正比增加, 但同时会随着它在语料库中出现的频率成反比下降. 通俗表达的意思是如果某个词或短语在一个类别中出现的频率较高, 并且在其他类别中很少出现, 则认为此词或者短语具有很好的类别区分能力, 适合用来分类[21]. TF-IDF 由词频和逆文档频率两部分统计数据组合而成, 即 TF×IDF. 词频 (Term Frequency, TF) 指的是某一个给定的词语在该文档中出现的频率, 计算公式如 (5) 所示:

$$tf_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}} \tag{5}$$

其中 $n_{i,j}$ 表示词语 t_i 在文档 d_j 中的出现次数, 分母则是在文档 d_j 中所有字词的出现次数之和, k 代表文档 d_j 中的总词数. 已知语料库中的文档总数, 一个词语的逆向文件频率 (Inverse Document Frequency, IDF) 可由总文档数目除以包含有该词语的文档的数目得到, 计算公式如 (6) 所示:

$$idf_i=\log\frac{|D|}{|\{j:t_i\in d_j\}|} \tag{6}$$

其中$|D|$表示语料库中的文档总数, $|\{j:t_i\in d_j\}|$代表包含词语 t_i 的文档数目 (即$n_{i,j}\neq 0$的文档数目), 如果该词语不在语料库中会导致分母为零, 因此一般情况下使用 $|1+\{j:t_i\in d_j\}|$. 由此可得出词语 t_i 的 TF-IDF 权重归一化后的计算公式如 (7) 所示:

$$tf-idf_i=\frac{tf_{i,j}\times idf_i}{\sqrt{\sum_{t_i\in d_j}[tf_{i,j}\times idf_i]^2}} \tag{7}$$

3 基于改进的 TF-IDF 算法的加权 Word2Vec 分类模型

短文本分类的关键在于短文本的表示, 一般的做法是利用向量空间模型将文档表示为 TF-IDF 的加权向量, 但这样得到的短文本向量往往有特征高维稀疏等问题. Word2Vec 模型提供了一种独特的方法解决特征稀疏问题, 而且引入了语义特征, 能训练出更加丰富准确的词向量, 短文本向量即可由这些包含语义关系的词向量表示.

在 Word2Vec 词向量的基础上, 结合改进的 TF-IDF 算法即 PTF-IDF 算法提出了短文本向量的表示方法及短文本分类模型.

3.1 PTF-IDF

传统的 TF-IDF 权重计算方法用于短文本分类时是将文档集作为整体考虑的, 未体现出词性对短文本分类的影响程度, 但在实际的分类过程中, 不同词性的词语对短文本分类的贡献度和影响力是不同的. 因此, 本文考虑在 TF-IDF 基础上根据词语的词性引入一个贡献因子, 通过调整词性的特征权重, 减少噪音项的干扰, 凸显特征词的重要程度, 使得不同类的短文本差别更明显.

通过已有的研究可以了解到, 名词、动词对文本内容的反映程度更强, 更能表征文本的主题, 而形容词、副词次之, 其他词性的词对于短文本分类的贡献更小. 文献[22]更是直接指出中文短文本主要依靠名词、动词、形容词、副词 4 种词性进行表达, 文中给出了各种词性的词语对短文本内容的表征能力, 其中动词和名词对短文本内容的表征能力最强, 对分类类别具有更高的贡献度. 基于此, 本文引入基于词性的贡献因子与 TF-IDF 算法结合作为词向量的权重, 改进的 TF-IDF 算法 (PTF-IDF 算法) 计算公式如 (8) 所示:

$$PTF-IDF_i=tf-idf_i\cdot e$$
$$e=\begin{cases}\alpha, \text{当}t_i\text{为名词或者动词}\\ \beta, \text{当}t_i\text{为形容词或者副词}\\ \gamma, \text{当}t_i\text{为其他词性的词项}\end{cases} \tag{8}$$

式中, t_i 表示短文本中的当前词, e 即为根据当前词的词性所分配的权重系数, 且满足$1>\alpha>\beta>\gamma>0$, $tf-idf_i$ 即为公式 (7).

3.2 Word2Vec 模型结合 PTF-IDF 算法表示短文本

将 Word2Vec 模型应用于文本分类解决了传统空

© 中国科学院软件研究所 http://www.c-s-a.org.cn

间向量模型的特征高维稀疏问题和“词汇鸿沟”问题, 但鉴于短文本具有篇幅短小、组成文本的特征词少等不同于长文本的特点, 单个词语的重要程度显得尤为重要, 因此与引入了词性贡献因子的 PTF-IDF 算法结合, 借助 PTF-IDF 算法从词频和词性两方面计算短文本中词汇的权重.

Mikolov 在文献[10]中指出词向量的学习不仅能学习到其语法特征, 还能利用向量相加减的方式进行语义上面的计算. 为了突出单个词语对文本内容的影响, 考虑其词频、词性特征作为权重, 可直接对短文本中的词语进行加权求和. 在分类效果相差不大的情况下, 相比于通过神经网络构建短文本向量具有较高的复杂度, 加权求和构造短文本向量数学模型构造简单且更容易理解. 对于每篇短文本$Dj(j=1,2,3,\cdots,m)$, 其短文本向量可以表示为如 (9) 所示的形式:

$$d_j=\sum_{t\in D_j} w_t\cdot PTF-IDF_i \tag{9}$$

其中, w_i 表示分词 t_i 经过 Word2Vec 模型训练出来的词向量, 通常将词向量的维数定为 200, 因此短文本向量同样是 200 维, 大大减少了分类过程中的计算量. $PTF-IDF_i$即为词语 t_i 引入了词性贡献因子的 PTF-IDF 权重, Word2Vec 词向量乘以对应的 PTF-IDF 权重得到加权 Word2Vec 词向量. 累加短文本中词语的加权 Word2Vec 词向量, 得到短文本向量 d_j.

3.3 短文本分类的工作流程

对未知短文本的分类过程如图 1 所示. 首先利用 Word2Vec 模型对大型分好词的语料库进行训练, 将所有词语根据其上下文语义关系映射到一个低维实数的空间向量, 即可获得每个词语对应的 Word2Vec 词向量. 利用结巴分词工具对训练集中的短文本进行分词并与训练 Word2Vec 模型得到的词向量一一对应. 结巴分词工具同样可以对分好的词语进行词性标注, 根据词语的词频和词性计算 PTF-IDF 值, 与 Word2Vec 词向量结合进行加权求和得到短文本向量.

很多研究表明, 与其他分类系统相比, SVM 在分类性能上和系统健壮性上表现出很大优势[23], 因此实验选用 SVM 作为分类工具, 根据短文本向量及其对应的标签训练出分类器. 测试过程与训练过程相似, 只是最后通过已训练好的分类器预测测试短文本的标签.

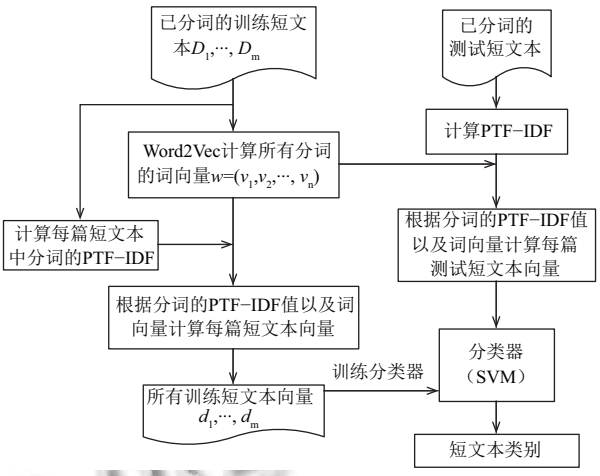

图 1 短文本分类的工作流程

4 实验

4.1 实验数据

实验数据集来自于由复旦大学计算机信息与技术系国际数据库中心自然语言处理小组收集发布的文本分类数据集. 原始数据集共 20 个分类, 包含文本 9804 篇文档, 每个类别中所包含的文档数量不等. 本文选取其中文档数量大于 500 的类别参与实验, 一共包含 3435 篇文档, 分别是艺术类、农业类、经济类和政治类, 每个分类下的文档数量如表 1 所示. 从中抽取新闻标题作为中文短文本分类数据集, 并把数据集随机划分成 5 份, 每次取其中 4 份作为训练集, 1 份作为测试集, 然后把 5 次分类结果的平均值作为最终结果. 所有 20 个类别的正文内容用 Word2Vec 模型训练词向量.

表 1 数据集各类别文档数量

	艺术类	农业类	经济类	政治类
文档数量	688	544	1487	716

4.2 分类性能评价指标

分类任务的常用评价指标有准确率 (Precision), 召回率 (Recall) 和调和平均值 $F1$. 其中准确率 P 是指分类结果中被正确分类的样本个数与所有分类样本数的比例. 召回率 R 是指分类结果中被正确分类的样本个数与该类的实际文本数的比例. $F1$ 是综合考虑准确率与召回率的一种评价标准. 计算公式分别如下所示:

$$P=\frac{A}{A+B} \tag{10}$$

© 中国科学院软件研究所 http://www.c-s-a.org.cn

$$R=\frac{A}{A+C} \tag{11}$$

$$F1=\frac{2PR}{P+R} \tag{12}$$

各参数含义如表 2 所示.

表 2 分类评价指标参数含义表

	分类为c类	分类非c类
实际为c类	A	C
实际非c类	B	D

4.3 PTF-IDF 算法的权重系数确定

本文提出的分类模型在短文本分类问题上的准确率受 PTF-IDF 权重系数的影响较大. 为得到较好的分类效果, 需要确定 PTF-IDF 算法中的最优权重系数. 当设置不同权重系数时, 基于 Word2Vec 模型与 PTF-IDF 算法结合表示的短文本向量在 SVM 分类器中的分类效果不同, 选取分类效果最好即 $F1$ 值最大时的系数值作为 PTF-IDF 算法的权重系数.

由于动词和名词对短文本内容的表征能力最强, 因此实验中将名词或者动词的权重系数 α 从 0.5 开始取值, 按$1>\alpha>\beta>\gamma$的规则, 采用三重循环依次以 0.1 的步长增大α, β, γ的值. 部分实验结果如表 3 所示.

表 3 F1 值与权重系数关系

α	β	γ	$F1$(%)	α	β	γ	$F1$(%)
0.5	0.4	0.3	92.15	0.7	0.6	0.5	92.03
0.5	0.4	0.1	90.12	0.7	0.6	0.1	89.39
0.5	0.3	0.2	91.52	0.7	0.5	0.2	91.26
0.5	0.2	0.1	89.98	0.7	0.4	0.3	91.99
0.6	0.5	0.3	90.03	0.7	0.3	0.1	89.79
0.6	0.5	0.1	89.96	0.8	0.7	0.5	90.38
0.6	0.4	0.1	91.72	0.8	0.6	0.4	91.70
0.6	0.3	0.2	93.01	0.8	0.4	0.2	92.79
0.6	0.2	0.1	90.08	0.8	0.2	0.1	88.98

实验结果显示当α, β, γ分别取 0.6、0.3、0.2 时, 分类效果最好, $F1$ 值可达 93.01%. 当α, β, γ取 0.8、0.4、0.2 时其次, $F1$ 值也达到 92.79%, 而当α, β, γ三者系数相近时, 如α, β, γ分别取 0.5、0.4、0.3 和 0.7、0.6、0.5 时类似于原 TF-IDF 算法与 Word2Vec 词向量加权求和, 分类效果适中, 由此也验证了引入词性贡献因子改进 TF-IDF 算法对短文本分类的有效性. 但并不是所有的词性贡献因子的组合都能取得不错的效果, 当过分看重名词和动词的权重而忽略其他词性的贡献度时结果反而差强人意. 因此通过合理调整词性贡献因子组合, 获得最优的词向量权重系数, 可以提升短文本的分类效果.

4.4 实验对比与分析

本文将分别使用 TF-IDF、均值 Word2Vec、TF-IDF 加权 Word2Vec 以及 PTF-IDF 加权 Word2Vec 四种模型对实验数据集中的新闻标题进行分类.

对于 TF-IDF 分类模型, 使用 Scikit-learn 提供的 TfidfVectorizer 模块提取文本特征并将短文本向量化. 均值 Word2Vec 模型是计算一篇短文本中所有通过 Word2Vec 模型训练出的 Word2Vec 词向量的均值. TF-IDF 加权 Word2Vec 模型是将短文本中词向量和对应词汇的 TF-IDF 权重相乘得到的加权 Word2Vec 词向量, 累加加权词向量得到加权短文本向量化表示. PTF-IDF 加权 Word2Vec 模型与 TF-IDF 加权 Word2Vec 模型类似, 只是引入词性贡献因子改进 TF-IDF 算法, 综合考虑词性与词频为词向量赋予不同的权重, 根据 4.3 小节中权重系数确定的实验, 将 α, β, γ 分别设置为 0.6、0.3、0.2.

实验中分类算法均使用 Scikit-learn 提供的 LinearSVM 算法, 所有实验采用五分交叉验证, 测试结果用准确率 (P)、召回率 (R)、$F1$ 指标进行测评, 测试结果如表 4–表 7 所列. 其中类别 C1、C2、C3、C4 分别代表艺术类、农业类、经济类、政治类, avg 代表 C1–C4 的平均值.

表 4 TF-IDF 模型 (单位: %)

类别	P	R	$F1$
C1	88.00	88.89	88.44
C2	90.65	93.03	91.82
C3	94.24	86.75	90.34
C4	86.35	92.11	89.14
avg	89.81	90.20	89.94

表 5 均值 Word2Vec 模型 (单位: %)

类别	P	R	$F1$
C1	86.49	91.43	88.89
C2	95.95	89.44	92.58
C3	88.54	92.98	90.71
C4	96.18	89.36	92.64
avg	91.79	90.80	91.21

表 6 TF-IDF 加权 Word2Vec 模型 (单位: %)

类别	P	R	$F1$
C1	94.78	87.20	90.83
C2	91.67	94.78	93.20
C3	89.84	96.14	92.88
C4	99.19	85.92	92.08
avg	93.87	91.01	92.25

© 中国科学院软件研究所 http://www.c-s-a.org.cn

表 7 PTF-IDF 加权 Word2Vec 模型 (单位: %)

类别	*P*	*R*	*F*1
C1	91.76	93.91	92.82
C2	91.43	94.12	92.76
C3	94.04	92.73	93.38
C4	95.29	92.30	93.77
avg	93.13	93.27	93.18

由表 4–表 7 的实验结果可以发现, 均值 Word2Vec 模型在 SVM 分类器上的准确率、召回率以及 *F*1 值比 TF-IDF 模型稍有提升, 由此也验证了 Word2Vec 模型应用于短文本分类的可行性以及 Word2Vec 模型所生成的词向量比传统模型所生成的词向量更能有效地表示文档特征.

基于 TF-IDF 加权的 Word2Vec 模型相比均值 Word2Vec 模型又有所提高, 在 SVM 分类器上所有类别的平均准确率、召回率、*F*1 值分别提升了 2.08%, 0.21%, 1.04%. 这归因于 TF-IDF 权重可以权衡 Word2Vec 模型生成的每个词向量在短文本中的重要性, TF-IDF 加权的 Word2Vec 词向量使用于文本分类的短文本表示更合理准确.

本文提出的引入词性贡献因子的 PTF-IDF 加权 Word2Vec 模型较对比的分类模型效果最好, 由图 2 也可以清楚地看出, 基于 PTF-IDF 加权的 Word2Vec 模型在多数类别上均有不错的表现, 所有类别的平均 *F*1 值验证了所提出的基于 Word2Vec 的 PTF-IDF 加权求和计算短文本向量表示方法在短文本分类方面的有效性.

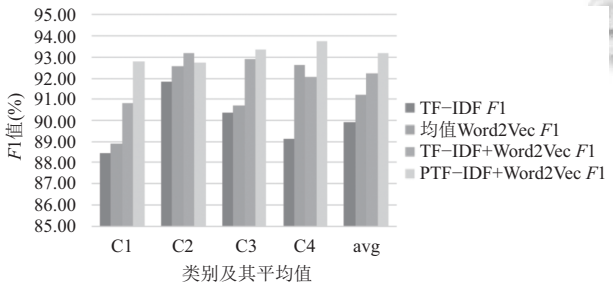

图 2 4 种短文本向量表示模型分类效果比较

5 结束语

针对当前短文本向量表示方法的不足, 借助 Word2Vec 模型的优点, 将 Word2Vec 模型与引入词性贡献因子的改进 TF-IDF 算法结合, 综合考虑词频和词性特征, 提出了一种基于 Word2Vec 的 PTF-IDF 加权求和计算短文本向量算法, 并应用于短文本分类问题, 在复旦大学中文文本分类语料库上的实验表明, 相较于传统的 TF-IDF 模型、均值 Word2Vec 模型以及 TD-IDF 加权 Word2Vec 模型, 本算法模型有更好的短文本分类效果. 但文章也有一些不足之处, 数据集较少, 实验中采用的类别不够丰富, 后续可在多个数据集上进行验证, 加强所提算法模型的可移植性; 在进行短文本向量表示时只是简单加权求和, 未考虑词与词之间的顺序及位置关系, 有待后续进一步的研究和实验.

参考文献

1 Manyika J, Chui M, Brown B, *et al*. Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute. https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation. [2015-09-05].

2 余凯, 贾磊, 陈雨强. 深度学习: 推进人工智能的梦想. 程序员, 2013, (6): 22–27.

3 Ling W, Luís T, Marujo L, *et al*. Finding function in form: Compositional character models for open vocabulary word representation. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal. 2015. 1520–1530.

4 朱征宇, 孙俊华. 改进的基于《知网》的词汇语义相似度计算. 计算机应用, 2013, 33(8): 2276–2279, 2288.

5 王荣波, 谌志群, 周建政, 等. 基于 Wikipedia 的短文本语义相关度计算方法. 计算机应用与软件, 2015, 32(1): 82–85, 92.

6 Rubin TN, Chambers A, Smyth P, *et al*. Statistical topic models for multi-label document classification. Machine Learning, 2012, 88(1-2): 157 –208. [doi: 10.1007/s10994-011-5272-5]

7 Dumais ST. Latent semantic analysis. Annual Review of Information Science and Technology, 2004, 38(1): 188–230.

8 Hofmann T. Probabilistic latent semantic indexing. Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Berkeley, CA, USA. 1999. 50–57.

9 Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. J Machine Learning Research Archive, 2003, (3): 993–1022.

10 Mikolov T, Sutskever I, Chen K, *et al*. Distributed representations of words and phrases and their compositionality. Proceedings of the 26th International Conference on Neural Information Processing Systems. Lake Tahoe, NV, USA. 2013. 3111–3119.

© 中国科学院软件研究所 http://www.c-s-a.org.cn

11 Zheng XQ, Chen HY, Xu TY. Deep learning for Chinese word segmentation and POS tagging. Proceedings of 2013 Conference on Empirical Methods in Natural Language Processing. Seattle, WA, USA. 2013. 647-657.

12 Tang DY, Wei FR, Yang N, *et al*. Learning sentiment-specific word embedding for twitter sentiment classification. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Baltimore, MD, USA. 2014. 1555–1565.

13 Kim HK, Kim H, Cho S. Bag-of-concepts: Comprehending document representation through clustering words in distributed representation. Neurocomputing, 2017, (266): 336–352. [doi: 10.1016/j.neucom.2017.05.046]

14 Socher R, Bauer J, Manning CD, *et al*. Parsing with compositional vector grammars. Proceedings of the 51st Meeting of the Association for Computational Linguistics. Sofia, Bulgaria. 2013. 455–465.

15 Lilleberg J, Zhu Y, Zhang YQ. Support vector machines and Word2vec for text classification with semantic features. Proceedings of the IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing. Beijing, China. 2015. 136–140.

16 Xing C, Wang D, Zhang XW, *et al*. Document classification with distributions of word vectors. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). Siem Reap, Cambodia. 2014. 1–5.

17 Le QV, Mikolov T. Distributed representations of sentences and documents. Proceedings of the 31st International Conference on Machine Learning. Beijing, China. 2014. 1188–1196.

18 唐明, 朱磊, 邹显春. 基于 Word2Vec 的一种文档向量表示. 计算机科学, 2016, 43(6): 214–217, 269. [doi: 10.11896/j.issn.1002-137X.2016.06.043]

19 Turian J, Ratinov L, Bengio Y. Word representations: A simple and general method for semi-supervised learning. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Uppsala, Sweden. 2010. 384–394.

20 Sun YM, Lin L, Yang N, *et al*. Radical-enhanced Chinese character embedding. In: Loo CK, Yap KS, Wong KW, *et al*. eds. Neural Information Processing. Cham: Springer, 2014, (8835): 279–286.

21 张玉芳, 彭时名, 吕佳. 基于文本分类 TFIDF 方法的改进与应用. 计算机工程, 2006, 32(19): 76–78. [doi: 10.3969/j.issn.1000-3428.2006.19.028]

22 黄贤英, 张金鹏, 刘英涛, 等. 基于词项语义映射的短文本相似度算法. 计算机工程与设计, 2015, 36(6): 1514–1518, 1534.

23 李玲俐. 数据挖掘中分类算法综述. 重庆师范大学学报 (自然科学版), 2011, 28(4): 44–47.

© 中国科学院软件研究所 http://www.c-s-a.org.cn