山西大学学报(自然科学版)37(4):595－600,2014
Journal of Shanxi University(Nat. Sci. Ed.)

文章编号:0253-2395(2014)04-0595-06

一种面向聚类的文本建模方法

唐晓丽,白宇*,张桂平,蔡东风
(沈阳航空航天大学 知识工程研究中心,沈阳 110136)

摘　要:借鉴主题模型的思想,利用 word2vec 训练数据的高效性以及词聚类结果的有效性,提出了一种基于 word2vec 的文本建模方法。该方法以 word2vec 算法得到的词聚类结果为基础,统计文本在词聚类类别上的概率分布,获得文本在类别空间上的特征向量,完成文本建模。将其与两种经典的文本建模方法 VSM 和 LDA 进行比较,实验结果显示在聚类效果上 F 值分别提高 6.01%、1.01%,在算法效率上有明显的提高。

关键词:主题模型;word2vec;文本建模;文本聚类
中图分类号:TP391　　**文献标志码**:A　　**DOI**:10.13451/j.cnki.shanxi.univ(nat.sci.).2014.04.019

A Text Modeling Method for Clustering

TANG Xiaoli,BAI Yu,ZHANG Guiping,CAI Dongfeng
(*Research Center for Knowledge Engineering,SAU,Shenyang* 110136,*China*)

Abstract:By referencing the idea of topic model,this paper proposed a word2vec-based approach that can model text by utilizing high efficient data training and valid clustering result of word2vec. This method represented a text as a mixed distribution of a category set on the basis of word clustering result of word2vec,and then obtained the feature vector of the text. Compared with two classical text modeling methods VSM and LDA,the experimental results show that F-value improves by 6.01% and 1.01% respectively on clustering result and the proposal method has obvious improvement on algorithm efficiency.
Key words:topic model; word2vec; text modeling; text clustering

0　引言

随着 Web 信息的爆炸式增长,计算机的信息处理已进入大数据时代。信息融合技术为 Web 信息处理提供了新的方法。信息融合[1]借鉴人脑的工作原理,利用计算机对具有相似或不同特征的多源数据和信息进行处理,为用户提供统一的信息视图和可综合利用的信息。文本形式是 Web 信息呈现的方式之一,而对文本信息的融合主要涉及两方面的问题:一是文本信息的挖掘,二是文本信息的组织。可见,文本挖掘是进行文本信息融合的前提与基础。

文本聚类[2]是文本挖掘的重要手段和方法。聚类过程主要通过三步来完成:(1)文本建模即文本表示;(2)文本相似度计算;(3)聚类算法的选择。可见,文本建模是聚类过程的第一步,是影响文本聚类效果的重要因素。当前,主要采用向量空间模型和主题模型来进行文本建模。

向量空间模型(Vector Space Model,VSM)[3]是 Salton 等在 1969 年提出的,它是目前最为成熟和应用

收稿日期:2014-08-28;**修回日期**:2014-09-18
基金项目:辽宁省教育厅科学技术研究项目(L2013066)
作者简介:唐晓丽(1989－),女,山东威海人,研究生,研究方向:自然语言处理。*通信作者:白宇,E-mail:baiyu@sau.edu.cn

最为广泛的文本表示方法。VSM 的基本思想[4]是将文本内容的处理转化为向量空间的运算,该向量的每一维分量用特征词的权重来表示。马晖男等[5]提出了一种新的修正的向量空间模型(MVSM),该模型将修饰词与中心词组成的合成短语引入到查询语句及传统的向量空间检索模型的信息表示中,并重新计算作为特征索引项的合成短语的权重值,有效提高了文本信息检索系统的检索性能。夏云庆等[6]提出以歌词作为歌曲情感分析的依据,采取基于情感单元的情感向量空间模型进行歌词情感分析,实验表明该模型在歌词情感分类中优于传统方法。VSM 虽然为文本处理带来了计算和操作上的方便,但其具有一定的局限性:一方面利用 VSM 进行文本建模无法表示文本的语义信息;另一方面用 VSM 表示文本数据,其数据空间是极度高维且稀疏的,这样容易引起维度灾难及存储空间浪费的问题。

主题模型相对 VSM,可以实现有效的降维,通过建立一个富含语义信息的数据空间,可有效解决 VSM 存在的上述问题。当前流行的主题模型有三种:LSI[7](Latent Semantic Indexing)、PLSI[8](Probabilistic Latent Semantic Indexing)和 LDA[9](Latent Dirichlet Allocation)。相比 LS1、PLS1,LDA 具有先验性假设,更加符合实际文本中主题分布情况,而且不易发生过拟合问题,更加适合处理大规模语料库[10]。

Blei 等人[4]利用 LDA 进行文本建模,使用 SVM 对建模后的文本进行分类,在文本维度大幅度下降的情况下文本分类的准确度得到有效提高。张志飞等[11]针对短文本的特征稀疏性和上下文依赖性两个问题,提出一种基于 LDA 的短文本分类方法。利用 LDA 生成的主题,一方面可以区分相同词的上下文来降低权重;另一方面关联不同词以减少稀疏性来增加权重。实验结果表明该方法在分类性能上得到有效的提高。石晶等[12]提出一种基于 LDA 的文本分割方法,以 LDA 为语料库及文本建模,利用 MCMC 中的 Gibbs 抽样进行推理,间接计算模型参数,获取词汇的概率分布,实验结果表明其片段边界的识别错误率远远低于其它同类算法。张小平等[13]通过一种高斯函数对特征词加权,改进了 LDA 主题模型的主题分布,实验结果显示改进的模型在主题表达和预测性能方面都有所提高。

本文借鉴主题模型文本建模的思想,充分利用 word2vec 训练数据的高效性以及词聚类结果的有效性,提出了一种基于 word2vec 的文本建模方法,该方法将文本表示为词聚类类别有限集合的混合分布,并使用 K－means 算法完成文本聚类。在复旦中文语料库上进行测试,实验结果表明,该方法在聚类效果和算法效率上均有明显的提高。

1 方法描述

1.1 word2vec 简介

word2vec[14]是 Google 在 2013 年开源的一款将词表示为实数值向量的高效工具。通过训练可以把对文本内容的处理简化为 K 维向量运算,而向量空间上的相似度可以用来表示文本语义上的相似度。word2vec 输出的词向量可以用来做很多 NLP 相关的工作,比如词聚类、找同义词、词性分析等。word2vec 另一大特点是高效性,Mikolov[15]指出一个优化的单机版本一天可训练上千亿词。

Word2vec 生成词向量的基本思想来自于 Bengio 提出的 NNLM(Neural Network Language Model),其原理示意图如图 1 所示。图 1 中,每个输入词都被映射为一个向量,该映射用 C 表示,所以 $C(w_{t-1})$ 即为 w_{t-1} 的词向量。g 为一个前馈或递归神经网络,其输出是一个向量,向量中的第 i 个元素表示概率 $p(w_t=i|w_1^{t-1})$。训练的目标是最大似然正则项,其计算公式如(1)所示。

$$\max Likelihood = \max \frac{1}{T}\sum_t \log f(w_t, w_{t-1}, \cdots, w_{t-n+2}, w_{t-n+1}; \theta) + R(\theta) \tag{1}$$

其中,θ 为参数,$R(\theta)$ 为正则项。

借鉴 NNLM 的思想,word2vec 采用的架构模型包含 CBOW(Continuous Bag-Of-Words)和 Skip-Gram 两种。其原理示意图如图 2 所示。

CBOW模型是利用上下文信息预测当前词的思想生成词向量,即将当前词上下文对应的连续词语表示

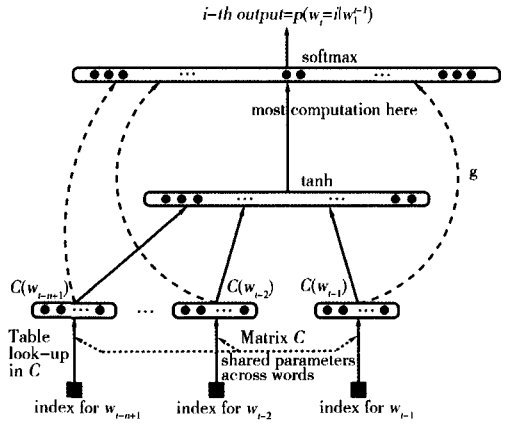

Fig. 1 Schematic of NNLM model[14]

图 1 NNLM 模型原理示意图[14]

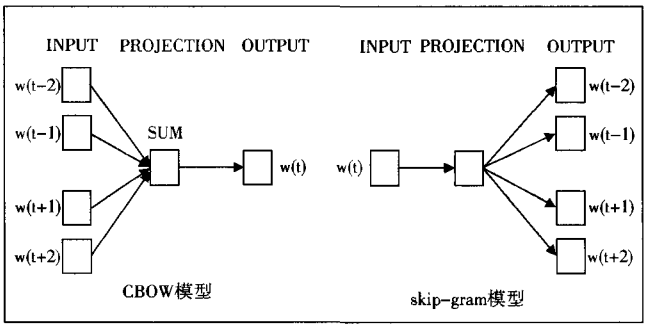

Fig. 2 Schematic of CBOW model and skip-gram model[14]

图 2 CBOW 模型和 skip-gram 模型原理示意图[14]

成词袋的形式，将训练的目标向量选为上下文词向量的求和，应用 Huffman 编码表示需要训练的目标函数，并利用随机梯度下降的方法对目标函数进行训练。为减少迭代次数，提高训练效率，随机选取上下文中一定数目的词语作为负例进行采样，最后使用层次 softmax 的表示形式生成目标词向量。

Skip-gram 模型则使用了与 CBOW 模型相反的方式生成词向量，即仅利用当前词向量来预测指定窗口内上下文的词向量，同样使用 Huffman 编码表示目标函数，采用随机梯度下降的方法对目标函数进行训练，最后生成使用层次 softmax 表示的上下文词语向量。

1.2 基于 word2vec 的文本建模

基于 word2vec 的文本建模方法主要包含 5 部分：预处理、词聚类、tf-idf 值统计、统计文本类别分布、文本向量化。其流程如图 3 所示。

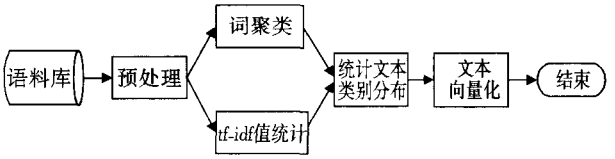

Fig. 3 Basic processes of text clustering based on word2vec

图 3 基于 word2vec 的文本建模流程

1.2.1 预处理

对于给定的文档集合，需进行预处理，主要包括分词、去停用词、文档合并，将处理后的语料以一篇文档的形式呈现出来，将其作为下一步的输入数据。

1.2.2 词聚类

在 Linux 环境下，运用 word2vec 工具对输入数据进行词聚类，其参数设置情况如表 1 所示。本文中，classes 参数设置的取值范围为 50～500，间隔为 50。将词聚类结果表示为向量的形式，如公式(2)所示：

$$Cluster\ Result = \ll word_1, C_1 > < word_2, C_2 > \cdots < word_n, C_n \gg \quad (2)$$

其中，n 表示词总数，C_i 表示词 $word_i$ 所属的类别编号。

表 1 word2vec 参数设置情况

Table 1 Parameter settings of word2vec

超参数	含义	取值
-size	向量维数	100
-window	上下文窗口大小	5
-sample	高频词亚采样的阈值	le-3
-hs	是否采用层次 softmax	1
-negative	负例数目	0
-cbow	使用 cbow 算法	0
-classes	词聚类类别数	50

1.2.3 tf-idf 值统计

遍历计算每篇文档中不同特征词的 tf-idf 值，计算公式如(3)所示。按照 tf-idf 值的大小将特征词由高到低进行排序，选取前 100 个特征词作为文本的特征。统计完成后，每篇文档的表示形式如公式(4)所示。

$$tfidf_{ij} = \frac{n_{ij}}{\sum_k n_{kj}} \times \log\left(\frac{|D|}{|\{j:t_i \in d_j\}|}\right) \tag{3}$$

其中，n_{ij} 表示第 j 篇文档 d_j 中词 t_i 的个数，$\sum_k n_{kj}$ 表示第 j 篇文档 d_j 中词总数，D 表示文档集合。

$$d_j = \ll word_{1j}, tfidf_{1j} > < word_{2j}, tfidf_{2j} > \cdots < word_{100j}, tfidf_{100j} \gg \tag{4}$$

1.2.4 统计文本类别分布

在 1.2.2 与 1.2.3 两阶段所得结果的基础上，统计每篇文档在不同类别下的分布。即对于每篇文档，顺序遍历所选出的 100 个特征词，统计每个词所属的类别，对同一类别下所有特征词的 tf-idf 值求和并进行归一化，计算公式如(5)和(6)所示，计算所得数值代表该篇文档在各个类别下的权重。该过程完成后，文本表示形式如公式(7)所示。

$$w_{jk} = \sum_i tfidf_{ij}(word_i \in C_k) \tag{5}$$

$$w'_{jk} = \frac{w_{jk}}{\sqrt{\sum_{k=0}^{1-N} w_{jk}^2}} \tag{6}$$

其中，w_{jk} 表示文档 d_j 在类别 C_k 上的权值，$tfidf_{ij}$ 表示文档 d_j 中属于类别 C_k 的词 $word_i$ 的 $tfidf$ 值，C 为词聚类类别集合；w'_{jk} 表示文档 d_j 在类别 C_k 上归一化后的权值，N 表示类别总数。

$$d_j = \ll C_1, w'_{j1} > < C_2, w'_{j2} > \cdots < C_N, w'_{jN} \gg \tag{7}$$

1.2.5 文本向量化

将每篇文档类别分布的统计结果转化为向量的形式，构建出“文本－类别分布”矩阵。通过以上操作，完成基于 word2vec 的文本建模。最终文本表示形式如式(8)所示：

$$\boldsymbol{D} = \begin{bmatrix} w'_{11}, w'_{12}, \cdots, w'_{1i}, \cdots, w'_{1N} \\ \vdots \quad \vdots \quad \vdots \quad \vdots \\ w'_{j1}, w'_{j2}, \cdots, w'_{ji}, \cdots, w'_{jN} \\ \vdots \quad \vdots \quad \vdots \quad \vdots \\ w'_{M1}, w'_{M2}, \cdots, w'_{Mi}, \cdots, w'_{MN} \end{bmatrix} \tag{8}$$

其中，N 代表类别数量，M 代表文本数。

2 实验及结果分析

2.1 实验语料

本次实验从复旦大学中文语料库中抽取 10 个子集作为实验语料，10 个子集分别为环境、计算机、交通、教育、经济、军事、体育、医药、艺术和政治，每个子集包含 200 篇文本，总计 2 000 篇文档。

2.2 评测指标

采用 F 度量值来衡量文本聚类效果，它是一种平衡准确率与召回率的评价指标。准确率及召回率计算方法如式(9)、(10)所示，n_i 为类别 i 的文本数量，n_j 为聚类后类别 j 的文本数量，n_{ij} 为聚类后类别 j 中隶属于类别 i 的文本数量。

$$P(i,j) = \frac{n_{ij}}{n_j} \tag{9}$$

$$R(i,j) = \frac{n_{ij}}{n_i} \tag{10}$$

对应的 F 值计算公式如(11)所示：

$$F(i,j) = \frac{2 \times P(i,j) \times R(i,j)}{P(i,j) + R(i,j)} \tag{11}$$

全局 F 值计算公式如(12)所示：

$$F = \sum_i \frac{n_i}{n} \max(F(i,j)) \tag{12}$$

其中,n 是文本集合中总的文本数目;通常 F 度量值越大,聚类效果越好。

2.3 实验结果与分析

本文采用了三种文本建模方法在实验语料上进行文本聚类实验。方法一为利用向量空间模型进行文本建模;方法二为利用 LDA 进行文本建模,即将文本表示为潜在主题有限集合的混合分布;方法三为第 1 节所提出的方法,即将文本表示为类别有限集合的混合分布。聚类算法采用 K-Means 算法,由于 K-means 算法聚类结果具有不稳定性,所以我们重复进行 5 次聚类实验,取其平均值作为最终实验结果。

在方法二中,采用 Gibbs 抽样算法来进行参数估计,先验超参数 α,β 取值分别为 $\alpha=50/K,\beta=0.01$,K 为主题数,迭代次数为 1 000。对于最优主题数 K 的选择,本文通过实验将聚类效果 F 值最高时的主题数量定为最优主题数。其结果如图 4 所示。由图 4 可知,在确定最优主题数的过程中,随着主题数目的增多,聚类效果 F 值先增大后减小,当 $K=200$ 时文本聚类效果最好。因此,在后续的聚类效果对比分析中选择主题数 K 的取值为 200。

在方法三中,同样采用实验的方法获取最优词聚类类别数量 C。确定的标准为聚类效果 F 值最高的类别数目。其结果如图 5 所示。从图 5 中可以看出,随着类别数量的增大,其曲线变化相对图 4 来说比较平稳,聚类效果相对比较稳定。在类别数量为 300 时 F 值最高,因此在后续的聚类效果对比分析中选择类别数量 C 的取值为 300。

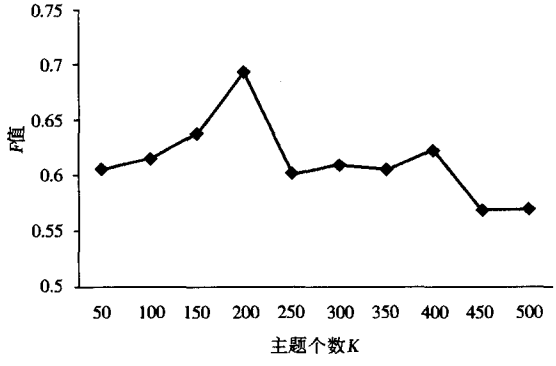

Fig. 4 Clustering results of different topic number

图 4 不同主题数的聚类效果

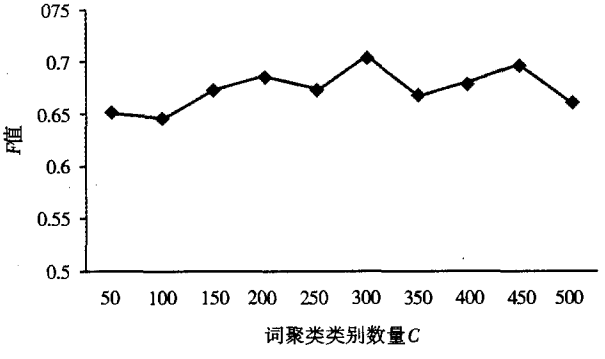

Fig. 5 Clustering results of different class number

图 5 不同词聚类类别数的聚类效果

在确定最优主题数量 K 和最优类别数量 C 的过程中,本文也对算法的执行时间进行了统计,其结果如表 2 所示,单位为秒。

表 2 两种算法执行时间比较

Table 2 Execution time comparison of two algorithms

主体/类别数量	50	100	150	200	250	300	350	400	450	500	总计
LDA_based(s)	517	993	1 466	1 941	2 407	2 882	3 422	3 863	4 328	4 836	26 655
word2vec_based(s)	21	22	29	34	50	62	71	65	95	82	531

由表 2 可以看出,方法三的算法效率相对方法二有明显的提高,这是因为本文所提出的方法在利用 word2vec 进行词聚类的过程中,采用了 Huffman 编码、层次 softmax 表示、负例采样以及高频词亚采样等方法实现了算法的高效性,而方法二在进行参数估计过程中,需要模拟 Dirichlet 过程,计算量较大,所以其算法执行时间远远高于本文所提出的方法。

将本文所提出的方法与方法一、方法二进行聚类效果对比,其结果如图 6 所示。

由图 6 可以看出,方法二和方法三的 F 值相对方法一分别提高了 5%、6.01%。主要原因是方法二与方法三在进行文本建模时均增加了语义信息,减少了其所包含的冗余、噪音信息,可以有效提高文本聚类的效果。方法二与方法三进行对比,在聚类效果上,方法三相对方法二提高 1.01%,说明本文提出的方法可以有效提高文本聚类效果;在算法效率上,由表 2 可知,方法三明显优于方法二。随着文本规模的增大,本文所提出方法的优势越来越明显。

3 结论

本文借鉴主题模型文本建模的基本思想,提出了一种基于 word2vec 的文本建模方法,既保证了主题模型在文本建模方面的优势,又将 word2vec 训练数据的高效性和词聚类结果的有效性两大特点融入其中,将文本表示为不同类别有限集合的混合分布,使得文本得到更好的表示。实验结果表明,该方法在聚类效果和算法效率上均有明显的提高。本文所提出的方法在统计每篇文档的类别分布时,是利用文档中特征词的 tf-idf 值进行统计的,下一步工作将对其进行改进,并尝试与基于 LDA 的文本建模方法相融合,以期达到更好的聚类效果和算法效率。

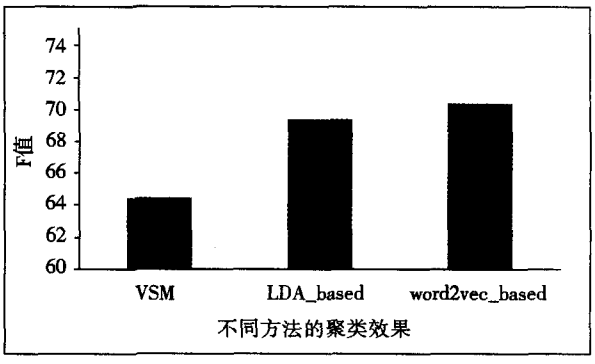

Fig. 6 F-value Comparison of three methods

图 6 三种方法效果比较

参考文献:

[1] 刘平峰,章佩璐,张军,等. 面向主题的 Web 信息融合模型[J]. 图书情报工作,2011,**8**:40-43.

[2] 姚清耘,刘功申,李翔. 基于向量空间模型的文本聚类算法[J]. 计算机工程,2008,**34**(18):39-41.

[3] Salton G,Wong A,Yang C S. A Vector Space Model for Automatic Indexing [J]. *Communications of the ACM*,1975,**18**(11):613-620.

[4] 孙昌年. 基于主题模型的文本相似度计算研究与实现[D]. 安徽:安徽大学,2012.

[5] 马晖男,吴江宁,潘东华. 一种修正的向量空间模型在信息检索中的应用[J]. 哈尔滨工业大学学报,2008,**40**(4):666-669.

[6] 夏云庆,杨莹,张鹏洲. 基于情感向量空间模型的歌词情感分析[J]. 中文信息学报,2010,**24**(1):99-103.

[7] Deerwester,S. Dumais,T. Landauer,*et al*. Indexing by Latent Semantic Analysis [J]. *Journal of the American Society of Information Science*,1990,**41**(6):391-407.

[8] Thomas H. Probabilistic latent semantic indexing[C]// Proceedings of SIGIR,Berkeley,CA,USA,1999:50-57.

[9] Blei D,Ng A,Jordan M. Latent dirichlet allocation [J]. *Journal of Machine Learning Research*,2003,**3**(1):993-1022.

[10] 王振振,何明,杜永萍. 基于 LDA 主题模型的文本相似度计算[J]. 计算机科学,2013,**40**(12):229-231.

[11] 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.

[12] 石晶,胡明,石鑫,等. 基于 LDA 模型的文本分割[J]. 计算机学报,2008,**31**(10):1865-1873.

[13] 张小平,周雪忠,黄厚宽,等. 一种改进的 LDA 主题模型[J]. 北京交通大学学报,2010,**34**(2):111-114.

[14] 邓澍军,陆光明,夏龙. Deep Learning 实战之 word2vec[R/OL]. 网易有道,2014-02-27.

[15] Mikolov T,Sutskever I,Chen K,*et al*. Distributed Representations of Words and Phrases and Their Compositionality[C]// Proceedings of NIPS,Lake Tahoe,Nevada,USA,2013:3111-3119.