文章编号:1008-1542(2015)05-0532-07 doi:10.7535/hbkd.2015yx05014

基于POS-CBOW语言模型的相似词分析

阮冬茹,潘洪岩,高 凯

(河北科技大学信息科学与工程学院,河北石家庄 050018)

摘 要:相似词分析是自然语言处理领域的研究热点之一,在文本分类、机器翻译和信息推荐等领域中具有重要的研究价值和应用意义。针对新浪微博短文本的特点,给出一种带词性的连续词袋模型(POS-CBOW)。该模型在连续词袋模型的基础上加入过滤层和词性标注层,对空间词向量进行优化和词性标注,通过空间词向量的余弦相似度和词性相似度来判别词向量的相似性,并利用统计分析模型筛选出最优相似词集合。实验表明,基于POS-CBOW语言模型的相似词分析算法优于传统CBOW语言模型。

关键词:自然语言处理;语言模型;词向量;相似词;POS-CBOW

中图分类号:TP391 **文献标志码**:A

Similar words analysis based on POS-CBOW language model

RUAN Dongru, PAN Hongyan, GAO Kai

(School of Information Science and Engineering, Hebei University of Science and Technology, Shijiazhuang, Hebei 050018, China)

Abstract: Similar words analysis is one of the important aspects in the field of natural language processing, and it has important research and application values in text classification, machine translation and information recommendation. Focusing on the features of Sina Weibo's short text, this paper presents a language model named as POS-CBOW, which is a kind of continuous bag-of-words language model with the filtering layer and part-of-speech tagging layer. The proposed approach can adjust the word vectors' similarity according to the cosine similarity and the word vectors' part-of-speech metrics. It can also filter those similar words set on the base of the statistical analysis model. The experimental result shows that the similar words analysis algorithm based on the proposed POS-CBOW language model is better than that based on the traditional CBOW language model.

Keywords: natural language processing; language model; word vector; similar words; POS-CBOW

相似词分析是近些年自然语言处理领域的研究热点之一,在文本分类、机器翻译以及信息推荐等领域中有着广泛应用。目前相似词的分析大都需要人为干预为主的方法,借助人工标注词典来设定词的相似性。但是随着社交网络中网络新词的不断涌现,基于人工标注的方法已无法完成庞大的标注任务,而且由于社交网络的短文本特征(如数据量庞大、书写不规范等),传统方法已无法得到较好的分析结果。现阶段,自然语言处理、深度学习等领域的相似词分析研究是解决这一问题的主要手段之一。

收稿日期:2015-04-14;修回日期:2015-06-26;责任编辑:陈书欣
基金项目:河北省社会科学发展研究课题资助项目(2015030344)
作者简介:阮冬茹(1967—),女,河北怀安人,副教授,主要从事自然语言处理、微博计算方面的研究。
通讯作者:高 凯副教授。E-mail:gaokai@hebust.edu.cn

阮冬茹,潘洪岩,高 凯. 基于POS-CBOW语言模型的相似词分析[J]. 河北科技大学学报,2015,36(5):532-538.
RUAN Dongru, PAN Hongyan, GAO Kai. Similar words analysis based on POS-CBOW language model[J]. Journal of Hebei University of Science and Technology,2015,36(5):532-538.

相关研究工作中，BENGIO 等[1]利用 2 个语句对相似词的概念进行了阐述，句子 1“The cat is walking in the bedroom”和句子 2“A dog was running in a room”，2 个语句在句式结构上非常相似，BENGIO 等定义了 the 和 a，cat 和 dog，is 和 was，walking 和 running，bedroom 和 room 等为相似词。文献[2]在神经网络语言模型基础上进行了完善，重点解决了高维空间的维度问题，利用词在高维空间的概率分布得到词与词的相似度。MIKOLOV 等[3-4]在 BENGIO 等工作的基础上对神经网络做了进一步优化，减少了神经网络的参数，提升了训练速度，并提出了一种新的神经网络语言模型，运用单隐含层的神经网络生成词向量，计算词的相似性，并在文献[5]中进一步优化提出了循环神经网络语言模型。MIKOLOV 等[6]提出了 CBOW 模型和连续 Skip-gram 模型，这两种模型都是一种类前馈神经网络语言模型，不同的是 CBOW 模型是预测相似词，而 Skip-gram 模型是预测相近词。随后，MIKOLOV 等[7]提出了将词的相似性关系应用于机器翻译领域，成功地预测低频率词汇，并提出了对短语的相似性分析和词向量的隐含语义关系分析[8-9]。LEVY 等[10]对稀疏空间向量的语义规律关系进一步进行研究和完善。QIU 等[11]针对 CBOW 和 Skip-gram 在词向量的相近性和歧义性上的缺陷提出了 PAS 模型，从而将准确率提高了约 16.9%。SOUTNET 等[12]利用 Skip-gram 与 LSTM 模型相结合的方法，提升了神经网络语言模型在长文本和短文本中的处理效果。文献[13]中在词袋模型的基础上提出了段向量，相对于词向量而言，它克服了原有模型的一些缺陷，在文本分类和语义分析等领域表现良好。ZHANG 等[14]利用 CBOW-CL-SimH 模型在语义层面对文本做重复性检测，利用混合模型将文本生成了向量，进而计算文本的相似度。MNIH 等[15]提出了基于 n-gram 语言模型的概率语言模型，该模型通过对有序词序列生成词向量，然后利用给出的词向量模型来预测词序列中下一个词向量，该模型是表现较好的 n-gram 模型。BLEI 等[16]提出了一种文档主题生成模型，利用词袋的方法，将文档的词生成向量，从而利用概率分布生成主题，该模型是一个三层贝叶斯概率模型，词袋中的词是无序的，所以简化了模型复杂度，提高了训练速度。MAAS 等[17]提出了一种 log-bilinear 模型来计算词向量的语义信息和情绪信息，利用监督与非监督的混合方法从语料中训练得到的词向量不仅有语义信息，还包含丰富的情绪信息，在情绪分类方面表现较好。

在中文研究领域中，词语的相似度计算大都依托于同义词林、HowNet 和中文 WordNet 等。文献[18]采用同义词词林作为语义体系，对中文词语的同义词识别进行了初步研究。吴思颖等[19]采用了一种基于中文 WordNet 的中英文词语相似度计算方法，解决了候选同义词集组合的权重和取舍问题，实现了一个可以计算英-英、汉-英、汉-汉词语之间相似度的算法。近年来，非监督方法成为了相似词研究的新兴方法，石静等[20]基于大规模语料的训练，在一定程度上提升了汉语语义相似度计算的准确率，并实现了不同语域的集成。文献[21]中通过构造领域类别核心词集，对词向量间语义关系进行了语义相似度的领域词语聚类分析。文献[22]提出了一种新的分词方法，利用 Word2vec 生成词向量对中文词进行聚类分析。

本文针对微博短文本的特点，对 MIKOLOV 等[5]在 Google Code 中的开源连续词袋模型 Word2vec 进行了调整和改进，给出了一种基于 POS-CBOW 的语言模型。该模型将结构调整为输入层、过滤层、投影层、词性标注层和输出层，通过过滤层对微博短文本进行修正，然后在词性标注层使生成的词向量带有词性信息，进而以空间向量的余弦值和向量的词性比较为条件，计算词的相似性。实验证明，该方法在微博短文本分析中有较高的准确率。

1　CBOW 语言模型

CBOW 语言模型是 MIKOLOV 等[6]阐述的一种类前馈神经网络语言模型，由输入层、投影层和输出层组成，模型结构如图 1 所示。CBOW 语言模型不同于标准词袋模型，其引入了连续分布式词表示的方法，形成了新的连续词袋模型。

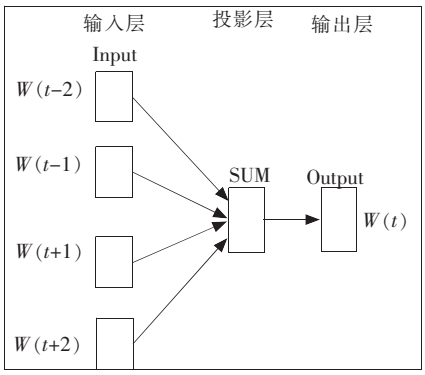

图 1　CBOW 语言模型

Fig. 1　CBOW language model

CBOW 语言模型通过将语料库词 $C(1)$，$C(2)$，…，$C(t)$ 映射到投影层，得到词典 $|V| \in R^m$。然后通过共享投影层，使语料库词 $C(t)$ 映射到投影层唯一的位置 $W(t)$，再通过 $W(t)$ 的上下文信息预测 $W(t)$，而其中每个 $W(t)$ 都与上文的

词序列无关。CBOW语言模型的基本训练步骤如下。

步骤1:在输入层,通过M限定输入层的上下文大小,然后在窗口中顺序读取语料库词序列$C(t-M)$,$C(t-M+1)$,…,$C(t+M)$,通过哈希表得到投影层的相应词位置$W(t-M)$,$W(t-M+1)$,…,$W(t+M)$,获得$W(t)$词的上下M个词Context($W(t)$)。

步骤2:在投影层,对$W(t)$的上下文Context($W(t)$)做步骤1操作,$V(t)$为$W(t)$上下文累加和。

$$V(t)=\sum_{t-N}^{t+N}\text{Context}(W(t))。\tag{1}$$

步骤3:从投影层到输出层,利用词$W(t)$的上下文信息,通过式(2)来生成词$W(t)$的向量值,其中式(3)为词向量回归分析操作,来完成对$W(t)$的判断。

$$P(W(t)\mid\text{Context}(W(t)))=\prod_{t-n}^{t+n}f(V(t)\theta),\tag{2}$$

$$f(V(t)\theta)=\frac{1}{1+\mathrm{e}^{-V(t)\theta x}}。\tag{3}$$

综上,CBOW语言模型与BENGIO等[1]提出的前馈神经网络语言模型相似,该模型去掉了隐含层,加快了语言模型的训练速度,且模型中每个词向量的计算只与滑动窗口限定的Context有关系,减少了模型的训练参数,降低了模型复杂度,提高了模型准确率。但是CBOW语言模型仍然需要大量的训练集,而且训练模型的好坏与上一级任务有密切关系。滑动窗口概念的引入,忽略掉了滑动窗口中的上下文内容对词的相似性计算产生的干扰。

2 POS-CBOW语言模型及相似词计算

2.1 POS-CBOW语言模型

本文给出的带有词性的连续词袋POS-CBOW语言模型,其结构如图2所示。该模型在CBOW语言模型的基础上增加了过滤层和词性标注层。过滤层的主要作用是修正短文本语料语序,优化词向量空间,从而使POS-CBOW语言模型相对于前一阶段的工作更加独立;词性标注层的作用是对所有的词向量进行词性标注,使空间词向量在潜在语义关系的基础上建立语法关系,从而提高相似词计算的准确率。

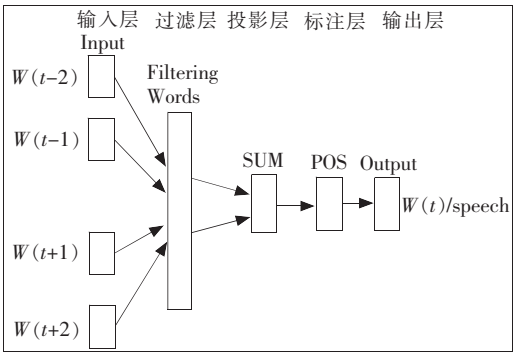

图2 POS-CBOW语言模型

Fig. 2 POS-CBOW language model

2.1.1 POS-CBOW语言模型的过滤层

微博作为一种抒发情感的载体,通常会加入一些符号来辅助情感的表达。这些附加信息虽然具有一定含义,但在本文的实际工作中,它改变了训练语料的正常语句结构,对训练模型产生了干扰。又因为微博的短文本这一特点,使得这些符号在语句中占据较大权重,而这对于传统的长文本分析算法来说是无法处理的难题。例如微博博文"I('▽')love you!",符号"('▽')"在未经过滤处理的情况下,极有可能会与"I"成为相似词。所以,需要修正其为"I love you!",去掉文本中的噪音信息。

POS-CBOW语言模型的过滤层介于输入层和投影层之间,使用整理的微博文本停用词表对训练语料进行语句修正,从而达到优化词向量空间的目的。过滤层算法步骤如下。

步骤1:初始化哈希表Vocab,初始值为-1。

步骤2:循环读取训练语料句子的词,计算哈希值,并以哈希值为下标,Vocab[Hash]=1。

步骤3:初始化过滤层Filter,大小为停用词表个数,循环读取停用词,并计算通用词哈希值,记录到过滤层Filter表。

步骤4:遍历读取Filter表每项的值h,并查询Vocab[h]是否等于1,等于1则将Vocab[h]=-1,否则循环继续,直到Filter遍历完毕,过滤完毕。

2.1.2　POS-CBOW 语言模型的词性标注层

CBOW 语言模型是一种概率模型，通过将语料中词的上下文信息生成相应的词向量映射到高维空间中，然后以词向量在高维空间中的关系来计算词与词之间的相似度。这样虽然提高了训练效率，加快了计算速度，但是，同时忽略了上下文信息中一些不符合相似词定义的词向量。例如图 3 是词向量“中国”的可视化图，在其临近向量中出现了“当今”、“近年来”、“国内”、“海外”等词，这些词的出现在句子中位置均与“中国”临近，但是严格地从相似词定义上来讲，这些词不是“中国”的相似词。

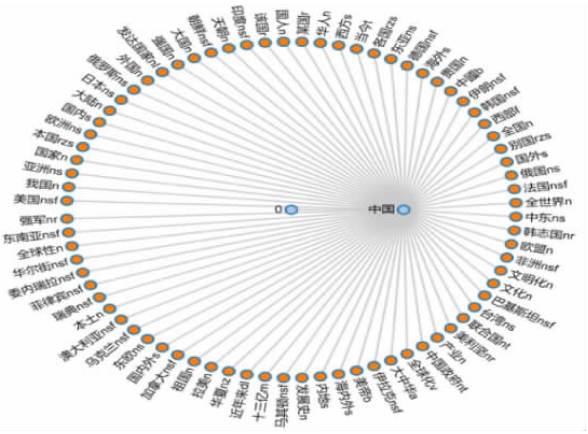

图 3　词向量“中国”可视化图

Fig. 3　Visualization of word vector “China”

为了排除上述词的干扰，得到更加准确的相似词结果，引入了词性标注层。

POS-CBOW 语言模型的词性标注层介于投影层和输出层之间，通过中文分词工具 NLPIR 对生成词向量进行词性标注，所用词性为计算所有汉语词性标记集。针对词的多词性这一性质，对词语的词性建立了词性体系，为相似词的计算提供词性参考体系，以便得出更加完善的相似词集。词性体系的构造步骤如下。

步骤 1：以 R 为根节点，创建以 R 为根节点的所有子节点。

步骤 2：以上一级子节点为根节点，创建相应节点下的子节点。

步骤 3：查看上一级节点是否有子节点，如果有，重复步骤 2，否则构造完毕，如图 4 所示。

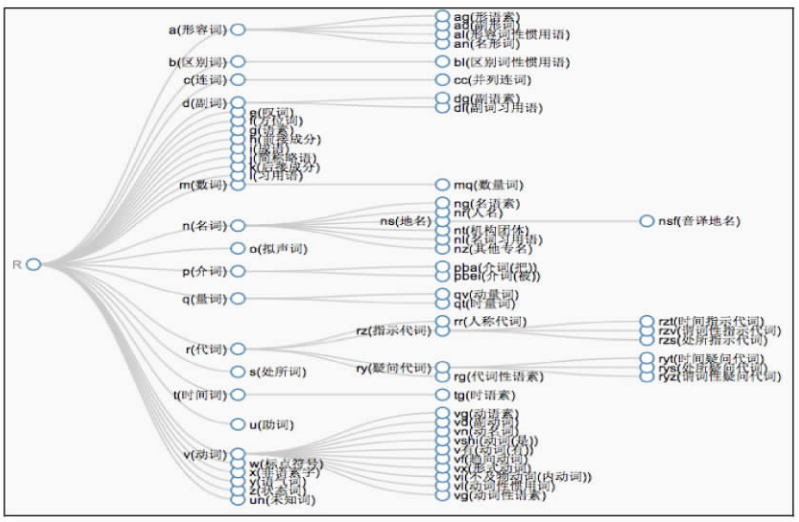

图 4　词性体系

Fig. 4　System of speech

2.2　基于 POS-CBOW 语言模型的相似词计算

POS-CBOW 语言模型生成的词向量不仅包含潜在的语义关系，还包含着语法关系。语法关系的加入完善了语义关系的不足。在相似词计算中以余弦相似度为计算方式，以语法关系为计算准则，进行词向量的相似性计算。例如，计算词向量“中国”的相似词，计算模型将会查找与词向量“中国”同在一个父类词性下的词，再计算词向量的余弦相似度，从而得到 2 个词向量的相似度。其中考虑到新词问题，把未知词性的词也加入结果集。本文采用了两种择优算法，一是 Top$N$ 算法，选择 N 个最优结果；二是通过建立统计模型，选出最优结果集。

2.2.1　Top$N$ 词向量计算

Top$N$ 算法是择优的经典算法之一，通过排名得出前 N 个最优项作为结果。本文在相似词的计算中利

用了 Top$N$ 的思想，通过结合余弦相似度和词性信息 2 个条件，对整个词向量空间遍历计算后，对相似词进行排序，选出前 N 个词作为结果集。Top$N$ 个相似词计算的基本步骤如下。

步骤 1：取词向量空间第 i 个词向量 V_i，在词性体系中查找 V_i 的父类词性，如果与 W 为同一父类词性或为“un”，进入步骤 2，否则，查看向量空间是否遍历完，是则结束计算，否，$i=i+1$，重复步骤 1 操作。

步骤 2：计算余弦相似度 $\mathrm{Sim}(W,V)=W\cdot V/(|W|\times|V|)$，如果 $\mathrm{Sim}(W,V)<0$，返回步骤 1，否则倒叙遍历 Set 集合，比较相似度值，如果小于 $\mathrm{Sim}(W,V)$，将该位置的值后移，插入 V 到该位置，重复步骤 1。

通过在 Top$N$ 算法中加入词性分析，使同一词性体系的词聚集在一起，而不同词性的词向量则被排除在外。例如：重新利用 POS-CBOW 语言模型对词向量“中国”做相似词计算，发现与词向量“中国”相邻近的词向量“当今”、“近些年”、“国内”和“海外”等被排出了结果集，如图 5 所示。

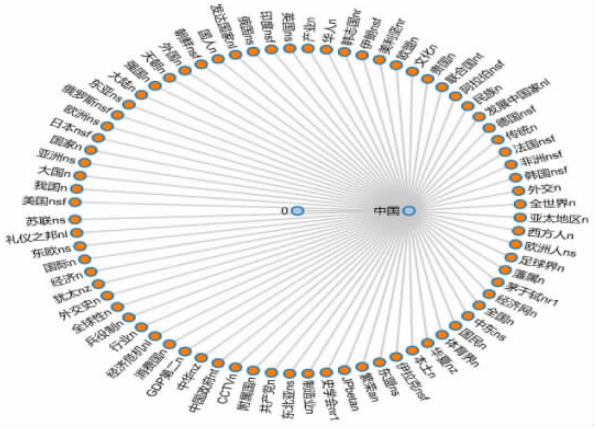

图 5 基于 POS-CBOW 语言模型相似词计算的 Top$N$ 结果

Fig. 5 Top$N$ result of similar word based on POS-CBOW language model

2.2.2 词向量的统计分析模型

Top$N$ 算法中相似词的计算结果往往受到 N 值限定，从而导致一些较优词向量的丢失。为了能够更加充分地获取最优结果集，本文给出了另一种相似词计算方法，采用一种动态阈值的统计分析模型来选出结果集。

首先，计算出余弦相似度大于 0 的所有词向量，获取相似度集合，计算集合的三阶标准化矩，统计分析出相似度值的概率分布，如图 6 所示。根据相似度集合的偏度，得出词向量相似度值的整体分布情况。如果词向量相似度集合为正偏态，阈值设为集合的平均数，即结果集合为平均数的右侧部分；如果集合偏度为负偏度，那么阈值就选择集合的中位数，即结果集合为中位数的右侧部分。表 1 为集合的概率分布和阈值选择情况。

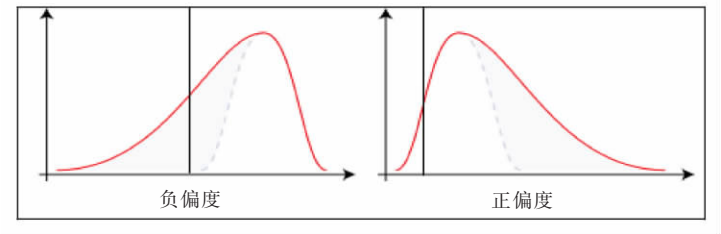

图 6 偏度与数值的概率分布

Fig. 6 Skewness and its probability distribution

表 1 Set 集合概率分布及阈值选取

Tab. 1 Probability distribution of Set and the threshold selecting

偏度	分布状态	阈值
<0	平均数<中位数<众数	平均数
>0	众数<中位数<平均数	中位数
$=0$	平均数=中位数	平均数或中位数

三阶标准化矩（偏度）公式为

$$\mathrm{Skew}(X)=E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right]=\frac{\mu^3}{\sigma^3}=\frac{k_3}{k_2^{3/2}}\ 。\tag{4}$$

推导后得出偏度值公式为

$$\mathrm{Skew}(X)=\frac{k_3}{k_2^{3/2}}=\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^{3/2}}\ 。\tag{5}$$

输入词向量 W，相似词计算算法步骤如下。

步骤 1：取词向量空间第 i 个词向量 V，在词性体系中查找 V 的父类词性，如果与 W 为同一父类词性或为“un”，进入步骤 2，否则，查看向量空间是否遍历完，是则结束计算，否，$i=i+1$，重复步骤 1 操作。

步骤 2：计算余弦相似度 $\mathrm{Sim}(W,V)=W\cdot V/(|W|\times|V|)$，如果 $\mathrm{Sim}(W,V)>0$，倒叙遍历 Set 集合，比较相似度值，如果小于 $\mathrm{Sim}(W,V)$，将该位置的值（后值）后移，插入 V 到该位置。重复步骤 1，否则进入步骤 3。

步骤 3：计算 Set 的偏度（Set 的三阶标准化矩），如果 Set 的偏度为正偏态，计算 Set 平均值，以平均值为阈值，筛选出最优词向量集合 Set1，如果 Set 偏度为负，计算 Set 中位数，以中位数为阈值，获取最优向量集合 Set2。

3　实验对比及分析

本文所用实验数据为新浪微博语料，语言为中英文混合，利用 NLPIR 分词工具对语料进行分词，整理出了 2 GB 文本，得到词数约 4.4 亿词的训练集，训练得出约 42 万条词向量。实验硬件环境为 2 台 Lenovo-Erazer X310，处理器 Intel 4 代 i5-4460 3.2 GHz，4 核 4 线程，内存 8 GB。分别利用 Word2vec 中的 CBOW 模型和 POS-CBOW 模型进行了训练，Window 值为 5，层数为 200 的情况下，没有加入反例训练，使用 4 线程训练。通过 2 个模型的训练过程与训练结果的对比分析可得如下结论（表 2）。

1）经过 POS-CBOW 模型的过滤层之后，微博语料集被过滤掉了约 1/4 的噪声，证明针对微博数据的过滤层是有效的，且生成的词向量空间相对来说压缩了约 0.7%；

2）从整个模型的训练过程分析，输入层的过滤与向量空间的压缩减少了训练过程中的迭代计算量，提高了模型的计算效率。从表 2 可知，在同样条件下，POS-CBOW 模型的训练时间也比 CBOW 模型的训练时间节省了 50%左右。

在接下来的实验中，分别利用 CBOW 语言模型和 POS-CBOW 语言模型对词向量“中国”进行了相似词计算，利用式（6）对 CBOW 语言模型和 POS-CBOW 语言模型的 Top$N$ 相似词结果进行评价。

$$\text{准确率}=\frac{\text{相似词数}}{\text{结果集总数}}\times 100\%\ 。\tag{6}$$

从图 3 和图 5 计算结果来看，POS-CBOW 语言模型加入词性分析后，与原 CBOW 语言模型相比，不同词性的词向量被排除之后，使得相同词性的词向量聚集在一起。例如，POS-CBOW 语言模型结果集中的“本国”、“西方”、“某国”、“当今”和“各国”等不符合定义的词向量被过滤除去。为了对比 2 个模型的准确率，对 CBOW 和 POS-CBOW 语言模型计算结果进行了分析，如表 3 所示。

表 2　CBOW 语言模型和 POS-CBOW 语言模型参数

Tab. 2　Parameters of CBOW language model and POS-CBOW language model

模型	CBOW	POS-CBOW
训练集总词数	437 286 850	437 286 850
过滤后的词数	未加入过滤层	309 745 076
词向量数	426 841	423 556
Window 值	5	5
网络层数	200	200
训练线程	4	4
时间/min	314	122

表 3　相似词计算分析

Tab. 3　Analysis of similar words

模型	CBOW	POS-CBOW
样本总词数	200	200
名词数	110	136
动词数	18	0
un	47	64
其他	25	0
相似词数	135	163
准确率/%	68	82

由表 3 分析可知，针对特定输入词“中国”，CBOW 和 POS-CBOW 两种语言模型的结果集都以名词性的词为主，但是 CBOW 语言模型往往会有一些与该名词相关的动词或者其他不相关的词在内，而这些词是不符合相似词定义的。通过对比两种模型的结果，按照定义进行人工标注，CBOW 语言模型的 Top200 中相似词数约为 135，准确率约为 68%，而 POS-CBOW 语言模型则为 163，准确率约为 82%。显然，在加入词性分析后，POS-CBOW 语言模型的准确率较高。

4　结　语

本文针对微博短文本给出了一种带有词性的连续词袋模型——POS-CBOW 语言模型，通过加入过滤层

和词性标注层，对词向量空间进行优化和提升相似词计算的准确率。实验表明，POS-CBOW语言模型在词向量空间质量和相似词计算方面优于CBOW语言模型。提出的模型通过加入词性标注层，对词向量空间起到了优化作用。但对于期望词量之间能包含更多的语义关系和语法关系的要求，目前尚有改进空间，这也是未来的工作重点。

参考文献/References：

[1] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model[J]. Journal of Machine Learning Research, 2003, 3(6):1137-1155.

[2] BENGIO Y, BENGIO S. Modeling high-dimensional discrete data with multi-layer neural networks[J]. Neural Information Processing Systems(NIPS),2000(12):400-406.

[3] MIKOLOV T. Language Modeling for Speech Recognition[D]. Brno:Brno University of Technology, 2007.

[4] MIKOLOV T, KOPECKY J, BURGET L, et al. Neural network based language models for highly inflective languages[C]//IEEE International Conference on Acoustics, Speech and Signal Processing. Taipei: IEEE, 2009: 4725-4728.

[5] MIKOLOV T, KARAFIAT M, BURGET L, et al. Recurrent neural network based language model[C]//Proceedings of Interspeech. Chiba:[s. n.], 2010:1045-1048.

[6] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient Estimation of Word Representations in Vector Space[EB/OL]. http://arxiv.org/abs/1301.3781,2013-01-16.

[7] MIKOLOV T, LE Q V, SUTSKEVER I. Exploiting Similarities Among Languages for Machine Translation[EB/OL]. http://arxiv.org/abs/1309.4168,2013-09-17.

[8] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed Representations of Words and Phrases and Their Compositionality[EB/OL]. http://arxiv.org/abs/1310.4546,2013-10-16.

[9] MIKOLOV T, YIH W T, ZWEIG G. Linguistic regularities in continuous space word representations[C]// Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. [S. l.]: Association for Computational Linguistics,2013:746-751.

[10] LEVY O, GOLDBERG Y. Linguistic regularities in sparse and explicit word representations[C]// Proceedings of the Eighteenth Conference on Computational Language Learning. Maryland: Association for Computational Linguistics,2014: 171-180.

[11] QIU L, CAO Y, NIE Z, et al. Learning word representation considering proximity and ambiguit[C]//Twenty-Eighth AAAI Conference on Artificial Intelligence. California:AAAI Press, 2014: 1572-1578.

[12] SOUTNET D, MÜLLER L. Continuous distributed representations of words as input of LSTM Network language model[J]. Lecture Notes in Computer Science, 2014,8655:150-157.

[13] LE Q V, MIKOLOV T. Distributed representations of sentences and documents[C]// Proceedings of the 31st International Conference on Machine Learning. Beijing:[s. n.],2014: 1188-1196.

[14] ZHANG Qi, KANG Jiahua, QIAN Jin, et al. Continuous word embeddings for detecting local text reuses at the semantic level[C]//Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval. New York: ACM, 2014: 797-806.

[15] MNIH A, HINTON G. Three new graphical models for statistical language modelling[C]//Proceedings of the 24th International Conference on Machine Learning. New York: ACM, 2007: 641-648.

[16] BLEI D, NG A, JORDAN M. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2003(3): 993-1022.

[17] MAAS A L, DALY R E, PHAM P T, et al. Learning word vectors for sentiment analysis[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. Portland:ACL, 2011: 142-150.

[18] 章成志. 词语的语义相似度计算及其应用研究[C]//第一届全国信息检索与内容安全学术会议(NCIRCS2004). 北京：中国学术期刊电子杂志出版社, 2004: 52-60.
ZHANG Chenzhi. Measuring and application of semantic similarty between words [C]// NCLRS 2004. Beijing: China Academic Journal Electronic Publishing House, 2004: 52-60.

[19] 吴思颖，吴扬扬. 基于中文WordNet的中英文词语相似度计算[J]. 郑州大学学报(理学版)，2010，42(2)：66-69.
WU Siying, WU Yangyang. Chinese and english word similarity measure based on chinese WordNet[J]. Journal of Zhengzhou University (Natural Science Edition), 2010, 42(2): 66-69.

[20] 石静，吴云芳，邱立坤，等. 基于大规模语料库的汉语词义相似度计算方法[J]. 中文信息学报，2013，27(1)：1-6.
SHI Jing, WU Yunfang, QIU Likun, et al. Chinese lexical semantic similarity computing based on largescale corpus[J]. Journal of Chinese Information Processing, 2013, 27(1): 1-6.

[21] 郑文超，徐鹏. 利用word2vec对中文词进行聚类的研究[J]. 软件，2013，34(12)：160-162.
ZHENG Wenchao, XU Peng. Research on chinese word clustering with word2vec[J]. Software, 2013, 34(12): 160-162.

[22] 罗杰，王庆林，李原. 基于Word2vec与语义相似度的领域词聚类[C]//第三十三届中国控制会议论文集. 北京：中国学术期刊电子杂志出版社，2014：517-521.
LUO Jie, WANG Qinglin, LI Yuan. Word clustering based on Word2vec and semantic similarity[C]//Proceedings of the 33rd Chinese Control Conference. Beijing: China Academic Journal Electronic Publishing House, 2014: 517-521.