计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(1):194−199 [doi: 10.15888/j.cnki.csa.006704]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 GloVe 模型的词向量改进方法①

陈珍锐, 丁治明

(北京工业大学 信息学部, 北京 100124)
通讯作者: 陈珍锐, E-mail: chenzhenrui@yeah.net

摘　要: 使用词向量表示方法能够很好的捕捉词语的语法和语义信息, 为了能够提高词向量语义信息表示的准确性, 本文通过分析 GloVe 模型共现矩阵的特点, 利用分布式假设, 提出了一种基于 GloVe 词向量训练模型的改进方法. 该方法主要通过对维基百科统计词频分析, 总结出过滤共现矩阵中无关词和噪声词的一般规律, 最后给出了词向量在词语类比数据集和词语相关性数据集的评估结果. 实验表明, 在相同的实验环境中, 本文的方法能够有效的缩短词向量的训练时间, 并且在词语语义类比实验中准确率得到提高.

关键词: 词向量; Word2Vec; GloVe; 共现矩阵; 无关词

引用格式: 陈珍锐,丁治明.基于 GloVe 模型的词向量改进方法.计算机系统应用,2019,28(1):194–199. http://www.c-s-a.org.cn/1003-3254/6704.html

Improved Word Representation Based on GloVe Model

CHEN Zhen-Rui, DING Zhi-Ming

(Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China)

Abstract: Word vector representation is a sound way to catch the grammatical and semantic information of words. In order to improve the accuracy of the semantic information of the word, this study proposes an improved training method model based on the GloVe by analyzing the characteristics of the co-occurrence matrix and using the distributed hypothesis. This method summarizes the general rules of irrelevant words and noise words in the co-occurrence matrix from analyzing the word frequency of Wikipedia statistics. Finally, we give the evaluation results of word vector in word analogy dataset and word correlation dataset. Experiments show that the method presented in this paper can effectively shorten the training time and the accuracy of the word semantic analogy experiment is improved in the same experimental environment.

Key words: word vector; Word2Vec; GloVe; cooccurrence matrix; unrelated words

1　引言

词向量表示技术是将自然语言中的每一个词语转换为稠密向量形式. 这种表示方法能够充分的发挥计算机的计算能力, 并且在现有的自然语言处理任务中具有广泛的应用, 例如通过计算向量之间的距离表示词语的相似程度可以应用在信息检索[1]、文档分类[2]和问答系统[3]等任务.

近几年来, 有许多关于词向量表示技术的相关研究. Mikolov[4]提出了 Word2Vec 模型通过引入负采样和哈夫曼编码, 使得训练速度比传统的神经网络模型[5]

① 基金项目: 国家重点研发计划 (2017YFC0803300); 北京市教委项目 (KM201810005023, KM201810005024, KZ201610005009); 国家自然科学基金 (61402449, 61703013, 91546111, 91646201); 北京市科技计划项目 (Z161100001116072)
Foundation item: National Key Research and Development Program of China (2017YFC0803300); Project of Beijing Education Commission (KM201810005023, KM201810005024, KZ201610005009); National Natural Science Foundation of China (61402449, 61703013, 91546111, 91646201); Science and Technology Program of Beijing Municipality (Z161100001116072)
收稿时间: 2018-06-04; 修改时间: 2018-06-27; 采用时间: 2018-07-10; csa 在线出版时间: 2018-12-26

得到了大幅的提升, 也使得 Word2Vec 模型得到了广泛的应用. Pennington[6]提出了 GloVe 模型, 该模型通过利用共现矩阵分解的方式得到词向量. Vilnis 等人[7]提出了一种概率模型训练词向量的方法, 他们将每一个词映射为一个多维高斯分布然后训练该高斯分布的均值和方差, 其中均值就是对应词语的词向量. 由于 Word2Vec、GloVe 等模型并不能有效表示自然语言中广泛存在的多义词, 为了解决这个问题, 提升词向量的表示, Huang[8]引入了多原型词向量训练模型, 通过对上下文词向量聚类给出目标词的准确语义, 然后利用全局得分和局部得分定义损失函数训练多原型词向量. Facebook 智能研究室提出了 fastText 文本分类器模型[9–11], 该模型可以在保证分类质量的同时, 大大缩短文本分类的训练时间. FastText 模型也能够用来训练词向量, 通过借鉴 Skip-gram 构建哈夫曼编码树思路加速模型的计算. 但是 fastText 通过加入 n-gram 特征保留了词语的语序信息提高分类的准确率, 同时也使得其训练的词向量能够对生成的低频词词向量有较好的表示效果, 并且对于词典以外的单词, 可以通过叠加它们字符级别的 n-gram 向量获得该词典外单词的词向量.

本文提出一种改进的 GloVe 模型训练词向量的方法. GloVe 模型是一种基于共现矩阵分解生成词向量的基本模型, 同 Word2Vec、wordrank[12]等模型一样可以根据语料库的统计信息训练词向量并捕捉词语间的语法和语义信息. 本文通过分析 GloVe 模型共现矩阵的特点, 利用分布式假说[13,14]过滤不能够代表目标词语义的无关词和噪声词, 减少共现矩阵中非零元素数目, 进而提高模型的训练速度. 最后,利用 Mikolov[15]提出的词汇类比方法和人为标注的词汇相似性数据集对词向量的训练效果给出评价.

2 研究基础

2.1 Global Vector 模型 (GloVe)

GloVe 模型是一种对 “词-词”共现矩阵 X 分解而得到的词向量表示方法. 共现矩阵 X 中的第 i 行第 j 列的值 X_{ij} 为目标词 V_i 与上下文词 V_j 在语料库中的共同出现次数. GloVe 模型由于只在全局的非零矩阵元素 X_{ij} 上进行训练, 其训练速度比 Word2Vec 更加高效. 该模型使用最小二乘法作为损失函数, 同时对共现矩阵 X 中的行和列加入了偏移项. 其损失函数为:

$$J=\sum_{i,j=1}^{|V|} f(X_{ij})(w_i^{\mathrm{T}}w_j+b_i+b_j-\log X_{ij})^2 \tag{1}$$

其中, $|V|$为词典的大小, w_i 为目标词的词向量, w_j 为上下文词的词向量, b_i, b_j 为共现矩阵 X 行和列的偏移值, $f(x)$ 是一个加权函数, 用于对从语料库中统计的低频词对进行衰减, 减少低频噪声带来的误差, 其定义为:

$$f(x)=\begin{cases}(x/x_{\max})^{\alpha} & \text{if}\, x\leqslant x_{\max}\\ 1 & \text{otherwise}\end{cases} \tag{2}$$

同时 GloVe 模型作者 Pennington 给出了 $x_{\max}$, α 的经验值分别为 100, 3/4.

2.2 分布式假设

我们知道, GloVe 模型的共现矩阵中的值 X_{ij} 是通过滑动窗口对语料库中所有存在窗口内的目标词与上下文词的词频统计信息. 根据分布式假说[13,14]如果两个词语具有相似的上下文那么它们的语义相近. 为了能够清晰的表达这种相似, 我们从维基百科中抽取出以下三个句子:

(1) Several species **of** **pear** are valued for their edible **fruit** and juices.

(2) The **peach** is seen as the **fruit of** happiness, riches, honours and longevity.

(3) One type **of** commonly known **gas** is **steam**.

对于上面句子划线的三个词 pear, peach 和 gas, 由我们的经验可知词对 (pear, peach) 关系要大于词对 (pear, gas) 或 (peach, gas) 之间的关系, 因为前者共同属于水果类别, 而后者并没有特别明显的相关性. 同时, 词对 (pear, peach) 相关性我们可以从句子 (1), (2) 中看出,它们拥有共同上下文词语 fruit、of 等, 如果当语料库足够大时在 pear 和 peach 的上下文中会有更多的词语代表它们的共同含义例如 tree、leaf、delicious 等, 但是 gas 的上下文中几乎不可能出现 tree, delicious 等上下文词语,这说明词语 fruit, tree, delicious 能够代表 pear 和 peach 的语义信息而不能表示 gas 的语义信息. 但是在上面的句子中也存在大量的既不能代表 pear 和 peach 的语义, 也不能代表 gas 语义的词语例如 of, and, is 等无关词或噪声词, 并且在对语料库做统计的过程中会存在大量的无关词或者噪声词, 它们的存在使得模型训练需要较长的时间, 同时由于引入噪声对词向量的训练质量造成一定的影响, 所以这些词语是没有必要参与公式 (1) 的运算.

3 研究方法

3.1 共现矩阵分析

设共现矩阵 X 中每一个元素 X_{ij} 表示上下文词 j 在目标词 i 窗口内出现的次数. $X_i = \sum_{k=1}^{|v|} X_{ik}$为共现矩阵 X 第 i 行的和, 即在目标词 i 窗口内所有上下文词出现的总次数. $P_{ij} = P\{j|i\} = X_{ij}/X_i$表示词 j 出现在词 i 周围的概率. 现在考虑三个目标词 a, b, c, 为了能够更加明确的说明词语之间的关系, 与上节类似假设 a 为 pear, b 为 peach 和 c 为 gas, 其中 a 与 b 代表的都是一种水果, 它们的相似度要远大于 a 和 c 或 b 和 c, 它们的关系可以通过利用共现矩阵中的比率得出. 表 1 给出了从维基百科中抽取的词 pear, peach, gas 和它们上下文词之间的共现概率统计信息.

表 1 pear, peach 和 gas 与他们上下文词共现概率及关系

概率	k=tree	k=the	k=fruit	k=steam	k=delicious	k=of
P(k|pear)	8.9×10^{-3}	7.8×10^{-2}	2.0×10^{-3}	1×10^{-5}	1.2×10^{-4}	3.8×10^{-2}
P(k|peach)	2.6×10^{-3}	7.5×10^{-2}	5.3×10^{-3}	3×10^{-5}	3×10^{-5}	3.1×10^{-2}
P(k|gas)	2.2×10^{-5}	7.4×10^{-2}	2.0×10^{-5}	5.5×10^{-4}	1×10^{-6}	3.5×10^{-2}
P(k|pear)/P(k|peach)	3.42	1.04	1/2.65	1/3	4	1.22
P(k|pear)/P(k|gas)	404.55	1.05	100	1/55	120	1.09

从表 1 可以看出 pear 和 peach 的上下文为 k 时, 比如当 k 为 tree, delicious, fruit 时它们的概率比 P(k|pear)/P(k|peach) 接近于 1, 但是对于 P(k|pear)/P(k|gas) 的比值却远远的大于 1 或小于 1, 也就是越相似的词它们的上下文词的共现概率比值越接近于 1, 越不相似的词它们的上下文词的共现概率比值越不接近与 1. 当 k 为 the, of 等无关词时, P(k|pear)/P(k|gas) 的比值也接近于 1. 通过分析可以概括出以下规律:

1) 对于词义相似的词 a,b 和它们的上下文词 k 有:

$$\begin{cases} \frac{P_{ak}}{P_{bk}} \approx 1 & \text{if } P_{ak} \geqslant P_{bk} \\ \frac{P_{bk}}{P_{ak}} \approx 1 & \text{if } P_{ak} < P_{bk} \end{cases} \tag{3}$$

2) 对于词义不相似的两个词 a,c 和上下文词 k, 当 k 不为无关词, 则有:

$$\begin{cases} \frac{P_{ak}}{P_{ck}} \gg 1 & \text{if } P_{ak} \geqslant P_{ck} \\ \frac{P_{ck}}{P_{ak}} \gg 1 & \text{if } P_{ak} < P_{ck} \end{cases} \tag{4}$$

当 k 为无关词时, 此时可以得到与公式 (3) 相似的公式:

$$\begin{cases} \frac{P_{ak}}{P_{ck}} \approx 1 & \text{if } P_{ak} \geqslant P_{ck} \\ \frac{P_{ck}}{P_{ak}} \approx 1 & \text{if } P_{ak} < P_{ck} \end{cases} \tag{5}$$

3.2 过滤共现矩阵

从上节的分析可以看出当词语 a,c 不相似时,给定上下文词 k 可以从他们的的共现概率比值中获得 k 是否为无关词信息. 例如从表 1 中我们发现当上下文词 k 为 tree, steam 时, P(tree|pear)/P(tree|gas)=404.55, P(steam|gas)/P(steam|pear)=55,它们的比值都要远远大于 1, 所以上下文词 tree, steam 不为无关词, 并且 tree 能够用来表示 pear 的语义, 而 steam 表示 gas 的语义. 而上下文词为无关词 the, of 时, P(the|pear)/P(the|gas)=1.05、P(of|gas)/P(of|pear)=1.09 它们的共现概率比值接近于 1. 如果假设词语 a、c 在给定上下文词 k 时, 共现概率比值为:

$$\begin{cases} \frac{P_{ak}}{P_{ck}} = \gamma & \text{if } P_{ak} \geqslant P_{ck} \\ \frac{P_{ck}}{P_{ak}} = \gamma & \text{if } P_{ak} < P_{ck} \end{cases} \tag{6}$$

其中, γ 为设置的超参数. 我们可以归纳以下结论:

若 a, c 为不相似的两个词, 给定上下文词 k 时:

1) 若共现概率比值 $\gamma\approx1$ 时 (公式 (5)), 此时 k 为无关词;

2) 若共现概率比值 $\gamma>>1$ 时 (公式 (4)), 此时 k 可以用于代表 a 或 c 的语义词.

根据上面的结论我们考虑如何选择与 a 不相似的词 c 来过滤无关词. 例如要过滤出 pear 上文中的无关词,如何选择与 pear 不相似的词语 gas. 这里给出了以下式子选择与 a 不相似的词语 c:

$$\text{set}(c) = \{c|\cos(w_a, w_c) < 0, c \in V\} \tag{7}$$

其中, $a, c \in V$, w_a, w_c 分别为 a, c 对应的词向量. 由分布式假设我们知道如果两个词语上下文越相似, 那么它们的语义越相似, 它们的余弦距离越大, 它的逆否命题为如果两个词语的余弦距离越小, 那么这两个词语的上下文越不相似, 它们的语义相差越远, 所以两个词语的相似关系可以使用余弦距离给出. 公式 (7) 给出选择

不相似词语的一般公式, 即从所有与 a 的余弦距离小于 0 的集合 c 中随机的选择 N 个不相似的词语过滤 a 中的无关词, 这样既可以减小共现矩阵中非零元素数量, 加快训练时间, 又可以使得实验效果得到改进, 然后结合公式 (6) 给出的结论过滤出无关词.

总而言之, 为了能够过滤出共现矩阵中的无关词与噪声词, 提高词向量的训练质量, 并加快词向量的训练效率, 本文方法可以概括为以下三个步骤:

1) 对于词典中的每个词 a 利用余弦距离找到与其余弦距离小于 0 的词语集合 set(c);

2) 从集合 set(c) 中随机的选择 N 个词语, 结合公式 (6) 及其结论从原共现矩阵中过滤出无关词、噪声词, 生成新的共现矩阵;

3) 将新的共现矩阵代入公式 (1) 训练新的词向量.

4 实验分析

我们使用维基百科数据集 (wiki2010) 训练词向量.该数据集包含大约 10 亿个单词, 用 NLTK 工具包中的方法对该数据集进行分词, 并且将所有大写字母转换为小写形成新的语料库, 然后使用 30 000 个出现最频繁的单词形成字典, 利用字典和左右为 10 的滑动窗口处理新的语料库构成共现矩阵.

在使用公式 (1) 训练后每个单词会得到两份词向量分别是目标词向量 W 和上下文词向量$\hat{W}$, 因为我们使用左右为 10 的滑动窗口生成的共现矩阵为对称矩阵, 所以 W, $\hat{W}$也是几乎相等的, 只是由于它们的初始值不同而造成稍微不同[6], 另外 Ciresan[16]指出对于某种类型的神经网络, 通过结合该网络中的多个参数可以帮助减少过度拟合和噪声的影响, 从而改善词向量的训练结果. 本文使用了 $W+\hat{W}$作为最终的词向量, 这样做可以在词语语义类比实验中增加准确率. 本文若无特别指出, 其他的相关的参数如 $x_{\max}$, α 等与 Pennington[6]在 GloVe 模型中设置相同.

在对词向量训练效果评价上, 我们使用 Mikolov[4]提出的词语类比数据集进行实验, 该类比数据集包含 19 544 个问题, 分为语义类比和语法类比两部分. 另外, 本文给出了超参数 N, γ 在不同取值时对词向量质量的影响, 同时在几个标准的词语相似度数据集上对训练的词向量质量进行评估.

4.1 词语相似度

我们使用了标准的词语相关性数据集对训练的词向量结果进行评估,其中包括 SimLex[17], WordSim-353、WS-S (similarity)、WS-R (relatedness)[18], MC[19], MEN[20], RG[21], YP[22]和 RW[23]. 这些数据集包含一系列的单词对列表, 每个单词对的相似度都有人为的评分, 我们通过计算人为打分和训练的词向量之间的皮尔逊系数[24]得出词向量与人为打分之间的相关性, 皮尔逊系数越高, 则相关性越大, 词向量的训练效果也就越好.

表 2 不同的词向量训练模型在词语相似度数据集上的皮尔逊系数

模型	SL	WS	WS-R	WS-S	MC	MEN	RG	RW	YP
Skip-gram	37.56	**68.6**	61.76	**79.08**	**84.80**	73.74	77.07	50.57	46.80
CBow	**38.87**	66.03	56.58	77.29	84.78	73.52	**80.06**	51.38	39.97
LG	32.23	65.49	58.96	76.15	70.41	71.31	71.00	**53.74**	41.50
Glove	35.90	68.08	**64.04**	76.97	82.81	73.53	78.08	50.52	43.83
Glove-r15	38.01	67.59	62.77	75.56	81.51	**74.59**	78.82	49.64	**50.10**

表 2 给出了 Skip-gram、CBow、LG (Gaussian embedding)[7]、Glove 以及 GloVe-r15 在词语相似度数据集的皮尔逊系数. 所有这些模型使用相同的维基数据集, 词向量维度均为 300 维, 其中 GloVe-r15 使用了 N=3, γ=15 的训练结果. 从总体上来看, 这些不同模型在所有的词语相似度数据集得到的皮尔逊系数都非常接近, 并且各个模型在不同的数据集皮尔逊系数都有最大值. Skip-gram 在 WS 和 MC 数据集上的皮尔逊系数最高. 本文模型 GloVe-r15 在数据集 MEN, YP 上取得较好的效果, 在其它不同的数据集上与其它模型差距也非常小. 从表格中看出, 我们的方法可以有效的捕捉词语相似性关系, 并且与其它模型训练在词语相似度皮尔逊系数也非常接近.

4.2 词语类比

Mikolov[4]提出了通过词语类比实验检验词向量的训练质量. 该方法主要通过回答类似于“如果 a 与 b 相似, 正如 c 与 d 相似”的问题, 这些问题又分为语义问题和语法问题两部分. 语义问题主要是对称谓和地点进行类比检验, 比如“boy”与“girl”相似, 正如“brother”与“sister”相似. 语法问题主要来对动词时态, 单复数等形式进行验证, 比如“dance”与“dancing”相似, 正如“decrease”与“decreasing”相似. 在计算的过程中, 方法是假设实验中的某个单词是未知的, 例如假设 d 未知, 通过余弦距离找到与 $w_b-w_a+w_c$ 最相近的词向量 $w_{d'}$, 检查词 d'是否与 d 一致, 若一致则认为类比正确.

表 3 给出了使用不同模型在不同的维度下词语类比实验准确率, 其中 GloVe-r15 是本文提到的方法将 N 设置为 3, γ 设置为 15 时所得到的实验结果. 我们主

要对比了流行的词向量训练方法 Skip-gram, CBow, GloVe 模型, 同时给出了 fasttext 在词向量 300 维时的训练结果.这些模型使用了 wiki2010 数据集进行训练, 并且词典大小、滑动窗口大小等超参数也都尽可能的保持相同. 从表 3 中可以得出以下结论.

表 3 不同模型在不同词向量维度下词语类比实验准确率 (%)

模型	词向量维度	训练时间 (s)	Sem	Syn	Tot
Skip-gram	100	15 300	62.99	43.46	51.04
CBOW	100	6003	24.42	17.54	20.21
GloVe	100	6667	78.35	**53.77**	63.31
GloVe-r15	100	**5341**	**80.99**	52.58	**63.61**
Fasttext-ngram	300	93 600	27.52	23.65	25.15
Skip-gram	300	18 720	68.70	48.45	56.31
CBOW	300	9060	34.37	28.98	31.07
GloVe	300	9360	83.00	**56.91**	67.04
GloVe-r15	300	**7440**	**84.77**	55.50	**67.86**
CBOW	1000	24 843	68.90	57.31	63.7
Skip-gram	1000	55 920	74.79	56.43	65.60
GloVe	1000	27 360	86.10	**59.26**	69.68
GloVe-r15	1000	**21 960**	**87.85**	58.90	**70.14**

1) 随着词向量维度的增加, 所有模型的词语类比实验准确率都在提高. 因为维度越大词向量对共现矩阵信息拟合的越准确, 故而词语类比实验准确率也会越高.

2) fasttext 训练时间最长, 除 fasttext 之外, 在其他模型的相互比较中 Skip-gram 在相同的维度下训练时间最长. fasttext 主要是用于文本分类, 通过使用字符间的 n-gram 信息提高分类的准确性. 我们在训练词向量时也加入词的 n-gram 特征, 其在训练过程中主要训练单词的组成成分[9], 需要训练的词向量也由原来的 K 扩大为 K'(K'的取值与 n-gram 的 n 取值范围有关), 由于 fasttext 采用了层次 Softmax 优化[10], 需要训练词向量 K 大小变化相对于算法复杂度$O(d\log_2(K))$影响较小. 其性能损失主要来源于损失函数计算和反向传播过程中由只对单词的计算变为对单词组成成分计算, 进而需要更多的处理器和内存交互而影响算法性能. 但是如果不在 fasttext 加入 n-gram 特征, 那么 fasttext 模型将退化为 Skip-gram 模型, 从而失去比较的意义. 通过加入 n-gram 语法可以获得词典外单词的词向量, 扩展词典的表示范围, 同时 Bojanowski 在论文第六节[9]中给出了 n-gram 语法在词语语义相似度和词素关系的定性分析.

3) 在相同的词向量维度下本文的方法是所有相关模型中用时最短, 语义类比实验结果最好的模型, 但是在语法类比实验中其准确率相对于原始 GloVe 模型在下降, 这是因为在使用分布式假设过滤共现矩阵时主要考虑的是语义信息, 使得模型在语法信息捕捉下降. 我们可以通过引入对单词的语法信息提高语法实验准确性, 本文第 5 节给出了处理方法.

由于词向量的训练质量会受到 N, γ 值的影响, 图 1 给出了当 N=3 时, γ 不同取值时词语类比实验准确率. 从图 1 可以发现, γ 对于词向量的训练质量影响较小. 但是当 γ 取较大值时准确率有稍微下降. 图 2 给出了 γ=15, N 的不同取值时对实验准确率的影响. 当 N 取较大值时, 词语类比实验的准确率有明显下降, 实验表明, 当 N 取值在[3, 7]之间, γ 在[10, 20]区间时可以取得相对较好的实验结果.

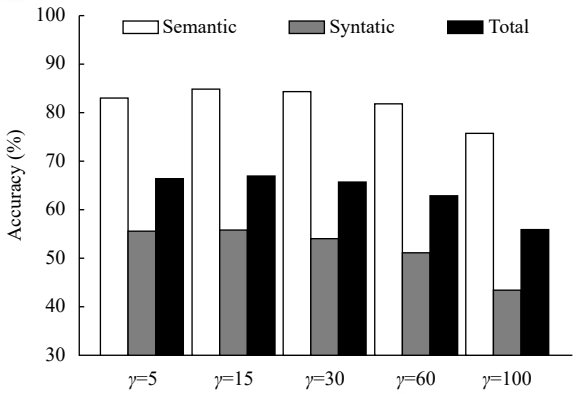

图 1 N=3, γ 不同取值时词语类比实验准确率

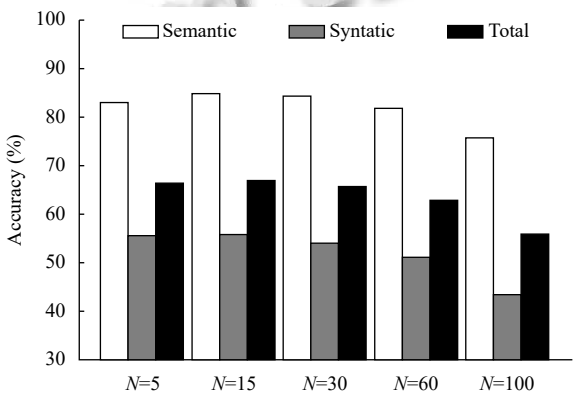

图 2 γ=15, N 不同取值时词语类比实验准确率

5 结论与展望

本文通过分析 GloVe 模型共现矩阵特点, 提出了一种过滤出共现矩阵中无关词的方法, 该方法可以在不影响词向量质量的前提下, 缩短词向量的训练时间, 并且能够更好的捕捉词语间的语义信息. 但是, 本文方

法虽然在语义类比实验得到提升,同时也会造成语法类比实验结果的下降, 未来可以从两个方向利用语法信息对实验结果进行改进: 1) 保留共现矩阵中形态变化词的词频统计信息. 即使用 NLTK 对词典进行词形归一化, 找到所有能够进行词形归一化的形态变化词,保留这些的词频统计信息, 只对非形态变化词进行共现矩阵中词频统计处理. 2) 借鉴 fasttext 在词语层面的 n-gram 思路. 将形态变化词进行词干提取形成词干+词尾的形式, 此时$w_i=w_s+w_l$其中 w_s 为词干词向量, w_l 为词尾词向量, 并且词尾词向量表示了语法信息. 由训练词向量 w_i 变为训练词向量 w_s 和 w_l.

最后, 本文给出一种提升词语语义相似度新方法, 实验表明, 该方法在语义相似度实验中比传统词向量训练方法能够获得了更好的结果, 也为深度学习在自然语言处理上层应用提供了更好的表示.

参考文献

1 Manning CD, Raghavan P, Schütze H. Introduction to Information Retrieval. New York, NY, USA: Cambridge University Press, 2008.

2 Sebastiani F. Machine learning in automated text categorization. ACM Computing Surveys, 2002, 34(1): 1–47. [doi: 10.1145/505282.505283]

3 Tellex S, Katz B, Lin J, *et al*. Quantitative evaluation of passage retrieval algorithms for question answering. Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. Toronto, Canada. 2003. 41–47.

4 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. arXiv: 1301.378, 2013.

5 Bengio Y, Ducharme R, Vincent P, *et al*. A neural probabilistic language model. The Journal of Machine Learning Research, 2003, 3: 1137–1155.

6 Pennington J, Socher R, Manning CD. GloVe: Global vectors for word representation. Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing. Doha, Qatar. 2014.

7 Vilnis L, McCallum A. Word representations via gaussian embedding. arXiv:1412.6623, 2014.

8 Huang EH, Socher R, Manning CD, *et al*. Improving word representations via global context and multiple word prototypes. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers. Jeju Island, Korea. 2012. 873–882.

9 Bojanowski P, Grave E, Joulin A, *et al*. Enriching word vectors with subword information. Transactions of the Association of Computational Linguistics, 2017, 5(1): 135–146.

10 Joulin A, Grave E, Bojanowski P, *et al*. Bag of tricks for efficient text classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers 2017. 2017. 427–431.

11 Joulin A, Grave E, Bojanowski P, *et al*. FastText.zip: Compressing text classification models. arXiv:1612.03651, 2016.

12 Ji SH, Yun H, Yanardag P, *et al*. WordRank: Learning word embeddings via robust ranking. Proceedings of 2016 Conference on Empirical Methods in Natural Language Processing. Austin, TX, USA. 2016. 658–668.

13 Harris ZS. Distributional structure. WORD, 1954, 10(2–3): 146–162. [doi: 10.1080/00437956.1954.11659520]

14 Firth JR. A Synopsis of Linguistic Theory 1930–1955. In: Studies in Linguistic Analysis. Oxford: The Philological Society, 1957: 1–32.

15 Mikolov T, Yih WT, Zweig G. Linguistic regularities in continuous space word representations. Proceedings of North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Atlanta, GA, USA. 2013. 746–751.

16 Cireşan DC, Giusti A, Gambardella LM, *et al*. Deep neural networks segment neuronal membranes in electron microscopy images. Advances in Neural Information Processing Systems, 2015, 25: 2852–2860.

17 Hill F, Reichart R, Korhonen A. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics, 2015, 41(4): 665–695.

18 Finkelstein L, Gabrilovich E, Matias Y, *et al*. Placing search in context: The concept revisited. ACM Transactions on Information Systems (TOIS), 2002, 20(1): 116–131. [doi: 10.1145/503104.503110]

19 Miller GA, Charles WG. Contextual correlates of semantic similarity. Language and Cognitive Processes, 1991, 6(1): 1–28. [doi: 10.1080/01690969108406936]

20 Bruni E, Tran NK, Baroni M. Multimodal distributional semantics. Journal of Artificial Intelligence Research, 2014, 49(1): 1–47.

21 Rubenstein H, Goodenough JB. Contextual correlates of synonymy. Communications of the ACM, 1965, 8(10): 627–633. [doi: 10.1145/365628.365657]

22 Yang DQ, Powers DW. Verb similarity on the taxonomy of WordNet. Proceedings of the 3rd International WordNet Conference. Jeju Island, Korea. 2006.

23 Luong MT, Socher R, Manning CD. Better word representations with recursive neural networks for morphology. Proceedings of the Seventeenth Conference on Computational Natural Language Learning. Sofia, Bulgaria. 2013. 104–113.

24 Spearman C. The proof and measurement of association between two things. The American Journal of Psychology, 1904, 15(1): 72–101. [doi: 10.2307/1412159]