利用 word2vec 对中文词进行聚类的研究

郑文超，徐鹏

（北京邮电大学网络技术研究院，北京 100876）

摘 要：文本聚类在数据挖掘和机器学习中发挥着重要的作用，该技术经过多年的发展，已产生了一系列的理论成果。本文在前人研究成果的基础上，探索了一种新的中文聚类方法。本文先提出了一种中文分词算法，用来将中文文本分割成独立的词语。再对处理后的语料使用 Word2Vec 工具集，应用深度神经网络算法，转化为对应的词向量。最后，将词向量之间的余弦距离定义为词之间的相似度，通过使用 K-means 聚类算法将获取的词向量进行聚类，最终可以返回语料库中同输入词语语意最接近的词。本文从网络上抓取了 2012 年的网络新闻数据，应用上述方法进行了实验，取得了不错的实验效果。

关键词：数据挖掘；聚类；分词；词向量；神经网络

中图分类号：TP39 **文献标识码：**A DOI：10.3969/j.issn.1003-6970.2013.12.040

本文著录格式：[1] 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

Research on Chinese word Clustering with Word2vec

ZHENG Wen-chao,XU Peng

(Beijing University of Posts & Telecommunications Institute of Network Technology, Beijing 100876*, China)*

【Abstract】Text clustering plays an important role in data mining and machine learning. After years of development, clustering technology has produced a series of theorey. This paper explored a new method of Chinese clustering. By putting forword a new method to Chinese word segments, this paper can split Chinese text into word segments. With Word2Vec toolset, we can transfrom word segments into vectors. To define the cosine distance between two vectors, we can apply K-means algorithm on the vectors to cluster words. In this paper, we downloaded network news text on the Internet, and applied the methods above, which shows good result.

【Key words】data mining; clustering; word segment; word vector; neural networks

0 引言

在自然语言处理领域，中文词聚类算法是被深入研究的课题。由一些属性相近的词组成的词可以看成是单个词语到语意一般概念的映射。词聚类算法对信息检索，语音识别等诸多领域都有使用价值。针对英语的研究中各种词聚类算法可以分为三种：第一，以各种启发式量度表示聚类过程中的元素的距离；第二，以统计模型给出距离量度并给定聚类结果的类总数；第三，同样以统计模型给出距离量度，但增加某种量度如困惑度的数目增长和减少。目前，针对中文已有一些研究，但计算结果似乎没有英语那么成功。

本文针对这种现状，本文将中文词语看成一系列独立词的“词袋模型”，这种模型将语言中词语之间的关系做了简化，仅仅考虑词语的统计特性；之后使用深度神经网络算法将词转化为 n 维向量，它在传统三层神经网络算法的基础上做了延伸，将网络从三层扩展到多层；最后用 k-mean 算法计算对这些向量进行聚类。本文使用这种方法，应用 word2vec 工具集进行了测试，最终取得了不错的结果。

1 算法设计

1.1 词袋模型

“词袋模型”是在自然语言处理和信息检索中的一种常见模型。它将文本中出现的词汇，想象成放在袋子中的零散而独立的物品，这样一来一个“袋子”就能代表一份文档。在这种模型中，文本、段落或者文档都被看作是无序的词汇集合，忽略语法甚至是单词的顺序。如果一个词在文档中出现不止一次，这可能意味着包含该词是否出现在文档中所不能表达的某种信息。[1] 在应用“词袋模型”之前，我们需要先将一段完整的文本处理成单个词的序列，即对文本进行分词。

1.2 中文分词

由于中文词之间是不存在明显的间隔的。我们设计了一种方法将连续的中文文本切成一系列词组的方法。现有的分词方法大致可以分为基于词典的匹配、基于概率统计的方法和基于语法规则的方法。本文使用的分词方法将词典和基于统计的方法结合起来：首先使用中文词库对原始文本进行过滤，如果某

作者简介：郑文超（1988-），男，硕士研究生，主要研究方向：云计算、信息检索

通信联系人：徐鹏（1977-），男，副教授，主要研究方向：下一代网络、云计算 .

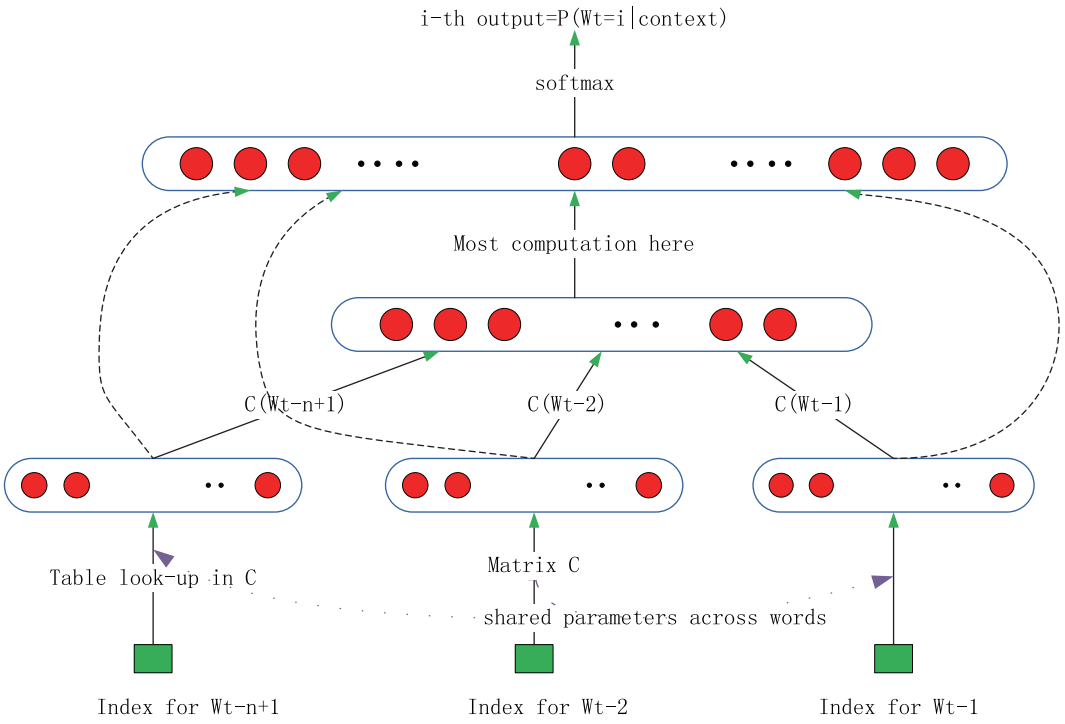

图 1 神经网络模型

Fig.1 neural networks model

个词汇在词典中被发现则直接作为单个词识别出来。否则，跳转到如下算法。每次从待处理的文本中，按照从左向右的顺序，识别出多种不同的 3 个词的组合；然后根据下面的 4 条消除歧义规则，确定最佳的备选词组合；选择备选词组合中的第 1 个词，作为 1 次迭代的分词结果；剩余的 2 个词进行下一轮的分词运算。采用这种办法的好处是，为传统的前向最大匹配算法加入了上下文信息，解决了其每次选词只考虑词本身，而忽视上下文相关词的问题。4 条规则包括：

1) 备选词组合的长度之和最大

2) 备选词组合的平均词长最大

3) 备选词组合的词长变化最小

4) 备选词组合中，单字词的出现频率统计值最高

经过分词处理之后，原始的文本会被处理成一系列由空格隔开的单个词。接下来可以使用神经网络算法构建语言模型，在构建语言模型的过程中得到词向量。

1.3 语言模型

语言模型是借由一个机率分布，指派概率给特定的字符串 $P(w_1,w_2,\cdots w_n)$，其中 w_n 代表语言模型中的某个词 [2]。语言模型多用于自然语言处理，如语音识别，机器翻译，词性标注，句法分析和资讯检索。由于字词与句子都是任意组合的长度，因此在训练过的语言模型中会出现未曾出现的字串 (资料稀疏的问题)，也使得在语料库中估算字串的机率变得很困难，本文在这里采用近似平滑的 n 元语法以避免零概率问题（N-gram 模型），作为基础模型。在 N-gram 模型下，每个词的概率仅与它前边的 N 个词有关，$P(w_i^T) = P(w_i|w_{t-i}+1,\cdots,w_{t-1})$。更进一步的，如果我们假设前 n 元词相互独立，可以将概率展开 $P(w_i^T) = \prod P(w_t|w_i^{t-1})$。

1.4 神经网络算法

本文采用的神经网络算法由 Bengio 在 2003 年提出 [3]。Bengio 用了一个三层的神经网络来构建语言模型。另外，他还假设这种语言遵循 n-gram 语言模型。(图 1)

最下方的 $w_{t-n+1},\cdots,w_{t-2},w_{t-1}$ 就是前 n－1 个词，模型会根据这 n－1 个词预测下一个词 wt。C(w) 表示词 w 所对应的词向量，整个模型中使用的词向量，存储于矩阵 C，C 的维数为 |V|×M。其中 |V| 表示词表的大小，M 表示词向量的维度。w 到 C(w) 的转化就是从矩阵中取出一行。

网络的输入层（第一层）是将 C(wt-n+1),…,C(wt-2),C(wt-1) 这 n-1 个向量首尾相接拼起来，形成一个 (n-1)×m 维的向量，下面记为 x。网络的隐藏层（第二层）如同普通的神经网络，直接使用 d+Hx 计算得到，其中 d 为一个偏置项，初始化值随机。使用 tanh 作为激活函数。网络的第三层（输出层）节点 y_i 表示，一共有 |V| 个节点。下一个词为 i 的未归一化 log 概率。最后使用 softmax 激活函数 [4] 将输出值 y 归一化成概率。最终 y 的计算公式为：y = b + Wx + U*tanh (d+Hx)。

式子中的 U 是隐藏层到输出层的参数，它是一个 |V|×h 维的矩阵，整个模型的多数计算集中在 U 和隐藏层的矩阵乘法中。式子中还有一个矩阵 W，这个矩阵包含了从输入层到输出层的

表 1 同“中国”最为接近的 top10 个词

Tab. 1 top 10 words most similar with ‘China’

接近度 top10 的词	余弦距离
亚洲	0.445374
海外	0.437304
核工业	0.436195
电信	0.420332
日本	0.414920
电影家	0.408968
我国	0.408262
世界	0.396693
乒乓球队	0.390933
美国	0.360211

表 2 同“计算机”最为接近的 top10 个词

Tab. 2 top 10 words most similar with ‘Computer’

接近度 top10 的词	余弦距离
应用	0.756215
微机	0.692575
编程	0.692512
传感	0.654025
嵌入式	0.634722
人工智能	0.630025
软件	0.629519
自动化	0.621523
软硬件	0.620579
电脑	0.614451

直连边。一般将将 W 置为 0。最后使用随机梯度下降法把这个模型优化，当优化结束之后，我们就从输出 y 中获取了某个词对应的词向量。获取到词向量表之后，我们就可以计算不同词向量直接的余弦距离，作为不同词之间的“距离”。在这之后，使用 K-means 聚类算法将近似的词聚集到一起。

1.5 K—means 聚类

K-means 算法接受输入量 k，然后将 n 种数据对象划分为 k 组聚类以便使得所获得的聚类满足：同一聚类中的对象相似度较高；而不同聚类中的对象相似度较小。聚类相似度是利用各聚类中对象的均值所获得一个“中心对象”来进行计算的。[5]

算法基本步骤：

1) 从 n 个数据对象任意选择 k 个对象作为初始聚类中心；

2) 根据每个聚类对象的均值，计算每个对象与这些中心对象的距离；并根据最小距离重新对相应对象进行划分；

3) 重新计算每个（有变化）聚类的均值；

4) 计算标准测度函数，当满足一定条件，如函数收敛时，则算法终止；如果条件不满足则回到步骤 2。

2 实验过程及结果分析

遵循以上过程，本文对改算法的效果进行了验证。语料库使用了“搜狗实验室”的“互联网语料库”，这个语料库中主要是互联网上过往的新闻数据，使用 XML 格式化的方式进行存储。总数据量为 2TB，本文使用了其中十分之一的文本数据，将数据格式化之后进行清洗，只保留文档的正文内容，去除标点符号。再应用上文中提到的中文分词方法进行分词，分词过程中使用的词典为“搜狗细胞词库”。

之后应用 word2vec 工具计算词向量，将输入的文本转换为词向量数据，写入文件。Word2vec 是由 Google 的研究人员发布的神经网络工具包，它完成了上文中所说的“连续词袋模型”，并且对这种方法进行了一些改进。能够将输入文本中的词转化为一系列词向量。这个工具集已经开始应用在自然语言处理的许多应用中。最后应用 K-means 算法，对输入的词进行聚类。将相同的词聚集在一起，表 1、表 2 中列出了部分实验结果。

3 结论

基于统计的中文词分类在自然语言处理领域有着重要的应用。机器自动生成的词类可以取代文法的词类；在分类基础上建立的语言模型可以应用于语音识别、OCR、汉字智能输入等许多领域。众所周知，基于词的语言模型在自然语言处理的许多方面取得了巨大的成功。然而，基于词的语言模型也存在着许多的问题，如参数空间庞大，训练数据不足，数据稀疏等。词的分类可以在一定程度上解决上述问题。本文使用 word2vec 工具集，找到了一种较为方便的方法对中文词进行聚类，并且取得了不错的效果。

参考文献

[1] 曾元颖 . 词袋模型 [OL]. [2012.10.12]. http://terms.naer.edu.tw/detail/1679006/
Zeng,Y Y. Word Bags Model[OJ][2012.10.12] http://terms.naer.edu.tw/detail/1679006/ (In Chinese)

[2] 维基百科 . 语言模型 [OL]. [2013-3-12]. http://zh.wikipedia.org/zh-cn/%E8%AA%9E%E8%A8%80%E6%A8%A1%E5%9E%8B
Wikipedia. Language Model[OL].[2013-3-12]. http://zh.wikipedia.org/zh-cn/%E8%AA%9E%E8%A8%80%E6%A8%A1%E5%9E%8B (In Chinese)

[3]Yoshua B,Rejean D,Pascal V,Christian J. A Neural Probabilistic Language Model[J]. Journal of Machine Learning Research, 2003, 3(0): 1137–1155

[4] 曾俊瑀，王方 . Softmax 回归 [OL]. [2013-8-13]. http://ufldl.stanford.edu/wiki/index.php/Softmax%E5%9B%9E%E5%BD%92
Zeng,J Y, Wang F. Softmax regression[OL].[2013-8-13]. http://ufldl.stanford.edu/wiki/index.php/Softmax%E5%9B%9E%E5%BD%92 (In Chinese)

[5] 袁方，周志勇，宋鑫 . 初始聚类中心优化的 k-means 算法 [J]. 计算机工程 , 2007, 33(3):65-69
Yuan F, Zhou Z Y, Song X. Initial Cluster Center Optimization Algorithm[J]. Computer Engineer, 2007, 33(3):65-99