无关语获取与语料聚类方法研究

周　峰[1],朱俊武[1],童　林[1,2],陈伟聪[3],陈　波[1]

(1. 扬州大学信息工程学院,扬州 225127)
(2. 中科院计算技术研究所智能信息处理开放实验室,北京 100190)
(3. 伊凡斯维尔大学计算机科学和应用数学系,印弟安纳州 美国 47722)

[**摘要**]　剔除无关语及语料聚类对提高自然语言理解的质量具有重要意义,也是自然语言理解的预处理关键技术. 鉴于无关语在语料中存在明显的特性,本文通过种子无关语推导出强无关语,并依据强无关语识别并导出新的无关语;然后,基于 2-gram 构建句子之间的相似性,利用层次法对语料进行聚类对 QA 语料进行问题相似的聚类. 最后,通过识别的新无关语实验及语料聚类实验,验证本文提出方法的有效性.

[**关键词**]　无关语,获取,识别,算法
[**中图分类号**]TP391　[**文献标志码**]A　[**文章编号**]1001-4616(2014)04-0150-08

Research on Method for Independent Languages Acquisition and Clustering Corpus

Zhou Feng[1], Zhu Junwu[1], Tong Lin[1,2], Chen Weicong[3], Chen Bo[1]

(1. College of Information Engineering, Yangzhou University, Yangzhou 225127, China)
(2. Institute of Computing Technology Chinese Academy of Sciences, Laboratory of Intelligent Information Processing, Beijing 100190, China)
(3. Department of Computer Science and Applied Mathematics, University of Evansville, Indiana USA, 47722)

Abstract: Eliminate irrelevant and corpora clustering is of great significance to improve the quality of the natural language understanding, and it is also the key technology for the pretreatment of the natural language understanding. Independent Languageshave has obvious features in the corpus, and this article through seeds independent language derived strong independent language, and based on the strong independent language recognition and derived a new independent language; then, based on the similarity between sentences constructed by the 2-gram, using the method of hierarchical clustering of corpus of QA corpora are similar clustering problem. Finally, by identifying new independent language experiment and corpus clustering experiment, the validity of the method proposed in this paper is verified.

Key words: independent languages, acquisition, recognition, algorithm

随着问答系统技术的发展,问答系统产生了庞大的用户咨询语句,这些咨询语句是一种可靠的用于知识获取的知识源,对海量的咨询语句进行深入分析,并从中获取不同粒度的知识已成为一个亟待解决的问题.

海量的咨询语句库具有以下特点:

(1)面向领域性:对于不同的领域侧重点不一样. 比如银行问答系统跟电信问答系统.

(2)实时行:咨询语句库都是当下用户咨询的语句.

(3)口语化:更加接近用户的习惯.

(4)易获取:不需要繁琐的预处理.

为满足从海量咨询语句中获取不同粒度知识的需要,需要对海量咨询语句的特点进行分析. 海量咨询语句特点的分析主要包含两方面,分别是面向领域性和口语化. 面向领域性让我们知道在咨询中的一些面

收稿日期:2014-07-16.
基金项目:国家自然科学基金(61170201、61472344)、江苏省高校自然科学基金(14KJB520041).
通讯联系人:朱俊武,博士,教授,研究方向:人工智能、机制设计等. E-mail:jwzhu@yzu.edu.cn

向领域的词和口语化中一些对于语句没有作用的词,识别出其中的这些词非常重要. 然而,人工总结这些词或者短语是十分困难的,人们容易想到的方法是从语料库中学习这些词.

在领域咨询语句中,其组成大多包括一些词或者短语. 通过领域关键词及句子语法结构可以把句子中的词或者短语剥离出来. 有些词或者短语对于句子的主体意思的表达及含义没有影响,我们将以这些词类定义为无关语.

为了更好地获取形式文法,在预处理阶段,需要对无关语剔除,并对语料聚类. 本文首先识别咨询语句中的无关语,其次聚类咨询语句. 实验结果表明,论文提出的方法切实有效.

1　相关工作

在生产技术和科学快速发展的今天,对于各个领域的类别的分类越来越细,不能只用专业知识来区分,于是引入数学方法,随着多元分析,聚类分析又从数值分类学中分离出来,形成一个独立的分支[1]. 聚类分析渐渐的被许多应用所使用,如模式识别、数据分析、图像处理与市场研究.

聚类分析已经成为数据挖掘中不可分割的组成部分,聚类是通过设定相似函数把类似的数据组成一起的过程,我们可以发现通过聚类后,同一组内的数据对象具有较高的相似度,而不同组中的数据对象则是不相似[2]. 各种各样的多维数据集会有多种多样的结构,不可能一种聚类可以适用于所有的数据集. 根据数据在聚类积聚规则与应用这些规则的方法,我们通常把聚类分成 4 个类别:划分法、层次法、基于密度的方法、基于模型的方法[3].

语料相似度计算越来越在各领域广泛应用,如自然语言处理、智能检索、文本聚类、文本分类、自动应答和机器翻译等,逐渐为越来越多的科研人员所研究[4-7],本文所作的句子相似度计算研究的背景是 FAQ 自动问答系统.

李彬[8]等人提出一种基于语义依存的汉语句子相似度计算的方法,由于汉语句子的表达形式是多种多样的,标准地理解句子所表达的含义必须深入到语义一层并结合语法结构信息. 该方法取得了一定的实验效果.

陈利人[9]等提出了这样的理论,应当从结构相似度、语义相似度两个方面来衡量句子相似度. 即计算句子相似度通过两个步骤完成. 第一步,计算词的结构相似度,并由此计算句子的结构相似度;第二步,对句子的语义相似度进行分析计算. 李素建提出了一种定量计算模型来衡量语句相似度,该理论基于知网和同义词词林[10].

秦兵等提出依据 TF-IDF 对语句处理的方法以及语句语义方法,对常用问题集计算句子相似度[11]. 杨思春[12]等人提出通过局部的格语义并在此基础上获取语句模式的方法. 这种模式与语句的意义的模式有相似之处,呈现了句子的结构成分和基本语法的意义特性. 车万翔等人通过改进编辑距离的方式实现检索中文相似语句的问题[13]. 吕学强[14]提出一种语句相似度计算方法,主要是用词的形式的相似度与词的顺序来决定句子之间的相似性. 其中词的形式相似度起主要作用,词的顺序相似度起次要作用. 俞士汶[15]依据对句子相似度定义和计算要求,设计了一种基于骨架依存数的语句相似度计算模型. 崔恒等[16]在某一个特定领域上整体考虑关键词之间的距离与关键词的顺序等信息,来计算其相似度.

还有基于 N 元模型的方法[17,18]和基于编辑距离的方法[19]等,并且这些方法在计算句子相似度的研究获得了很多进展. 然而这类方法对于机器翻译系统的小语料时效果很好,进行 EBMT 翻译时的大语料时往往难以应付.

黄河燕[20]等人提出一种多层次句子相似度计算方法,先从基于句子的词表层特征和信息熵大规模语料库中获得具有特征的候选项实例,其次再将这些候选实例进行泛化匹配,在多策略机器翻译系统 IHSMTS 中得到了很好的效果. 该方法对于分词、词类标注有很高要求,然而很难满足大规模语料的分词标准及词类标注的标准,这是一个相当庞大的工作.

2　无关语及自动获取算法

2.1　无关语基本概念

为了对无关语有个直观的认识,首先考虑下面的句子:

您好！请问电脑黑屏怎么办,谢谢!

这是对于计算机相关的咨询语句,其中“您好！请问”和“谢谢”都是无关语,它们对于问题的主体意思没有联系,句子的主体意思“电脑黑屏怎么办”,然而识别出这些无关语对于本文后面的聚类及文法学习都是极其重要的.

依据无关语将句子标注出3个语块,即:

【您好！请问】电脑黑屏怎么办【,谢谢!】

其中无关语部分用全角的方括号标记,定义如下:

定义1(无关语语块)

无关语语块是句子中连续的词的序列.

定义2(无关语特性)

无关语是满足如下条件的词的序列:

(1)领域无关性. 它们是脱离于领域词的序列,但并不是领域无关性词的序列就是无关语. 它是一个充分非必要条件.

(2)主体意思无关性. 咨询语句中不涉及语句的主体意思.

例如:【您好！请问】与【！谢谢】是两个无关语语块,它们仅仅是在询问对方时礼貌用语,独立于计算机咨询领域,并且去除这两个无关语语块后咨询语句的主体意思并没有丢失,符合无关语的特性.

但是在海量的咨询语句中识别出这些无关语语块,人们首先想到的是手工整理出无关语语块,手工整理的好处是正确性高,但是局限性太大,不能面向海量咨询语句,工作量太繁琐,本章在结合手工及机器学习的方式识别出更多的无关语语块.

定义3(种子无关语)

由人工依据无关语特性,手工整理的无关语.

通过咨询语句的观察与分析,我们发现种子无关语中的有些在咨询语句中具有相对固定的特征位置,如表1,通过实验分析种子无关语中的“你好”、“请问”在句头的情况比例很大.

表1　种子无关语在咨询语句中位置及该位置的比例

Table 1　Position and percent of irrelevant word in Q&A sentence

种子无关语	特征位置	百分比
你好	句头	95%
请问	句头	82%

定义4(特征位置)

一条咨询语句被分为3个特征位置:句头、句中、句尾. 限定元素个数阀值来判断位置. 假如一条有咨询语句“[头]xsi….”,其中x很短|x|<k,即si句头特征位置. 同理句尾特征位置.

通过定义我们知道不一定是贴在两个边上位置才是特征位置,它们可以偏离两个边位置,但是偏离的位置很小. 如果满足在咨询语句中大多数情况下载句头特征位置或者句尾特征位置. 我们认为这些种子无关语具有很强的无关性,定义它们为强无关语.

定义5(强无关语)

依据定义4,如果出现在句头特征位置的次数与所有位置之和的比例大于阀值,我们认为它是一个强无关语. 同理句尾特征位置. 例如:咨询语句“[头]xsi”中的无关语si,在整个咨询语料中,处于句头的次数为head(si),总共出现si的次数count(si),如果head(si)/count(si)>threshold,即si是强无关语.

强无关语没有严格的局限,通过定义3我们知道强无关语是一些具有特征位置句头、句尾的无关语. 还有一类无关语我们也认为是强无关语,当该无关语的长度大于阀值时我们也认为是强无关语.

2.2　获取强无关语算法

通过强无关语的定义,我们依据强无关语的两个特性,即位置与长度,获取无关语中的强无关语.

种子无关语中无关语为si,有测试咨询语句集合S,si在S中出现的次数count(si),其中

出现在句头特征位置的次数head(si),满足

$$\begin{cases} \text{head}(si) > \text{threshold1} \\ \dfrac{\text{head}(si)}{\text{count}(si)} > \text{threshold2} \end{cases}$$

或者

|si|>hreshold3

其中,threshold1 表示 si 特征位置次数的阀值,threshold2 特征位置的比例阀值,threshold3 表示 si 长度的阀值.

根据以上的定义,提出了获取强无关语的方法,提出了 SeekStrongSeed 算法. 算法思想:

首先判定种子无关语的长度是否大于阀值,然后统计出种子无关语在测试语料中各个特征位置的次数,判定句头特征位置次数占各个特征位置的总次数是否大于阀值,同理句尾特征位置.

```
算法 1:SeekStrongSeed
输入:测试语料 arrCorpus,种子无关语 arrSeed
输出:强无关语 arrStrongSeed
1:SeedToPCount ⇓ arrSeed;
2:foriin 0:arrSeed. size( ) do
3:StringseedTemp = arrSeed. get( i) ;
4:for j in 0:arrCorpus. size( ) do
5:StringcorpusTemp = arrCosrpus. get( j) ;
6:ifcorpusTemp. contain( seedTemp)
7:ifSeedToPCount. contain( seedTemp)
8:              String PCount = SeedToPCount. get( seedTemp)
9:String[ ] array = PCount. split( "-") ;
10:ifseedTemp 在句头特征位置
11:intSCount = Integer. parseInt( array[ 0] ) +1;
12:PCount = String. valueOf( Scount) +"-" +array[ 1] +"-" +array[ 2] ;
13:SeedToPCount. put( seedTemp,PCount) ;
14:end if
15:同理句尾位置、句中位置
16:else ifseedTemp 在句头特征位置
17:intSCount = 1;
18:StringPCount = String. valueOf( Scount) +"-" +0+"-" +0;
19:SeedToPCount. put( seedTemp,PCount) ;
20:end if
21:同理句尾位置、句中位置
22:end if
23:end if
24:end for
25:end for
26:for String seedtemp:seedToPCount. keyset( )
27:StringPCount = seedToPCount. get( seedtemp) ;
28:String[ ] array = PCount. split( "-") ;
29:intScount = Integer. parsetInt( array[ 0] ) ;
30:intMcount = Integer. parsetInt( array[ 1] ) ;
31:intEcount = Integer. parsetInt( array[ 2] ) ;
32:doubleSrate = ( double) Scount/Scount+Mcount+Ecount;
33:doubleMrate = ( double) Mcount/Scount+Mcount+Ecount;
34:doubleErate = ( double) Ecount/Scount+Mcount+Ecount;
35:ifSrate>threshold&&Scount>sup|| Erate>threshold&&Ecount>sup
36:arrStrongSeed. add( seedTemp) ;
37:    end if
38:   end for
```

测试语料是随机抽取的咨询语句,然而实验没有用庞大的测试语料,存在稀疏的问题. 在以上方法统计次数时存在稀疏,不能确保无关语在语料中的出现次数,会丢失一些强无关语. 识别出的强无关语中加

入人工校验. 让强无关语更具正确性,这对于识别新无关语的工作意义很大. 人工校验准则是符合无关语特性. 强无关语也是无关语中的一种,只是它们在语料中具有强的特性,本文主要分析长度与位置特性. 如图 1 为获取强无关语的流程图. 输入为测试语料与种子无关语,然后基于长度与位置的统计方法找到两种强无关语,即两种输出文件,人工对这两个文件进行校验并最终合并两个文件组成一个强无关语文件.

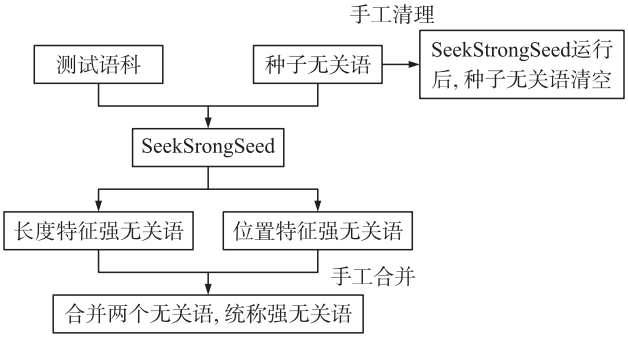

图 1　获取强无关语流程图

Fig. 1　Flow chart to extract strong irrelevant words

2.3　基于种子无关语的无关语识别

上面已经能通过测试语料识别出种子无关语中的强无关语. 强无关语在语料中具有很强的特性,如何用强特性识别出测试语料中新的无关语? 本节将是主要研究这个问题.

直观地,我们知道句子[您好! 请问]、[谢谢]是无关语语块,但是人工整理的无关语存在局限性,假设[您好]无关语语块没有被整理进种子无关语. 表 3. 1 中[请问]是强无关语并且具有句头位置特征,即|您好|<k(阀值 k). 我们认为[您好]与[您好! 请问]为无关语.

命题 1　咨询语句 p:[头]xsi…. ,其中 si,满足 si 在句头,|x|<threshold2,则:x 与 x+si 为无关语.

依据命题 1,提出基于种子无关语的新无关语识别 SeekNew,算法思想为通过种子无关语中的强无关语位置特性,无关语是具有连续的语块序列,具有强特性的种子无关语周围肯定还存在新的无关语,特别是在特征位置的强无关语,用强无关语带出新的无关语. 该类新无关语具有句头或句尾的特征位置并且周围有强无关语. 输入为测试语料与种子无关语中的强无关语,输出为识别的新无关语.

```
算法 2:SeekNew
输入:测试语料 arrCorpus,强无关语 arrStrongSeed
输出:新无关语 arrNew
1:for i in 0:arrCorpus. size( ) do
2:String corpusTemp=arrCosrpus. get(i);
3:for j in 0:arrStrongSeed. size( ) do
4:String strongseedTemp=arrStrongSeed. get(j);
5:ifcorpusTemp. contain(strongseedTemp)
6:String k=strongseedTemp 在 corpusTemp 距离句头的字符串
7:if |k|<threshold
8:          arrNew. add(k);
9:          arrNew. add(k+strongseedTemp);
10:end if
11:end if
12:end for
13:end for
```

强无关语是经过手工校验,具有正确性高. 完全用统计的方法识别出的新无关语也是需要经过手工校验. 如图 2 本文给出了识别新无关语的流程图,其中加入了人工校验的过程. 人工校验的好处让新无关语可以作为种子无关语使用,加入种子无关语后又可以用这些新种子无关语学习出新的强无关语,我们发现这是一个迭

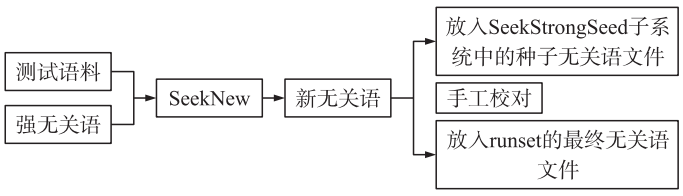

图 2　获取新无关语流程图

Fig. 2　Flow chart to extract new irrelevant words

代的过程. 所以经常校验后的新无关语会被放入种子无关语文档中,等待下一次学习用.

2.4　系统实现与实验结果分析

从种子无关语到强无关语获取再利用强无关语识别新无关语,这是一遍识别新无关语流程,我们会用新无关语进行迭代的识别出新无关语,直到不能迭代出新无关语. 在无关语学习结束,分别用种子无关语及新无关语标注测试语料. 并对整体无关语击中情况进行分析.

我们随机抽取 100 条测试语料,手工标注出无关语,共标注出 213 个无关语. 表 2 中可以看出学习后语料无关语击中情况有所增加.

上面实验告诉我们无关语击中次数增加,然而不知道是完全被击中了还是部分击中. 我们又随机抽取 100 条测试语料,手工标注出无关语,共 101 个无关语. 表 3 中学习后部分击中从 22 降低到 14,降低的部分击中变成完全击中. 完全击中增加 34.

表 2　100 条语料无关语击中情况

Table 2　100 extracted and irrelevant words

	击中个数	击中比率
原始无关语	107	50.2%
更新后无关语	163	76.5%

表 3　100 条语料部分完全击中情况

Table 3　100 perfectly extracted and irrelevant words

	部分击中个数	完全击中个数
原始无关语	22	40
更新后无关语	14	74

3　语料聚类

3.1　咨询语料聚类概述

咨询语料呈现各种的客户问题,直接分析这些语料会很浪费时间,上一节已经为我们识别出无关语,让咨询语料干扰性降低,这对接下来聚类相似的语料对于语料的分析会有很大的帮助. 文本的聚类有划分法、层次法、基于密度的方法、基于模型的方法. 本节主要基于 2-gram 构建句子之间的相似性,利用层次法对语料进行聚类对 QA 语料进行问题相似的聚类.

3.2　基于 2-gram 的句子相似函数

句子相似性度量是一个重要的问题,很多和自然语言相关的应用都依赖于句子相似性的度量,句子相似度研究源于信息检索(Information Retrieval)领域. 在语言研究领域,句子相似性研究也起着重要的作用,例如,在双语平行语料库中,运用语句的相似度可以提取具有不同相似等级的句子. 在以往的句子相似性研究中,需要体现句子的结构和语义信息,这些都需要花费更大的工作时间,然而本文下面的文法学习不需要考虑具体的结构与语义,通过一些策略体现出结构和语义,也体现了统计的特性.

为了有效度量句子的相似性,从以下两个角度挖掘能体现句子的特征:

(1)句子的形式相似度.

(2)句子的语义相似度.

基于以上的分析,本节将按如下的步骤定义句子的相似性度量:

(1)采用 2-gram 模型化句子,通过相邻两个词的组合作为形式相似度策略.

(2)采用 2-gram 模型化句子,通过句子的之间的比值作为语义相似度策略.

统计式的语言模型是借由一个机率分布,而指派机率给字词所组成的字串:$P(w_1,\cdots,w_m)$

在语音辨识和在资料压缩的领域中,这种模式试图捕捉语言的特性,并预测在语言串行中的下一个字.

当用于资讯检索,语言模型是与文件有关的集合. 以查询字"Q"作为输入,依据机率将文件作排序,而该机率 $P(Q|M_d)$ 代表该文件的语言模型所产生的语句之机率.

N 元语法的 n 体现了该词间的独立性,n 越小独立性越强. 则可根据不同语料的独立性特点选择不同的模型了. 通常 $n=3$. 直观上讲,第 i 位置的词与前面多少个词的相关性并不一定,另外,"词"是一个笼统的概念(可以代表字、词短语等),它的选取也不确定,而一个模型直接赋予 n 一个确定的值,这本身是一种近似. 所以说,模型不可能精确表达,根据这种局限性,一个好的模型的重要性就可想而知了. 模型提出后也要检验,至少要满足已知定理,比如 n 元语法模型就加上了<BOS>和<EOS>,以使 $i-1$ 有意义并满足概率的归一性.

3.3　基于句子相似度矩阵的层次聚类

2-gram 模型基本这样一种假设,在词串 $w=w_1,w_2,\cdots,w_n$ 中第 i 个词的出现只与前面 $i-1$ 个词相关($i=1,2\cdots n$),而与其它任何词都不相关. 词串开头加<BOS>结尾加<EOS>,基于 2-gram 模型是词串用一个集合表示 $S=\{$<BOS>w_1、w_1w_2、$\cdots$、w_n-iw_n、w_n<EOS>$\}$,其中每个原子,如<BOS>w_1,我们称为 2-gram.

假设有两个词串集合 S_1,S_2,其中 S_1 的 2-gram 个数为 C_1,S_2 的 2-gram 个数为 C_2,共同的 2-gram 个数为 C_{com}. 则两个词串的相似度为

$$P_{sim<S1,S2>}=\frac{C_{com}}{C_1+C_2-C_{com}}.$$

依据词串之间的相似度计算方法,我们对于咨询语句进行语句之间的矩阵建模:

$$\boldsymbol{A}=\begin{pmatrix} 1 & P_{sim<S1,S2>} & P_{sim<S1,S3>} & \cdots & P_{sim<S1,Sn-1)} & P_{sim<S1,Sn)} \\ P_{sim<S2,S1>} & 1 & P_{sim<S2,S3>} & \cdots & ? & ? \\ P_{sim<S3,S1>} & P_{sim<S3,S2>} & 1 & P_{sim<Sx,Sy>} & ? & ? \\ ? & ? & P_{sim<Sy,Sx>} & 1 & P_{sim<Sn-2,Sn-1>} & ? \\ P_{sim<Sn-1,S1>} & \cdots & \cdots & P_{sim<Sn-1,Sn-2)} & 1 & P_{sim<Sn-1,Sn>} \\ P_{sim<Sn,S1>} & \cdots & \cdots & \cdots & P_{sim<Sn,Sn-1>} & 1 \end{pmatrix}.$$

以上咨询语句模型是可以看成一个上三角矩阵,每个位置的值为语句之间的相似度,值越大我们认为相似性越高.

假设 N 条咨询语料待聚类,对于层次聚类来说,基本步骤如下:

(1)(初始化)把每条咨询语句归为一类,计算它们之间也就是咨询语句之间的相似度;

(2)寻找各个咨询语句之间,把他们归为一类,咨询语句的总数就少了 1 个;

(3)重新计算新生成的咨询语句整体与各个旧的咨询语句之间的相似度;

(4)重复(2)和(3)直到结束.

如图 3 这是一个层次聚类的例图,更加形象地对层次聚类进行了说明,它是一种自顶向下的方式聚类方式. 首先由一个一个独立的语句,然后进行一层一层的聚类,最终生成一个大的集合,是否能执行到最后一层取决于你的阀值. 显然地,阀值的选取对于聚类的好坏有很大的影响. 层次聚类最大的优点,就是它一次性地得到了整个聚类的过程,只要得到了下面那样的聚类树,想要分多少个集合都可以直接根据树结构来得到结果,改变集合数目不需要再次计算数据点的归属.

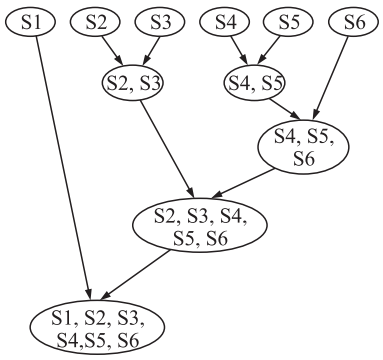

图 3　语句聚类例图

Fig. 3　Case graph of clustering sentence

3.4　实现与实验分析

本节实现主要是对语句之间的相似度的计算及建模. 过程如下:

算法 3:*similarmodle*

输入:咨询语料 S

输出:$similar_{<Sm,Sn>}$

1:*fori in* 0:$|S|$ *do*

2:用 2-*gram* 表示两个语句 pi,pi+1 即 Si,Si+1.

3:计算 $P_{sim<Si,Si+1>}$,并放入 $similar_{<i,i+1>}$

4:*end for*

本文采用的 java 在 myeclipse 上编程实现系统的主要算法,实验环境 AMD E1-1200 APU with Radeon(tm)HD Graphics 1.40 GHZ /4.00 GB/Win7. 为了测试该现实. 咨询语料有 1 万条,首先进行语料的相似度建模,然后用已有的层次聚类源码进行测试,进行相似度最小值阀值设定,计算聚类的结果我们从每个集合中随机抽

表 4　聚类结果分析

Table 4　Analysis of clustering results

集合	准确率/%
S_1(电脑黑屏)	100
S_2(电脑死机)	80

取 50 条语料进行实验分析.

表 4 中集合 S_1(电脑黑屏)表示随机抽取 S_1 中主体意思是电脑黑屏的集合. 其中 S_2 中“我的电脑无缘无故关机了”,标注无关语语块“【我的】电脑【无缘无故】关机【了】”其中主体意思为“电脑关机”与“电脑死机”意思不同,但是两者之间的 $P_{sim<S1,S2>}$ 为 3/7 大于设定的域值. 由于我们会基于频繁项学习,而作为关机只是很少的一部分存在集合中,这对我们后面的学习不会产生严重的影响,我们可以对集合中语料依据 $P_{sim<S1,S2>}$ 进行排序. 在这边我们会建立一个领域的词典用来区分关键词,用来区分语料之间的区别,如关机与死机. 在聚类之间先通过领域关键词区分语料然后再进行聚类,这样就可以避免此类问题发生. 通过实验我们会建立领域关键词词典,对于语料句型相似的用关键词比较来区分语料的相似,然后再进行相似度的计算,最后聚类.

4　结论

本文讨论了无关语的识别问题. 鉴于手工整理无关语是最容易获取无关语的方法,但是会存在局限性. 在整理的无关语中,易于发现其中的一些无关语在语料中存在强的特性,大部分情况下无关语出现在语料的句头或者句尾. 本文借助种子无关语确定强无关语,并依据强无关语识别出新的无关语,经过人工校验,然后迭代产生新无关语. 最后对识别的新无关语进行了实验结果分析.

本文研究了咨询语句相似度计算的方法. 虽然相似度计算已经很多,但是要让聚类的结果符合本文的文法学习,由此找到了 2-gram 的相似度计算,并在此基础上用层次聚类的方法进行相似咨询语句的聚类. 实验表明,本文提出的文本聚类有效性.

[参考文献]

[1] Sambasivam,Theodosopoulos. Advanced data clustering methods of mining web documents[J]. Issues in Informing Science and Information Technology,2006,8(3):563-579.

[2] Han J,Kamber M. 数据挖掘概念与技术[M]. 第二版. 范明,孟小峰,译. 北京:机械工业出版社,2006.

[3] 孙吉贵,刘杰,赵连宇. 聚类算法研究[J]. 软件学报,2008,19(1):48-61.

[4] 段良涛,郭曙超. 中文文本校对技术研究[J]. 电脑知识与技术,2014,10(19):4 601-4 604.

[5] 陈智鹏. 基于统计的搜索引擎中文输入纠错技术研究[D]. 北京:北京邮电大学电子工程学院,2010.

[6] 来社安,蔡中民. 基于相似度的问答社区问答质量评价方法[J]. 计算机应用与软件,2013,30(2):266-269.

[7] 李晨,巢文涵,陈小明,等. 中文社区问答中问题答案质量评价和预测[J]. 计算机科学,2011,38(6):230-236.

[8] 李彬,刘挺,秦兵,等. 基于语义依存的汉语句子相似度计算[J]. 计算机应用研究,2003,20(12):15-21.

[9] 陈力为,袁琦. 计算语言学进展与应用[M]. 北京:清华大学出版社,1995.

[10] 刘群,李素建. 基于《知网》的词汇语义相似度计算. Http://www.keenage.com.

[11] 王盛,樊兴华,陈现麟. 利用上下位关系的中文短文本分类[J]. 计算机应用,2010,30(3):603-606.

[12] 刘汉兴,林旭东,田绪红. 基于本体的自动答疑系统的研究与实现[J]. 计算机应用,2010,30(2):415-418.

[13] 冯成,陈智敏. 领域本体建模方法的研究[J]. 科学技术与工程,2009,9(2):455-459.

[14] 骆正华,樊孝忠,刘林. 本体论在自动问答系统中的应用[J]. 计算机工程与应用,2005,41(32):229-232.

[15] 俞士汶. 基于骨架依存树的语句相似度计算模型[C]//中文信息处理国际会议(ICCIP'98),北京,1998.

[16] 崔恒,蔡东风,苗雪雷. 基于网络的中文问答系统及信息抽取算法研究[J]. 中文信息学报,2004,18(3):24-31.

[17] Keiji Yasuda,Fumiali Suagya,etc. An automatic evaluation method of translation quality using translation answer candidates queried from a parallel corpus[C]//Proceeding of MT Summit's conference,Santiago de Compostela,2001.

[18] Yao Jianmin,Zhou Ming. An automatic evaluation method for localization oriented lexicalised EBMT system[C]//Proceeding of the 19th International Conference on Computational Linguistics,Taipei,2002.

[19] Yasuhiro Akiba,Kenji Imamura,EiichiroSumita. Using multiple edit distances to automatically rank machine translation output [C]//Proceeding of MT Summit's conference,Santiago de Compostela,2001.

[20] 黄河燕,陈肇雄,张孝飞,等. 大规模句子相似度计算方法[J]. 中文信息学报,2006,(z1):47-52.

[责任编辑:陆炳新]