辽宁大学学报
自然科学版
第38卷 第4期 2011年

JOURNAL OF LIAONING UNIVERSITY
Natural Sciences Edition
Vol. 38 No. 4 2011

《知网》义原相似度计算的研究

袁晓峰
(盐城师范学院 信息科学与技术学院，江苏 盐城 224002)

摘 要:词语之间相似度的计算广泛应用于信息检索、文本主题抽取、文本分类、机器翻译等研究领域. 词语之间的相似度的计算通常有两方法,基于统计的方法和基于世界知识的方法. 对于中文的词语相似度计算,有人提出一种利用《知网》计算词语相似度的方法,该方法通过计算《知网》义原的相似度进而计算词语的相似度,但是该方法在计算义原相似度时没有考虑义原在层次体系树上的深度以及区域密度. 在此基础之上深入研究《知网》的义原层次体系,将义原在层次体系树上的深度和区域密度两个因素添加到义原相似度计算中. 最后,实现了该计算方法并得到实验结果,将实验结果与改进前的计算方法的结果比较,发现考虑义原在层次体系树上的深度和区域密度得到的结果比不考虑这两个因素得到结果更符合实际.

关键词:知网; 义原; 相似度; 自然语言处理

中图分类号:TP391.1 文献标识码:A 文章编号:1000-5846(2011)04-0358-04

0 引言

词语相似度是指两个词在不同的上下文中可以互相替换而不改变文本的句法语义结构的程度[1],词语相似度广泛应用于信息检索、信息抽取、文本分类、词义排歧、基于实例的机器翻译等领域[2-5]. 词语相似度计算通常有两种方法,一种是基于统计的方法,一种是基于某种世界知识的方法. 统计方法是在大规模语料中统计分析每个词的特征词向量,然后利用这些向量之间的相似度(用向量的夹角余弦计算)作为这两个词的相似度[6]. 基于世界知识的方法依赖于某种世界知识库,目前英文世界知识库研究的较多的是Wordnet,中文世界知识库研究得较多的是《知网》[7].

刘群、李素建在文献1中利用《知网》提出一种计算词义相似度的方法,该方法根据《知网》义原上下位关系生成了一个义原层次体系树,通过计算义原在该体系树上的距离从而计算其相似度,为计算中文词义相似度提出了一种可行的方法. 但该方法在计算义原的距离时,完全根据两个义原在层次体系树上的路径长度来计算,而没有考虑其他因素. 事实上,义原在层次体系树上的位置不同它们之间的距离也会不同. 位置因素主要包括义原在层次体系树上的深度和区域密度. 例如 "动物"和"植物"、"水果"和"蔬菜",这两对概念之间的路径长度都是2,但前一对义原处于层次树的较高层,后一对处于层次树的较低层,两对义原的相似度不应该相同,前者应该较后者小. 再如:路径长度相同的两对结点,如果一对位于概念层次树中低密度区域,另一对位于高密度区域,那么前者的语义距离应大于后者. 本文在文献1的基础之上着重研究义原在层次体系树上的深度以及区域密度对利用《知网》计算义原相似度的应用和影响.

1 《知网》简介

《知网》是一个以汉语和英语的词语所代表

* 作者简介:袁晓峰(1978-),男,助教,硕士,自然语言处理、信息检索.
收稿日期:2011-10-21

的概念为描述对象，以揭示概念与概念之间以及概念所具有的属性之间的关系为基本内容的常识知识库.

1.1　义原的概念

《知网》中有两个主要的概念 "概念"和"义原". 其中，"概念"是对词汇语义的一种描述，每一个词可以表达为几个概念. "概念"用一种特定的"知识表示语言"来描述，这种"知识表示语言"所用的"词汇"叫做"义原". "义原"是用于描述一个"概念"的最小意义单位[8].

《知网》一共采用了 1500 个义原，分为: Event|事件、entity|实体、attribute|属性、aValue|属性值、quantity|数量、qValue|数量值、SecondaryFeature|次要特征、syntax|语法、EventRole|动态角色、EventFeatures|动态属性十大类.

1.2　义原层次体系

《知网》中义原之间存在着复杂的关系，如: 上下位关系、同义关系、反义关系、对义关系等，不过，最重要的还是义原的上下位关系. 根据义原上下位关系，义原可以组织成一个义原层次体系，如事件类和实体类义原的层次体系如图 1 和图 2. 这个层次体系是一个树状结构，这种结构是进行词语相似度计算的基础.

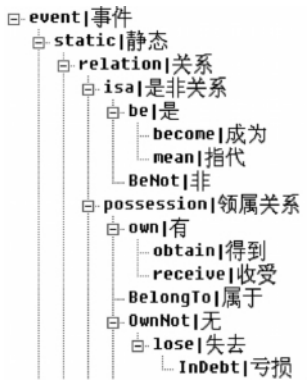

图 1　事件类义原层次体系树

2　义原相似度计算

由于义原可以根据上下位关系进而组织成树状结构的层次体系，因此可以通过计算义原之间的距离从而计算义原之间的相似度，文献 1 中给出其计算公式为:

$$sim(p_1,p_2) = \frac{\alpha}{\alpha + d} \tag{1}$$

图 2　实体类义原层次体系树

其中 p_1、p_2 分别表示两个义原，d 是 p_1 和 p_2 的距离. α 是一个可调节的参数，通常 α 是指相似度为 0.5 时的词语距离值.

2.1　义原距离

定义 1　义原 p_1、p_2 的距离 d 定义为: 在同一类义原层次体系树上 d 为 p_1 到 p_2 的路径长度，否则 d 统一设为 20(根据参考文献 1).

《知网》为广大研究者提供了许多有用的资源，包括义原层次结构文件—WHOLE. DAT. WHOLE. DAT 文件用{编号 义原 双亲结点编号}格式组织，如: { 1 static|静态 0 }，其中"static|静态"为义原，"1"是"静态"的编号，"0"为"静态"的双亲结点的编号(见图 1). 我们设计求义原距离算法如下:

(1) 在 WHOLE. DAT 中搜索 p_1 及其双亲结点得到 $R = \{R_1, R_2, \cdots, R_i, R_j, \cdots R_m\}$，$R_j$ 为 R_i 的双亲结点.

(2) p_2 的处理同(1).

(3) 找到 p_1 和 p_2 的最近共同双亲结点 cp，如果 cp 存在，则 $d = dis(p_1, cp) + dis(p_2, cp)$; 否则，$d = 20$.

2.2　义原的深度及区域密度

定义 2　义原 p 的深度是指所在义原层次体系树的根结点到该义原的路径长度，用 deep(p) 表示.

定义 3　义原 p 的区域密度是指 p 所在层上的结点与一个可调节参数 β 的比例，公式为:

$$density(p) = \frac{nc(p)}{\beta} \tag{2}$$

其中 nc(p) 为义原 p 兄弟结点的个数; β 是一个调节系数，β 值的设置要尽量使得 density

(p) 的值覆盖区间(0 ~ 0.5) .

2.3 相似度计算

在计算两个义原相似度时,我们综合考虑两个义原的路径长度以及它们分别在层次体系树上的深度以及区域密度. 我们假设义原的深度和区域密度对义原相似度的贡献是独立的,深度对相似度的影响比密度对相似度的影响要小.

$$con(p) = \gamma deep(p) + \eta desity(p) \quad (3)$$

其中 $\gamma < \eta$ 且 $\gamma + \eta = 1$.

根据义原深度越大相似度越大和区域密度越大相似度越大的原则,我们可以得出(1) 式中的 d 为:

$$d = \delta \cdot \frac{dis(p_1, cp) + dis(p_2, cp)}{con(p_1) + con(p_2)} \quad (4)$$

其中,如果 cp 不存在,即 p_1 和 p_2 不在同一个类层次树上,则 $dis(p_1, cp) + dis(p_2, cp) = 20$. δ 为一个调节参数.

3 实验

我们将上述方法通过 VC 6.0 实现出来,经过反复测试,我们将上文中的几个可调节参数设置如下:

$\alpha = 1.6$, $\beta = 50$, $\gamma = 0.3$, $\eta = 0.7$, $\delta = 3$

我们将改进后的方法得到实验结果同改进前方法得到的结果比较,其结果如表 1 所示:

表 1 部分义原相似度比较

义原 1	义原 2	路径长度	文献 1 方法结果	本文方法的结果
动物	植物	2	0.444444	0.451273
水果	蔬菜	2	0.444444	0.501197
切削	破开	2	0.444444	0.637261
相等	相像	2	0.444444	0.493654
关系	状态	2	0.444444	0.331074
万物	空间	2	0.444444	0.263261
方	圆	2	0.444444	0.452555
静态	行动	2	0.444444	0.255066
生物	虫	3	0.377559	0.477559
物质	虫	4	0.285714	0.295113
切削	水果	20	0.074074	0.121327

从表 1 可以看出,文献 1 中的计算方法比较粗糙,只要路径长度相等则义原相似度相等. 如,“动物”和“植物”、“水果”和“蔬菜”的路径长度都为“2”,相似度则都为“0.444444”,这显然不太合理. 根据词语相似度的概念—两个词在不同的上下文中可以互相替换而不改变文本的句法语义结构的程度—可以看出,“水果”和“蔬菜”的相似度应该更大. 而改进后的方法得到的结果中前者为“0.451273”,与“0.444444”非常接近; 后者为“0.501197”,显然比改进前的方法得到的结果更合理. 这是因为改进后的方法考虑了义原的深度和区域密度,“水果”和“蔬菜”的深度和区域密度较“动物”和“植物”的深度和区域密度大. “切削”和“破开”的深度和区域密度更大,所以它们的相似度更大,为“0.637261”,从直观上看,这是合理的.

表 2 中再给出一些本文计算方法得到的结果,供读者参考.

表 2 一些义原的相似度

义原 1	义原 2	相似度	义原 1	义原 2	相似度
机器	电脑	0.54435	吃	喝	0.640598
资金	货币	0.538064	暴动	反抗	0.639631
陆地	水域	0.534855	触	抚	0.637261
食品	饮品	0.531601	出现	发生	0.497454
问题	原因	0.455733	朝向	通往	0.453512
部件	配件	0.331074	明暗	清浑	0.386252
方向	位置	0.331074	主题	焦点	0.324324
新闻	音乐	0.329399	颜色	气味	0.324324
精神	事情	0.280966	多	少	0.324324
时间	空间	0.264361	体育	教育	0.242424

从表 2 来看,绝大部分结果还是比较合理的. 由于《知网》义原体系的组织方法对计算义原相似度影响很大,因此,随着《知网》的不断完善,本文设计的计算方法的结果也会相应得到改善.

4 结论

本文中,我们在文献 1 中的《知网》义原相似度计算方法的基础之上,仔细研究《知网》中义原的层次体系结构,通过考察义原在层次体系树上的深度和区域密度对义原相似度的影响,提出一种对计算义原相似度的改进方法. 我们认为,义原在层次体系结构树上的深度越大、义原所在区域密度越大,表明该义原和兄弟义原被划分得越详细,所以也应该越相似. 实验表明,改进后的方法得到的结果绝大部分都是比较合理的,也比改进前的方法得到的结果更符合实际.

参 考 文 献 :

[1] 刘群,李素建. 基于《知网》的词汇语义相似度计算

[1] . 计算语言学及中文信息处理,2007,7:59 – 76.

[2] Gauch S. and Chong M. K. . Automatic Word Similarity Detection for TREC 4 Query Expansion [C] . Proc. Of TREC – 4: The 4th Annual Text Retrieval Conf., Nov. 1995, Gaithersburg, MD, 1995:527 – 536.

[3] LI Xiaobin, Szpakowicz S., and Matwin S. . A WordNet – based algorithm for word sense diambiguation [C] . Proc. of the Twelth International Joint Conference on Artificial Intelligence(IJCAI) . 1995:1368 – 1374.

[4] 王斌. 汉语双语语料库自动对齐研究 [D] . 中国科学院计算技术研究所博士学位论文,1999.

[5] 李涓子. 汉语词义排歧方法研究 [D] . 清华大学博士论文,1999.

[6] Dagan I, Marcus S. Contextual word similarity and estimation from sparse data [A] . Collins M. Processing of the Annual Meeting of the Association for Computational Linguistics [C] . New Mexico: American Association for Artificial Intelligence, 1993: 164 – 171.

[7] 董振东,董强. “知网”, http://www.keenage.com [OL] .

[8] 林丽,薛方,任仲晟. 一种改进的基于《知网》的词语相似度计算方法 [J] . 计算机应用,2009,(1):217 – 218.

Research of Word Relativity Based on HowNet

YUAN Xiao – feng

(*School of Information Science and Technology, Yancheng Teachers College, Yancheng*,224002,*China*)

Abstract: The similarity computation between words is widely used in many research area, such as information retrieval, extracting subject of documents, text clutering, machine translation and so on. There used to be two ways to compute the similarity between words, one is based on statistics, another is base on the ontology. There is a method based HowNet to calculate the similarity between Chinese words already. This method calculate the similarity between words thought calculate the similarity between primitives of HowNet. But this method have ignored the depth and density of primitives. We add the factor of primitive depth and density to the method above though researching of HowNet carefully. We realize our method and got the experimental data, and we find our method is more practical than the method already existent.

Key words: HowNet; Primitive; Similarity; Natural Language Processing

(责任编辑　郑绥乾)