基于语义计算的语句相关度研究

李素建
(中国科学院计算技术研究所,北京 100080)
E-mail:lisujian@ict.ac.cn

摘 要 该文在中文问题回答系统中引入了语义计算。基于《词林》和 hownet 两种语言资源,提出计算词与词之间的相似度和相关度,然后得到语句间的相关度,系统通过对语句相关度的比较从而得到查询问题的最优答案。该方法采用了定量计算,易于结合到 QA 系统中,同时避免了很多传统的自然语言处理问题。试验结果表明该方法是有效的。

关键词 自然语言处理 问题回答 语句的相关度

文章编号 1002-8331-(2002)07-0075-02 **文献标识码** A **中图分类号** TP391.1

Research of Relevancy between Sentences Based on Semantic Computation

Li Sujian
(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)

Abstract: This paper introduces semantic computation into our Chinese Question-Answering system. Based on two kinds of language resources Cilin and hownet, this paper presents an approach to compute the similarity and relevancy between words. Using these results, the relevancy between two sentences can be calculate and the optimal answer for the query can be get in the system. The method adopts quantitative methods and can be incorporated into QA systems easily, avoiding some difficulties in conventional NLP problems. Finally some examples are presented to show that the results are satisfying.

Keywords: Natural Language Processing, Question Answering, Relevancy

1 引言

Internet 的快速发展,网上资源急速增长。为了满足用户从网上得到对于特定问题的准确答案,QA(Question-Answering)系统成为目前一个重要的研究课题[1,2]。由于 TREC(Text Retrieval Conference)和 MUCs(Message Understanding Conferences)的支持,英语方面的 QA 系统已经有很大进展,在 TREC-8 的评测报告中可以看到一些系统表现出了很好的性能[3]。

由于汉语自身的语言特点,在实现汉语 QA 系统中有自身的困难和特点。首先,英语 QA 系统的实现,目前大部分是进行框架结构填充,英语的特点决定一般的事件是围绕动作扩展信息的;汉语的独特性,它以词作为实体表现各种关系,词与词之间不象英语那样具有较强的语法依附性,而是通过语义建立起相应的关联。因此这里可以直接通过计算词的语义距离来比较相关性,而不是象英语中那样得到框架后再进行比较。其次,英语可以通过词形和时态等的变换来帮助表达意义,而汉语很少具有这些形态的变化,只能以字义和词义为中心表达含义。通过借鉴机器翻译领域内一些对语句相似度的研究[4],不对语句进行语法结构分析,主要利用句子的表层信息,即组成句子的词的语法、语义信息,从而可以避免考虑句子的整体结构。该文针对 QA 系统提出一种新的衡量机制-语句的相关度,并利用知网和同义词词林作为语言资源进行语义计算。

2 语义计算的基本思想

Hownet[5]是在 Internet 上发布的一个汉英双语资源,它着力描述了概念与概念之间以及概念所有的特性之间的关系,这些关系都隐含在知网的知识词典和义原的特征文件中[5,6]。义原在知网中是个重要的概念,它是从所有汉词中提炼出的可以用来描述其它词汇的不可再分的基本元素。在知网的汉语知识库中包括了 66,681 个词义。每个词义定义如下:

W_X=词语

DEF=概念定义

其中的 DEF 部分是表示与义原的关系。因此可以简单地认为词是由义原通过某种关系构成的。对一些概念形式化定义如下:

$SS=\{s_1,s_2,K,s_n\}, n=1541$

$WS=\{c_1,c_2,\cdots,c_m\}, m=66,681$

$REL=\{*,@,?,!,\sim,\#,\$,\%,\hat{},\&,NULL\}$

$c_i \Rightarrow r_{i1}s_{i1},r_{i2}s_{i2},\cdots,r_{ik}s_{ik},r_{ij}\in REL, s_{ij}\in SS(1<t<k)$

SS 表示义原集合,含有 1541 个义原。知网中的所有词义构成一个词义集合 WS,有 66,681 个元素,每个义原也应该看作是一个特殊的词义、它的定义为自身。集合 REL 是概念之间或义原之间的可能关系集合。对于某个词义 c_i,它的解释若由 k 项组成,每一项为一个关系符号加上一个义原。

通过知网的组织关系,引入了语义计算,它分为三步进行,首先建立起义原间相似度和关联度的计算机制;然后根据义原之间的计算结果得到词语间的相似度和相关度;第三步由词间的相似度和相关度得到语句的相关度。

作者简介:李素建,博士生,研究方向为自然语言理解、机器翻译、知识挖掘。

```
entity实体
  ├ thing|万物 [#time|时间,#space|空间]
  │  ├ physical|物质 [!appearance|外观]
  │  │  ┄ ├ animate|生物 [*alive|活着,!age|年龄 *die|死 *metabolize|代谢]
  │  │  │  ├ AnimalHuman|动物 [!sex|性别,*AlterLocation|变空间位置 *StateMental|精神状态]
  │  │  │  │  ├ human|人 [!name|姓名,!wisdom|智慧,!ability|能力,!occupation|职位 *act|行动]
  │  │  │  │  │  └ humanized|机人 [fake|伪]
  │  │  │  │  └ animal|兽 [*GetKnowledge|认知]
  │  │  │  │     ├ beast|走兽 [*GetKnowledge|认知]
  │  │  │  │     ┄
```

图 1 特征义原的网状结构示例

3 语义计算过程

在知网中义原的关系是通过几个特征文件建立起来的，每个特征文件如图 1 所示，整体为一个树状结构，每一个节点代表了一个义原称为主义原，后面所跟方括号里的内容是该义原的一些解释义原，每个解释义原前有一个符号表示解释义原和主义原之间的关系。解释义原和主义原都具有上下位节点，这样不同特征集中的义原就产生了不同的关联度。一棵树中上下位义原之间的关系，称作是纵向联系，纵向关系的义原与义原之间存在着相似度；义原之间的其它可能关系则是跨越体系的义原之间的联系，这里称作是横向联系，用关联度来进行定量计算。

对于两个义原，进行相似度的计算公式如下：

$$sim(s_1,s_2)=\begin{cases}\alpha_1/dist(s_1,s_2) & t(s_1)=t(s_2)\\ 0 & t(s_1)\neq t(s_2)\end{cases} \tag{1}$$

s_1,s_2 为义原集合中的任意两个元素，$t(s_1)=t(s_2)$ 说明义原 $sem1$ 和 $sem2$ 处于一个分类体系中，只有当两个义原处于一个分类体系中，才具有相似度，相似度和义原之间的距离成反比。

在义原构成体系中，每个义原和不在同一个树中的义原也可能有一定的关系，这样就为原来义原体系的树状层次结构增加了横向联系，从而使整个义原体系呈现了一种网状结构。根据继承性，下位义原继承上位义原的解释义原，而解释义原本身也存在着一定的层次结构，因此这样就存在着义原的横向关联扩展和纵向关联扩展。横向关联扩展就是扩展到解释义原的上位义原；纵向关联扩展就是扩展到上位义原的解释义原。计算两个义原的关联度公式为：

$$\begin{cases}ext(s_j)=\{s_i|REL(s_j,s_i)\}\\ Asso(s_1,s_2)=\sum_{s_i\in ext(s_1)} w_i sim(s_i,s_2)+\sum_{s_j\in ext(s_2)} w_j sim(s_1,s_j)\end{cases} \tag{2}$$

公式中 $ext(s_i)$ 表示义原 s_i 的扩展集合，在计算任意两个义原 s_1,s_2 的关联度时，首先计算一个义原和另一义原扩展集合元素的相似度，然后根据义原间的相应关系加权求和，得到两个义原最终的关联度。公式中 w_i 和 w_j 表示相应关系的权值，对于关系集合 REL 中的不同关系设置不同的权值，表示不同关系对关联度的影响。

由于知网中义原的组织结构是一种网状结构，建立义原之间的相似度和关联度比较容易，但它的词义是通过与义原的关系建立起来的，义原的相似性和关联性并不能代表词义的近似。所以这里采用了另一种语言资源——同义词词林[4]，它对汉词进行了语义分类，整个框架基本是一棵树状层次结构。图 2 为同义词词林词义结构的部分示例，每一个节点为一个语义类，从根部语义类逐渐细化，离根部越近的语义类越抽象，所有语义类都表示一个概念，下对应汉语中的词，只有和叶节点对应的语义类是一个具有相似意义汉词的词群。

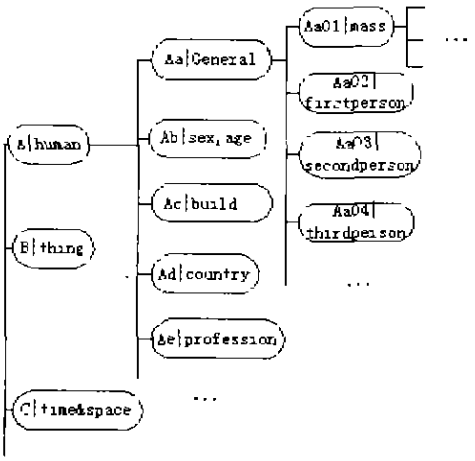


图 2 词林结构示例

同样和计算义原相似度一样，笔者根据该结构计算词义的相似度，其计算公式如下：

$$sim(c_1,c_2)=\begin{cases}\alpha_1/dist(c_1,c_2) & t'(c_1)=t'(c_2)\\ 0 & t'(c_1)\neq t'(c_2)\end{cases} \tag{3}$$

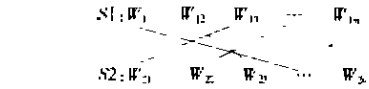
c_1,c_2 为词林中的任意两个词义，$t'(s_1)=t'(s_2)$ 计算 c_1,c_2 是否在词林的一个大类中，如果在一个大类中，则具有相似度，相似度和词义的距离成反比。词义距离是和结构图中的距离成正比的。

知网中由于每一个词义都是由若干义原进行解释的，进行词义之间相关度计算时，基于义原相似度和关联度进行。为了和义原间的关联度区分开，这里使用 relevancy 表示两个词义的关联度，计算时采用如下公式：

$$\begin{cases}Rele(c_1,c_2)=Rele(def(c_1),def(c_2))\\ Rele(def(c_1),def(c_2))\approx\sum_{s_i\in def(c_1),s_j\in def(c_2)}\max Rele(s_i,s_j)\\ def(c)=\{s_i|REL(c,s_i)\}\\ Rele(s_i,s_j)=w_1 sim(s_i,s_j)+w_2 asso(s_i,s_j)\end{cases} \tag{4}$$

$Rele(c_1,c_2)$ 表示两个词的相关度，$def(c)$ 是词 c 的解释义原集合，通过得到两个义原集合中关系最为密切的一些义原对，得到两个词的近似度和相关度，关系密切程度根据义原的近似度和关联度加权求和来衡量。

计算两个语句 s_1 和 s_2 之间的相关度时，假设它们已经经过词切分、同指消歧、语义标注，分别得到 m 个和 n 个的关键词序列：$w_{11}w_{12}..w_{1m}$ 和 $w_{21}w_{22}..w_{2n}$，下面用直线表示符合两句相关度最大互相匹配的词对。



由于在计算过程中不知道词对搭配的情况，因此采用动态优化的方法。计算公式如下：

$$\begin{cases}Rele(S_1,S_2)=M_{0,n}\\ 1/d_{ij}=\alpha\, Rele(w_{1i},w_{2j})+\beta Sim(w_{1i},w_{2j})\\ M_{0,j}=M_{i,0}=0\\ M_{1,1}=1/d_{11}\\ M_{i,j}=\max_{1\le k\le n}\{1/d_{ik}+M_{i-1,j}\}\end{cases} \tag{5}$$

（下转 83 页）

理解,测试和修改;

这7个类的DIT和NOC都不大,属于合理取值范围;

类DEVICE的LCOM值较大,说明该类的内部联系不大,不符合程序的"高内聚"要求,即模块化程度不高。

由此得出结论,类DEV_LIST_TYPE和DEVICE的可靠性和可维护性不太好,应该加强测试,或者重新进行设计。

表4 软件A/B传统复杂性度量与MOOD度量

度量	软件A	软件B
MHF	0.157	0.428
AHF	0.709	0.777
MIF	0.606	0.109
AIF	0.867	0.872
POF	0.041	0.001
COF	0.113	0.166
圈复杂性	83.333	54.125
注释率	0.195	0.334

(2)根据表4,笔者对软件B和A进行比较可以看出:软件B的方法隐藏和属性隐藏程度高,继承机制使用程度相近,多态机制使用程度低,圈复杂性好,耦合程度稍高。综合以上各种因素,不仅可以从定性的角度得出软件B的可靠性和可维护性较软件A好的结论,而且可以从定量的角度来比较软件B与软件A质量。

3 结束语

该文首先分析并讨论了软件质量度量模型,然后根据此模型分析了SQEvaluate中着重评测的可靠性、可维护性这两个质量特性及其子特性,在此基础上,对系统实现的度量进行分析并讨论了这些度量对相关软件质量子特性的影响,最后用一个实例分析如何用度量结果对软件质量进行评测。

(收稿日期:2001年1月)

参考文献

1.Roger S Pressman.软件工程——实践者的研究方法[M].北京:机械工业出版社,1999

2.朱三元.软件质量及其评价技术[M].北京:清华大学出版社,1990

3.Information technology-Software product evaluation-Quality characteristics and guidelines for their use[S].ISO/IEC 9126,1998

4.王振宁.程序复杂性度量[M].北京:国防工业出版社,1997

5.T j McCabe.A Complexity Measure[J].IEEE Transactions on Software Engineering,1976.12;SE-2(4)

6.M H Halstead.Elements of Software Science 1976

7.Shyam R Chidamber,Chris F Kemerer.A Metrics Suite for Object-Oriented Design[J].IEEE Trans.Software Engineering,1994.6;20(6)

8.Fernando Brito e Abreu,Walcelio Melo.Evaluating the Impact of Object-Oriented Design on Software Quality[C].IEEE Proceeding of METRICS'96,1996

(上接76页)

公式(5)中,α和β分别表示词相似度和相关度影响句子相关度的权值,$1/d_{ij}$表示第一句中的第i个词和第二句中的第j个词的语义关联度。

4 实验结果分析和讨论

文章利用IR系统,根据每个询问的关键词检索得到20个文档,每个文档中平均抽取50个语句,即对于每个询问存在约1,000个语句,然后和询问句进行相关度计算。表1列举了五个询问的结果,第二列为抽取相关语句的数目,第三列得到与询问语句具有最大相关值的查询语句,这里列出了相关度的值。

表1 Q-A系统查询结果实例

Query Nr.	Relevant sentences	Largest Relevancy
1st	1,029	205.127
2nd	986	232.411
3rd	997	334.826
4th	1003	602.133
5th	1002	603.329

从表1可以看到不同的查询所得到答案的相关值的大小可能有很大差别,相关度的值与句子的长度以及句子中的词都有很大关系,因此只纵向比较检索结果中句子与查询句子相关度的值,而不能横向得到各个查询最优答案的相关度值之间的关系。

文中进行相似度和相关度计算的公式(1)(2)(3)(4)中都有一些权值,这些权值的设置都是根据经验值取最优,同时从表1可看出句子相关度的值也没有进行归一化。这是因为语言本身带有不确定因素,而语言资源的编辑也带有相当的主观因素,对于不同词定义的标准和角度都可能是不同的,因此在整个语义计算中也就难以消除一些主观因素。

对于QA结果的评测也是一个难题,因为制定标准是一个非常繁琐的工作,因此笔者尽量简化标准,评判人判断查询答案实行0-1标准,如果认为答案回答了问题,则为1,即答案正确;反之为0,答案错误。通过约100人的评测,对于多数回答结果,98%以上的人认为是合理的。

5 结束语

该文通过一个QA系统介绍了语义计算方法,主要利用义原、词义和句子的语义相似度和相关度的计算,得到查询的最优答案。这种语义计算针对了汉语的特点:由语素构成词,由词构成句。并且把语义信息结合到计算过程中,从而减少了语句结构分析和框架填充过程的复杂度和歧义性,避免中间过程的损耗,同时,对义原间的相似度和关联度,词义间的近似度和相关度的有关计算,对于其它领域例如文本聚类,词义消歧,语句对齐等研究工作也可以提供具体而有效的支持。

(收稿日期:2001年3月)

参考文献

1.B Katz.From Sentence Processing to Information Access on the World Wide Web[C].AAAI Spring Symposium on Natural Language Processing for the World Wide Web,Stanford University,Stanford CA,1997

2.Rohini Srihari,Wei Li.Information Extraction Supported Question Answering[C].Proceedings of the 8th Text Retrieval Conference(TREC-8),National Institute of Standards and Technology,Gaithersburg MD,1999

3.E Voorhees.The TREC-8 Question Answering Track Report[R].National Institute of Standards and Technology,1999:77

4.穗志方.基于骨架依存树的语句相似度计算模型[C].计算语言学文集,1998;(3):176-184

5.董振东,董强.知网简介.http://www.keenage.com

6.周强,冯松岩.Building a relation network representation for how-net[C].Proceedings of 2000 International Conference on Multilingual Information Processing,Urumqi,China,2000:139-145

7.Gan K W,Wong P W.Annotating information structures in Chinese texts using HowNet[M].Hong Kong:Second Chinese Language Processing Workshop,2000:85-92

8.梅家驹,竺一鸣等.同义词词林[M].上海辞书出版社,1983