基于抽象知识点模型的句子相似度计算①

江长柱, 明晓乐, 张东香

(江苏科技大学 计算机科学与工程学院, 镇江 21200)

摘 要: 在认真研究用户咨询问题的基础上, 提出了一种新的计算用户咨询文本相似度的模型—抽象知识点模型. 利用该模型, 把用户咨询的问题分解为中心词和抽象知识点两部分. 从而用户咨询的文本相似度计算就转化为中心词和抽象知识点相似度计算. 另外, 本文对以往计算文本相似度的算法进行了改进. 实验结果表明, 提出的抽象知识点模型对文本相似度计算的准确率有很大提高作用, 计算结果更接近人工判断结果.

关键词: 中心词; 抽象知识点; 相识度计算

Sentence Similarity Computing Based on Abstract Knowledge Model

JANG Chang-Zhu, MING Xiao-Le, ZHANG Dong-Xiang

(School of Computer Science, Jiangsu University of Science and Technology, Zhen jiang 212000, China)

Abstract: User consultation questions are carefully analyzed and the new calculate of the text similarity model are pointed out—Abstract Knowledge Model. Using this model, the user's problem is decomposed into two parts: the Key-words and the Abstract Knowledge. Thus, the text similarity computing of user consult is transformed into Key-words and Abstract Knowledge similarity calculation. In addition, the paper has been improved the algorithm of the text similarity calculate in the past. Experimental results show that the proposed Abstract Knowledge Model plays a greatly role in text similarity calculation accuracy and the results closer to human judgment results.

Key words: key-words; abstract knowledge; similarity calculation

1 引言

随着网络的发展和人们生活节奏的加快, 现有的书本及图书馆资源已经远远满足不了人们学习、查询信息的需求. 而面对浩瀚的网络资源, 如何快速、准确的搜索出自己想要的学习、信息资源, 一直以来是广大科研人员追寻的永恒主题. 文本相似度计算是解决该问题的关键. 纵观国内外, 许多专家、学者已提出多种文本相似度的计算模型及方法, 并取得了一些成就, 大致有以下几种方法: 1)基于词层面的文本相似度计算[1,2]; 2)基于句子结构的文本相似度计算[3,4]; 3)基于语义层面的文本相似度计算[5-7].

然而现存的相似度计算方法及模型都存在其各自的不足之处, 因此本文提出一种新的计算模型—抽象知识点模型.

2 抽象知识点模型介绍

2.1 中心词、抽象知识点介绍

定义 1. 中心词: 用户咨询文本中人为理解不可少的事件对象.

定义 2. 抽象知识点: 用户咨询文本的主旨, 对应为中心词所要表达的意向.

先举个例子, 直观的说明下本文所提的抽象知识点模型.

举例: 地球和月球相距多少公里?

利用该模型可以把该用户咨询的语句拆分成如下的"中心词"+"抽象知识点"形式.

中心词 1	中心词 2	抽象知识点
地球	月球	距离查询

① 收稿时间:2014-09-17;收到修改稿时间:2014-10-16

利用该模型, 可以把用户咨询的所有文本的相似度计算转化为中心词和抽象知识点的相似度计算. 为此, 如何把用户咨询问题的抽象知识点提取出来,如何半自动存储及快速查询又是一个很重要的问题.

2.2 抽象知识点树的介绍

为了实现抽象知识点的半自动添加及快速查询, 我们构建了抽象知识点树. 本知识点树是用户咨询文本相似度计算模型的核心, 并定义了各个知识点之间的关系. 其结构为分层次的树状结构, 规则如下[8]:

1) 抽象知识点树是由手工总结出来的, 其知识点的增加将半自动实现; (半自动实现方法详见 1.3.3 知识点树自动扩展)

2) 抽象知识点的根节点定义为抽象父类, 子节点定义为抽象子类, 各抽象子类包含的信息相互无交集;

3) 抽象子类必须是抽象父类的某一个属性, 且各个属性涵盖的内容相互独立;

4) 抽象知识点树同层次的结点由专有的字符表示, 该字符就代表抽象知识点在抽象知识点树层数.

2.3 抽象知识点树的构建及扩展

2.3.1 构建抽象知识点树

构建抽象知识点树的结点一般都是用户咨询问题的核心及意向(下面会给出解释), 用知识点树的上下层关系来表示知识点的关系, 根节点通常用一个行业的虚概念表示(如:银行领域), 用 T 表示, 记为抽象父类, 位居第 0 层. 能表示该领域的特有属性知识点位居第一层, 用标号 B 表示, 如: 信用卡、支付宝、网上银行、股票、汇款、存款、网银限额等; 第二层, 由能表示第一层各知识点的属性组成, 用标号 A 表示, 如: “信用卡”, 其下又有“开通方法”、“审核”等属性; 第三层, 由能表示第二层各知识点的属性组成, 用 A_1 表示, 如: “开通方法”, 其下又有“查询”等属性.

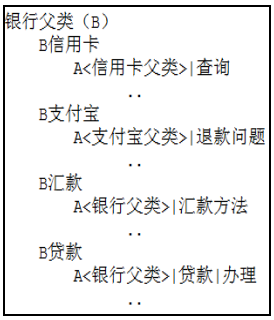

图 1　银行业抽象知识点树模型

一般认为, 第一层及其以上的知识点为非通用知识点, 是本领域的特有属性, 而从第二层开始往下的知识点为通用知识点, 为所有领域通用. 银行业抽象知识点树模型如图 1.

2.3.2 抽象知识点树系统存储

在我们的知识管理系统中, 我们的抽象知识点是以<*父类>、<*近类>存储的(*代表对应的知识点), 这为下一步自动添加知识点做准备. 银行业抽象知识点树系统存储如图 2 所示.

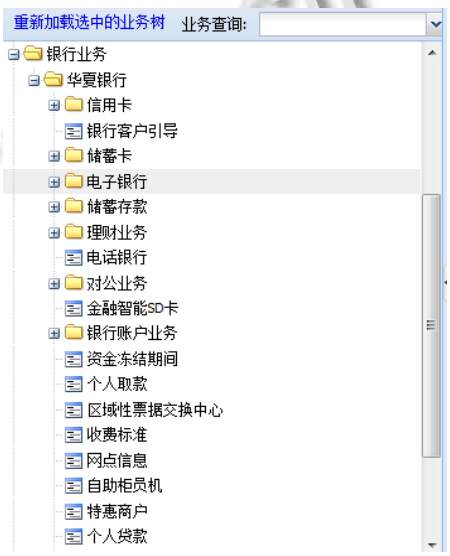

图 2　抽象知识点树系统存储

2.3.3 知识点树自动扩展

用户咨询的一句话经过系统的分词系统(本次采用的是中科院汉语词法分析系统 ICTCLAS2009)去停用词后, 得到对应知识点父类或近类序列(S_1,S_2, S_3,…,S_n)并输入到我们的知识点树存储系统, 知识点树系统遍历各自的结点, 得到抽象知识点集序列(W_1,W_2,W_3,...,W_s),W_i 中包括 w_i、w_i 的抽象父类和抽象子类. 采取深度优先遍历的方法自动扩展抽象知识点树, 规则是在距根节点最近的属性层加入对应新属性. 若 w_i 在抽象知识点树中无抽象父类, 则只遍历 w_i 及抽象子类; 若无抽象子类, 则只遍历 w_i 及抽象父类; 抽象知识点树的根结点不参与自动扩展.

3　抽象知识点相似度

3.1 抽象知识点相似度定义

抽象知识点相似度计算方式有别于以往的语义树模型的计算方式, 语句 1 为用户咨询的一句话经过去

停用词、分词后的抽象知识点父类或近类序列($S_1,S_2,S_3,\ldots,S_n$), 用集合 Q_1 表示, 语句 2 类似的得到抽象知识点父类或近类序列($S_1,S_2,S_3,\ldots,S_m$), 用集合 Q_2 表示, 若 Q_1 和 Q_2 相似, 则我们定义语句 1 和语句 2 相似, 用 $Sim(Q_1,Q_2)$来表示抽象知识点序列 Q_1,Q_2 的相似度. $Sim(Q_1,Q_2)$应满足如下定义:

1)对称性, 即: $Sim(Q_1,Q_2)=Sim(Q_2,Q_1)$.

2)区间性, 即: $Sim(Q_1,Q_2)$的值在[0,1]区间.

3)完全相似性, 即: 抽象知识点序列 Q_1、Q_2 完全相同, $Sim(Q_1,Q_2)=1$.

4)完全相反性, 即: 抽象知识点序列 Q_1、Q_2 完全不相同, $Sim(Q_1,Q_2)=0$.

3.2 抽象知识点相似度计算

抽象知识点表达了用户咨询的主题内容, 在语义树中计算两个抽象知识点相似度的核心是如何计算两个抽象知识点的语义距离. 通过学习可知道, 目前基于树状层次结构计算语义相似度研究方法相当成熟, Li 等人提出语义依存图的相似度模型[9],刘群提出了自己的语义相似度计算公式[10]. 其研究方法主要分为以下两类: a. 基于语义树两节点的语义距离; b. 基于语义树两节点所包含的共有信息.

1) 基于语义树两节点的语义距离

$$Sim(w_1,w_2)=\frac{\beta}{(\beta+Dist(w_1,w_2))} \tag{1}$$

公式(1)中, w_1,w_2 表示抽象语义树中的两个抽象知识点, $Dist(w_1,w_2)$表示抽象知识点 w_1,w_2 之间的语义距离, β 为调节参数, 取值为 $Sim(w_1,w_2)=0.5$ 时的语义距离.

2)基于语义树两节点所包含的共有信

$$Sim(w_1,w_2)=\frac{2*\log T(w_f)}{(\log T(w_1)+\log T(w_2))} \tag{2}$$

公式(2)中: w_f表示抽象知识点 w_1,w_2 最近的共同抽象父类, T(w)表示抽象知识点的抽象子类个数与语义树中所有抽象知识点个数的比值.

本文认为, 抽象知识点应该同时体现两个知识点的语义距离和两个知识点所包含的共有信息, 公式(3)是我们改进了抽象知识点相似度计算公式: (其中 β 是可调节参数, a、b、c 分别代表抽象知识点 1、抽象知识点 2 和抽象知识点 1 与抽象知识点 2 的最近的共同抽象知识点父类的层数.)

$$Sim(W_1,W_2)=\frac{2\times\sum_{\beta=1}^{c}\frac{1}{(\alpha+\beta)}}{\sum_{\chi=1}^{a}\frac{1}{(\alpha+\chi)}+\sum_{\delta=1}^{b}\frac{1}{(\alpha+\delta)}} \tag{3}$$

其中, α -可调节参数, a -抽象知识点 1 的层数, b -抽象知识点 2 的层数, c -抽象知识点 1 和抽象知识 2 的最近的共同抽象知识点父类的层数.

4 中心词相识度

前面我们已经阐述过, 我们会把用户咨询的一句话分成“中心词+抽象知识点”模式, 如何获取“中心词”及“抽象知识点”是本模块的关键. 我们主要采用以下两种方法:

1) 利用系统自动返回中心词

在我们的知识管理系统中, 增加了自动返回中心词的功能, 即以“中心词=<!*>”形式存储在书写的词模串里. 例如: “熊猫烧香是啥？”, 经过我们知识管理系统后(即去冗余词、分词类等)被分解为: “熊猫烧香(!网络病毒父类) 是(!是近类) 啥(!什么近类) ？(!问号父类) (4 words)”, 那么其对应的存储模式是: <!网络病毒父类>*<!是近类>*<!什么近类>@2#中心词=<!网络病毒父类>&编者=“江长柱”, 在这里大家看到的“中心词=<!网络病毒父类>”就是我们想要的中心词, 剩余部分即为抽象知识点.

2) 首先定位抽象知识点

该情况适用于系统中无用户所问的提问语句结构, 经过知识管理系统后(即去冗余词、分词类等)遍历抽象知识点树定位对应的抽象知识点, 未定位成功部分我们就把其作为中心词.

上述两种方法中, 方法 1)是最理想方法, 无论是速度还是准确率都能达到很高水准. 因此, 在电信、移动行业, 正在加大语句结构词模构建, 一般用户咨询的问题语句结构系统都已包含. 方法 2)查询速度没有方法 1)快, 但准确率也很高.

4.1 中心词相似度计算

用户咨询的文本一般都是短句, 故其中心词一般不会超过四个, 在这里我们取 K_1, K_2 为两个用户咨询的中心词集合, 假设 K_1,K_2 分别包含 X,Y 个中心词, 设 K_1 中的第 x 个中心词和 K_2 中第 y 个中心词之间的相似度为 Sim(x,y),我们可以构建中心词相似度矩阵如下:

$$\begin{pmatrix} S_{11} & \cdots & S_{1y} \\ \vdots & \ddots & \vdots \\ S_{x1} & \cdots & S_{xy} \end{pmatrix}$$

上述矩阵中: S_{xy} 表示中心词集合 K_1 第 x 个中心词与中心词集合 K_2 第 y 个中心词的相似度值, 其中: x 的取值范围为(1, X),y 的取值范围为(1, Y).

取中心词 x,y 最大相似度方法如下:

$$S_x = \max\left(S_{x1}, S_{x2}, S_{x3}, \cdots, S_{xY}\right) \tag{4}$$

$$S_y = \max\left(S_{y1}, S_{y2}, S_{y3}, \cdots, S_{yX}\right) \tag{5}$$

中心词集合 K_1, K_2 之间的相似度计算公式定义如下:

$$Sim(K_1, K_2) = \frac{1}{2} \times \left(\frac{\sum_{x=1}^{X} S_x}{X} + \frac{\sum_{y=1}^{Y} S_y}{Y} \right) \tag{6}$$

公式(6)中:

K_1 -----语句 1 的中心词序列集合

K_2------语句 2 的中心词序列集合

S_x------中心词集合 K_1 与中心词集合 K_2 相似度的最大值

S_y------中心词集合 K_2 与中心词集合 K_1 相似度的最大值

X------中心词集合 K_1 中中心词个数

Y------中心词集合 K_2 中中心词个数

5 用户咨询的语句相似度计算

由上述中心词和抽象知识点的相似度计算, 我们可以定义用户咨询的两语句 Sen_1、Sen_2 相似度计算公式如下:

$$\begin{aligned} &Sim(Sen_1, Sen_2) \\ &= \theta \times Sim(K_1, K_2) + (1-\theta) \times Sim(W_1, W_2) \end{aligned} \tag{7}$$

公式(7)中:

K_1——语句 1 的中心词序列集合

K_2——语句 2 的中心词序列集合

W_1——语句 1 抽象知识点

W_2——语句 2 抽象知识点

θ——可调节参数(θ 为 0.53 是本次试验的最佳确定值)

6 相关实验及实验结果分析

通过网络爬虫我们从百度知道和百度爱问抓取大量的文本资料, 通过聚类, 我们从中选取了通信行业和银行行业的语料, 人工过滤后选取 2000 条常用语句(分成四组, 每组 500 条语句)作为实验集, 该实验集具有以下特征:

1) 一个中心词+一个知识点

如: 5 元 GPRS 套餐业务如何开通?

2) 一个中心词+两个知识点

如: 5 元 GPRS 套餐业务如何取消? 到哪里可以办理?

3) 两个中心词+一个知识点

如: 5 元 GPRS 套餐改为 10 元 GPRS 套餐如何办理?

4) 两个中心词+两个知识点

如: 5 元 GPRS 套餐改为 20 元 GPRS 套餐如何办理? 有优惠吗?

当然, 在用户咨询的问题中可能有多个中心词和抽象知识点的情况, 其计算方法不变. 利用本文提出的方法 1)、2)、3)、4)中心词和抽象知识点如下:

中心词 1	中心词 2	抽象知识点 1	抽象知识点 2
5 元套餐		业务开通方法	
5 元套餐		业务取消方法	办理地点查询
5 元套餐	10 元套餐	办理方法查询	
5 元套餐	20 元套餐	办理方法查询	是否优惠查询

实验结果:

本次试验数据结果由以下四个实验集测试数据组成, 分别为: 实验集 1: 一个中心词+一个抽象知识点; 实验集 2: 一个中心词+二个抽象知识点; 实验集 3: 一个中心词+二个抽象知识点; 实验集 4: 两个个中心词+两个抽象知识点. 测试数据生成的折线图如图 3 所示.

表 1　一个中心词+一个抽象知识点

方法	准确率(%)
基于 VSM 的方法	90.22
基于语义树的方法	89.78
本文提出的方法	93.89

表 2　一个中心词+二个抽象知识点

方法	准确率(%)
基于 VSM 的方法	88.04
基于语义树的方法	87.67
本文提出的方法	93.46

表 3 两个中心词+一个抽象知识点

方法	准确率(%)
基于 VSM 的方法	89.37
基于语义树的方法	84.64
本文提出的方法	93.12

表 4 两个个中心词+两个抽象知识点

方法	准确率(%)
基于 VSM 的方法	86.45
基于语义树的方法	81.38
本文提出的方法	92.78

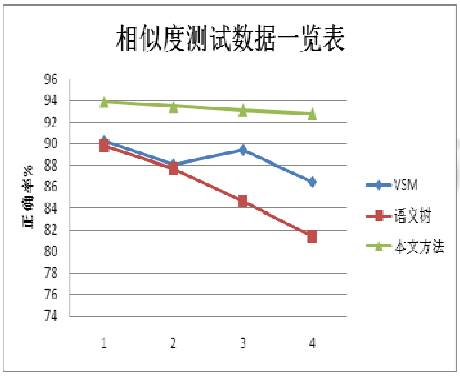

图 3 三种方法相似度测试数据

通过该实验结果图, 我们可以看到, 随着中心词和抽象知识点的增加, 各个方法的测验数据呈下降趋势, 其中基于语义树的方法下降较为明显, VSM 受中心词数量的影响较大, 基于本文提出的方法测试数据较稳定, 且保持较高的正确率.

7 结论与展望

本文基于用户咨询的问题提出了抽象知识点模型的概念, 首先把用户咨询问题分解为中心词和抽象知识点两部分, 在计算抽象知识点的部分时我们构建了抽象知识点树, 不仅考虑抽象知识点的语义距离同时也考虑了抽象知识点的语义, 更有效的分析和表示了用户咨询问题的真实表达意思. 文中给出的多个中心词和抽象知识点语料中, 本方法测试结果比别的方法准确率更高. 但目前此抽象知识点语义树涵盖的范围有限, 仅限通信、银行和保险行业, 而且抽象知识点树的完善是半自动进行的, 需要不断总结并完善, 是一个长期过程.

基于本文抽象知识点树已在中国电信和某银行智能查询系统中投入使用, 反馈信息良好, 下步我们将扩大该语义树覆盖范围, 使其投入到更多的行业中.

参考文献

1 秦兵,刘挺,王洋,等.基于常问问题集的中文问答系统研究.哈尔滨工业大学学报,2003,35(10):1179–1182.

2 Ho C, Azrifah M, Murad A. Word Sense Disambiguation-Based Sentence Similarity. Coling, 2010: 418–426.

3 黄莉.基于动态特征词的中文句子相似度计算.宝鸡文理学院学报(自然科学版),2013,3(33):49–52.

4 梁正平,纪震,刘小丽.基于语义模板的问答系统研究.深圳大学学报理工版,2007,3(24):281–284.

5 李素建.基于语义计算的语句相关度研究.计算机工程与应用,2002,38(7):75–76,83.

6 穗志方,俞士汶.基于骨架依存树的语句相似度计算模型.中文信息处理国际会议(ICCIP`98).北京,1998.

7 安建成,武俊丽.基于语义树的概念语义相似度计算方法研究.微电子学与计算机,2011,28(1):138–141,146.

8 张映海.基于概念树扩展的中文文本检索研究.计算机工程与应用,2008,44(26):154–157.

9 Li R, Li SH, Zhang Z. The semantic computing model of sentence similarity based on Chinese FrameNet. Proc. of Web Intelligence/IAT Workshops. Los Alamitos, CA. IEEE Computer Society. 2009. 255–258.

10 张亮,尹存燕,陈家骏.基于语义树的中文词语相似度计算与分析.中文信息学报,2010,24(6):23–30.