基于抽象概念的知网词语相似度计算

朱新华[1,2]，郭小华[1]，邓 涵[1]，马润聪[1]

（1. 广西师范大学 计算机科学与信息工程学院，广西 桂林 541004；
2. 广西区域多源信息集成与智能处理协同创新中心，广西 桂林 541004）

摘 要：针对基于知网的词语相似度算法进行研究，提出一种基于抽象概念的词语相似度的快速计算方法。将《知网》义项语义表达式中带有关系约束的第一独立义原定义成抽象概念，将义项语义表达式转换成一个多层次的抽象概念组；根据义项定义中的抽象概念将义项挂到《知网》现有的义原树中，形成一棵包含义原、抽象概念与义项等概念的义项树；利用义项树中的深度与路径，在现有优秀算法基础上，通过适当的义项定义的预处理与参数调节，直接计算义项间的语义相似度，避免复杂的意义相似性计算。实验结果表明，该方法对于 MC30 词对的相似度计算值与人工判定值相比，取得了 0.84 的 Pearson 相关系数，达到了目前优秀词语相似度算法的水平。

关键词：词语相似度；知网；义项树；抽象概念；最短路径；深度

中图法分类号：TP391 **文献标识号**：A **文章编号**：1000-7024（2017）03-0664-07

doi：10.16208/j.issn1000-7024.2017.03.020

Word similarity calculation based on abstract concept in HowNet

ZHU Xin-hua[1,2], GUO Xiao-hua[1], DENG Han[1], MA Run-cong[1]

(1. College of Computer Science and Information Technology, Guangxi Normal University, Guilin 541004, China;
2. Collaborative Innovation Center of Multi-Source Information Integration and Intelligent Processing in Guangxi, Guilin 541004, China)

Abstract: By researching word similarity algorithms based on HowNet, an efficient method for calculating word similarity based on the abstract concept was put forward. The first independent sememe with relationship constraints in the semantic expression of a sense was defined as an abstract concept, and the semantic expression of the meanings was transferred into a multi-level abstract concept group. According to the abstract concept in the definition of a sense, the senses were hung on the existing sememe tree in the HowNet and a sense tree that containing all the concepts was formed. Through the proper pretreatment and parameter adjustment of the senses' definition, using depth and path in the sense tree and improving the existing outstanding algorithm, the semantic similarity between the senses was directly calculated and the complicated meaning similarity calculation was avoided. That the Pearson correlation coefficient between the human judgments in MC30 dataset and the computational measures presented in this approach is 0.84 is showed in the experiments, which achieves the cevel of good word similarity algorithms.

Key words: word similarity; HowNet; sense tree; abstract concept; shortest path; depth

0 引 言

目前词语相似度计算方法大致可以分为以下两种：一种是基于大规模语料库进行统计和计算；另一种是基于世界知识进行计算[1]。目前，国内基于世界知识计算词语相似度的方法主要有基于同义词词林[2]的词语相似度计算和基于知网[3]的词语相似度计算。同义词词林的结构清晰，所有义项都在同一棵大树中，因此非常便于使用直观的深度与路径方法计算词语相似度；而在《知网》中，只存在一棵义原树，并没有给出义项树，而是给出了义项的定义

收稿日期：2016-01-06；**修订日期**：2016-03-10

基金项目：国家自然科学基金项目（61462010、61363036）

作者简介：朱新华（1965-），男，广西桂林人，教授，研究方向为自然语言处理、智能教学系统等；郭小华（1992-），女，湖北武汉人，硕士研究生，研究方向为自然语言处理；邓涵（1991-），女，湖北荆州人，硕士研究生，研究方向为自然语言处理；马润聪（1989-），男，湖北洪湖人，硕士研究生，研究方向为自然语言处理。E-mail：zxh429@263.net

与意义解释，因此在《知网》中目前主要依据意义的相似性来计算义项的相似度，由于意义相似性的计算需要分组计算义项定义中的义原组，其计算过程较为复杂。本文提出了一种基于抽象概念的知网词语相似度的快速计算方法。该方法通过义项语义表达式中的抽象概念，将所有义项全部挂到义原树中，形成一棵包含所有概念的义项树，采用一种基于深度与路径的方法直接计算义项的相似度。

1　相关知识

1.1　知网简介

《知网》[4]是董振东先生为实现中英文机器翻译建设的一个中英双语常识库，目前仍在更新发展中。《知网》中主要概念有：义原、义项、语义表达式。义原是《知网》描述“概念”的基本单位，即原子概念，用于对其它义项进行意义解释。义项又称“概念”，是一个词语的一种解释，一个词语允许有多个义项。

语义表达式可以简写为 DEF，它是义项的主体，由一个个结合知识描述符号的基本义原所组成，用于解释义项的意义。按照刘群等[3]的划分方法，DEF 中的义原被划分为如下 4 个部分：

（1）排在最前面的称为第一独立义原，用于描述义项概念的主要语义特征与直接上位。

（2）除第一独立义原之外，不带任何描述符号的义原称为其它独立义原，用于描述概念的次要语义特征。

（3）用“=”连接的义原称为关系义原，用于描述概念和概念之间的关系。

（4）以符号“～！@＃￥%&.＊”等开头的义原称为符号义原。其中，每一种符号表示一种特殊的关系，从而形成多种概念之间的关系。

义原在知网中分为事件、实体、属性、属性值、数量、数量值、次要特征、语法、动态角色与动态属性 10 大类，在 2000 版本中共计有 1500 多个义原，目前还在不断地扩充之中，在每个大类中，义原根据上下位关系构建出树状结构如图 1 所示[4]。

本文所述知网如无其它特殊说明，均指目前在知网官方网站可免费下载的 2000 年版。

1.2　基于知网的词语相似度的研究现状

刘群等[3]在基于知网的词语相似度研究中开辟了先河，他们对知网的义项文本做了相应调整，提出了将知网词语相似度的计算转换为词语语义表达式（DEF）相似性计算的方法，即根据词语的意义相似性计算词语的相似度。他们采取把整体相似度还原为部分相似度的加权平均策略，得出两个义项间的语义相似度计算公式如下

$$Sim(S_1,S_2)=\sum_{i=1}^{4}\beta_i\prod_{j=1}^{i}Sim_j(S_1,S_2) \qquad (1)$$

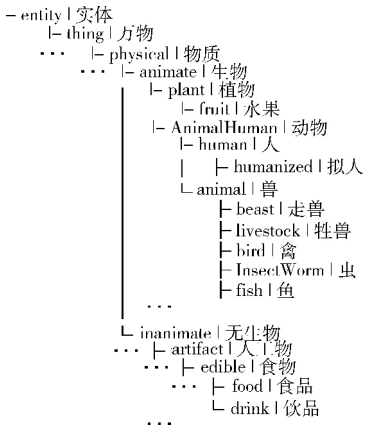

图 1　义原的树状层次结构

其中，β_i（$1\leqslant i\leqslant 4$）是调节参数，且有：$\beta_1+\beta_2+\beta_3+\beta_4=1$，$\beta_1\geqslant\beta_2\geqslant\beta_3\geqslant\beta_4$。$Sim_j(S_1,S_2)$ 表示两个义项中特定义原间的相似度，具体定义和公式请参见文献［3］。

此后基于知网词语相似度的研究大多都是在刘群等方法的基础上进行的改进。例如，葛斌等[1]从义原的深度、区域密度等方面改进了义原的相似度算法，该方法计算出的词语相似度呈两端扩散且能更精确的区分微小语义之间的差异。张瑞霞等[5]研究了基于知网的未收录词语的相似度计算方法。朱征宇等[6]认为语义表达式中的义原有一定的线性关系，并提出了利用加权策略计算知网词语相似度等。这些计算词语相似度的方法虽然对刘群的方法有所改进，但是本质上依然是基于意义相似度计算词语相似度的方法。

2　基于抽象概念的义项树

2.1　抽象概念的定义

根据《知网》作者的定义，《知网》中的知识描述符号事实上是一种简化的描述逻辑（DL）[7]，例如《知网》中的义项定义：

保镖＝＝“human | 人，＃occupation | 职位，＊protect | 保护”

相当于描述逻辑中的概念定义[7]：

保镖＝＝人 ⋂ ∃ 相关概念．职位 ⋂ ∃ 施事．保护

描述逻辑中有 5 个最基本的构造算子：合取（⋂）、析取（⋃）、非（ ）、存在量词（∃）、全程量词（∀）。

在义项的 DEF 中，第一独立义原用于描述义项的主要语义特征与直接上位，其它独立义原用于描述概念的次要语义特征，关系义原与符号义原用于描述概念和概念之间的关系[3]。事实上，我们也可以将其它独立义原转化成某种关系义原或符号义原，例如在“和尚”的 DEF＝＝{human | 人，religion | 宗教，male | 男} 中，其它独立义原“宗教”可以转换成符号义原“＃宗教”，表示“和尚”与“宗教”之间具有相关关系，而独立义原“男”可以转

化成关系义原“性别＝男”。因此，我们应用 DL 的语法规则到《知网》义项定义中，将第一独立义原作为它的直接上位（父结点），将其它义原均转换成描述逻辑中带存在量词（∃）的关系约束[8]，并将义原间的逗号转换为描述逻辑中的交集运算符“∩”，从而形成描述逻辑表示的义项定义。以义项“保镖”为例，由于“人”是第一独立义原，而“职位”和“保护”是用来修饰人的符号关系义原，根据符号“#”与“*”所代表的关系，使用描述逻辑义项“保镖”的定义为：保镖＝＝人∩∃相关概念．职位∩∃施事．保护。在该逻辑表达式中，“保镖”可以理解为：保镖是一个人，也是一种职位，它的工作性质是保护。

在本文中，为了便于建立我们计算词语相似度所需的义项树，我们将《知网》中被定义的义项以及所有的义原称为实概念，表示该概念在现实中有对应的词语存在。在描述逻辑中，一个关系约束就是一个原子概念[9]，因此，我们将义项定义中第一独立义原与关系约束的交集（或并集）称为抽象概念。抽象概念在现实中并无具体的词语对应，仅用于对另一概念的解释。在上述“保镖”定义的例子中，“保镖”、“人”、“职位”与“保护”都是实概念，在该定义等式的右侧中存在两个抽象概念：“人∩∃相关概念．职位”、“（人∩∃相关概念．职位）∩∃施事．保护”，其中后者为义项“保镖”的等价抽象概念。

2.2 搭建基于抽象概念的义项树

为了构造出一棵我们所需要的带有抽象概念的义项树，根据义项 DEF 中的抽象概念，我们可以在义原树的基础之上，按照如下步骤构造出义项树：

（1）根据 DEF 中的第一独立义原，确定义项在义原树的直接上位。

（2）按顺序，依次将第一独立义原与 DEF 中的关系约束构成抽象概念，每形成一个抽象概念就在义项对应义原树的直接上位下增加一个结点。例如，根据“保镖”的 DEF，其在义原树中的直接上位为“人”，可在“人”的下方添加抽象概念“人∩∃相关概念．职位”、并在“人∩∃相关概念．职位”的下方添加抽象概念“（人∩∃相关概念．职位）∩∃施事．保护”。

（3）重复（2），直到第一独立义原与 DEF 中的全部关系约束构成一个与义项等价的抽象概念，此时该抽象概念就是义项在义原树中的位置，如图 2 所示。

（4）重复（1）～（3），直至将知网中的所有义项全部挂到义原树中，此时的义原树就被成功地扩展为包括知网所有概念的义项树。

在图 2 中，我们增加了一个虚拟的结点作为根结点连通不同大类的义原树，从而形成一棵完整的词语义项树，以便于计算不同词性的词语之间的相似度。

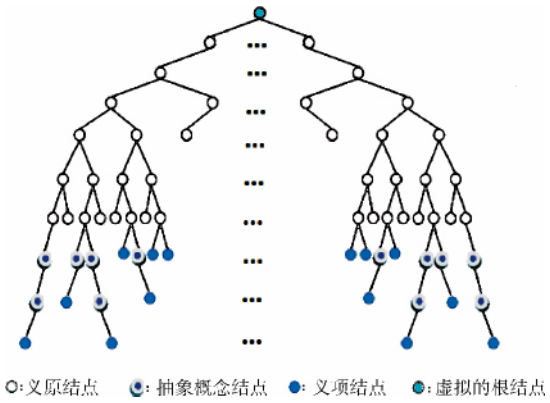

图 2 义项树

3 基于抽象概念的知网词语相似度的计算

3.1 基于抽象概念的知网词语相似度的计算公式的提出

在构造出义项树后，利用该义项树便可直接计算知网语义相似度。在目前基于本体的词语相似度算法中，深度与路径是两个主要被考虑的因素[10,11]。Wu 等[10]在基于 WordNet[12]机器翻译研究中，为解决词语选择问题，首先提出了这种方法。他们定义词语 w_1 和 w_2 的相似度计算公式为

$$Sim(w_1,w_2)=\frac{2H}{2H+N_1+N_2} \tag{2}$$

这里 N_1 和 N_2 分别表示词语 w_1 和 w_2 到它们最近公共父结点的路径距离，H 表示词语 w_1 和 w_2 最近公共父结点在本体中的深度。

我们采用这个算法计算基于抽象概念的知网义项相似度，同时规定深度与路径距离不限制为整数。经实验，我们发现式（2）的计算值普遍高于人工判定值，公式中取一倍最近公共父结点的深度时，计算出的词语相似度更接近于人工判定值。鉴于此，我们只取一倍最近公共父结点的深度来反映义项间的共性。我们提出的基于抽象概念的知网义项的初始相似度计算公式为

$$InitSim(s_1,s_2)=\frac{Deep(LCP(s_1,s_2))}{Deep(LCP(s_1,s_2))+Path(s_1,s_2)} \tag{3}$$

其中，$Path(s_1,s_2)$ 表示义项 s_1 与 s_2 在义项树中的最短路径距离，$Deep(LCP(s_1,s_2))$ 表示义项 s_1 和 s_2 的最近公共父结点 $LCP(s_1,s_2)$ 在义项树中的深度。

考虑到我们在构造义项树时加了一个虚拟的根结点，在计算义项相似度之时，为避免因为增加了一个根结点而使义项最近公共父结点深度增加所带来的误差，我们给每个相似度乘以一个取值范围为 0 到 1 的余弦值，以此对相似度进行平滑调节。这样，最后义项间语义相似度的计算公式如下

$$FinalSim(s_1,s_2)=InitSim(s_1,s_2)\times Cos((1-InitSim(s_1,s_2))\times\frac{\pi}{2}) \tag{4}$$

其中，$InitSim(s_1, s_2)$ 是式（3）计算的义项 s_1 与义项 s_2 之间的相似度值，$FinalSim(s_1, s_2)$ 是我们最后计算出的义项间的语义相似度。另外，为了克服式（3）中，因义项间的最近公共父结点为根结点而造成的相似度为 0 的情况，我们把义项间最近公共父结点为根结点的义项最终相似度取值为 0.01。

考虑到有的词语对应多个义项，我们取两个词语所有义项对中的最大相似度作为两个词语的最终相似度。设词语 w_1 有 n_1 个义项，词语 w_2 有 n_2 个义项，则我们提出的词语 w_1 与 w_2 在知网中的相似度计算公式为

$$sim(w_1, w_2) = \begin{cases} \max\limits_{i=1\cdots n_1, j=1\cdots n_2} \{FinalSim(s_{1i}, s_{2j})\}, & w_1 \neq w_2 \\ 1, & w_1 = w_2 \end{cases} \tag{5}$$

其中，$FinalSim(s_{1i}, s_{2j})$ 表示式（4）计算出的词语 w_1 的第 i 个义项与词语 w_2 的第 j 个义项间的相似度值。

3.2 义项间最短路径距离的计算

我们认为如图 2 所示的义项树中，义原结点与抽象概念结点对义项间距离的贡献程度是不相同的，其中义原结点是义项间距离的主要贡献者，而抽象概念结点只是辅助贡献者。因此，根据义项 s_1 与 s_2 的 DEF 定义，我们把式（3）中，义项 s_1 到 s_2 的最短路径距离定义成第一独立义原之间的距离与基于抽象概念的路径补偿两部分之和，如式（6）所示

$$Path(s_1, s_2) = path(FS(s_1), FS(s_2)) + AbstComp(s_1, s_2) \tag{6}$$

其中，$FS(s)$ 表示义项 s 的 DEF 定义中的第一独立义原，$path(FS(s_1), FS(s_2))$ 表示两个义项的第一独立义原间的最短路径距离，其值等于该最短路径中边的条数。$AbstComp(s_1, s_2)$ 表示基于抽象概念的路径补偿。

李峰等[13]认为不带符号的独立义原是对义项的一种直接描述，而带有符号标识的义原是一种间接描述，直接描述比间接描述的作用更大，并且是识别该义项必不可少的属性特征。为了区分直接描述（其它独立义原）和间接描述（关系义原与符号义原）对计算词语语义的影响，我们在路径补偿计算中添加相应的参数来调节这些影响。另一方面，在《知网》2000 版的义项定义中，存在许多只包含第一独立义原的不完全定义的情况，如“轿车”与“汽车”的 DEF 中都只包含第一独立义原 {LandVehicle | 车}，为避免这种不完全定义给义项相似度计算所带来的不利影响，我们在路径补偿计算中引入初值函数。最终，我们得出如下的路径补偿计算公式

$$AbstComp(s_1, s_2) = \frac{1}{SameAbs(s_1, s_2) + 1} + Sum(SS) \times \alpha + Sum(OS) \times \beta \tag{7}$$

其中，α 用于调节符号关系义原与符号义原对路径补偿的影响，β 用于调节其它独立义原对路径补偿的影响，$SameAbs(s_1, s_2)$ 表示义项 s_1 和 s_2 定义中相同的概念结点个数，$Sum(SS)$ 表示两个义项的 DEF 定义中不相同的符号义原与关系义原的个数之和，$Sum(OS)$ 表示两个义项的 DEF 定义中不相同的其它独立义原的个数之和，$1/(SameAbs(s_1, s_2)+1)$ 为路径的初始补偿函数。

由于义原概念中并不包含抽象概念，当计算两个义原概念之间的相似度时，基于抽象概念的路径补偿取 0。例如当计算“水果”和“食物”的相似度时，由于二者都是义原，因此计算过程中基于抽象概念的路径补偿取值为 0。当计算义原概念与义项概念之间的相似度时，我们只取一半的路径初始补偿，例如当计算“水果”和“梨”的相似度时，由于“水果”是义原而“梨”是义项，此时路径补偿的为 $(1/2) * (1/(SameAbt(s_1, s_2)+1))$。

3.3 最近公共父结点在义项树中深度的计算

在式（3）中，$Deep(LCP(s_1, s_2))$ 表示义项 s_1 和 s_2 的最近公共父结点 $LCP(s_1, s_2)$ 在义项树中的深度。根据义项不同性质的公共父结点，我们将其在义项树中的深度分为以下几种情况分别进行计算：

（1）如果两个义项定义的第一独立义原相同，且存在相同的抽象概念，则我们将最后相同的抽象概念定义成义项间的最近公共父结点，例如，对于义项库文件中的“中午”和“正午”的 DEF 定义：中午＝＝ {time | 时间，afternoon | 午}、正午＝＝ {time | 时间，afternoon | 午}，二者的第一独立义原相同，且存在相同的抽象概念“时间 ∩ ∃ 与相关．午”，因此它们最近公共父结点为抽象概念，其深度按如下公式进行计算

$$Deep(LCP(s_1, s_2)) = Deep(SameFS) + SameSum(SS) \times \alpha + SameSum(OS) \times \beta \tag{8}$$

其中，$SameFS$ 表示两个义项定义中相同的第一独立义原，$Deep(SameFS)$ 表示相同的第一独立义原在义项树中的深度，其值等于该独立义原在义项树中到根结点之间的边的条数。$SameSum(SS)$ 表示两个义项的 DEF 定义中相同的符号义原与关系义原的个数之和。$SameSum(OS)$ 表示两个义项的 DEF 定义中相同的其它独立义原的个数之和。α、β 的值与式（7）相同。

（2）如果两个义项定义中第一独立义原相同，但不存在相同的抽象概念，则我们将相同的第一独立义原定义成义项间的最近公共父结点，其深度等于该独立义原在义项树中到根结点之间的边的条数。

（3）如果两个义项定义中无任何相同的成分，则分别按两个义项定义中的第一独立义原，在义原树中向上查找最近公共父结点，直到根结点为止。此时，义项间的最近公共父结点为义原树中的某个义原结点，其深度等于该义原在义项树中到根结点之间的边的条数。

3.4 语义表达式的预处理

在按上述公式计算义项语义相似度之前，对于一些特殊的义项语义表达式的定义，需要进行适当的预处理，以确保得出合理的相似度计算结果。

（1）在知网的义项库文件中，同时包含义项与义原两类概念的定义，我们按以下规则对二者进行识别与处理：如果第一独立义原与被定义的概念不相同，则被定义的概念为义项，其定义表达暂不作处理；如果第一独立义原与被定义的概念相同，则被定义的概念为义原，其定义表达处理为空。例如，对于义项库文件中的“梨”和“水果”概念的 DEF 定义：梨＝＝｛fruit｜水果｝、水果＝＝｛fruit｜水果｝，按照上述规则可判定出概念“梨”为义项，而概念“水果”为义原。

（2）当待比较的两个义项定义的第一独立义原相同，或者其中一个义项定义中的第一独立义原与另一个义项定义中的其它独立义原相同时，依次继续从两个义项定义中寻找相同的其它独立义原、关系义原或符号义原，并将相同的部分同时依次移到两个义项定义中的前部，此时调整后两义项 DEF 定义中分别排在最前面的义原为它们的第一独立义原，义项树中的这两个义项及其 DEF 定义中义原将以新的第一独立义原为直接上位，按照 DEF 重新排序后的顺序重新接到义项树中去，以确保在义项树中正确找出相同的抽象概念作为义项的公共父结点。例如，对于义项库文件中的“小伙子”和“男孩子”的 DEF 定义：小伙子＝＝｛human｜人，young｜幼，male｜男｝，男孩子＝＝｛human｜人，male｜男，young｜幼｝，按照上述规则男孩子的 DEF 重新排序后为：男孩子＝＝｛human｜人，young｜幼，male｜男｝。对于义项库文件中的“宝石”和“宝物”的 DEF 定义：宝石＝＝｛stone｜土石，treasure｜珍宝｝，宝物＝＝｛treasure｜珍宝｝，按照上述规则宝石的 DEF 重新排序后为：宝石＝＝｛treasure｜珍宝，stone｜土石｝。

（3）如果待比较的两个义项定义的第一独立义原不相同，但存在一个义项定义中的其它独立义原、关系义原或符号义原与另一个义项的第一独立义原相同，此时表明这两个义项之间存在某种关联性，为体现这种关联性对词语相似度计算的贡献，我们把这种情况下的两个义项的定义处理成省略相同义原后面的所有其它义原。例如，在计算义项“海岸＝＝｛part｜部件，%land｜陆地，#waters｜水域，edge｜边｝”与“丘陵＝＝｛land｜陆地｝”时，由于义项“海岸”中的符号义原“陆地”与义项“丘陵”中的第一独立义原相同，因此在计算这两个义项的相似度之前，将二者的语义表达式的定义处理成“海岸＝＝｛part｜部件｝”与“丘陵＝＝｛land｜陆地｝”。

注意，上述语义表达式的预处理只是临时性的，当计算完义项相似度后又恢复其原状。

3.5 参数的设定

为了设定合理的参数值，我们对 5151 组参数值进行实验测试。鉴于直接描述比间接描述的作用更大，我们认为 α 不应该超过 β，而且为了区分第一独立义原和其它独立义原的作用，我们规定 β 不应该超过 1。因此，α 与 β 之间存在以下关系

$$0\leqslant\alpha\leqslant\beta\leqslant1$$

在实验过程中，我们要求这 5151 组参数中 α 和 β 的值满足上述关系，基于本文篇幅的限制，我们暂且给出我们实验中不同参数取值的部分实验数据见表 1，用该数据制成的三维图如图 3 所示。由于在参数以 0.01 为变化幅度时，Pearson 相关系数值平均变化幅度为 0.000 002，表 2 中数据均保留 6 位小数，便于数据变化的观测。在表 2 中，每一行从左到右、每一列从上到下都是不断增大的，可以看出随着 α 和 β 值的增大，用本文方法计算得到的 MC30 词语对间的相似度与其对应的人工判定值计算得到的 Pearson 相关系数值是越来越大（如图 3 所示）。因为 $0\leqslant\alpha\leqslant\beta\leqslant1$，为了使我们的实验效果达到最优化，我们将 α 和 β 值均取它们可以取的最大值，即 $\alpha=1$，$\beta=1$。

表 1 本文方法中不同参数对应的 Pearson 相关系数

	β=0.92	β=0.93	β=0.94	β=0.95	β=0.96	β=0.97	β=0.98	β=0.99	β=1.00
α=0.88	0.838 838	0.839 480	0.840 112	0.840 735	0.841 350	0.841 955	0.842 552	0.843 140	0.843 719
α=0.89	0.838 840	0.839 481	0.840 114	0.840 737	0.841 351	0.841 957	0.842 553	0.843 141	0.843 721
α=0.90	0.838 842	0.839 483	0.840 115	0.840 739	0.841 353	0.841 958	0.842 555	0.843 143	0.843 722
α=0.91	0.838 843	0.839 485	0.840 117	0.840 740	0.841 355	0.841 960	0.842 557	0.843 145	0.843 724
α=0.92	0.838 845	0.839 486	0.840 119	0.840 742	0.841 356	0.841 962	0.842 558	0.843 146	0.843 726
α=0.93		0.839 488	0.840 120	0.840 743	0.841 358	0.841 963	0.842 560	0.843 148	0.843 728
α=0.94			0.840 122	0.840 745	0.841 359	0.841 965	0.842 562	0.843 150	0.843 729
α=0.95				0.840 747	0.841 361	0.841 967	0.842 563	0.843 151	0.843 731
α=0.96					0.841 363	0.841 968	0.842 565	0.843 153	0.843 733
α=0.97						0.841 970	0.842 567	0.843 155	0.843 734
α=0.98							0.842 568	0.843 156	0.843 736
α=0.99								0.843 158	0.843 738
α=1.00									0.843 739

表 2　不同方法计算的 MC30 词语相似度

词语 1	词语 2	刘群等[3]	李峰等[13]	本文方法	本文方法（不做预处理）	人工判定值
轿车	汽车	1	1	0. 998 750	0. 998 75	0. 98
宝石	宝物	0. 1455	0. 6	0. 856 822	0. 856 822	0. 96
旅游	游历	1	1	1. 000 000	0. 999	0. 96
男孩子	小伙子	1	1	0. 999 583	0. 999 583	0. 94
海岸	海滨	1	1	0. 999 583	0. 999 583	0. 925
庇护所	精神病院	0. 5792	0. 5376	0. 758 705	0. 758 705	0. 9025
魔术师	巫师	0. 676	0. 7503	0. 834 093	0. 834 093	0. 875
中午	正午	1	1	0. 998333	0. 998 333	0. 855
火炉	炉灶	0. 5896	0. 5584	0. 874 779	0. 874 779	0. 7775
食物	水果	0. 1263	0. 4077	0. 333 333	0. 166 314	0. 77
鸟	公鸡	1	1	0. 998 571	0. 998 571	0. 7625
鸟	鹤	1	1	0. 998 571	0. 998 571	0. 7425
工具	器械	1	1	0. 999 286	0. 999 286	0. 7375
兄弟	和尚	0. 8611	0. 8718	0. 857 504	0. 857 504	0. 705
起重机	器械	0. 3692	0. 4953	0. 691 622	0. 691 622	0. 42
小伙子	兄弟	0. 8	0. 7333	0. 857 504	0. 857 504	0. 415
旅行	轿车	0. 0741	0. 3902	0. 010 000	0	0. 29
和尚	圣贤	0. 6825	0. 7436	0. 484 593	0. 484 593	0. 275
墓地	林地	0. 1221	0. 3907	0. 196 260	0. 196 26	0. 2375
食物	公鸡	0. 1116	0. 3892	0. 136 197	0. 135 935	0. 2225
海岸	丘陵	0. 1	0. 2576	0. 019 252	0. 019 252	0. 2175
森林	墓地	0. 1116	0. 3487	0. 113 096	0. 113 096	0. 21
岸边	林地	0. 0965	0. 2647	0. 019 252	0. 019 252	0. 1575
和尚	奴隶	0. 6611	0. 6051	0. 576 322	0. 576 322	0. 1375
海岸	森林	0. 1116	0. 2718	0. 019 252	0. 019 252	0. 105
小伙子	巫师	0. 6	0. 4181	0. 691 622	0. 691 622	0. 105
琴弦	微笑	0. 0741	0. 1431	0. 010 000	0	0. 0325
玻璃	魔术师	0. 1219	0. 3653	0. 135 935	0. 135 935	0. 0275
中午	绳子	0. 0999	0. 2023	0. 015 612	0. 015 612	0. 02
公鸡	航行	0. 0741	0. 359	0. 010 000	0	0. 02

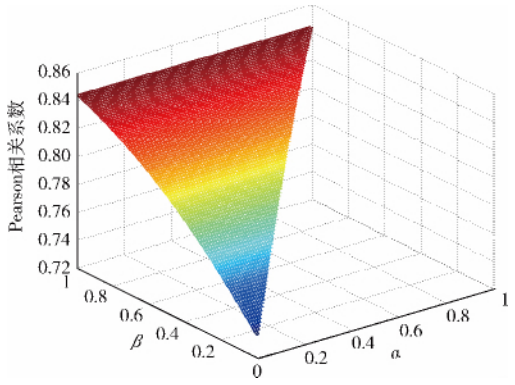

图 3　α、β 参数与 Pearson 相关系数构成的三维图

4　实验与分析

4. 1　实验比较

MC30 测试集是目前国际普遍采用的词语相似度的测试平台，有着规范的人工判定标准。其定义具体请参见文献［14］。在实验中，我们采用该测试集进行测试，我们首先将 MC30 中的英语词对按照意义最接近的原则翻译成对应的中文词对，然后分别采用刘群等[3]、李峰等[13]、本文以及在不做任何预处理的情况下本文方法通过《知网》计算出词对的相似度（见表 2），最后分别计算出这些方法的相似度计算结果与 MC 人工判定值之间的 Pearson 相关系数[15]（见表 3）。为增加对比的全面性，在表 3 中，我们还列出了若干基于英文语义词典 WordNet 与基于中文《同义词词林》的算法结果。

4. 2　实验结果分析

从表 2 中可以看出，本文方法计算出的词语相似度值的覆盖面是最广的，而且与 MC30 人工判定值最为接近。本文方法克服了非同义词语相似度为 1 的情况，而在刘群等[3]和李峰等[13]计算方法中会出现许多非同义词的词语对相似度为 1 的情况，这是不合理的。另外本文方法是直接基于抽象概念的义项树计算词语相似度，而不需要进行复杂的意义相似性计算，因此本文方法在计算精度与计算效率两个方面都要优于对比的计算方法。

表 3 不同方法与 MC30 人工值的 Pearson 相关系数

方法名	相似度方法	使用的语义词典	Pearson 相关系数
Wu 等[10]	基于深度与路径	英文 WordNet	0.7464
Hao 等[11]	基于深度与路径	英文 WordNet	0.8161
CP/CV[16]	基于信息内容	英文 WordNet	0.8138
Mohamed 等[17]	杂合方法	英文 WordNet	0.8460
Hao 等[11]	基于深度与路径	中文《同义词词林》	0.8252
刘群等[3]	基于意义的相似性	中文《知网》	0.6991
李峰等[13]	基于意义的相似性	中文《知网》	0.793
本文（不做预处理）	基于深度与路径	中文《知网》	0.825
本文（做预处理）	基于深度与路径	中文《知网》	0.8443

从表 3 中可以看出，文本算法在不做任何预处理时计算得到的 Pearson 相关系数为 0.825，和其它词语相似度的计算方法相比，本文算法的计算效果是相对较好的，这验证了本文算法自身的优越性。另外本文方法在进行预处理时计算得到的 Pearson 相关系数为 0.8443，已达到了基于英文语义词典 WordNet 与基于中文词典《同义词词林》的优秀算法的计算水平，这一方面表明本文提出的方法是成功的，另一方面也说明《知网》结构较为合理，只要有优秀的算法相配合，是可以在自然语言处理中发挥更大作用的。

但是本文提出来的方法也存在一些不足的地方，例如："水果"和"食物"的相似度偏小，原因是往上搜索最近公共父结点需要走 6 步，造成义项间的最短路径的长度过大；"庇护所"和"精神病院"的相似度也偏小，原因是本文方法在计算路径时，考虑了符号义原和其它关系义原对路径的补偿，造成义项间的最短路径长度过大；而"和尚"和"奴隶"、"小伙子"和"兄弟"以及"巫师"和"小伙子"的相似度却偏大，原因是它们的最近公共父结点都为第一义原，造成义项间的最短路径长度过小。

另外，基于知网的词语相似度的计算方法和基于《同义词词林》的词语相似度的计算方法，在自然语言处理应用中是可以相互补充的。由于知网和词林各自的应用目标不同，导致它们收录的词条差别较大，具体请参见文献[13]。尽管知网和词林都在不断地优化扩充中，但是由于其结构和性质的差别，知网和词林在很多方面还是存在很大差异，因此将知网和词林结合利用，可以有效促进国内词语相似度的研究。

5 结束语

本文提出了基于抽象概念的知网词语相似度计算方法，该算法有以下几个优点：①基于带有抽象概念的义项树直接计算义项相似度，而不需要计算意义相似度；②合理地考虑了语义表达式中的各类义原对词语相似度的影响；③充分考虑了各方面的影响，例如添加一个根节点的影响；④添加初始函数避免不完全定义带来的结果单一化。另外在本文方法的研究过程中，我们发现在基于树的词语相似方法中，一些词语对的相似度会相互制约，当一个词对相似度精确度提高时，另一个词对的就会减少，以至于目前还没有理想的方法能够计算出所有词对都与人工判定值完全接近，而且这种情况的出现还和词典结构中存在词语的分类不合理有关，因此，词语相似度的改进需要同时结合改善词典的分类结构。

参考文献：

[1] GE Bin, LI Fangfang, GUO Silu, et al. Word semantic similarity algorithm research based on HowNet [J]. Application Research of Computers, 2010, 27 (9): 3329-3333 (in Chinese). [葛斌，李芳芳，郭丝路，等. 基于知网的词汇语义相似度计算方法研究 [J]. 计算机应用研究，2010，27 (9)：3329-3333.]

[2] MEI Jiaju, ZHU Yiming, GAO Yunqi, et al. Synonym CiLin [M]. Shanghai: Shanghai Lexicon Publisher, 1983 (in Chinese). [梅家驹，竺一鸣，高蕴琦，等. 同义词词林 [M]. 上海：上海辞书出版社出版，1983.]

[3] LIU Qun, LI Sujian. Lexical semantic similarity calculation based on HowNet [C] //The 3rd Session of Chinese Vocabulary Semantics Seminar, 2002: 59-76 (in Chinese). [刘群，李素建. 基于《知网》的词汇语义相似度计算 [C] //第三届汉语词汇语义学研讨会，2002：59-76.]

[4] DONG Zhendong, DONG Qiang. HowNet [DB/OL]. [2007-01-01/2009-12-20]. http://www.keenage.com/zhiwang/c_zhiwang.html(in Chinese). [董振东，董强. HowNet [DB/OL]. [2007-01-01/2009-12-20]. http://www.keenage.com/zhiwang/c_zhiwang.html.]

[5] ZHANG Ruixia, YANG Guozeng, WU Huixin. Semantic similarity calculation of Chinese unlisted words based on HowNet [J]. Journal of Chinese Information Processing, 2012, 26 (1): 16-21 (in Chinese). [张瑞霞，杨国增，吴慧欣. 基于《知网》的汉语未登录词语义相似度计算 [J]. 中文信息学报，2012，26 (1)：16-21.]

（下转第 713 页）

Chinese). [李春华，尤志翔，闫吉辰，等. 调整量最小的多图像校正算法 [J]. 信号处理，2013，29 (11)：1495-1503.]

[9] Yang Jiachen，Ding Zhiyong，Guo Fei，et al. Multiview image rectification algorithm for parallel camera arrays [J]. Journal of Electronic Imaging，2014，23 (3)：6-8.

[10] Vincent Nozick. Camera array image rectification and calibration for stereoscopic and au-tostereoscopic displays [J]. Annal of Telecommunications，2013，68 (11)：581-596.

[11] Choi M，Kim J，Cho W K，et al. Low complexity image rectification for multi-view video coding [C] //IEEE International Symposium on Circuits and Systems，2012：381-384.

[12] Cheng Hao，An Ping，Li Hejian，et al. Stereo image rectification algorithm for multi-view 3D display [C] //International Conference on 3D Imaging，2011：1-5.

[13] He W，Guozhong W，Liliang L，et al. Fast automatic elimination of vertical parallax of multiview images [C] //IEEE 10th International Conference on Signal Processing Proceedings，2010：1004-1007.

(上接第 670 页)

[6] ZHU Zhengyu，SUN Junhua. Improving word semantic similarity calculation based on HowNet [J]. Journal of Computer Applications，2013，33 (8)：2276-2279 (in Chinese). [朱征宇，孙俊华. 改进的基于《知网》的词汇语义相似度计算 [J]. 计算机应用，2013，33 (8)：2276-2279.]

[7] Lee TB，Hendler J，Lassila O. The semantic Web [J]. Scientific American，2001，284 (10)：34-43.

[8] LIU Chunchen，LIU Dayou，WANG Shengsheng，et al. Modern and application of improving semantic similarity calculation [J]. Journal of Jilin University (Engineering and Technology Edition)，2009，39 (1)：119-123 (in Chinese). [刘春辰，刘大有，王生生，等. 改进的语义相似度计算模型及应用 [J]. 吉林大学学报 (工学版)，2009，39 (1)：119-123.]

[9] Baader F，Calvanese D，Mcguinness DL，et al. The description logic handbook：Theory，implementation and applications [M]. Cambridge：Cambridge University Press，2007：43-95.

[10] Wu ZB，Palmer M. Verbs semantics and lexical selection [C] //Proc of the 32nd Annual Meeting on Association for Computational Linguistics，1994：133-138.

[11] Hao D，Zuo WL，Peng T，et al. An approach for calculating semantic similarity between words using WordNet [C] //Digital Manufacturing and Automation. IEEE，2011：177-180.

[12] Fellbaum C. Semantic network of English：The mother of all WordNets [M]. Berlin：Springer Netherlands，1998：137-148.

[13] LI Feng，LI Fang. Chinese word semantic similarity calculation based on HowNet of the 2000 version [J]. Journal of Chinese Information Processing，2007，21 (3)：99-105 (in Chinese). [李峰，李芳. 中文词语语义相似度计算—基于《知网》2000 [J]. 中文信息学报，2007，21 (3)：99-105.]

[14] Miller GA，Charles WG. Contextual correlates of semantic similarity [J]. Language and Cognitive Processes，1991，6 (1)：1-28.

[15] LIU Hongzhe. Text semantic similarity algorithm research [D]. Beijing：Beijing Jiaotong University，2012 (in Chinese). [刘宏哲. 文本语义相似度计算方法研究 [D]. 北京：北京交通大学，2012.]

[16] Kim JW，Candan KS. CP/CV：Concept similarity mining without frequency information from domain describing taxonomies [C] //Proc of the 15th ACM International Conference on Information and Knowledge Management，2006：483-492.

[17] Mohamed AHT，Mohamed BA，Hamadou AB. Ontology-based approach for measuring semantic similarity [J]. Journal of Engineering Applications of Artificial Intelligence，2014，36 (C)：238-261.