文章编号:1000-5471(2013)12-0101-06

成对排序本体学习算法①

朱林立[1], 戴国洪[2], 高 炜[3]

1. 江苏理工学院 计算机工程学院, 江苏 常州 213001;
2. 常熟理工学院 机械学院, 江苏 常熟 215500; 3. 云南师范大学 信息学院, 昆明 650500

摘要: 本体作为一种结构化数据模型已广泛应用于知识表示和信息处理, 并成为近几年计算机领域的研究热点. 提出基于成对排序学习方法的本体相似度计算和本体映射算法, 利用 Mahalanobis 距离函数得到计算模型, 通过梯度下降策略得到模型的最优解, 从而将本体图或多本体图中的顶点对映射成实数来表示它们的相似程度. 通过两个实验表明, 新算法对特定的应用领域具有较高的效率.

关 键 词: 本体; 相似度; 本体映射; 成对排序; 距离函数; 梯度下降策略

中图分类号: TP301 **文献标志码**: A

本体这一术语源自哲学, 用于描述事物间的本质关联. 引入计算机领域后, 本体作为一种数据结构表示模型已广泛用于计算机科学的各个领域. 随着本体技术的日渐成熟, 本体又被广泛应用于生物医学[1]、地理科学[2]、社会科学[3]等诸多领域. 本体应用的核心任务计算来自同一本体或不同本体间代表概念的顶点间的相似度. 因此, 本体算法的核心是相似度计算.

随着信息处理数据量的日渐庞大, 本体相似度计算算法已不再是原先启发式的方法, 通过样本学习得到本体相似度计算函数已成为本体研究的热点. 例如: 文献[4-6]提出了根据谱图理论设计本体相似度计算和本体映射算法, 其计算模型最后将问题转化为计算带权本体图拉普拉斯矩阵次小特征值对应的特征向量. 通过该特征向量, 本体图中的每个顶点被映射成实数, 两顶点间的相似程度通过它们对应实数的差值来确定. 此类算法的收敛性理论分析可参考文献[7-8].

另外一种本体学习算法是通过排序学习算法得到的, 例如文献[9-11]. 通过学习最优排序函数, 从而将本体图映射成实直线, 将图中每个顶点映射成实数, 再通过顶点对应实数间的差值来判定本体图中两顶点对应概念间的相似度.

但是, 通过类似文献[9-11]顶点排序算法得到的本体相似度是一种间接相似度而非绝对相似度预测. 我们知道本体算法所真正要得到的相似度函数 $f: V\times V\to \mathbb{R}^+\cup\{0\}$ 是将每一对顶点映射成一个实数, 也就是它们的相似度. 在信息检索中, 通过领域专家人为确定参数 M, 将满足 $\mathrm{Sim}(A, B)>M$ 的所有顶点 B 对应的概念集合返回给用户作为顶点 A 对应概念的查询扩展. 对于本体映射而言, 设图 $G_1, G_2, \cdots, G_m$ 分别对应本体 $O_1, O_2, \cdots, O_m$. 对于每个 $A\in V(G_i)$, 其中 $1\leqslant i\leqslant m$, 在 $G-G_i$ 中找出所有满足 $\mathrm{Sim}(A, B)>M$ 的所有顶点 B 对应概念的集合返回给用户作为顶点 A 对应概念的查询扩展.

本文将从另外一个角度来思考排序算法如何应用于本体相似度计算和本体映射. 我们的算法区别于其它基于排序思想的本体算法的本质之处在于: 本文的目标不是将每个顶点映射到实数轴, 而是将每一对顶

① 收稿日期: 2013-06-07
基金项目: 国家自然科学基金项目(60903131); 教育部科学技术研究重点项目(210210); 江苏省高校自然科学研究项目(10KJD52002).
作者简介: 朱林立(1975-), 男, 湖北巴东人, 硕士, 高级工程师, 主要从事人工智能及云计算等的研究工作.

点映射到实数轴，最后得到 $\| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2$. 同时，我们引入降维的思想来处理顶点对应向量维数过高带来的计算量庞大的问题. 本文的组织结构如下：在1节中，给出了成对排序本体学习算法的设计思路及模型求解策略；2节中，给出了总体算法的框架；3节中，通过2个具体实验来说明新本体算法对特定的应用领域是有效的.

1 本体图上成对排序学习

首先考虑单本体图 $G=(V, E)$ 的情况，其中 V 是本体图 G 的顶点集，E 是本体图 G 的边集. 将本体图中每个顶点的信息用一个向量表示. 选取 V 的一个子集 V'，定义在 V' 上的样本集为

$$S=\{(v_i, v_j, y_{ij}): v_i \in V', v_j \in V', v_i \neq v_j, y_{ij} \in \mathbb{R}^+ \cup \{0\}\}$$

即 y_{ij} 为 V' 中顶点 v_i 和 v_j 的相似度标记，y_{ij} 越小表示顶点 v_i 与 v_j 之间的相似度越大. 为了方便起见，我们用粗体字符 $\boldsymbol{v}_i$ 来表示顶点 v_i 所对应的向量. 从而样本集中标记 y_{ij} 的值可以由距离函数来确定，例如：在欧氏距离下有 $y_{ij} = \| \boldsymbol{v}_i - \boldsymbol{v}_j \|^2$.

在排序算法中，关心的不是 y_{ij} 值的精确度，而是顶点对标记之间的相对大小关系，这是排序和回归的本质区别所在. 例如：顶点 v_i 与顶点 v_k 对应概念之间的相似度要高于顶点 v_i 与顶点 v_j 之间的相似度，则它们的标记应该满足 $y_{ij} > y_{ik}$. 为了衡量排序函数的优劣性，定义如下布尔函数：

$$I_{\{y_{ij}>y_{ik}\}} = \begin{cases} 1 & \text{若 } v_k \text{ 比 } v_j \text{ 更相似于 } v_i \\ 0 & \text{否则} \end{cases} \tag{1}$$

设 $V'=\{v_1, v_2, \cdots, v_n\}$. 在欧氏距离函数下的排序学习经验模型可表示为：

$$R = \sum_{i=1}^{n-2} \sum_{j,k \in \{i+1, \cdots, n\}} I_{\{\| v_i - v_j \|^2 > \| v_i - v_k \|^2\}} \tag{2}$$

为解决高维向量信息表示带来的计算复杂问题，我们需要引入一些降维的思想. 具体的做法是采用Mahalanobis距离函数来取代欧氏距离. 即，

$$y_{ij} = \| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2$$

其中矩阵 $\boldsymbol{A} \in \mathbb{R}^{K \times D}$，$D$ 是特征空间的维数(即向量 v_i 的维数)，$K \leqslant D$ 是新特征空间的维数. Mahalanobis距离是从 $\mathbb{R}^D$ 到 $\mathbb{R}^K$ 的线性投影，其本质是用低维向量来对每个顶点的信息进行重新表示.

由此，(2)可转化为如下经验模型：

$$R(\boldsymbol{A}) = \sum_{i=1}^{n-2} \sum_{j,k \in \{i+1, \cdots, n\}} I_{\{\| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2 > \| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_k \|^2\}} \tag{3}$$

排序函数通过最小化经验模型(3)得到. 但由于 $R(\boldsymbol{A})$ 中包含不可导的布尔函数，直接最小化(3)非常困难. 为此，我们采用指数函数来取代布尔函数(1)，得到

$$R(\boldsymbol{A}) = \sum_{i=1}^{n-2} \sum_{j,k \in \{i+1, \cdots, n\}} \exp\{\| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2 - \| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_k \|^2\} \tag{4}$$

从而 $R(\boldsymbol{A})$ 可导，并且可用标准梯度下降策略最小化(即求解 $\frac{\partial R}{\partial \boldsymbol{A}} = 0$). 计算 $R(\boldsymbol{A})$ 时，可令 $\boldsymbol{v}_{ij} = \boldsymbol{v}_i - \boldsymbol{v}_j$，$\boldsymbol{v}_{ik} = \boldsymbol{v}_i - \boldsymbol{v}_k$，并对(4)如下变换：

$$\begin{aligned} R(\boldsymbol{A}) = & \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \exp\{\| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2 - \| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_k \|^2\} = \\ & \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \sum_{k=j+1}^{n} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2 - \| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} = \\ & \sum_{i=1}^{n-2} \sum_{j=i+1}^{n-1} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \sum_{k=j+1}^{n} \exp\{- \| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} \end{aligned}$$

从而可得：

$$\frac{\partial R}{\partial \boldsymbol{A}} = -2A \sum_{i=1}^{n-2} \Big(\sum_{j=i+1}^{n-1} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \times \sum_{k=j+1}^{n} v_{ik} v_{ik}^T \exp\{- \| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} +$$

$$\sum_{j=i+1}^{n-1} \boldsymbol{v}_{ij}\, \boldsymbol{v}_{ij}^{T} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \times \sum_{k=j+1}^{n} \exp\{-\| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\})$$

由此易知 $R(\boldsymbol{A})$ 的计算复杂度为 $O(n^2K+nDK)$，$\dfrac{\partial R}{\partial \boldsymbol{A}}$ 的计算复杂度为 $O(n^2KD)$.

对于本体映射，设 $V'=\bigcup_{i=1}^{m} V_i'$，其中 $V_i'=\{v_i^1, v_i^2, \cdots, v_i^{n_i}\}$ 是 G_i 的顶点子集，$n=\sum_{j=1}^{k} n_j$. 从而训练集可表示为

$$S=\{(v_i, v_j, y_{ij}): i,j \in \{1, \cdots, m\}, i \neq j, v_i \in V_i' v_j \in V_j', y_{ij} \in \mathbb{R}^+ \cup \{0\}\}$$

2 ～ 4 式可分别表示为：

$$R=\sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j, v_k \in V'-\bigcup_{t=1}^{i} V_t'} I_{\{\| v_i-v_j \|^2 > \| v_i-v_k \|^2\}} \tag{2'}$$

$$R(\boldsymbol{A})=\sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j, v_k \in V'-\bigcup_{t=1}^{i} V_t'} I_{\{\| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_j \|^2 > \| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_k \|^2\}} \tag{3'}$$

$$\begin{aligned} R(\boldsymbol{A}) &= \sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j, v_k \in V'-\bigcup_{t=1}^{i} V_t'} \exp\{\| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_j \|^2 - \| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_k \|^2\} = \\ &\sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j \in V'-\bigcup_{t=1}^{i} V_t' v_k \in V'-\bigcup_{t=1}^{i} V_t', v_k \neq v_j} \exp\{\| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_j \|^2 - \| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_k \|^2\} = \\ &\sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j \in V'-\bigcup_{t=1}^{i} V_t' v_k \in V'-\bigcup_{t=1}^{i} V_t', v_k \neq v_j} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2 - \| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} = \\ &\sum_{i=1}^{m-1} \sum_{v_i \in V_i} \sum_{v_j \in V'-\bigcup_{t=1}^{i} V_t'} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \sum_{v_k \in V'-\bigcup_{t=1}^{i} V_t', v_k \neq v_j} \{-\| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} \end{aligned} \tag{4'}$$

$$\begin{aligned} \frac{\partial R}{\boldsymbol{A}} = -2A \sum_{i=1}^{m-1} \sum_{v_i \in V_i} (& \sum_{v_j \in V'-\bigcup_{t=1}^{i} V_t'} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \times \sum_{v_k \in V'-\bigcup_{t=1}^{i} V_t', v_k \neq v_j} \boldsymbol{v}_{ik}\, \boldsymbol{v}_{ik}^{T} \{-\| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\} + \\ & \sum_{v_j \in V'-\bigcup_{t=1}^{i} V_t'} \boldsymbol{v}_{ij}\, \boldsymbol{v}_{ij}^{T} \exp\{\| \boldsymbol{A}\boldsymbol{v}_{ij} \|^2\} \times \sum_{v_k \in V'-\bigcup_{t=1}^{i} V_t', v_k \neq v_j} \{-\| \boldsymbol{A}\boldsymbol{v}_{ik} \|^2\}) \end{aligned}$$

2　基于成对排序学习的本体算法

由以上分析可知，通过最小化 $R(\boldsymbol{A})$ 得到最优投影矩阵 $\boldsymbol{A}$，从而将本体图或多本体图中的每一对顶点映射成实数，来表示它们对应概念的相似程度. 具体算法描述如下：

算法 1：基于成对排序学习的本体相似度计算算法

步骤 1：预处理. 将本体图中每个顶点的信息用一个向量表示.

步骤 2：选取 V' 生成 S.

步骤 3：利用标准梯度下降策略，通过最小化经验模型(4) 得到投影矩阵 $\boldsymbol{A}$.

步骤 4：给出 G 中每对顶点所对应的实数，即 $\| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_j \|^2$.

算法 2：基于成对排序学习的本体映射算法

步骤 1：预处理. 设图 $G_1, G_2, \cdots, G_m$ 分别对应本体 $O_1, O_2, \cdots, O_m$，令 $G=G_1+G_2+\cdots+G_m$，将 G 中每个顶点的信息用一个向量来表示.

步骤 2：从每个分支的顶点集中选取一部分顶点，构成 V'，并得到 S.

步骤 3：利用标准梯度下降策略，通过最小化经验模型(4') 得到投影矩阵 $\boldsymbol{A}$.

步骤 4：给出 G 中来自不同分支的顶点对所对应的实数，并选择本体映射策略，生成本体映射.

需要注意的是，算法最后通过求解 A，得到每对顶点的 $\| \boldsymbol{A}\boldsymbol{v}_i-\boldsymbol{A}\boldsymbol{v}_j \|^2 \in \mathbb{R}^+ \cup \{0\}$，其值越小表示 v_i 与 v_j 之间的相似度越大. 此类排序算法的核心思想是降维，可用于高维数据信息表示的情形. 此外，由于排

序算法关注的是不同顶点对之间的数值谁大谁小，因此最后算法得到的是一个关于顶点对相似度大小的排序列表. 在查询扩展等具体应用中，这种列表以及具体 $\|\boldsymbol{Av}_i-\boldsymbol{Av}_j\|^2$ 的值为我们选择查询扩展策略提供了依据. 可以设定参数 $M>0$，将所有满足 $\|\boldsymbol{Av}_i-\boldsymbol{Av}_j\|^2<M$ 的顶点 v_j 返回给用户，作为对顶点 v_i 对应概念的查询扩展；也可以设定 $N\in\mathbb{N}$，针对顶点 v_i，在算法得到的排序列表中查询 $\|\boldsymbol{Av}_i-\boldsymbol{Av}_j\|^2$ 值最小的前 N 个 v_j 返回给用户，作为对顶点 v_i 对应概念的查询扩展.

3 实 验

3.1 本体相似度实验

第一个实验是验证算法1对生物本体相似度计算的效率，数据集采用生物GO本体 O_1. 由于本文的算法最终是将本体图中每一对顶点映射成相似度并排序，因此可采用 $P@N$[12]平均准确率来评价实验结果. 即根据算法得到的 $\|\boldsymbol{Av}_i-\boldsymbol{Av}_j\|^2$ 值，列出与每个顶点相似度最高的 N 个对象，再和领域专家列出的每个顶点相似度最高的 N 个对象进行对比，分别计算每个顶点的命中率，最后得到所有顶点的平均命中率，见图1.

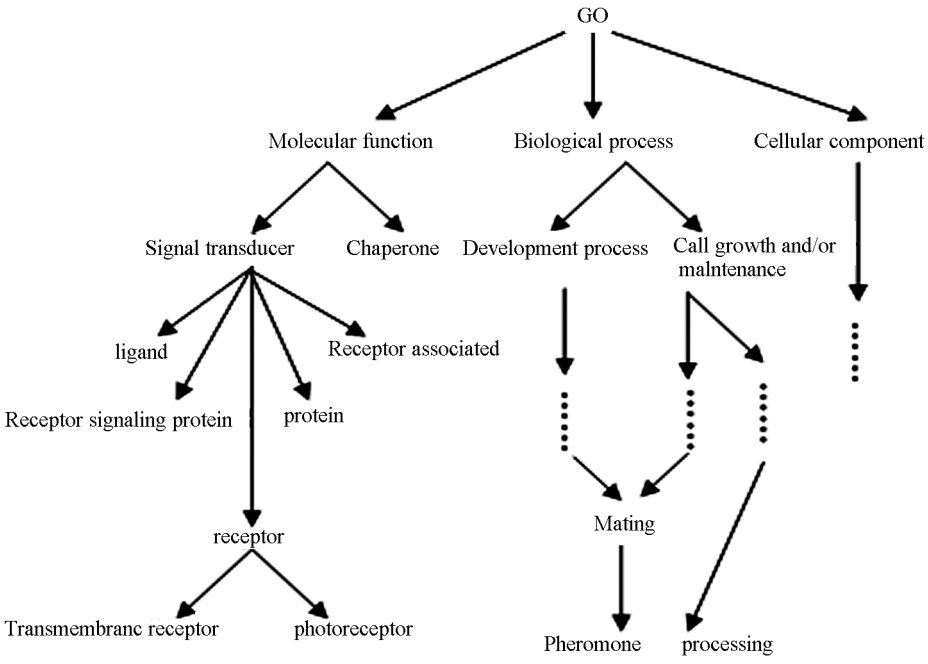

图1 GO本体 O_1

在具体实验过程中，我们首先选取大约1/3的顶点，并通过计算两两顶点对之间的欧式距离来确定不同顶点对之间标记的大小关系. 并运用算法1得到顶点对相似度排序列表，从而得到每个顶点与之相似度最高的 N 个顶点. 同时，我们还分别采用快速排序算法[13]和标准本体排序算法[9]应用于GO本体，将每个顶点映射成实数，通过比较实数间的差值来确定与每个顶点相似度最高的 N 个顶点. 计算这3种算法得到的 $P@N$ 准确率并对实验结果进行比较，部分数据如表1：

表1 实验1部分数据

算法名称	$P@3$ 平均准确率/%	$P@5$ 平均准确率/%	$P@10$ 平均准确率/%
本文算法1	54.31	62.42	73.86
快速排序算法	47.73	55.52	69.93
标准排序算法	52.37	60.62	72.96

由表1可知，本文所提出的算法1对于GO本体的效率高于快速排序算法和标准排序算法.

3.2 本体映射实验

本文第二个实验是采用文献[9]中构造的两个“计算机软件”本体 O_2 和 O_3 来验证算法2的效率，见图2,3.

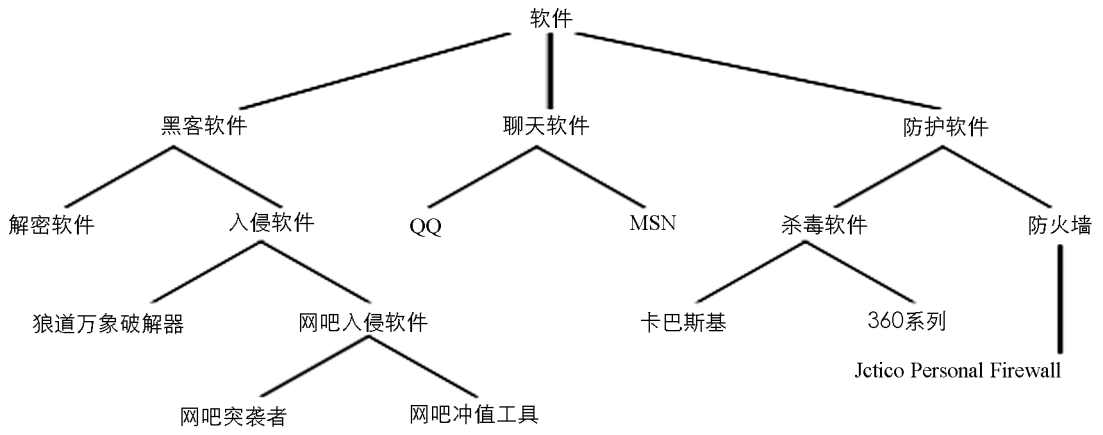

图 2　“计算机软件”本体 O_2

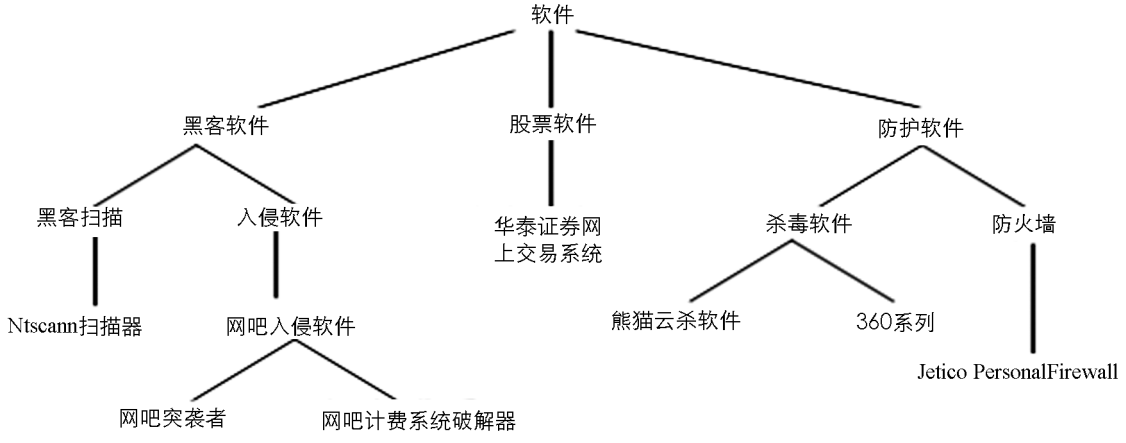

图 3　“计算机软件”本体 O_3

选取大约一半数量的顶点，通过欧式距离函数来确定不同顶点对之间标记的大小关系. 并运用算法 2 得到不同分支顶点对($\boldsymbol{v}_i$，$\boldsymbol{v}_j$)对应的 $\| \boldsymbol{A}\boldsymbol{v}_i - \boldsymbol{A}\boldsymbol{v}_j \|^2$ 值，由此得到与每个顶点与之相似度最高(不同分支中)的 N 个顶点. 同样地，采用快速排序算法[13]和标准本体排序算法[9]应用于“计算机软件”本体，将每个顶点映射成实数，通过比较实数间的差值来计算与每个顶点相似度最高(不同分支中)的 N 个顶点. 计算这 3 种算法得到的 $P@N$ 准确率并对实验结果进行比较，部分数据如表 2.

表 2　实验 2 部分数据

算法名称	$P@1$ 平均准确率/%	$P@3$ 平均准确率/%	$P@5$ 平均准确率/%
本文算法 2	57.58	71.72	81.82
快读排序算法	39.39	51.52	58.79
标准排序算法	54.55	66.67	78.82

由表 2 可知，本文所提出的算法 2 对于建立“计算机软件”本体 O_2 和 O_3 间本体映射的效率高于快速排序算法和标准排序算法.

4　结束语

本文提出了基于成对排序学习的本体相似度计算和本体映射算法. 由该算法得到的实数直接衡量了顶点间的相似度. 由于算法的本质还属于排序算法，注重顶点对与顶点对之间相似度的大小关系，由此通过排序序列可以确定与某个概念 A 相似度最大的前 N 个概念. 因此，该算法继承了回归和排序各自的优点，在类似查询扩展的具体应用中，给了用户多种策略选择余地.

参考文献:

[1] MORK P, BERNSTEIN P. Adapting a Generic Match Algorithm to Align Ontologies of Human Anatomy [C] //20th International Conf. on Data Engineering. Los Alamitos, CA, USA: IEEE Comput Soc, 2004: 787－790.

[2] FONSECA F, EGENHOFER M, DAVIS C, et al. Semantic Granularity in Ontology-Driven Geographic Information Systems [J]. AMAI Annals of Mathematics and Artificial Intelligence-Special Issue on Spatial and Temporal Granularity, 2002, 36(1－2): 121－151.

[3] BOUZEGHOUB A, ELBYED A. Ontology Mapping for Web-Based Educational Systems Interoperability [J]. Interoperability in Business Information Systems, 2006, 1(1): 73－84.

[4] 高 炜, 梁 立, 张云港. 基于图学习的本体概念相似度计算 [J]. 西南师范大学学报: 自然科学版, 2011, 36(4): 64－67.

[5] 高 炜, 梁 立. 基于超图正则化模型的本体概念相似度计算 [J]. 微电子学与计算机, 2011, 28(5): 15－17.

[6] 高 炜, 朱林立, 梁 立. 基于图正则化模型的本体映射算法 [J]. 西南大学学报: 自然科学版, 2012, 34(3): 118－121.

[7] 高 炜, 张云港, 梁 立. C 相似度函数下正则谱聚类的收敛阶 [J]. 兰州大学学报: 自然科学版, 2011, 47(2): 109－111.

[8] 高 炜, 周定轩. 与一般相似度函数相关的谱聚类的收敛性 [J]. 中国科学: 数学, 2012, 42(10): 985－994.

[9] 高 炜, 兰美辉. 基于排序学习方法的本体映射算法 [J]. 微电子学与计算机, 2011, 28(9): 59－61.

[10] 王雅玡, 高 炜, 张云港, 等. 图推进排序学习算法 [C]. Shanghai: Proceedings of 2010 International Conference on Circuit and Signal Processing, 2010: 368－371.

[11] GAO W, LIANG L. Ontology Similarity Measure by Optimizing NDCG Measure and Application in Physics Education [J]. Lecture Notes in Electrical Engineering, 2012, 142(2): 415－421.

[12] CRASWELL N, HAWKING D. Overview of the TREC 2003 Web Track [C]. Proceedings of the Twelfth Text Retrieval Conference. Gaithersburg, Maryland: NIST Special Publication, 2003: 78－92.

[13] HUANG X, XU T, GAO W, et al. Ontology Similarity Measure and Ontology Mapping Via Fast Ranking Method [J]. International Journal of Applied Physics and Mathematics, 2011, 1(1): 54－59.

On Ontology Learning Algorithm of Pairwise Ranking

ZHU Lin-li[1], DAI Guo-hong[2], GAO Wei[3]

1. *School of Computer Engineering, Jiangsu University of Technology, Changzhou Jiangsu 213001, China;*
2. *School of Mechanical Engineering, Changshu Institute of Technology, Changshu Jiangsu 215500, China;*
3. *School of Information Science and Technology, Yunnan Normal University, Kunming 650500, China*

Abstract: As a structured data model, ontology has been widely used in knowledge representation and information processing, and it becomes a hot topic of computer science in recent years. In this paper, new ontology similarity calculation and ontology mapping algorithm based on the pairwise ranking learning method have been discussed before a model is given by means of Mahalanobis distance function. The optimal solution has been obtained in views of gradient descent strategy. Thus, each pair of vertices in ontology graph or mutil-ontology graph is mapped into real number, which represents their similarity. Two experiments show that the new algorithm has a higher efficiency for specific applications.

Key words: ontology; similarity; ontology mapping; pairwise ranking; distance function; gradient descent strategy

责任编辑 周仁惠