基于聚类的本体块匹配方法

张 鹏, 杨 峰, 吕 帅, 刘 磊
(吉林大学 计算机科学与技术学院, 长春 130012)

摘要: 提出一种新的处理 $n:m$ 映射的方法, 该方法将 $n:m$ 映射问题转化为聚类问题, 利用 Hownet 中的资源使本体中的实体基于语义关系聚合, 并重新给出了查全率和查准率的计算公式. 使用 Hownet 及其相关工具对 OAEI 组织给出的一组本体对进行实验, 实验结果表明, 该方法对块匹配问题效果较好.

关键词: 本体映射; 块匹配; 聚类
中图分类号: TP311 **文献标志码**: A **文章编号**: 1671-5489(2011)03-0493-05

Clustering-Based Ontology Block Matching Approach

ZHANG Peng, YANG Feng, LÜ Shuai, LIU Lei
(*College of Computer Science and Technology, Jilin University, Changchun* 130012, *China*)

Abstract: A new approach to deal with $n:m$ mappings was proposed, which translates the ontology matching problem into a clustering issue. It uses the information in Hownet to make the entities in ontologies cluster based on semantic relationship. The formulae of precision and recall were redefined. We used Hownet and its related tools to carry out experiments on a pair of ontologies provided by OAEI, and the experimental results demonstrate that our approach is feasible on block matching problem.

Key words: ontology mapping; block matching; cluster

随着语义网的发展, 本体已成为描述资源信息的一种重要方法. 网络中异构本体大量存在, 为了保证信息的一致性, 需要在相关本体间建立一种关联关系, 即本体映射.

本体映射是指把两个本体作为输入, 计算本体元素间映射关系的过程[1]. 根据映射元素的数目, 将本体映射分为 $1:1$ 映射、$1:n$ 映射, $n:$ null 映射、null$:m$ 映射及 $n:m$ 映射. $n:$ null 映射和 null$:m$映射是两种特殊情况, 是指一个本体中没有另外一个本体中元素的映像, 由于其自身的特殊性, 在本体映射研究中一般不考虑.

目前, 关于本体映射的研究已取得了许多成果, 如: 唐杰等[2]基于贝叶斯决策理论提出了风险最小化的本体映射模型, 设计并实现了能处理 $1:1$ 映射和 $1:n$ 映射问题的系统 RiMOM; 文献[3]基于本体中实体描述信息的语义和实体间层次结构影响本体映射结果的思想, 设计并实现了能处理大型本体映射任务的 Falcon-AO 系统; Doan 等[4]基于将机器学习方法引入到本体映射中的思想, 设计并实现了用实例信息计算概念间相似度的 GLUE 系统; Euzenat 等[5]基于将匹配策略动态组合对实体进行比对的思想, 开发了使本体匹配问题转化为最优化问题的 OLA 系统.

收稿日期: 2010-03-30.

作者简介: 张 鹏(1986—), 男, 满族, 硕士研究生, 从事语义网与本体映射的研究, E-mail: zhangpeng0521@jlu.edu.cn. 通讯作者: 刘 磊(1960—), 男, 汉族, 教授, 博士生导师, 从事软件形式化、语义网与本体工程的研究, E-mail: Liulei@jlu.edu.cn.

基金项目: 国家自然科学基金(批准号: 60873044)、中央高校基本科研业务费专项基金(批准号: 200903174; 200903183)和浙江师范大学计算机软件与理论学科开放基金(批准号: ZSDZZZZXK11).

上述研究均与 1∶1 映射或 1∶n 映射相关，而 1∶1 映射和 1∶n 映射只是 n∶m 映射的特例. 本文称n∶m映射为块映射或块匹配，块是一个实体集合，块匹配为源于不同本体的两个块建立映射关系的过程. 实体包括概念、属性、关系及实例. 由于块匹配问题的复杂性，因此当前关于块匹配的研究较少. 目前只有 PBM 方法[6]和 BMO 方法[7]处理过 n∶m 映射问题. PBM 一般用于处理大型本体映射任务中计算复杂度较高的问题，利用结构信息和字符串匹配技术为两个大型本体中的概念建立关联，根据关联对两个本体的层次结构分别进行分块操作，使用锚(anchor)技术和实体文本计算块间的相似度. BMO 为两个本体中的每个实体建立实体文本，利用 VSM(vector space model)[8]为每个实体建立实体向量，用向量夹角余弦计算实体的关联程度，并用层次对分算法对两个本体同时进行逐层分割，形成以节点为实体块的树状结构，使用动态规划算法在树状实体块图中抽取最优映射.

本文将本体块映射问题转化为聚类问题，将映射过程简化为实体在语义相似关系下的聚合，在完成实体聚合的同时也完成了块映射. 与 PBM 相比，该方法使分块和映射同时完成，降低了块映射问题的复杂程度. 与 BMO 相比，避免了 BMO 在本体分割时遇到的 NP 问题. 此外，在计算语义相似度时，该方法使用了 Hownet(知网)中的资源，由于 Hownet 描述两种语言(英语和汉语)的概念，所以可以解决汉语本体的映射问题.

1　基本定义

定义 1　令 A 为一个本体，A 中所有实体构成的集合 S 称为源集.

定义 2　令 S 为一个源集，d_i 为其中的实体，可以是概念、属性或实例，S 的子集 $D=\{d_i \mid d_i \in S\}$ 称为块.

定义 3　令块 D_1 和 D_2 分别是源集 S_1 和 S_2 的子集，且 $S_1 \neq S_2$，集合 $O=D_1 \cup D_2$ 称为类簇.

当 D_1 或 D_2 为空时，O 仍然是一个类簇，块是一种特殊的类簇.

定义 4　如果源于不同本体的两个块 D_1 和 D_2 被分到一个类簇 O 中，则称块 D_1 与 D_2 匹配.

定义 5　实体和类簇统称为匹配元.

2　语义相似度计算

为了发现块与块之间的映射关系，需要一种有效的方法衡量匹配元之间的语义距离. 语义距离与语义相似度间有密切联系. 两个词语的语义距离越大，语义相似度越低；两个词语的语义距离越小，语义相似度越高. 一般情况下，语义距离是一个$[0,\infty)$间的实数，语义相似度是一个$[0,1]$间的实数，二者可以简单地建立一种对应关系. 这种对应关系应该满足如下条件约束：1）当两个词语的语义距离为 0 时，其语义相似度为 1；2）当两个词语的语义距离为∞时，其语义相似度为 0；3）语义相似度随着语义距离的增加而减小.

目前，已有许多学者对语义相似度的计算方法经行了研究，如 tf-idf[ ]方法[9]、PHSS 模型[10]、HMM 模型[11]及基于 WordNet 的词汇语义相似度算法[12]等. 但这些方法只能在一定程度上对英语词汇间的语义相似度进行适当衡量，而对汉语词汇的效果则不明显. 本文使用的计算语义相似度方法可以有效解决该问题，充分利用了 Hownet 词库中的资源. Hownet 将汉语和英语的词语所代表的概念作为描述对象，以揭示概念与概念间及概念所具有的属性间的关系为基本内容的常识知识库. 对于知网中的每个词语，都有一个 DEF 与之对应（DEF 是 Hownet 概念的定义，为一个语义表达式，是 Hownet 的核心）. 利用 Hownet 计算词汇语义相似度的方法由以下四方面计算词汇的语义相似度：1）两个 DEF 的包含关系；2）两个概念主类间的相似度；3）DEF 各个节点的相似度；4）两个 DEF 主类义原框架的相似度. 该方法可以有效衡量英语和汉语两种语言中词汇间的语义距离，计算词汇间的语义相似度. 对于两个词汇 a 和 b，利用上述方法求得 a 和 b 的语义相似度用 Sim $L(a,b)$ 表示. 本文的匹配元计算方法就是以这种方法为基础. 根据前文给出的匹配元定义可知，匹配元间的语义相似度分为 3 种情况：1）实体与实体的语义相似度；2）实体与类簇的语义相似度；3)类簇与类簇的语义相似度.

定义 6　令 a 和 b 分别是本体中实体 d_1 和 d_2 的名称，则实体间相似度定义为

$$\mathrm{Sim}(d_1,d_2) = \mathrm{Sim}\,L(a,b).$$

定义7 令 d_{ij} 是类簇 O_i 中第 j 个实体且 $|O_i| = n$，则实体 d 与类簇 O_i 间的相似度定义为 $\mathrm{Sim}(d,O_i) = \sum_{j=1}^{n}\mathrm{Sim}(d,d_{ij})$.

定义8 令 d_{1i} 是 O_1 中第 i 个实体且 $|O_1| = n$，d_{2j} 是 O_2 中第 j 个实体且 $|O_2| = m$，则类簇 O_1 与类簇 O_2 间的相似度定义为 $\mathrm{Sim}(O_1,O_2) = \sum_{i=1}^{n}\sum_{j=1}^{m}\mathrm{Sim}(d_{1i},d_{2j})$.

3 语义聚类分块

为了将实体根据语义距离聚合成 k(用户期望)个类簇，本文基于层次聚类算法的思想设计了聚类分块算法. 算法对参与映射两个本体中的实体同时操作，先令所有实体构成一个集合，置每个实体为一个类簇，然后不断寻找语义距离最近的两个类簇，将它们合并，类簇数目减1，直至剩下类簇的数目等于 k 为止.

为方便算法的形式化表示，在给出算法前，先对算法中使用的概念给出符号化描述：在算法中，令 S 表示当前的类簇集，n 表示当前类簇的数目，O_i 表示第 i 个类簇，d_{ij} 表示第 i 个类簇中第 j 个实体，则有 $S = \{O_1,O_2,\cdots\}$，$O_i = \{d_{i1},d_{i2},\cdots\}$.

Algorithm Semantic_Cluster

begin

initialization;//置两个本体中的每个实体为一个类簇，对类簇进行编号

loop

$(i,j) = \arg\max_{(i,j)}\{\mathrm{Sim}(O_i,O_j)\,|\,O_i \in S,\ O_j \in S\}$; //找到语义相似度最大的两个类簇

$O_i = O_i \cup O_j$; //将语义相似度最大的两个类簇合并为一个类簇

$n = n - 1$; //类簇数目减1

until ($k = = n$) //若类簇数目等于 k，则算法中止

end

该算法将块匹配问题转化成聚类问题，算法的过程实际上是用 $|A| + |B|$ 个叶节点生成 k 棵二叉树的过程，这里二叉树的节点是类簇，叶节点是实体，根节点是算法最终形成的类簇，$|A|$ 表示本体 A 中的实体数目，$|B|$ 表示本体 B 中的实体数目. 在最终形成的 k 棵二叉树中，任何一颗二叉树叶节点的集合恰好是这棵二叉树的根节点，即同一棵二叉树上的叶节点之间彼此相似.

该方法研究的语义相似显然是一种等价关系，如果把两个本体中实体构成的集合视为全集 H，则各个类簇 O_i 即可视为 H 在语义相似关系下的等价类，即在同一个类簇中的两个实体是语义相似的.

4 方法实现

基于聚类的本体块匹配方法实现过程如图1所示，该方法以两个本体作为输入，经过预处理、语义距离计算及语义聚类分块的处理，最终形成 k 个类簇作为输出. 其中 k 由用户预先指定.

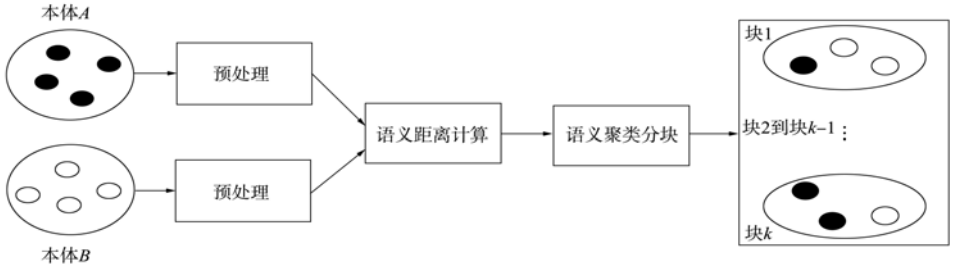

图1 块映射方法的实现过程

Fig. 1 Procedure of block matching approach

1）预处理是为语义相似度计算做准备，从 OWL 或 RDF 文件中抽取本体信息，分为语义信息完善和同义信息转换两步. 语义信息完善是对一些特殊实体的信息进行完善，包括三种操作：对于名称使用缩写的实体，名称还原为全称；对于使用 URL 为名称的实体，使用其描述信息重新命名该实体；对于本体中的空节点，使用其相邻本体的信息命名该节点. 同义信息转换是为了更好地使用 Hownet 中的资源，由于词汇的较多并且复杂，Hownet 不会对所有的概念都进行描述，但一般会包含与之同义的概念. 因此需要将本体中不被 Hownet 接受的概念同义转换为被 Hownet 接受的概念. 该过程需用户参与，使用 Hownet 词库对参与映射的每个实体信息进行检索，如果该信息可以被 Hownet 描述，则将其保留，否则使用 Hownet 词库中的同义信息将其代替.

2）语义距离计算是语义聚类分块的基础，主要工作是计算匹配元间的语义距离，采用语义相似度衡量，利用 Hownet 计算词汇语义相似度的方法计算匹配元间的语义相似度. 该阶段计算 3 种匹配元间的语义相似度：实体与实体的语义相似度；实体与类簇的语义相似度；类簇与类簇的语义相似度.

3）语义聚类分块是整个方法的核心，主要工作是使用聚类分块算法对两个本体中的实体同时实施分块操作. 根据匹配元间的语义相似度，使用聚类分块算法不断缩减类簇的个数，直至类簇的个数等于 k.

最后得到的每个类簇中的实体源于两个本体，其中源于一个本体的所有实体与源于另外一个本体中的所有实体语义相似，它们之间可以建立一种映射关系，这种映射关系即为 $n:m$ 映射. 源于同一本体中的所有本体构成了一个块，一个类簇中包含两个这样的块，这两个块匹配.

5　实验结果与分析

本体中实体要聚合成的类簇数目由用户指定. 当将本体中的实体聚合成 k 个类簇时，本文同时进行两种操作，一种是由专家将两个本体中的实体根据语义关系分成 k 个类簇，一种是根据本文提出的方法将两个本体中的实体聚合成 k 个类簇. 本文的实验以专家分块生成的结果为标准，与算法生成的结果相比，通过计算查全率和查准率评价实验结果. 由于衡量 1∶1 映射和 1∶n 映射问题结果的标准已不再适用块映射效果的评测，本文将重新给出查全率和查准率的计算公式.

定义 9　当将实体聚合成 k 个类簇时，令集合 $\{B_i \mid 1 \leqslant i \leqslant k\}$ 表示算法生成的 k 个类簇，集合 $\{C_i \mid 1 \leqslant i \leqslant k\}$ 表示专家分成的类簇，$|B_i \cap C_j|$ 表示类簇 B_i 与类簇 C_j 含有相同实体的个数，对于一个确定的类簇 B_i，若类簇 C_j 使 $|B_i \cap C_j|\ (1 \leqslant j \leqslant k)$ 取最大值，则称类簇 C_j 为类簇 B_i 的相似类簇，用 $S(B_i)$ 表示.

定义 10　令 B_i 为算法生成的第 i 个类簇，λ_i 为 B_i 在查准率中所占的权重，则查准率定义为

$$\mathrm{prec} = \sum_{i=1}^{k}\left(\lambda_i \times \frac{|B_i \cap S(B_i)|}{|B_i|}\right).$$

定义 11　令 C_i 为专家划分产生的第 i 个类簇，ω_i 为 C_i 在查全率中所占的权重，则查全率定义为

$$\mathrm{rec} = \sum_{i=1}^{k}\left(\lambda_i \times \frac{|C_i \cap S(C_i)|}{|C_i|}\right).$$

在实际问题中，λ_i 和 ω_i 可以有多种取值方法，若取 $\lambda_i = \dfrac{|B_i|}{\sum_{j=1}^{k}|B_j|}$，则 $\mathrm{prec} = \sum_{i=1}^{k}\dfrac{|B_i \cap S(B_i)|}{\sum_{j=1}^{k}|B_j|}$.

同理，若取 $\omega_i = \dfrac{|C_i|}{\sum_{j=1}^{k}|C_j|}$，则 $\mathrm{rec} = \sum_{i=1}^{k}\dfrac{|C_i \cap S(C_i)|}{\sum_{j=1}^{k}|C_j|}$. 在本文的实验中，令

$$\lambda_i = \frac{|B_i|}{\sum_{j=1}^{k}|B_j|}, \qquad \omega_i = \frac{|C_i|}{\sum_{j=1}^{k}|C_j|}.$$

为了评价本文提出的方法，本文使用 Hownet 及其相关工具对 OAEI（ontology alignment evaluation

initiative)组织给出的一组本体对(animals A 和 animals B)进行实验,通过 k 值的变化观察查全率和查准率的变化,实验结果列于表1. 由表1可见,本文提出的算法对块匹配问题效果较好,随着 k 值的变化,查准率和查全率基本在80%以上,查全率和查准率变化不大,趋于稳定. 但当类簇数目较少时,由于类簇包含的实体较多,类簇间的语义相似度较低,不同类簇间语义相似度的差值较小,偶尔会出现查准率较低的情况,如当 $k=6$ 时,查准率仅为65.4%.

表1 实验结果

Table 1 Experimental results

k 值	查准率/%	查全率/%
20	88.4	88.4
18	84.6	80.7
16	84.6	80.6
14	88.4	88.4
12	88.4	84.4
10	80.7	88.4
8	88.7	92.3
6	65.4	84.6

综上所述,本文利用Hownet中的资源计算匹配元间的语义相似度,基于层次聚类的思想设计了实体分块算法;并重新给出了查全率和查准率的计算公式. 实验结果表明,该方法对块匹配问题效果较好. 但本文的研究仅从词语的语义关系上去求解问题,未考虑本体内部的层次结构和实体间的逻辑联系,所以实验结果中存在着查准率有时较低的问题.

参 考 文 献

[1] LIU Qiang, ZHAO Di, ZHONG Hua, et al. Ontology-Aided Automatic Schema Matching [J]. Journal of Software, 2009, 20(2): 234-245. (刘强, 赵迪, 钟华, 等. 本体辅助的自动化匹配技术 [J]. 软件学报, 2009, 20(2): 234-245.)

[2] TANG Jie, LIANG Bang-yong, LI Juan-zi, et al. Automatic Ontology Mapping in Semantic Web [J]. Chinese Journal of Computers, 2006, 29(11): 1956-1976. (唐杰, 梁邦勇, 李涓子, 等. 语义 Web 中的本体自动映射 [J]. 计算机学报, 2006, 29(11): 1956-1976.)

[3] HU Wei, QU Yu-zhong. Falcon-AO: A Practical Ontology Matching System [J]. Web Semantics: Science, Services and Agents on the Worlds Wide Web, 2008, 6(3): 237-239.

[4] Doan A, Madhavan J, Domingos P, et al. Learning to Map between Ontologies on the Semantic Web [C]//WWW'02 Proceedings of the 11th International Conference on World Wide Web. New York: ACM, 2002: 662-673.

[5] Euzenat J, Valtchev P. Similarity-Based Ontology Alignment for OWL-Lite [C]//European Conference on Artificial Intelligence (ECAI'04). Amsterdam: IOS Press, 2004: 333-337.

[6] HU Wei, ZHAO Yuan-yuan, QU Yu-zhong. Partition-Based Matching of Large Class Hierarchies [C]//Asian Semantic Web Conference (ASWC'06). Berlin: Springer-Verlag, 2006: 72-83.

[7] HU Wei, QU Yu-zhong. Block Matching for Ontologies [C]//Proceedings of the 5th International Semantic Web Conference (ISWC'06). Berlin: Springer-Verlag, 2006: 300-313.

[8] Raghavan V V, Wong K S M. A Critical Analysis of Vector Space Model for Information Retrieval [J]. Journal of the American Society for Information Science, 1986, 37(5): 279-287.

[9] QU Yu-zhong, HU Wei, CHENG Gong. Constructing Virtual Document for Ontology Matching [C]//WWW'06 Proceedings of the 15th International Conference on World Wide Web. New York: ACM, 2006: 23-31.

[10] LIU Chun-chen, LIU Da-you, WANG Sheng-sheng, et al. Improved Semantic Similarity Calculating Model and Application [J]. Journal of Jilin University: Engineering and Technology Edition, 2009, 39(1): 119-123. (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.)

[11] JI Xiang, LIU Hua-xiao, WU Fen-fen, et al. Information Extraction Based on Character Extraction and HMM [J]. Journal of Jilin University: Information Science Edition, 2009, 27(4): 396-399. (纪祥, 刘华虓, 吴芬芬, 等. 基于特征和 HMM 的信息提取 [J]. 吉林大学学报: 信息科学版, 2009, 27(4): 396-399.)

[12] WU Jian, WU Chao-hui, LI Ying, et al. Web Service Discovery Based Ontology Similarity of Words [J]. Chinese Journal of Computers, 2005, 28(4): 595-602. (吴健, 吴朝晖, 李莹, 等. 基于本体论和词汇语义相似度的 Web 服务发现 [J]. 计算机学报, 2005, 28(4): 595-602.)