

基于 VS-Adaboost 的实体对齐方法

万 静¹ 李 琳¹ 严欢春¹ 王少华²

(1. 北京化工大学 信息科学与技术学院, 北京 100029; 2. 武汉大学 国际软件学院, 武汉 430074)

摘 要: 针对现有实体对齐方法大多以本体模式匹配为基础, 处理异构关联数据集间对齐关系存在局限性且实体链接缺失问题严重的现状, 在分析关联数据语义的基础上, 提出了一种独立于模式的基于属性语义特征的实体对齐方法, 对关联数据集中实体属性根据语义标签特征及统计特征建模, 并采用有监督的可变样本集 VS-Adaboost 算法实现分类器优化。实验结果表明, 该方法的时间效率、准确率、查全率较高, F 测度效果较好。

关键词: 关联开放数据; 语义网; 机器学习; VS-Adaboost

中图分类号: TP391 **DOI:** 10.13543/j.bhxbzr.2018.01.012

引 言

知识图谱^[1]的出现为信息获取带来新的转变, 它通过为查询词赋予丰富的语义信息建立与现实世界实体的关系, 从而帮助用户更快找到所需的信息。关联开放数据(linked open data, LOD)^[2]作为知识图谱结构化数据的主要来源, 随着规范化的开放数据发布标准^[3]的逐步统一, 可获得的 LOD 数量在不断增加。实体对齐技术通过建立异构关联数据集实体间的共指链接来进行实体消歧, 从而实现高效的信息共享。所以实体对齐技术是实现语义标准化、促进数据集间互操作化的关键技术, 也是当前知识图谱构建技术中的研究热点和难点。

实体对齐问题可看作一个二分类问题。现有实体对齐方法主要分为基于模式层和基于实例层两类。基于模式层的实体对齐主要针对本体定义中的概念、属性、命名规则进行度量, 且根据是否与匹配数据集构成本体一致又可分为内部和外部两种。内部方法主要依赖于领域内本体定义规则, 外部方法主要依赖与外部知识库的模式映射实现对齐。Tummarello 等^[4]提出的 Sindice 查找索引通过一种明确定义的内部属性来推断实体间等价关系; Nikolov 等^[5]利用关联数据集中各数据集之间现有的 OWL:sameAs 链接来推断实体间等同关系; Nath

等^[6]提出的 Anchor-flood 算法利用本体中概念层次关系进行实例聚类进而实现实体互联。

基于实例层的对齐方法根据论证数据来源不同也可以分为内部和外部两种, 相同点是都利用多种基于属性的相似度度量函数进行合计度量, 算法模型与记录链接^[7]领域的算法较为类似。内部方法主要通过直接对比数据源中相应实体相似度实现, 外部方法则是利用 OWL 的传递性, 通过各数据源中现存 OWL 链接进行聚合作出等价推断。Volz 等^[8]提出的 Silk 系统利用不同相似度匹配方法(如 Avg、Max、Min、Euclidean 距离等)实现链接发现, 进而实现实体对齐; Hu 等^[9]针对实体 URI 提出了一种结合 OWL:sameAs、逆函数属性及最大基数的内核函数方法, 并通过两阶段的自训练算法来实现数据集实体间的对齐。

上述基于模式层或实例层的方法大多依赖于前阶段的模式匹配结果, 易于过滤掉语义异构属性中的重要信息, 链接构建存在缺失; 同时, 关联数据集间实体对齐工作因其本体描述语言(OWL)的多样性和复杂性, 使用基于传统方法的数据预处理时间成本较高。本文提出一种基于属性文本语义特征的实体对齐方法(semantic features based entity alignment, SFEA)。在特征向量建模阶段, 针对关联数据集中具有语义异构特征的实体属性进行分类, 抽取得到非结构化文本的同时保留了一般模式识别方法中易于过滤掉的有价值实体关系信息; 同时在分类器生成阶段, 结合 C4.5 算法和改进的可变样本集 Adaboost(VS-Adaboost)分类学习算法生成分类器来进行实体对齐关系分类。

收稿日期: 2017-03-30

基金项目: 国家科技支撑计划(2015BAK03B04)

第一作者: 女, 1975 年生, 副教授

E-mail: wanj@mail.buct.edu.cn

1 实体对齐定义

关联数据一般采用 resource description framework (RDF) 执行逻辑推理^[10]、描述 web 服务。定义 1 描述了一个使用 RDF 三元组定义的关联数据集。

定义 1 关联数据集形如式(1)

$$D_{LD} = \{O, L, R, P, R_F, P_F\} \quad (1)$$

式(1)中 O 为实例, L 为文本信息, R 为关系, P 为属性, $R_F \subseteq O \times R \times O$ 是一个由主谓宾三元组表示的、宾语为实例的关系事实, $P_F \subseteq O \times R \times L$ 是一个由主谓宾三元组表示的、宾语为文本信息的属性事实。

实体对齐通过对比各个异构或同构关联数据集中的实体^[11] 来确定它们是否指代同一现实实体。根据定义 1, 本文中独立于模式的实体对齐方法主要是利用实例层信息, 将每一个 RDF 三元组数据文本看作整体。基于此方法进行的实体对齐可由定义 2 描述。

定义 2(实体对齐 (Entity Alignment)) 给定数据集

$$D_{LD_1}, D_{LD_2}, E_A(D_{LD_1}, D_{LD_2}) = \{(a_i, b_j, \text{sim}(a_i, b_j)) \mid a_i \in D_{LD_1}, b_j \in D_{LD_2}, (a_i, b_j) \in D_{LD_1} \times D_{LD_2}, \text{sim}(a_i, b_j) \in [0, 1]\}$$

其中, $\text{sim}(a_i, b_j)$ 为表述实体 a_i 和 b_j 语义相似度的数值, 通过判断 $\text{sim}(a_i, b_j)$ 与阈值 ε 的大小关系进而判定实体间是否有指代一致的等同关系。

2 基于属性文本语义特征的实体对齐 (SFEA) 方法

2.1 SFEA 方法框架

本文提出的 SFEA 方法通过分析、抽取实体对属性特征建模得到特征向量, 再采用有监督的机器学习分类方法训练生成分类器, 进而将实体关系进行有效分类, 通过将等同和不等同两类映射到两个不同标号的类来实现实体对齐。即找到一个分类器 C_1 , 将非对齐实例分到 -1 类中, 将对齐实例分到 1 类中。SFEA 方法框架如图 1 所示。

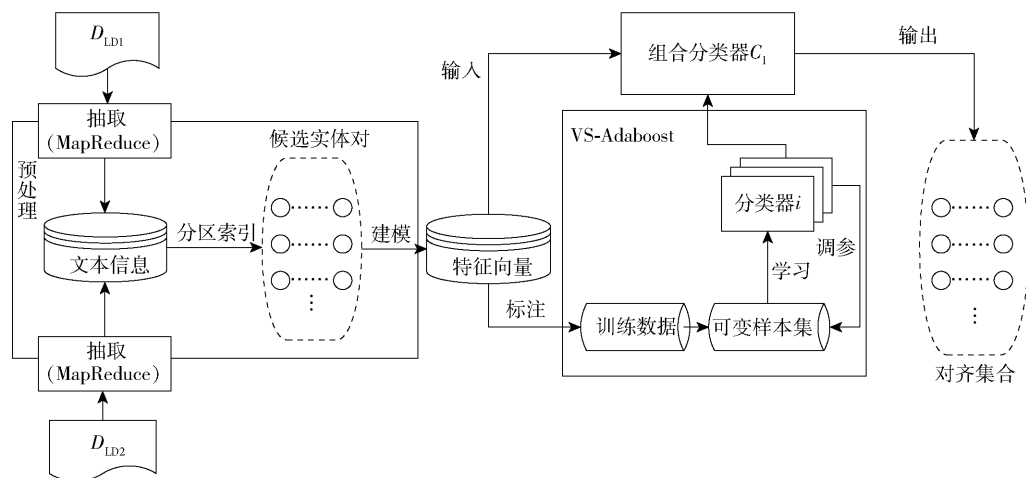


图 1 SFEA 方法流程图

Fig. 1 SFEA approach flow chart

SFEA 方法主要实现步骤如下。

1) 数据预处理。针对给定数据集中三元组数据包含的丰富文本信息, 在文本抽取过程中使用 MapReduce 框架建立 key-value 对, 并进行排序获得有价值的文本信息。对每一实体抽取独立于模式的属性文本信息后, 按类别赋予不同权重并筛选出候选实体对集合, 降低计算复杂度。

2) 特征向量建模。对数据预处理阶段抽取得到的文本信息, 选用合适的特征相似度计算函数生成特征向量。

3) 分类学习。标注已有对齐关系作为训练数据, 使用 C4.5 算法迭代生成并测试基分类器, 并通过改进的 VS-Adaboost 算法更新样本集规模, 得到平衡总分类器, 进而实现数据集实体对齐。

2.2 数据预处理

数据预处理包括数据标准化抽取及分区索引建立。

1) 数据标准化抽取

本文数据标准化抽取工作中通过轻量级 JavaScript object notation (JSON) 技术解析 RDF 三元组数

据^[12]得到 JSON 字符串,再提取(key,value)集并分类。具体属性文本信息抽取过程中,将待对齐的实体对看作一系列非结构化的字符串的组合,根据不同属性中文本描述特征的不同抽取下列几类数据。

(1) l_{label} 类:实例名称等标识性的词语通过 $\langle \text{rdf:label} \rangle$ 属性识别。

(2) l_{property} 类:实例属性中进行文本描述。

(3) l_{value} 类:实例属性值中的文本描述。某一属性值含义由于描述词语数量的不同存在差异,所以将属性值的文本抽取根据词语数量的不同抽取为 l_{single} 、 l_{short} 、 l_{long} 3 类。

(4) 针对三元组实例可能存在多样性数据信息,抽取 l_{date} 、 l_{number} 、 l_{link} 作为补充。

2) 分区索引建立

由定义 2 可知,对于各自包含 $|A|$ 和 $|B|$ 个实例的数据集来说,实体对齐过程中要处理的实例对的个数是 $|A| \times |B|$,这对于本来就拥有百万级数据的数据集来说工作量十分巨大,通过设计高效的索引技术可以避免数据集规模较大带来的计算复杂度问题,因此进行筛选预处理是很有必要的。本文通过为描述数据集中实体的关键词设置倒排索引(inverted value)筛选得到候选相似对齐实例集 \mathcal{S} ,其中关键词的获得是在文本信息抽取中通过 tokenizer() 方法得到的。

2.3 特征匹配

为保证数据评估的可靠性,本文选用多种相似形函数来构建实体属性特征向量并计算实体间相似关系。在此过程中采用 MapReduce 并发处理模型框架能在筛选有效信息的同时,最大程度保证候选实体对的完整性、降低计算复杂度。

相似度特征计算方法为:以数据预处理结果为输入,分别使用 MapReduce 计算框架的 Map()、Reduce() 过程提取文本特征项,生成特征向量。在提取文本特征项过程中,选用具有较好类别区分能力的 TF-IDF 统计方法度量字、词重要性及相似程度,并结合三元组数据的文本语义标签特征,针对不同类型文本选用合适的度量方法得到特征向量 $e_i(p_1, (p_1), p_2, (p_2), \dots, p_n, (p_n))$ 。

由于文本特征存在区别,分别选用式(2)~(6) 5 大类度量函数计算各个维度上的向量值(范围[0, 1])。

1) 属性标签文本(l_{single} 、 l_{short} 、 l_{long} 、 l_{label})由于含有较多文本描述,适用于余弦相似度核函数 SimCos

(), 通过计算描述词的 TF-IDF 值($F_{\text{tf-idf}}()$)得到其特征值,计算公式如式(2)

$$\text{SimCos}(T_1, T_2) =$$

$$\frac{\sum_{t_i \in T_1, t_j \in T_2} [F_{\text{tf-idf}}(t_i) \times F_{\text{tf-idf}}(t_j)]}{\left[\sum_{t_i \in T_1} F_{\text{tf-idf}}(t_i)^2 \right] \times \left[\sum_{t_j \in T_2} F_{\text{tf-idf}}(t_j)^2 \right]} \quad (2)$$

2) 短描述属性(l_{single} 、 l_{short} 、 l_{label})中常含关键词,适用于逆文档频率度量函数 SimIdf(), 公式如式(3)

$$\text{SimIdf}(T_1, T_2) =$$

$$\frac{\sum_{t_i \in T_1, t_j \in T_2} [F_{\text{idf}}(t_i) \times F_{\text{idf}}(t_j)]}{\left[\sum_{t_i \in T_1} F_{\text{idf}}(t_i)^2 \right] \times \left[\sum_{t_j \in T_2} F_{\text{idf}}(t_j)^2 \right]} \quad (3)$$

3) 对于短描述特性,为更好权衡单个词汇的重要性,进一步使用字逆文档频率函数 SimTopidf() 计算,公式如式(4)

$$\text{SimTopidf}(T_1, T_2) =$$

$$\frac{\sum_{w \in W_1 \cap T_2} F_{\text{idf}}(w) + \sum_{w \in W_2 \cap T_1} F_{\text{idf}}(w)}{\sum_{w \in W_1} F_{\text{idf}}(w) + \sum_{w \in W_2} F_{\text{idf}}(w)} \quad (4)$$

4) 由于对每一实例仅抽取一个 label 属性,而且 label 中的描述词对于对齐与否的判断较为重要,所以对于 l_{label} 属性的度量适用计算 Levenshtein 相似度($S_{\text{levenshtein}}(l_{\text{label}}^a, l_{\text{label}}^b)$)的度量函数 SimEdit() 及 SimCount(), 计算公式如式(5)、(6)

$$\text{SimEdit}(l_{\text{label}}^a, l_{\text{label}}^b) = 1 - \frac{S_{\text{levenshtein}}(l_{\text{label}}^a, l_{\text{label}}^b)}{\max(|l_{\text{label}}^a|, |l_{\text{label}}^b|)} \quad (5)$$

$$\text{SimCount}(l_{\text{label}}^a, l_{\text{label}}^b) = \frac{1 - 2^{-|w|w \in l_{\text{label}}^a \cap l_{\text{label}}^b|}}{1 - 2^{-(|l_{\text{label}}^a| + |l_{\text{label}}^b|)/2}} \quad (6)$$

另外, l_{single} 一般是 ISBN 类极具标识性的字符, l_{short} 中少量词的出现对实例对齐与否的判断起着较关键的作用,所以 l_{single} 、 l_{short} 、 l_{long} 在特征向量的相似度计算中应赋予较高权重。

2.4 VS-Adaboost 分类学习

2.4.1 二分类问题

假设两实例对 (p_1, p_2) 和 (q_1, q_2) , 特征相似度向量分别用 μ 、 y 表示, $u_6 = v_6 = 0.3$, $u_5 = 0.1$, $v_5 = 0.5$ 。若 p_1 和 p_2 分别是由一些长文本和短文本组成的,而 q_1 和 q_2 可能是由同领域内的一些长文本构成的,那么只根据 $u_5 < v_5$ 并不能判定 q_1 、 q_2 为同一现

实实体的可能性大于 p_1, p_2 , 也就是说它们之间的关系并不是线性可分的。

特定领域的关联数据集^[13]具有丰富的上下文语义关系,特征相似度向量的维度之间也包含隐含关系,因此本文选择决策树中的 C4.5 算法生成基分类器^[14],并在训练中迭代更改样本集规模、优化分类器性能;另外待处理的关联数据集数据量较大、实体个数较多是实体对齐的关键问题,因此神经网络、支持向量机等具有复杂内核的学习模型并不适用于此场景,因此本文提出了一种有监督的可变样本集 VS-Adaboost 算法训练模型来优化学习模型、生成训练数据集。二分类学习过程总体流程如算法 1 所示。

算法 1 二分类学习算法

输入 正训练数据集 P , 负训练数据集 N , 候选集 \mathcal{S} , 基分类器 M , 迭代次数 N , 可变样本影响因子 f_P, f_N

- 1) 初始化
 $S_L = |\mathcal{S}|$;
 $S_P = |P|$;
 $S_N = |N|$;
- 2) 将 P, N 作为训练数据使用 C4.5 算法及 VS-Adaboost 训练 M , 得到总分类器 M'
- 3) $\text{int } n = 0$; // 迭代计数
- 4) **while** $n < N$ **do**
 $S_P = S_P \times f_P$;
 $S_N = S_N \times f_N$;
if $i = 1$
 $P = L_i$; // 使用 L 中第一个 S_D 作为新的正训练数据
else
 $N = L_i$; // 其余作为负训练数据
 $n = n + 1$;
end while

输出 \mathcal{S} 中的有序列表 L

2.4.2 总分类器生成

C4.5 算法将信息增益率函数作为分类标准,避免了对多重取值属性的选择。该方法使用有放回的方法进行抽样,生成差异性较大、准确性较高的基分类器,Adaboost 算法通过不断迭代更新样本权重来更新样本集,进而得到分类器权重,最后得到总分类器 M' 。 M' 训练生成流程如算法 2 所示。

算法 2 VS-Adaboost 分类算法

输入 训练样本集 $\{(x_1, y_1), (x_2, y_2), \dots\}$,
C4.5 分类器

- 1) $\sigma_i^1 = w(i), i = 1, \dots, m$ // 初始化样本权重
- 2) **for** $t = 1, \dots, T$ // 迭代次数

$d^t = w^t / \sum_{i=1}^m w_i^t$ // 产生分布
 $\lambda_t: X \rightarrow [0, 1]$ // 基于分布 d^t 调用 C4.5 得到

$\phi_t = \sum_{i=1}^m d_i^t |\lambda_t(x_i) - y_i|$ // λ_t 的误差率
 $\psi_t = \phi_t / (1 - \phi_t)$ // 计算新误差
 $\sigma_i^{t+1} = \sigma_i^t \times \psi_t$ // 更新

end for

输出

$$M'(x) = \begin{cases} 1 & \sum_{t=1}^T \left(\log \frac{1}{\psi_t} \right) \lambda_t(x) \geq \frac{1}{2} \sum_{t=1}^T \left(\log \frac{1}{\psi_t} \right) \\ 0 & \end{cases}$$

3 实验与结果分析

3.1 实验设置

实验条件设置为:VMware Workstation 12 Pro;系统 Oracle_Linux_7u2_64bi;jdk 版本 jdk-8u77-linux-x64;Hadoop 版本 Hadoop1.2.1;控制台 XShell5;虚拟机为 1 个 master、3 个 node,每台机器配置均为 2G + 20G。

实验数据取自 British Museum(BM)、Amsterdam Museum(AM)、DBpedia^[15](DB) 和 New York Times(NY)4 个 LOD 数据集,数据集信息如表 1 所示。

表 1 实验数据集		
Table 1 Experimental data set		
数据集名称	三元组个数	已有链接数 (to DBpedia)
British Museum	1806753903	
Amsterdam Museum	5000000	43
DBpedia	1200000000	750
New York Times	345889	10359

以 DBpedia 为对齐标准,与其他 3 个数据集分别进行两两实验,由于 AM、NY 中已存在与 DBpedia 的 owl:sameAs 链接,将已有链接作为训练数据进行分类器的学习,从而进行 AM-DB、NY-DB 两组实验,最后根据已有标注数据进行 BM-DB 上的实验。

3.2 评估方法

使用精确率(p)、召回率(r)与 F1 测度(F_{1-m})作为分类结果的评价标准,计算公式如式(7)~(9):

$$p = \frac{P_1}{P_1 + N_1}$$
(7)

$$r = \frac{P_1}{P_1 + N_2}$$
(8)

$$F_{1-m} = 2 \times \left(\frac{pr}{p+r} \right)$$
(9)

其中, P_1 为实验中被分类器正确标记的对齐数据; P_2 为被正确标记的非对齐数据; N_1 为实验中被分类器错误标记的非对齐数据; N_2 为实验中被分类器错误的标记对齐数据。

另外在索引构建的实验过程中,发现候选对筛选的召回率是不完美的(为 0.9),这意味着最大可实现分类的 F_{1-m} 严格低于 100%。因此相应的 F_{1-m} 只是悲观估计,分类器不可能完全错误地标记所有遗漏的匹配对。因此本文使用加权悲观 F-Measure(F_m)公式(10)来估计真实 F_{1-m} 。

$$F_m = \frac{100F_{1-m}}{F_{max}}$$
(10)

3.3 结果分析

为了对比本文算法(VS-Adaboost)与其他算法之间的性能差异,实验中同时使用 RiMOM 框架进行对齐链接生成。在使用 VS-Adaboost 方法分类学习过程中,通过多组实验,得到最佳迭代次数为 7 时分类器效果最佳(可避免过拟合现象)。最终实验结果如表 2 所示。

表 2 实体对齐结果
Table 2 Entity alignment results

数据集	方法	r	p	F_m
NY-DB	RiMOM	0.342	0.961	0.504
	VS-Adaboost	0.859	0.952	0.903
AM-DB	RiMOM	0.315	0.837	0.458
	VS-AdaBoost	0.726	0.875	0.794
BM-DB	RiMOM	0.246	0.293	0.267
	VS-AdaBoost	0.373	0.377	0.375

由表 2 可知,3 组实验中基于 VS-Adaboost 算法的实体对齐方法在召回率与 F_{1-m} 测度值上均比 RiMOM 方法有一定程度的提高,从而可以有效发现模式匹配中易于过滤掉的链接关系,提高召回率;但准

确度方面,在 BM-DB 实验中有错误分类的情况出现,原因可能是缺乏较为充分、有效的标注训练数据。整体来说利用本文方法可以在较少人工干预前提下实现对关联数据集中数据实例的高效链接。

4 结论

(1) 通过分析关联数据的属性语义特征,应用独立于模式的匹配算法将属性值信息看做纯文本信息,可以发现在模式匹配中被排除的隐含对齐信息。

(2) 利用大规模并行 MapReduce 框架处理文本特征,为处理大规模结构化数据集上的实体对齐问题带来新的可能;同时在保证数据完整性的前提下,利用一种有监督的机器学习算法降低了对齐过程的计算复杂度。

(3) 经过验证发现本文方法在存储空间方面所需缓冲存储空间大约仅为 40 M 左右,这为处理高数量级数据及带来更多可能,同时可以极大保证等同链接构建的质量,在未来知识图谱构建中有着广泛前景。

参考文献:

[1] Ji G L, Liu K, He S Z, et al. Knowledge graph completion with adaptive sparse transfer matrix[C] // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. Phoenix, USA, 2016: 985-991.

[2] Bizer C, Heath T, Berners-Lee T. Linked data-the story so far[J]. International Journal on Semantic Web and Information Systems, 2009, 5(3): 1-22.

[3] Cyganiak R, Jentzsch A. The linking open data cloud diagram[EB/OL]. [2017-08-22]. <http://lod-cloud.net>.

[4] Tummarello G, Delbru R, Oren E. Sindice. com: weaving the open linked data[C] // ISWC/ASWC 2007. Busan, South Korea, 2007: 552-565.

[5] Nikolov A, Uren V, Motta E. Data linking: capturing and utilising implicit schema-level relations[C] // 2010 Wordshop on Linked Data on the Web. Raleigh, USA, 2010.

[6] Nath R P D, Seddiqui H, Aono M. An efficient and scalable approach for ontology instance matching[J]. Journal of Computers, 2014, 9(8): 1755-1768.

[7] Hassanzadeh O, Lim L, Kementsietsidis A, et al. A declarative framework for semantic link discovery over relational data[C] // Proceedings of the 18th International Conference on World Wide Web. Madrid, Spain, 2009:

- 1101–1102.
- [8] Volz J, Bizer C, Gaedke M, et al. Silk-a link discovery framework for the web of data[C] // 2009 Workshop on Linked Data on the Web 2009. Madrid, Spain, 2009: 538–543.
- [9] Hu W, Chen J F, Qu Y Z. A self-training approach for resolving object coreference on the semantic web[C] // Proceedings of the 20th International Conference on World Wide Web. Hyderabad, India, 2011: 87–96.
- [10] Bazoobandi H R, Rooij S D, Urbani J, et al. A compact in-memory dictionary for RDF data[C] // European Semantic Web Conference 2015. Portoroz, Slovenia, 2015: 205–220.
- [11] Liu S L, Liu K, He S Z, et al. A probabilistic soft logic based approach to exploiting latent and global information in event classification[C] // Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence. Phoenix, USA, 2016: 2993–2999.
- [12] 易军凯, 周育彬, 万静. 一种基于关联数据的数字博物馆语义融合方法[J]. 北京化工大学学报: 自然科学版, 2014, 41(6): 103–108.
- Yi J K, Zhou Y B, Wan J. Linked data-based semantic integration for digital museum[J]. Journal of Beijing University of Chemical Technology: Natural Science, 2014, 41(6): 103–108. (in Chinese)
- [13] Wang Z, Zhang J W, Feng J L, et al. Knowledge graph embedding by translating on hyperplanes[C] // Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence. Québec, Canada, 2014: 1112–1119.
- [14] Schapire R E. The boosting approach to machine learning: an overview[M] // Nonlinear estimation and classification. New York, USA: Springer New York, 2003: 149–171.
- [15] Bizer C, Lehmann J, Kobilarov G, et al. Dbpedia-a crystallization point for the web of data[J]. Web Semantics Science Services and Agents on the World Wide Web, 2009, 7(3): 154–165.

An entity alignment approach based on the VS-Adaboost algorithm

WAN Jing¹ LI Lin¹ YAN HuanChun¹ WANG ShaoHua²

(1. College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029;

2. International Software Institute, Wuhan University, Wuhan 430074, China)

Abstract: Entity alignment is the key technology needed to realize knowledge graph construction, information integration and sharing. Most of the existing entity alignment approaches are based on ontology pattern matching, and have limitations when dealing with the alignment of heterogeneous linked open data (LOD), where the problem of missing entity links is serious. In this paper, a schema-independent entity alignment approach based on attribute semantic features is proposed. The entity attributes of the LOD are modeled according to their semantic label characteristics and statistical features. The supervised variable set VS-Adaboost algorithm is used to realize classifier optimization. The experiments executed on selected datasets show that compared with conventional methods the approach is more efficient and has better efficacy in terms of accuracy, recall and F measurement.

Key words: linked open data; semantic web; machine learning; VS-Adaboost

(责任编辑: 汪 琴)