一种面向 Deep Web 数据源的重复记录识别模型

申德荣,刘丽楠,寇 月,聂铁铮,于 戈
(东北大学信息科学与工程学院,辽宁沈阳 110004)

摘 要: 重复记录是指描述现实世界中同一实体的不同的记录信息.由于从同一个领域的不同 Deep Web 数据源中抽取的记录信息通常存在许多重复记录,本文针对半结构化的重复记录的识别进行研究.在已知全局模式和全局模式与各 Deep Web 数据源查询接口映射关系的基础上,提出了一种重复记录识别模型.基于从 Deep Web 中抽取出的半结构化的数据,采用查询探测方法确定所抽取数据所匹配的属性,通过分析抽取的实例数据确定属性重要度,结合多种相似度估算器和多种算法计算记录间的相似度,进而识别重复记录.实验表明,该重复记录识别模型在 Deep Web 环境下是可行且有效的.

关键词: 重复记录; 深层 web; 数据清洗
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2010) 02-0275-07

A Duplicate Records Identification Model for Deep Web Data Sources

SHEN De-rong, LIU Li-nan, KOU Yue, NIE Tie-zheng, YU Ge
(*School of Information and Science and Engineering, Northeastern University, Shenyang, Liaoning* 110004, *China*)

Abstract: Duplicate records are multiple different records describing the same entity in the real world. Since some of the records extracted from different Deep Web sources in the same domain usually are duplicates, the paper focuses on duplicate records identification and a duplicate records identification model is proposed on the basis of known global schema and the relationship between the global schema and the interface attributes of each Deep Web data source. Based on the semi-structured data extracted from Deep Web data sources, the attributes that these data matching to are annotated by using a query probing method and the dominance of attributes of global schema is specified by analyzing these extracting instance data. Moreover, multiple estimators and multiple similarity algorithms are adopted to identify the duplicates. The experiment results show our duplicate record identification model is feasible and efficient.

Key words: Duplicate records identification; deep web; data extraction

1 引言

重复记录是指描述现实世界中同一实体的不同形式的记录信息,如来自不同数据源的描述同一本书的记录信息.从不同 Deep Web 数据源抽取出的记录信息可能存在很多重复数据,需要在提交给用户之前进行识别,目的是为用户提供高质量的结果记录.然而,从各个网站中抽取出的数据信息主要是用 XML 或 HTML 标签描述的半结构化数据,并且其所对应的结构化的信息事先不可知,因此,传统的基于关系数据的重复记录识别方法并不适用.

在特定领域内,已知全局模式和全局模式与 Deep Web 数据源局部接口模式映射关系的基础上,本文提出了一种重复记录识别模型[7].针对从 Deep Web 中抽取出的半结构化数据,分析抽取数据所匹配的全局模式属性,基于实例数据确定属性重要度,并结合多种相似度估算器和多种算法计算记录间的相似度,进而识别重复记录.在计算来自不同数据源的实体记录间的相似度时,我们提供了一个可扩展的相似度算法库,可以针对不同领域制定相应的相似度计算策略和选择不同的相似度计算方法.新的相似度算法也可方便补充到相似度算法库中.实验表明,该重复记录识别方法是准确而有效的.

2 相关工作

重复记录识别,也被称作数据清洗或去重,作为一

收稿日期:2008-04-18;修回日期:2009-06-20
基金项目:国家自然科学基金(No.60973012,No.60693139);国家 863 高技术研究发展计划(No.2008AA01Z146,No.2009AA01Z131)

项技术已有许多 ETL(Extracting, Transforming, Loading)工具,如 Data Stage[1]和 CoSort[2].传统的重复记录识别方法[3,4],主要针对结构化数据,是在表的模式信息已知的前提下,比较两个元组在对应属性上文本的相似度.而对半结构化数据,其并没有明确的属性名称和含义,有些甚至无属性名称,因此传统的基于关系数据的处理方法不能很好适用.对于 XML 对象的重复记录的识别[5,6],文献[5]只使用了单一的字符串的比较方法.文献[6]从 XML 文档中抽取的数据被存储在称为对象描述的关系形式中,若两个对象描述的元组数据含有相同的 XPath,则认为是相似的对象.以上方法中,都是只使用单一的相似度计算方法计算记录相似度,并没有针对不同的领域特征而采用不同的相似度计算策略,也没有根据不同数据类型的数据特性而采用不同的比较方法.

本文提出了一种重复记录识别模型,处理从不同 Deep Web 数据源中抽取出来的半结构化的重复数据.主要特点如下:(1)该重复记录模型不仅适用于关系型数据,也适用于半结构和无结构化的数据;(2)可根据需求结合多种相似度估算器来识别重复记录;(3)每个相似度估算器由支持特定数据类型的多种相似度计算方法构成.

3 重复记录识别模型

本文提出的适应于 Deep Web 环境下的重复记录识别模型框架如图 1 所示.重复记录识别模型主要由数据预处理模块、同构记录处理模块、异构记录处理模块组成.在该模型中,首先将查询的结果记录输入到数据预处理模块,在该模块中创建结果记录的 DOM 树并将其转化成实体记录的形式;然后,在同构记录处理模块中确定各实体记录的属性值与全局模式的匹配关系并计算全局模式各属性的权重;在异构记录处理模块中,计算异构记录间的相似度并由此生成重复记录集.其相关概念及定义描述如下.

定义 1 属性值(r_j)和实体记录(O_i). r_j 是组成实体记录的一个属性的值(即 DOM 树中每个叶子结点文本内容). O_i 是从数据源中抽取出的一条结果记录或称实体记录,它是由多个属性值组成.其中,i 表示从某一数据源中抽取出第 i 个实体记录,j 表示组成实体记录的第 j 个属性值.

定义 2 同构记录是指从同一数据源中抽取出的实体记录,它们有相同的模式(DOM 树结构).

定义 3 异构记录是指从不同数据源中抽取出的实体记录.它们的模式(DOM 树结构)可能不同.

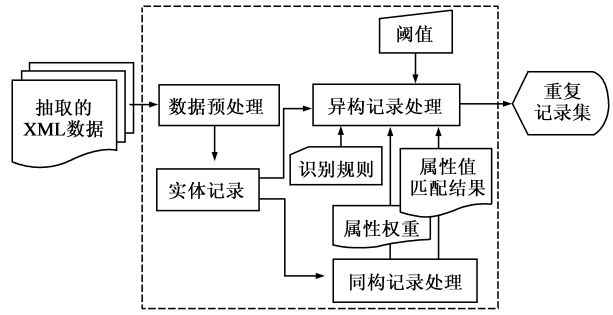

图1 重复记录识别模型框架

4 数据预处理模块

重复记录识别模型在进行同构记录和异构记录处理前,首先要对所抽取出的 XML 或 HTML 标签表示的结果数据进行预处理.对结果记录的预处理主要包括两部分:各结果记录 DOM 树的创建和实体记录的生成.

4.1 DOM 树的创建

本文采用 DOM(Document Object Module 文档对象模型)来解析 XML 文档.首先对从数据源中抽取出的每条结果记录所对应的 XML 文档在逻辑上创建一个 DOM 树模型,树中的每个结点代表一个对象,这样通过操作这棵树和树的结点就可以完成对 XML 文档的操作.例如,图 2 为所抽取出的一条查询结果记录的 DOM 树.

4.2 实体记录的创建

针对来自不同网站的查询结果的 DOM 树结构不同、DOM 树各结点的属性含义及匹配关系未知等特征,我们分别分析每条记录的 DOM 树,将其转化成统一的实体记录的形式.

(1)初始化实体记录

对于每条记录所对应的 DOM 树,考虑各数据源 DOM 的结构特点,抽取其叶子结点下的文本信息并组成集合,表示为 $O_i = \{r_1, r_2, \cdots r_j\}$. 例如,对应图 2 中 DOM 树的实体记录为 $O_1 = \{r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}, r_{11}\}$,其中 r_1 = "Think in Java(4th Edtion)", r_2 = "By", r_3 = "Bruce Eckel", r_4 = "Paperback", r_5 = "-Feb 20,

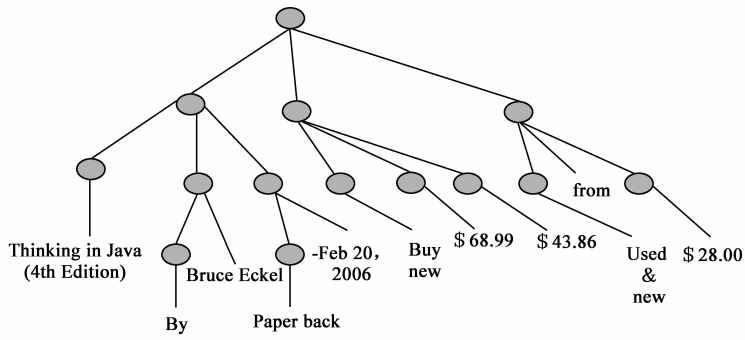

图2 查询结果的DOM树示例

2006", r_6 = "Buy new", r_7 = "$68.99", r_8 = "$43.86", r_9 = "Used&new", r_{10} = "from", r_{11} = "$28.00".

(2)删除噪声属性值

经过步骤(1)获得的实体记录中,一些属性值如图2中 By,Buy new 等,是无用的,甚至影响处理的效率和准确度,我们称这些属性值为噪声属性值,需要删除它们.经观察,这些噪声属性值并不是 Deep Web 数据源存储的实体本身的信息,并且这些信息重复出现在该数据源抽取出的实体信息中.通过判断其是否重复出现在该数据源抽取的所有实体中(即比较 $O_m.r_j$ 的值与 $O_n.r_j$ 的值($m,n=1,2,\cdots$且 $m\neq n$)是否都相同),若重复出现,则为噪声属性,将其删除.通过删除噪声属性值后,基于图2中记录获得的新实体记录为 $O_1=\{r_1,r_2,r_3,r_4,r_5,r_6\}$,其中 r_1 = "Think in Java(4th Edtion)", r_2 = "Bruck Eckel", r_3 = "Feb 10,2006", r_4 = "$40.94", r_5 = "$33.00", r_6 = "Get it…".我们将删除噪声属性值后的数据作为数据预处理模块的结果数据,输入到同构和异构记录处理模块.

5　同构记录处理模块

同构记录处理模块中,我们主要利用从相同数据源中抽取出来的实体记录,基于全局模式来确定实体记录属性值与全局模式属性间的匹配关系,并确定全局模式各属性在计算实体记录相似度时的权重.

5.1　实体记录属性值模式的确定

对于从每个数据源获取的实体记录,在计算异构记录间的相似度之前,需要确定各记录 DOM 树叶子结点值对应的全局模式属性的匹配关系(即各异构记录属性值间的匹配关系).已知全局模式查询接口到各 deep Web 查询接口的映射关系,因此我们利用抽取出的同构的样本实体记录,基于全局模式来确定各实体属性值间的匹配关系.其主要步骤为:

(1)向全局查询接口提交实体属性值.从每个 Deep Web 数据源中抽取 N 个样本实体记录(本模型实验 N 取值为20),经过数据预处理后,得到了由各个属性值组成的实体记录 $O_i=\{r_1,r_2,r_3,\cdots,r_j\}$,其中,每个属性值 r_j 都对应于 DOM 树的第 j 个叶子结点.将各属性值 r_j 分别单独提交给全局查询接口,通过全局接口映射到该实体记录所在数据源的查询接口进行查询[8,9].在提交过程中,遵循以下规则:①每次只向全局接口提交某个实体记录中的一个属性值 r_j;②全局查询接口的其余属性的值为空或默认值;③属性值 r_j 要提交给全局查询接口的所有属性来确定该属性值与全局接口属性的匹配关系.

(2)查询结果的获取与分析.由于属性值的提交是将从本数据源中抽取出的结果通过全局查询接口再次提交到本数据源,因此,只要全局接口属性与提交的属性值匹配就会得到查询结果,反之将没有或有少量的查询结果.我们根据此性质对查询结果进行分析.对属性值 r_j 的每次提交,我们将其查询结果记录为 $r_{j\text{-}attri}=\{result,result\text{-}num\}$,其中 $attri$ 为全局查询接口中的属性 i(即表示该结果是 r_j 提交给全局查询接口中的属性 i 的结果),$result$ 表示是否有查询结果(其取值为布尔值:有查询结果 $result$,取值为1,否则,为0),$result\text{-}num$ 为结果记录个数.

(3)属性值间匹配关系的确定.通过将各样本实体记录的属性值 r_j 提交到全局接口可得到查询结果值 $r_{j\text{-}attri}.result$,统计同一数据源下各样本实体记录的属性值 r_j 的提交结果之和 $\sum r_{j\text{-}attri}.result$,则对于一个含有 i 个属性的全局查询接口和一个各实体记录都含有 j 个属性值的实体记录集间可以生成一个查询结果矩阵,如图3所示,矩阵中的元素值就是将实体记录 DOM 相同叶子结点值提交到全局模式属性查询接口返回结果布尔值之和.则与属性值 r_j 所匹配的全局模式的属性应为其所在行取值最大的元素所对应的属性,并且该元素的值也应是其所在列的最大值.否则,则需专家来确定与该属性值所匹配的全局属性.若 r_j 所在行所有元素的值都为0(即 $\sum_{n=1}^{i}\sum r_{j\text{-}attri}.result=0$),则全局属性中没有与该属性值 r_j 匹配的属性,则将该类属性值放到待匹配属性值集合 R 中.对 R 的处理将在第5节说明.可见,通过对样本实体记录的分析就可得到某一数据源下实体记录各属性值与全局模式的属性间的匹配关系.

$$\begin{array}{c} \quad attr_1 \quad\quad attr_2 \quad\quad \cdots \quad\quad attr_i \\ \begin{matrix} r_1 \\ r_2 \\ \\ r_j \end{matrix}\begin{bmatrix} \sum r_{1\text{-}att1}.result & \sum r_{1\text{-}att2}.result & & \sum r_{1\text{-}atti}.result \\ \sum r_{2\text{-}att1}.result & \sum r_{2\text{-}att2}.result & & \sum r_{2\text{-}att2}.result \\ \vdots & \vdots & \ddots & \vdots \\ \sum r_{j\text{-}attj}.result & \sum r_{j\text{-}attj}.result & & \sum r_{j\text{-}attj}.result \end{bmatrix} \end{array}$$

图3　查询结果矩阵

5.2　全局模式属性权重的确定

在计算实体记录相似度时,不同的属性在判定相似度时所占的权重是不同的,因此,在基于全局模式计算实体记录相似度前,要确定特定领域中全局模式各属性的权重.

性质1　当向全局模式的某一属性接口提交精确查询时,若该属性的唯一性越高,返回的结果就越少,则该属性在识别记录是所占的重要度就越高.

例如,在图书领域中,ISBN 属性的重要度高于出版社属性,因为 ISBN 唯一标识一本书的实体,而出版社不能.

根据性质 1,本模型提出以下计算特定领域全局模式权重的方法:

$$\omega_{attr\text{-}name} = 1 - \frac{\sum result\text{-}num_{attr\text{-}name}}{\sum result\text{-}num}$$

其中,$\omega_{attr\text{-}name}$为某全局属性的权重,$\sum result\text{-}num_{attr\text{-}name}$为提交到该属性所有查询的查询结果的个数,$\sum result\text{-}num$ 为总的查询结果的个数(即提交的所有查询的查询结果的个数).

6 异构记录处理模块

在异构记录处理模块中,我们要在所抽取出的查询结果中找出描述现实世界同一实体的重复记录.其具体算法如下:

算法 1 重复记录识别算法

输入:实体记录,即 $O_i = \{r_1, r_2, \cdots\}$;

处理步骤:

(1)根据实体记录属性值与全局模式属性的匹配关系,利用多种相似度估算器计算实体记录的相似度值;

(2)整合相似度值;

(3)识别重复记录;

输出:重复记录集.

经过数据预处理模块处理后的实体记录,作为异构记录处理模块的输入数据.通过计算异构记录间的相似度,识别重复记录.

在异构记录处理模块中,我们采用了一系列的针对不同数据类型的相似度估算器.每个相似度估算器仅处理某种特定类型的实体记录属性值.使用多种相似度估算器主要有以下两点好处:首先,采用多种相似度估算器使相似度比较更具有针对性.其次,使用相似度估算器这种模式的可扩展性好,可以随时加入新的相似度估算器.另外,针对不同的领域我们还可以选择特定的相似度估算器.

6.1 多种相似度估算器

传统的计算两个实体记录相似度值的方法是将属性值逐一做比较或将整个记录的值看作一个字符串进行比较,这种方法的时间代价相对较高且准确度不好.为了减少比较次数和提高计算的准确度,我们根据特定领域各全局属性的特征,针对不同属性类型采用不同的相似度估算器来计算两个实体记录各属性值间的相似度值.

本模型中,已实现了四种类型的相似度估算器:文本类型、数字类型、日期类型和价格类型.每个相似度估算器针对一种属性类型计算其相似度.同时,还提供了一系列的匹配算法,通过结合不同的匹配算法,多种相似度估算器可以构成更多的模式来支持不同类型的属性值相似度的计算.已实现的四个相似度估算器描述如下:

①文本型相似度估算器.主要计算字符类型的数据间的相似度值,如人名、书名等.在该估算器中我们主要采用了以下三种匹配算法:Q-gram 算法[10]、Affine gap distance 算法[11]、Jaro Distance[12].

②数字相似度估算器.主要针对数字类型数据间的相似度值,如图书的 ISBN 号(数字型的数据)、房屋的面积等.传统的计算数字型数据相似度的算法相对简单,典型的就是将数字看成字符串进行简单的比较.我们提出了两种数字型相似度估算器的方法,具体描述如下:

精确距离算法:若两个数字型字符串 n_1 和 n_2 完全相同,则它们的相似度为 1,否则为 0.

范围距离算法:若两个数字型字符串 n_1 和 n_2 在数值上的差小于一个阈值 δ,则认为它们是相似的.两个数字型数据 n_1 和 n_2 的相似度计算方法如下:

$$s(n_1, n_2) = 1 - \sqrt{\frac{(n_1 - \bar{n})^2 + (n_2 - \bar{n})^2}{2}} \Big/ \bar{n}$$

其中,$\bar{n}$ 是 n_1 和 n_2 的平均值.

③日期型相似度估算器.用来计算日期类型数据的相似度值.首先要将所有的日期型数据都转换成统一的表示形式"yyyy. mm. dd",其中"yyyy"表示年份,"mm"表示月份,"dd"表示日.根据日期的比较要求,可精确到年或月或日,相应地,比较两个属性值的"yyyy"或"yyyy-mm"或"yyyy-mm-dd",若两个属性值相等,则两个日期型数据的相似度为 1,否则为 0.

④价格型相似度估算器.根据观察,现实电子商务网站中很多相同商品的原价相同,售价不同但很贴近,因此,价格类型属性的相似度的计算算法为:

假设 p_i 和 p_i' 是两条价格类型数据,则 p_i 和 p_i' 的相似度值是:

$$s(p_i, p_{i_i}') = 1 - \sqrt{\frac{(p_i - \bar{p})^2 + (p_{i_i}' - \bar{p})^2}{2}} \Big/ \bar{p}$$

其中,$\bar{p}$ 是 p_i 和 p_i' 的平均值.

综上,在重复记录识别过程中,根据不同领域数据的特征,可以组合出不同的相似度估算器,也可根据属性的特征采用不同的匹配算法,还可根据需要将新的相似度估算器和算法方便地加入到重复记录识别模型中.

6.2 相似度的整合

重复记录的识别主要是依据两个实体记录间相似度的值,因此,我们通过整合采用不同相似度估算器所计算得出的实体记录各属性值间的相似度来计算两条

实体记录间的初始相似度,其公式如下:

$$s(O_i, O_j) = \sum \omega_{attr-name} \times s(r_i, r_j)$$

其中,$s(O_i, O_j)$是两条实体记录的相似度,$s(r_i, r_j)$是相应属性值 r_i, r_j 依据指定的相似度估算器计算的相似度,$\omega_{attr\text{-}name}$是该领域全局模式中相应属性的权重.

对于来自不同数据源的实体记录,其与全局模式相匹配的属性值的个数各不相同,匹配的个数较多,则得到相似度值的可信度就越高,反之,则越低.因此,在最终确定两条实体记录相似度时,应考虑与全局模式相匹配的属性值的个数对相似度判定的影响,则两个异构记录 O_i 和 O_j 的相似度改为:

$$sim(O_i, O_j) = s(O_i, O_j) \times \frac{NUM_{attr}}{\max\{O_i.\ NUM_{attr\text{-}value}, O_j.\ NUM_{attr\text{-}value}\}}$$

其中,NUM_{attr}为两条实体记录属性值与全局模式匹配的属性个数,$NUM_{attr\text{-}value}$表示某个数据源的实体记录的属性值的个数,$\max\{O_i.\ NUM_{attr\text{-}value}, , O_j.\ NUM_{attr\text{-}value}\}$是两个实体记录属性值个数的最大值.

通过以上方法,可以得到来自不同数据源的实体记录间的相似度值.

6.3 重复记录识别

(1)不确定相似记录的处理

通过以上步骤得到两条实体记录的相似度后,可以根据专家提供的两个阈值(相似阈值 θ_1 与不相似阈值 θ_2)来判定两个实体记录间是否为重复记录.若实体记录间的相似度在相似阈值 θ_1 与不相似阈值 θ_2 之间,将它们放在待匹配属性值集合 R 中,再次计算实体记录间的相似度.具体算法如下:

①将待匹配属性值集合 R 中的所有属性值看作一个字符串,利用 Q-gram 算法计算两个实体记录 O_i 和 O_j 中的待匹配属性值集合 R 的相似度 $s(R_i, R_j)$.

②根据与全局模式匹配的属性值间的相似度 $\sum \omega_{attr\text{-}name} \times s(r_i, r_j)$ 和待匹配属性值集合 R 的相似度 $s(R_i, R_j)$,得到再次计算的实体记录 O_i 与 O_j 的相似度:

$$sim(O_i, O_j)' = \omega \times sim(O_i, O_j) + (1 - \omega) \times s(R_i, R_j)$$

其中,$\omega = \dfrac{\max\{O_i.\ NUM_{attr\text{-}value}, O_j.\ NUM_{attr\text{-}value}\}}{NUM_{G\text{-}attr}}$,$NUM_{G\text{-}attr}$是本领域中全局模式属性个数.

(2)重复记录集的确定

在计算完异构记录的相似度后,采用如下步骤比较实体记录间的相似度,并得到重复记录集,具体步骤如下:

①对于初始得到的结果数据(假设第一次提交的结果是从少量数据源抽取出的数目较少的记录)逐一比较,得到各实体记录间的相似度.

②在计算得到的各实体记录的相似度中,找出相似度值最大且相似度值大于相似阈值 θ_1 的两个记录,作为"图"中的两个结点(该图为有向右环图),并将两结点用带权的边连接,权值即为两个记录的相似度.按此方法向图中添加结点,直至所有结点都加入到了图中.这样,每个图为一个重复记录集,图中各结点为该重复记录集中的元素.

③对于之后查询到的结果记录,在初始查询结果的基础上,我们依据领域专家定义阈值和前两步得到的初始的由各重复记录组成的"图"进行处理.首先将后查询到的结果和各"图"中的特征结点(指"图"中出度最大的结点,若有多个结点的出度相同,则选择各边权值和最高的那个结点).进行比较,计算出它们之间的相似度,若相似度不小于相似阈值 θ_1,则将其加入与其相似度最大的结点所在的"图"中,若相似度在相似阈值 θ_1 与不相似阈值 θ_2 之间,则利用 6.3(1)中的算法再次计算相似度值,若比较后所有相似度都小于相似阈值 θ_1,则新生成一个"图","图"中结点为该实体记录.例如,图 4 中,初始得到 5 个结果记录有 $a_1, a_2, \cdots, a_5$,计算各记录间的相似度如图 4(a)所示,图 4(b)为最后的结果.由此得到重复记录集为$\{a_1, a_2, a_5\}$和$\{a_3, a_4\}$.

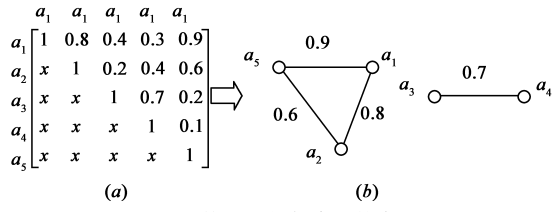

图4 异构记录相似度比较举例

7 实验

为了验证本重复记录识别模型的可行性和有效性,本实验采用了来自图书、电影和二手车两个领域的 60 个网络数据库中的数据,对比分析了只采用一种相似度计算方法和采用多种相似度估算器识别重复记录的实验结果.

7.1 数据集和评价标准

数据集:本实验采用了 30 个网上书店的网络数据库、15 个电影网站的网络数据库和 15 个在线二手车市场的网络数据库.我们通过向查询接口提交特定的查询来收集查询结果.在获得了来自不同网站的查询结果页面后,抽取出每条结果记录,对这些结果记录进行预处理,得到统一形式的实体记录.实验数据集分测试数据集和训练数据集.训练数据集主要用来确定实体属性值与全局模式属性间的匹配关系,并且估算特定

领域中全局模式各属性的权重;使用测试数据集来验证本模型的可行性和有效性.

评价标准:采用了信息检索领域中常用的两个评价标准:查准率(precision)和查全率(recall).在本实验中,查准率是指正确识别的重复记录的个数和识别出的重复记录的个数的比值.查全率是指正确识别的重复记录的个数和实际数据集中重复记录总数的比值.假设正确识别的重复记录的个数为 C,错误识别的重复记录的个数为 F,未识别出的重复记录的个数为 M,则在本实验中,查准率和查全率分别为 $C/(C+F)$和 $C/(C+M)$.

7.2 实验结果分析

分析并比较了只采用一种相似度计算方法和采用多种相似度估算器识别重复记录的实验结果.

(1)只采用一种相似度计算方法

图 5 给出了只使用一种相似度计算方法时所得到的实验结果,可以看出其查准率和查全率都要低于使用多种相似度估算器的方法.

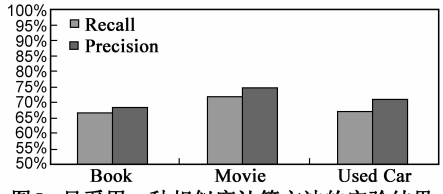

图5 只采用一种相似度计算方法的实验结果

(2)使用多种相似度估算器

图 6 给出了采用多种相似度估算器识别重复记录的结果.在本实验中,针对图书领域主要采用了文本型相似度估算器、日期型相似度估算器和价格估算器,在电影领域主要采用了文本相似度估计器和日期型相似度估算器,在二手车领域主要采用了文本型相似度估算器和价格相似度估算器.从实验结果中可以看出,本模型提高了整个识别的准确度,尤其是在图书领域.

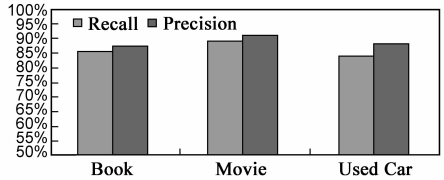

图6 采用多种相似度估算器的实验结果

(3)记录相似度特征

非重复记录和重复记录的相似度特征如图 7 所示,根据分析,采用本模型中的方法,重复记录的相似度主要集中在[0.4,0.7]之间,而非重复记录的相似度主要集中在[0,0.2]之间.因此,可以看出只有一小部分实体无法判断其是否重复.

(4)数据源数量的影响

图 8 给出了增加数据源数量时实验结果的改变.如图 8 所示,当数据源数量为 5、10、15、20、25 和 30 时,识别结果的查准率和查全率的变化情况.从实验结果中可以看到,实体记录的数量越多,所获得的实验的查准率和查全率就越高.

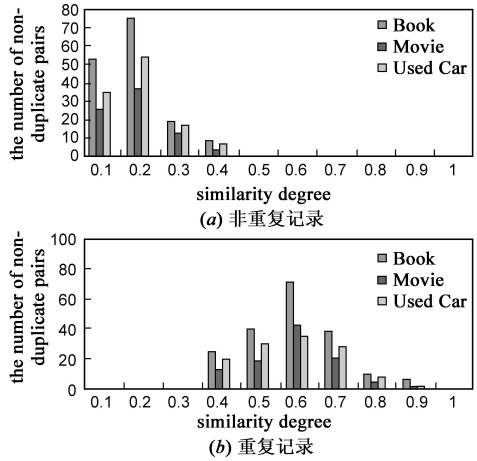

图7 非重复记录和重复记录的相似度特征

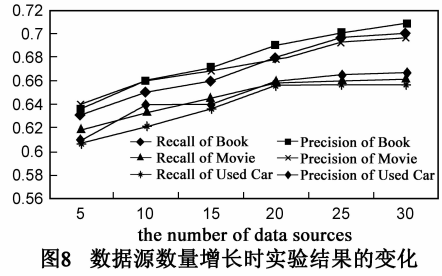

图8 数据源数量增长时实验结果的变化

8 结论

本文提出了一种重复记录识别模型.该模型主要针对来自不同 Deep Web 数据源的半结构化数据,采用多种相似度估算器相结合的方法来识别重复记录,获得了较好的准确度和有效性.

在未来的工作中,我们希望实现更多类型的相似度估算器,并且提高重复记录识别模型的自适应性和处理效率.

参考文献:

[1] Data Stage [EB/OL]. http://www.ardentsoftware.com/datawarehouse/datastage,2007.

[2] CoSORT[EB/OL]. http://www.iri.com/external/ dbtrends.htm,2007.

[3] WE Winkler. The State of Record Linkage and Current Research Problems [EB/OL]. http://citeseer.ist.psu.edu/255199.html,1999.

[4] S Tejada, CA Knoblock, S Minton. Learning domain-independent string transformation weights for high accuracy object identification[A]. Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Min-

ing[C]. New York: ACM, 2002. 350 – 359.
[5] M Weis, F Naumann. Detecting duplicate objects in XML documents[A]. Proceedings of the 2004 International Workshop on Information Quality in Information Systems[C]. New York: ACM, 2004. 10 – 19.
[6] K Zhang. A constrained edit distance between unordered labeled trees[J]. Algorithmica, 1996, 5(3): 205 – 222.
[7] LN liu, Y Kou, GS Sun, et al. Duplicate identification model for deep web[J]. Journal of Southeast university (English Edition), 2008, 24(3): 315 – 317.
[8] W Y Ma, JR Wen, F Lochovsky, et al. Instance-based schema matching for web databases by domain-specific query probing [A]. Proceedings of the Thirtieth International Conference on Very Large Data Bases[C]. New York: ACM, 2004, Vol. 30, 408 – 419.
[9] B He, K C-C Chang. Making, holistic schema matching robust: an ensemble approach[A]. Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining[C]. New York: ACM, 2005. 429 – 438.
[10] E Ukkonen. Approximate string matching with q-grams and maximal matches[J]. Theoretical Computer Science, 1992, 92 (1): 191 – 221.
[11] MS Waterman, TF Smith, W A Beyer. Some biological sequence metrics[J]. Advances in Math, 1976, 20(4): 367 – 387.
[12] MA Jaro. Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida[J]. Journal of the American Statistical Association, 1989, 84(406): 414 – 420.

作者简介:

申德荣　1964 年生于辽宁铁岭,东北大学教授、博士生导师.中国计算机学会高级会员.主要研究方向为分布式数据库、Web 数据管理和 Web 服务.
E-mail: shenderong@ise.neu.edu.cn

刘丽楠　1983 年生于沈阳,东北大学硕士研究生.主要研究方向为重复实体识别.
E-mail: everina520@163.com