计算机研究与发展, 卷43(增刊):46-53,2006 .（第23届中国数据库学术会议,广州.）

DEEP WEB 数据集成中的实体识别方法

凌妍妍 刘伟 王仲远 艾静 孟小峰

（中国人民大学信息学院 北京 100872）

（lingyy@ruc.edu.cn）

摘 要 互联网上存在着大量可访问的Web数据库，不同Web数据库之间存在着内容上的重叠。来自不同Web数据库的记录虽然在网页上的表现形式不同，但是可能描述的是同一实体。因此实体识别是 Deep Web 数据集成中数据合并过程里一个必不可少的环节，而且是一个很具有挑战性的工作。本文对这个问题进行了深入的探讨，提出了一种新颖的方法自动完成实体识别，该方法克服了传统的实体识别工作以模式匹配为前提的弊端，并且与领域无关。实验表明，这种方法在 Deep Web 环境下可以达到相当高的准确性。

关键词 Deep Web; Web数据库; 实体识别; 数据合并

中图法分类号 TP391

Entity Identification for Deep Web Data Integration

Ling Yan-Yan, Liu Wei, Wang Zhong-Yuan, Ai Jing and Meng Xiao-Feng

(School of Information, Renmin University of China, Beijing, 100872)

Abstract Nowadays, growing number of Web Databases emerge from the web with their contents duplicated. Two or more instances from different sources may refer to a single entity in the real world, though they are presented variously on WebPages. Therefore, entity identification is a crucial step in Web Databases integration but it's also a challenging task. In this paper, we have probed into this issue and proposed a novel automatic approach which is domain independent. Unlike traditional approaches, our approach is implemented without schema matching. The intensive experiments on real web sites show that the proposed approach can achieve high accuracies.

Key words Deep Web; Web databases; Entity identification; Data merge

1. 引言

随着 Web 飞速发展，其所蕴含的信息量也在急剧增长。整个 Web 按照信息深度的不同，可分为 Surface Web 和 Deep Web 两大类。根据 Brightplanet 在 2000 年发布的调查[1]，Deep Web 中包含的信息量超过 Surface Web 上千倍，而且这个比例仍在持续地上升。UIUC 大学在 2004 年的调查[2]对整个 Deep Web 的规模作了一次估计，结果表明目前 Deep Web 中可访问的 Web 数据库的数量超过了 45 万个。

为了能够有效利用 Deep Web 中丰富的信息，建立 Deep Web 数据集成系统成为了当前最迫切的需求。由于 Web 数据库的异质性和自主性，对从各个 Web 数据库中抽取结果的合并是一项十分具有挑战性的工作。为了对抽取结果进行清洗和去重，实体识别则是数据合并过程中的一个必不可少的环节。

作为 Deep Web 数据集成的一个应用，商品价格比较系统，要求把来自不同购物网站表示现实世界同一商品的信息识别出来并进行价格比较。以购书为例，如果我们希望从出售图书的电子商务网站（比如 Amazon 和 Bookpool）购买到最便宜的关于数据挖掘方面的图书，那么需要将来自这两个不同售书网站的查询结果中表示同一本书的记录识别匹配在一起。

传统的实体识别的工作中所提出的方法都是以模式匹配为前提的，即通过对实体之间在同一属性上的值进行比较来判断两个记录是否为同一实体。Web 中信息主要是以 Html 页面的形式发布，由于 Html 页面主要的作用是信息的表现，因此结构化程度很差，在这个前提下难以完成准确的模式匹配。本文在这个基础上，提出了一种在 Deep Web 数据集成环境下进行实体识别的方法。该方法不同于传统的实体识别方法，没有试图在结构化程度很差的 Html 文档上进行模式匹配，而是把每个结果记录看作一个文本文档，通过比较结果记录之间在文本上的相似性将表示现实世界中相同实体的结果记录对识别出来。此外，由于电子商务网站在 Deep Web 占有很大的比例，而在电子商务网站在结果记录中都包含有价格信息，因此我们针对这个特点在前一步的基础上进而把价格因素考虑进来，进一步提高了实体识别的准确性。通过实验表明，我们提出的这种方法在 Web 的环境下可以达到相当高的准确性。

本文其余部分组织如下：第 2 节阐述问题描述；第 3 节提出了 Deep Web 数据集成环境下自动实体识别的方法；第 4 节是相关工作的比较；第 5 节给出实验结果及分析。第 6 节是对全文的总结。

2. 问题描述

在关系数据库领域存在大量成熟的实体识别、数据清洗的工作，但是这些成果在 Web 环境下并不直接适用。因为不同于关系数据库中结构化的数据，Web 中的数据主要是通过网页发布的，因此是一种特殊的半结构化的数据。如图 1 中出现的记录，现存的工作很难以较高的准确性完成记录内部各数据项的分割，以及在各个数据项上的语义添加（Annotation）。所以以往基于结构化数据的实体识别方法并不能直接应用于 Web 数据集成这个新环境。

以购书为例（图 1）假设我们要在 Amazon 返回的 M 条记录和 Bookpool 返回的 N 条记录上进行匹配，单从某一项(比如书名)是不能保证一定能够判断出是不是同一本书的。因为在购书领域，即使存在同样的书名，也有可能是源自不同作者的不同的书。所以在一个领域，仅凭单独的某一个数据项并不能提供给我们足够的信息去判定实体间的匹配关系，而且在半结构化的 Web 数据上，现有的工作还无法将某个数据项（如书名）准确地提取出来。

在购书这个领域，书名和作者一起构成了唯一确定一个实体的主码，如果能在主码字段上进行类似 group by 的操作，似乎就能将表示同一实体的记录匹配在一起，但是我们不能忽略实体识别问题所处的 Web 大环境。首先，对于不同的领域，想要准确确定每一个领域（如航空领域，餐饮领域等）的主码，并非易事，而且，我们致力于产生一种一般性的领域无关的方法，试图能够适应 Web 上的不同领域。其次，即使我们能确定某个领域中唯一标识一个实体的主码，主码字段也不一定都在 Web 页面上返回的记录中出现（比如“作者”可能不出现在记录中，也许在 detail 页面中才可能找到，或者根本就不出现）。况且即使出现了的字段也还无法通过现有的工作准确地提取出来。于是，在我们的方法中，我们保留了半结构化数据本身的特点，在对来自两个网站的结果集中的数据进行匹配的过程中，综合考虑了出现在记录范围内的所有字段的内容，并采用了比较文本相似性的方法，在不需要进行传统模式匹配（Schema Match）和语义添加（Annotation）的基础上，有效地实现了 Deep Web 数据集成中的实体识别，而且是领域无关的。

3. Deep Web 数据集成中的自动实体识别

假设我们要在两个任意的网站 A 和 B 上建立实体之间的匹配关系，对于网站 A 中的每一个实体 A_i，我们都试图在另一个网站 B 中找到与之匹配的实体。于是在我们的方法中，我们计算 A_i 与 B 中所有可能匹配实体之间的相似度值，并考虑最大相似度值与阀值之间的关系，从而判断 B 中与 A_i 相似度值最大的实体是否就是真正与 A_i 匹配的实体。

在上述基础之上，我们通过以下 3 个步骤来解决 Deep Web 数据集成中实体识别的问题：

a) 记录块划分
b) 相似度计算
c) 迭代的训练

3.1 记录块划分

通过观察来自任意一个网站的查询结果列表，可以发现在每一条记录中，不仅包含被描述实体本身的信息，也包含了一些元数据（Metadata）信息。以图 1（A）为例，其中“ourprice:”“published”“you save”等都是显示记录时用到的一些描述性模版信息，不是在 Web 数据库中存储的实体本身的信息，因此这些元数据信息对于实体本身的识别并不起到积极作用，反而由于它们在每个记录中同样出现，会干扰实体间相似度的判别。所以我们要事先将这些元数据信息当成类似于 IR 中的阻止词（Stop Words）进行去除。

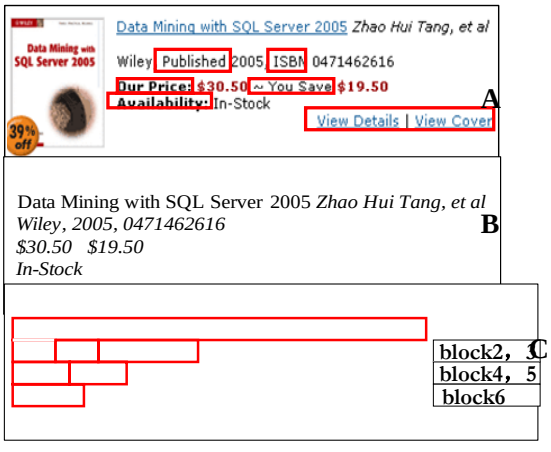


图 1 记录块划分示例

现在图 1（B）所示记录中只保留了进行实体识别所必需的实体本身的信息。如果我们将整个记录看作是描述实体的一段文本，然后在文本间进行相似度的比较，从而确定实体间的相似度，这样的方法准确度低，错误率高，因为它忽略了一个很重要

的因素，那就是记录文本中各个部分在决定实体是否匹配的过程中，发挥着不同的重要性。直观地看，决定两个记录是否表示同一本书，“书名”起到的作用似乎比“图书所属分类”起到的作用要大。如果采用传统的在记录内部划分语义块的方法，其复杂度高，将会成为整个系统效率的瓶颈。

我们采用了一个比较巧妙的办法来解决这个问题，由于 W3C HTML 规范定义了 93 个标签，其中有些标签(如 TABLE、P、DIV、SPAN 等)是用来将网页进行布局、划分为语义上的结构的。我们利用这类布局标签来对记录进行语义块划分不仅效率高，而且能使整体的准确性达到很高的水平。图 1（C）给出了基于布局标签的划分结果——图 1（A）中的记录被划分为 6 块。

3.2 相似度计算

Deep Web 数据集成中，对任两个网站返回的结果进行实体识别，我们需要一种合理的方法衡量来自不同网站的两条记录所代表的实体之间的匹配程度。3.1 节中已经介绍了基于标签对记录块进行划分，每个划分之后的语义块在决定实体之间匹配程度的重要性上存在着一定的区别，于是我们采用一组权值来量化地表示这种区别。对于图 1(c)中划分得到的 6 个语义块，我们分别赋予权值 W_1，W_2…W_6。在计算此记录与来自另一网站的记录之间的相似度值时，问题就转化为计算此记录中各语义块与另一记录之间相似度值的加权相加之和。

定义 1（实体相似度） 实体 A 和实体 B 的相似度值等于实体 A 内各语义块（设为 m 块：$A\text{-}block_1$，$A\text{-}block_2$……$A\text{-}block_m$）与实体 B 之间相似度值的加权相加之和。其中 W_i 代表语义块 $A\text{-}block_i$ 在决定实体匹配程度上的重要性。

公式 1： $$S(A,B)=\sum_{k=1}^{m} W_k \times S(A\_block_k, B)$$

具体到实体 A 内的各语义块，我们并不试图在与之比较的实体 B 内部进行划分，从而找到相匹配的块进行对应块之间相似度计算，这是因为：(1) 在结构性很差的 Html 文档之间进行模式匹配的工作难度大，而且一旦在我们的方法中引入模式匹配出现匹配错误时，会严重影响实体识别工作的准确性。(2) 我们的方法以简单高效的基于标签划分语义块为基础，不同于严格意义上划分记录的各个属性，加剧了模式匹配工作的难度。因此，为了提高实体识别效率，避免由于引入模式匹配造成的错误匹配，我们采用了基于文本比较的标准 TFIDF 余弦-相似度计算的方法[5]计算语义块与另一个记录的相似度值。

在 Deep Web 数据集成的应用中，电子商务的网站占到了很大比例。在处理这类网站的实体识别问题时，价格可以看作是一个很有价值的线索，因为不同电子商务网站出售的同一种商品在价格上一般是非常接近的。对于价格这种特殊的数据类型，标准的计算文本相似度的方法是不适用的，我们需要一种新的衡量标准来计算价格之间的相似度。

首先，在记录块内部我们要将价格正确地识别出来。不难发现，价格在页面上的出现必然伴随着一些特殊的前缀或后缀信息，如“￥”,“＄”,“Price”，“价格”,“元”等。通过识别出这些有限的前、后缀信息，就能很容易地将出现在记录块内部的价格识别出来。其次，考察两个价格之间的匹配程度，绝对的数值差异是不恰当的，我们需要考察数值的相对差值。如$28 与$30 绝对差值是$2，而$2800 与$3000 绝对差值是$200，价格的匹配程度不能由绝对差值来衡量。差异系数可以很好的解决这一问题。

定义 2（价格的相似度） 价格 A 与价格 B 的相似度值 Sp 等于价格 A 与价格 B 的差异系数的补值：$S(P_1,P_2)=1-DC=1-\left(\frac{\sqrt{(P_1-\overline{P})^2+(P_2-\overline{P})^2}}{\overline{P}}\right)$ **公式 2**

其中，DC(Differential Coefficient)指的就是两个价格 P_1 和 P_2 的差异系数，式中的 $\overline{P}$ 指的是两个价格 P_1 和 P_2 的平均值。至此，利用上述实体相似度的定义（定义 1），来自两个网站的记录两两之间的相似度值可由各语义块相似度值加权相加得出。于是，如何量化地衡量各语义块不同的权值，即它们在决定实体匹配性中不同的重要性，就成为了一个关键问题。下面将详细介绍在我们的方法中，是如何通过积极有效的迭代训练得到这组关键权值的。

3.3 迭代的训练

在 Deep Web 数据集成的过程中，对任意两个网站进行实体识别的工作。在已知不同记录之间的相似度值计算方法的前提下，有两组未知数亟待确定：

[1]. 用一组量化的权值来衡量各语义块在实体匹配过程中不同的重要性。其中，每个语义块对应一个权值；

[2]. 用一组阀值来衡量实体匹配的最终结果。依据这组阀值，我们可以将经过相似度计算的两个实体划分到不同的类别中去——依据匹配程度由高到低，分为三类：匹配/疑似匹配/不匹配。

3.3.1 训练样本

我们试图在两个给定的网站上（网站 A 和网站 B）建立实体之间的匹配关系，首先我们在这两个网站上手动选取出 N 对匹配的记录（彼此描述同一实体）作为训练样本。假设 A_1，A_2……A_n 是来自网站 A 的 N 个样本记录，同样 B_1，B_2……B_n 是来自网站 B 的 N 个样本记录，相应地，A_i 与 B_i 是匹配的记录对，于是 A_i 与 B_j（$j \neq i$）都是不匹配的记录对。总的来说，在我们的训练样本中，来自网站 A 的每一个记录 A_i，在 B 中都能唯一地找到一个与之匹配的记录 B_i，而与 B 中剩余的 $N-1$ 个记录都是不匹配的。

3.3.2 权值的确定

在对分别来自网站 A 和 B 的记录 A_x 和 B_y 进行相似度值计算的时候，先将 A_x 进行语义块划分，并给每一个划分得到的语义块赋予不同的权值，这里假设得到 M 个块分别对应 M 个权值（W_1，W_2……W_m）。在我们的训练样本中，对于每一个 A_i，在网站 B 中都能找到一个与之匹配的 B_i，以及 $N-1$ 个与之不匹配的 B_j（$j \neq i$）。可以肯定的是，A_i 与匹配记录 B_i 之间的相似度值一定大于 Ai 与不匹配记录 Bj（$j \neq i$）之间的相似度值。于是对于来自网站 A 的每一个 A_i，都能得到如下的一组不等式：

公式 3

$$\begin{cases} S(A_i, B_i) \geq S(A_i, B_1) \\ \cdots\cdots \\ S(A_i, B_i) \geq S(A_i, B_{i-1}) \\ S(A_i, B_i) \geq S(A_i, B_{i+1}) \\ \cdots\cdots \\ S(A_i, B_i) \geq S(A_i, B_n) \end{cases}$$

同时，记录 A_i 与 B_j 之间的相似度值又可以由 A_i 中各语义块与 B_j 的相似度值加权相加得到，于是我们利用相似度计算公式（公式 1）对公式 3 中的每一个不等式进行展开。对于训练样本中网站 A 的每一个记录 A_i，都能得到以 W_1，W_2…W_m（M 个语义块对应的 M 个权值）为未知数的一组 $N-1$ 个不等式。而网站 A 中共有 N 个记录，于是一共能得到 $N*(N-1)$ 个不等式。实际情况中，不等式的个数如果像这样按照 N 的指数倍增长，那将是不可忍受的。于是我们观察由每一个 A_i 得到的 $N-1$ 个不等式，只需保证 A_i 与匹配记录 B_i 之间的相似度值大于 Ai 与所有不匹配记录 Bj（$j \neq i$）之间的最大相似度值，就可以保证 A_i 与匹配记录 B_i 之间的相似度值一定大于 Ai 与每一个不匹配记录 Bj（$j \neq i$）之间的相似度值。于是，对于每一个 Ai，如下的一个不等式足以保证上述 $N-1$ 个不等式成立：

公式 4 $$S(Ai, Bi) \geq MAX\{S(Ai, Bj)\} \quad (i, j = 1, 2..., n; j \neq i)$$

相应地，对于网站 A 的 N 个记录，不等式数量由原来的 $N*(N-1)$ 个迅速降为 N 个。

我们知道，对于一个不等式组，它的解集对应的是空间中的一个凸多面体。每一个未知数（在这里就是 W_1，W_2…W_m）对应解空间里的一维。我们取其取值范围的平均值作为相应权值的解，最终可以保证这个 M 维的向量必然是解空间中的一个解。初始计算出来的每个权值往往都落在一个很大的取值范围内，因此并不能精确地量化反映出每个权值对应的语义块在决定实体是否匹配重要性上的权重。之后，我们通过迭代训练的方法来不断地将每个权值都缩小在一个更精确的范围内。

3.3.3 阀值的确定

根据上述由训练样本得出的初始不等式组，我们得到一组初始的权值。由公式 1 中实体相似度的定义，实体 A_x（来自网站 A）和实体 B_y（来自网站 B）之间的匹配程度，可以根据这组权值，对实体 A_x 内各语义块与实体 B_y 的相似度值进行加权相加。于是，对于训练样本中任两条来自不同网站的记录都可以计算出相似度的值，共有 $N \times N = N^2$ 个匹配值。其中，每一个匹配值对应一个可能的组合。

我们知道在训练样本中 N^2 个可能的组合里，有 N 个组合被认为是匹配的，另外的 $N \times (N-1)$ 个组合是不匹配的，因为对来自网站 A 的每一个记录 A_i，在 B 中能且只能找到一个与之匹配的记录 B_i。图 2 中对于每一个 Ai 我们在 0-1 的数轴上标示出它与 B 中每一个实体的相似度的值（介于 0 到 1 之间）。圆圈表示 Ai 同与之匹配的实体 Bi 之间的相似度，方块表示 Ai 同不匹配实体 Bj（$j \neq i$）之间的相似度，代表匹配实体的圆圈必然在每个数轴上最接近 1。N 个匹配的组合对应于图中的 N 个圆圈，我们将这 N 个匹配值中的最小值作为阀值 ***T1***（虚线），同样 $N \times (N-1)$ 个不匹配的组合对应于图中的 $N \times (N-1)$ 个方块，再将这 $N \times (N-1)$ 个匹配值中的最大值作为另一个阀值 ***T2***（实线）。如图，训练样本中的 N^2 个组合就被这两个阀值成功地划分为两类：匹配/不匹配。对于样本以外等待判断的两个实体，相似度值超过阀值 ***T1*** 被认为是匹配的；相似度值小于阀值 ***T2*** 被认为是不匹配的；而相似度值介于两个阀值之间的就被认为是疑似匹配的，还需要用户手动来判断。

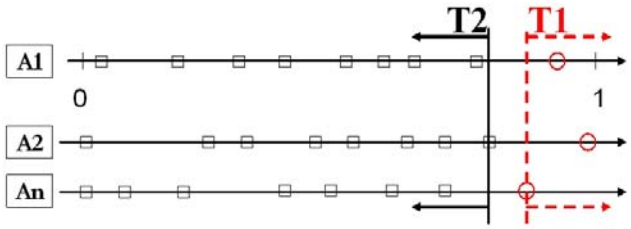


图 2 不交叉情况

图 3 交叉情况

我们考虑图 3 中的情况。图 3 在图 2 的基础上对代表相似度值的数轴上的点进行了一些调整，在每个 *Ai* 对应的数轴上仍然满足代表匹配实体的圆圈比代表不匹配实体的方块更接近 1。但是综合来看所有的实体对应的数轴，我们并没有保证所有 *N* 对匹配样本的相似度（圆圈表示）都大于所有 $N\times(N-1)$ 对不匹配样本的相似度（方块表示），即出现图中所示不匹配记录对 A_2 和 $B_y(y\neq 2)$之间的相似度 $S(A_2,B_y)$ 反而大于匹配记录对 A_1 与 B_1 之间的相似度 $S(A_1,B_1)$ 的情况。在这种情况下，训练样本中的 N^2 个组合就无法被这两个阀值 ***T1***（虚线）和 ***T2***（实线）成功地划分为两类：匹配/不匹配。相似度值落在交叉区域内的两个实体在是否匹配上具有二义性。

因此，一旦原先的不等式组得到的一组权值导致样本记录之间的相似度值在数轴上出现了交叉情况，我们就要重新对不等式组做出限制，使得调整之后的权值不会导致二义性价差区域的出现。图 3 中我们找到造成交叉情况出现的两个点——最小匹配样本匹配值 $S(A_1,B_1)$和最大不匹配样本匹配值 $S(A_2,B_y)$——我们将 $S(A_1,B_1)\geqslant S(A_2,B_y)$作为一个新的限制加入到原不等式组中去。这样就保证了新解出来的一组权值必然使得这个新不等式成立，于是造成交叉的原因得到了破除。

3.3.4 迭代训练器

至此，我们成功地得到了一组阀值去衡量实体匹配的最终结果。依据最小匹配样本相似度值 ***T1*** 和最大不匹配样本相似度值 ***T2***，我们可以判断经过相似度计算的两个实体是否匹配。但是，我们对权值的训练还没有停止。通过以权值作为未知数的不等式组可以得到空间中的一个凸多面体作为解空间，每个权值对应解空间中的一维。我们希望通过迭代训练的方法尽量使解空间缩小到一个更精确的范围内，并最终趋于稳定。

在原不等式组中，我们观察每一个不等式的含义。不等式左边是匹配的记录，右边是不匹配的记录，整个式子表示匹配记录对的相似度值大于不匹配记录对的相似度值。这个条件只是表示了大小关系，没有量化地衡量它们之间匹配值的差距。观察图 2 可以发现，当保证了匹配样本序列（所有的圆圈）与不匹配样本序列（所有的方块）不交叉时，匹配记录对和不匹配记录对之间的匹配值就至少相差了 ***T1-T2***（两个阀值的间距）。我们将这个新的限制加入到原不等式组（公式 3）中，得：

公式 5

$$\begin{cases} S(A_i,B_i)\geq S(A_i,B_1)+(T_1-T_2) \\ \cdots\cdots \\ S(A_i,B_i)\geq S(A_i,B_{i-1})+(T_1-T_2) \\ S(A_i,B_i)\geq S(A_i,B_{i+1})+(T_1-T_2) \\ \cdots\cdots \\ S(A_i,B_i)\geq S(A_i,B_n)+(T_1-T_2) \end{cases}$$

由于每一个不等式中都加入了匹配值差距至少为 ***T1-T2*** 的约束，产生出来的一组新的权值 W_1，$W_2\ldots W_m$构成的解空间逐渐被控制在一个比较精确的范围内。通过这组更精确反映各语义块重要性的权值，我们又可以重新计算样本中匹配样本序列和不匹配样本序列的匹配值，投影到数轴上获得一组新的阀值——***T1*** 和 ***T2***。重复上述过程，将新的限制 ***T1-T2*** 作为新的约束重新规范不等式组，产生新的一组权值，将解空间控制在更精确的范围内。这样循环反复的过程就是迭代训练的方法，直到两阀值 ***T1*** 和 ***T2*** 趋于稳定，这时得到的权值是对各语义块在决定实体是否匹配中重要性的真实反映。

4. 相关工作

在传统数据库领域，实体识别工作也被称为数据清洗和去重。[6，7]就是在同一个表内寻找等价的元组，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程度。这些工作都是在具体的领域上开展的，扩展性差而且代价很高。在 Web 环境下同样也存在一些工作试图将不同数据源提供的数据匹配起来。[8]是较为完善的指导在多个异质数据源上如何进行实体识别工作的方法，它提出了多种减少匹配代价提高匹配效率的策略，但是在结构化很差的 Web 数据上很难直接应用；[9]提出了一

系列被赋予权值的字符串转换规则，并将其应用在共享属性上从而进行实体是否匹配的判别，这是以模式匹配和发掘共享属性为前提的。在[10]中提出了一种 PROM 的方法，利用专家制定的或实际中发掘出来的各属性间的限制来协助实体识别工作，但是针对 Web 的各个领域，完整地制订出不同属性间依赖关系的规则库对开发者来说是一个很大的负担。而我们的工作一个很突出的优点就是能自动地学习并获取针对不同数据源的不同的权值，尽可能减少用户参与，而且准确性很高。具体到 Deep Web 数据集成环境下，鉴于不同数据源的异质性和自主性，模式匹配是很难进行的工作。我们的方法则是在模式匹配关系未知条件下进行实体识别的首次尝试。

5. 实验

为了验证本文所提出实体匹配方法的正确性，我们利用 Lucene(基于 Java 的全文索引引擎工具包)实现了一个研究原型，并在本节给出了实验结果。

5.1 数据集

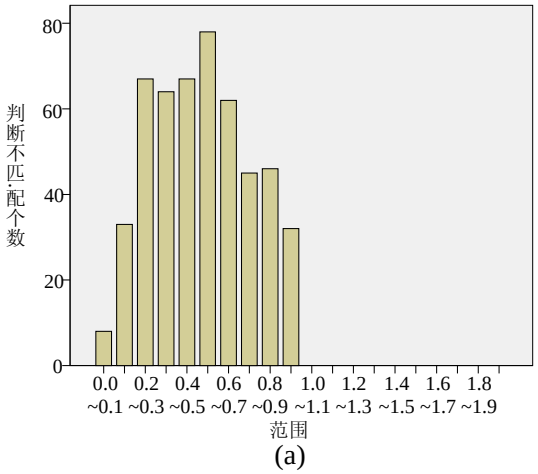
本实验的数据集在图书这个主题下从大家熟悉的 Amazon 和 Bookpool 两个购书网站获取。我们通过对它们的查询接口提交 8 个特定查询的方式获取数据记录。对每个查询，从两个网站的查询结果中的起始处各选取大致相同的记录数目。数据集共分为两类：训练集和测试集。训练集用来选取样本训练权值和阀值，包括来自 3 个查询的记录，每个查询在两个网站各选取 40 个记录。测试集用来验证本文所提出方法的正确性，包括来自 5 个查询的记录，

该数据集具有以下几个特点：第一，同一个查询下两个网站的查询结果具有较高重复比例；第二，足够规模，整个数据集超过 2000 个记录；第三，查询之间相互独立，多个互不相同的查询保证彼此查询结果基本没有交叉。基于这三个特点，该数据集可以保证实验结果的客观性。

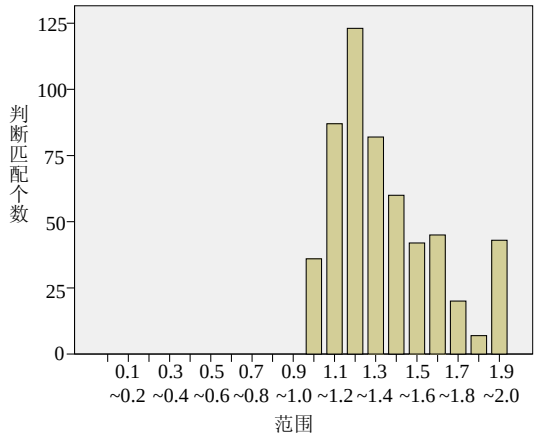
5.2 评价标准

- 匹配准确率：判断为匹配的所有记录对中判断正确的比例；
- 不匹配准确率：判断为不匹配的所有记录对中判断正确的比例；
- 总体准确率：所有进行判断的记录对中判断正确的记录对所占的比例，包括匹配与不匹配的记录对；
- 无法判断率：所有进行判断的记录对中无法判断的记录对所占的比例。

5.3 实验分析



(a)



(b)

图 4 相似度值的分布

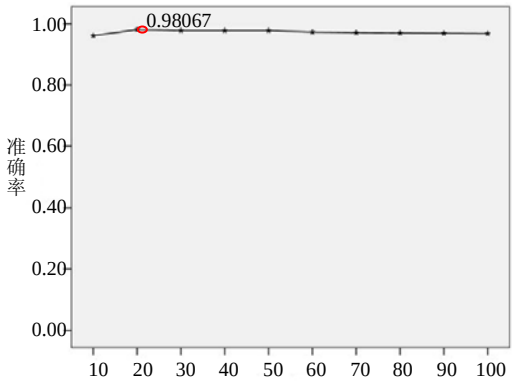


图 5 样本数量与准确率之间的关系

我们首先在训练集上进行训练得到记录中每个块的权值，然后计算得到匹配阀值与不匹配阀值。根据匹配阀值与不匹配阀值，所有记录对相似值集合可分为三个部分：不匹配记录对(小于不匹配阀值)、匹配记录对(大于匹配阀值)和无法判断记录对(介于不匹配阀值和匹配阀值之间)。将每个实体与另一个网站的所有实体的相似度值从大到小排序，取其最大值，将其投影到数轴上进行观察。从图中可以观察

到记录对的相似度值分布的两个明显的特征：第一，匹配记录对的相似度值主要聚集在 1.0-1.5 之间，而不匹配记录对则主要集中在 0.2-0.7 之间；第二，只有很小比例的记录对被判断为无法判断。

表 1 是实验结果的详细数据。从表中可以得出三个结论：第一，准确性高，在四个标准上都达到了比较理想的效果，匹配准确率、不匹配准确率和总体准确率都在 98%以上；第二，人工干预量小，对于无法判断的记录对需要进行人工判断，而无法判断率仅有不到 2%；第三，稳定性好，不同的数据集在匹配准确率、无法判断率和总体准确率相差很小，在不匹配准确率上只有 xml 与其它相差略大。总的来说，我们所提出的自动的判断方法能够达到令人满意的效果。

	匹配正确数量	匹配准确率	不匹配正确数量	不匹配准确率	无法判断数量	无法判断率	总体正确数量	总体准确率
Database	95	0.9895833	91	1	4	0.0209424	190	0.9947644
Linux	139	0.972028	58	0.983050	2	0.0098039	199	0. 97549
OS	121	0. 968	73	0.986486	5	0.0245098	199	0. 97549
Software	88	0.9777778	111	0.991071	2	0.0098039	201	0.9852941
xml	98	0. 989899	73	0.948051	4	0.0222222	175	0. 972222
总计	541	0. 9783	406	0.983051	17	0.017294	964	0. 980671

表 1 具体实验数据

另外在实验中我们发现了一个有趣的问题：(整体)准确率与训练集的数量并非正相关的。图 5 给出了二者之间的关系，从图中可以发现训练集数量在 20 附近准确率达到峰值，然后会缓慢下降。这说明训练样本数过多过少都不好，关键在于样本选取得当。训练样本数过少，无法真正体现各 block 的重要性；训练样本数过多，可能出现一些干扰不等式，影响权值及阀值的确定。

综上所述，我们所提出的自动的实体识别方法能够在较少训练集的情况下达到非常高的准确性和极低的人工干预量。

6. 总结

传统实体识别工作都是在良好的模式匹配前提下实现的，而在 Deep Web 环境下模式匹配仍未得到很好的解决。本文提出了一种在 Deep Web 环境下进行实体识别的方法。该方法无需以模式匹配为前提，通过对表示实体的记录进行划分和比较记录对之间文本的相似性来达到实体识别的目的。实验结果表明，这种方法可以达到非常高的准确性。

7. 参考文献

[1] http://www.brightplanet.com/technology/DeepWeb.asp

[2] Kevin Chen-Chuan Chang, Bin He, Chengkai Li, Mitesh Patel, Zhen Zhang: Structured Databases on the Web: Observations and Implications. SIGMOD Record 33(3): 61-70 (2004)

[3] W.Frakes and R. Baeza-Yates. Information retrieval: Data structures and algorithms, Prentice Hall 1992.

[4] William W. Cohen: Integration of Heterogeneous Databases Without Common Domains Using Queries Based on Textual Similarity. SIGMOD Conference 1998

[5] Sunita Sarawagi, Anuradha Bhamidipaty: Interactive deduplication using active learning. KDD 2002

[6] E.Winkler, W.: The state of record linkage and current research problems. In: Proceedings of of the Survey Methods Section. (1999)

[7] Sheila Tejada, Craig A. Knoblock, Steven Minton: Learning domain-independent string transformation weights for high accuracy object identification. KDD 2002

[8] Doan, A., Lu, Y., Lee, Y., Han, J.: Object Matching for Information Integration: A Profiler-Based Approach. IIWeb 2003

凌妍妍，女，1985 年生，硕士研究生，研究方向：Deep Web 数据管理。

刘伟，男，1976 年生，博士研究生，研究领域：Deep Web 数据管理。

王仲远，男，1985 年生，本科三年级。

艾静，女，1985 年生，本科三年级。

孟小峰，男，1964 年生，教授（博导），研究领域：Web数据管理，XML数据库，移动数据管理。