第 34 卷 第 13 期 计 算 机 工 程 2008 年 7 月
Vol.34 No.13 Computer Engineering July 2008
·软件技术与数据库· 文章编号：1000—3428(2008)13—0051—03 文献标识码：A 中图分类号：TP311

基于规则集的 Deep Web 信息检索

杨巨峰[1]，史广顺[1]，赵玉娟[1,2]，王庆人[1]
(1. 南开大学机器智能研究所，天津 300071；2. 天津市气象信息中心，天津 300074)

摘 要：提出一种基于规则集的新型 Deep Web 信息检索模型。该模型包含 4 个层次，主要处理环节如任务分派、信息提取、数据清洗等引入了 Deep Web 特有的结构规则、逻辑规则和应用规则协助工作。把该模型应用于科技文献检索、电子机票定购和工作简历搜索 3 个领域，实验结果证明该模型灵活、可信，有效信息查全率达到 96%以上。

关键词：信息检索；深层网络；规则集；数据提取

Rules-based Deep Web Information Retrieval

YANG Ju-feng[1], SHI Guang-shun[1], ZHAO Yu-juan[1,2], WANG Qing-ren[1]
(1. Institute of Machine Intelligence, Nankai University, Tianjin 300071; 2. Tianjin Meteorological Information Center, Tianjin 300074)

【Abstract】This paper proposes a novel rules-based model to extract data from Deep Web pages. The model comprises four layers, main processing parts as task allocation, information extraction, data cleaning which work based on the rules of structure, logic and application. It applies the new model to three intelligent system, scientific paper retrieval, electronic ticket ordering and resume searching. Experimental results show that the proposed method is robust and feasible.

【Key words】information retrieval; Deep Web; rules set; data extraction

1 概述

网络上大部分内容不能通过静态链接直接获取，特别是大部分隐藏在搜索表单之后的页面只有通过用户键入一系列关键词才可以获得。与Surface Web相比，Deep Web中蕴含的信息量达到其 400 倍~500 倍，访问量高出 15%，而且数据质量相对更高[1]。

随着网络技术的发展，Deep Web信息检索技术成为研究的热点。这类研究致力于帮助人们自动地获取并利用自由分布在Deep Web中的丰富信息。一些论文探讨从Internet上发现Deep Web数据库的技术[2]，另一些则研究从查询接口中分析和提取属性并构建统一模式的方法[3]，这项研究有助于集成多个Deep Web，并向用户提供访问异构站点和数据库的统一途径。Deep Web研究的另一个重要领域是数据提取，即将用户感兴趣的信息从半结构或无结构的Web页面上抽取出来，并保存为XML文档或关系模式[4-5]。目前，研究者开始关注语义信息对Deep Web的影响[6]。

上述文献的研究覆盖了 Deep Web 信息检索的各主要环节。但研究者在讨论理论模型和理想化的算法时，往往忽略了相关技术应用于实际时可能遇到的问题。这些问题包括：在驱使爬虫访问 Deep Web 时，如何选择最有可能得到理想结果的目标站点集合；如何优化现有解析方法使其面对结构各异的页面时通用并且鲁棒，因为这些页面通常是使用不同技术构建的；对获得的 Deep Web 数据做怎样的处理使其能更好地应用于实际。

2 基于规则集的 Deep Web 信息检索模型

本文在前人工作的基础上，针对 Deep Web 信息检索技术在特定领域的应用，设计了一种基于规则集的 Deep Web 信息检索模型。利用领域知识和接口相似度两种规则对用户提出的查询任务进行分派，选择访问最有可能得到有效结果的一组 Deep Web 站点。以所得页面的结构特征为基础提取数据，并利用逻辑规则协助校正。最后对原始数据集进行清洗和排序，使其符合应用习惯并可以被用户接口直接使用。

2.1 模型结构

本文提出的模型自下而上依次包含 4 个主要层次，如图 1 所示。

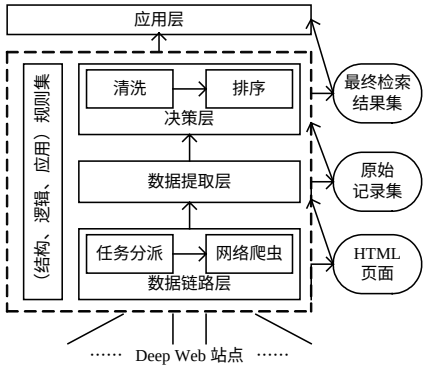

图 1 基于规则集的 Deep Web 信息检索模型

(1)数据链路层：处于模型的最底层，是模型与 Deep Web 站点之间的通信接口。任务分派模块负责选择相关度最高的目标网站，网络爬虫并行实施访问，并获取初始结果页面。

(2)数据提取层：分析 HTML 结果页面的结构特征和逻辑特征，确定各单元数据的含义，生成原始数据集。

(3)决策层：对上一步生成的原始数据进行清洗得到有价

基金项目：天津市自然科学基金资助项目(05YFJMJC01500)
作者简介：杨巨峰(1980－)，男，博士研究生，主研方向：Web 信息检索，模式识别；史广顺，副教授；赵玉娟，助理工程师；王庆人，教授、博士生导师
收稿日期：2008-03-20 **E-mail：**mcward@yahoo.cn

值的最终数据，根据用户的查询倾向对记录排序。

(4)应用层：处于模型的最高层，是模型与操作者的接口，负责接收用户的查询请求；并将结果以被期望的形式呈现出来，供用户查看或进一步处理使用。

2.2 规则集

通过对大量 Deep Web 页面进行学习，总结了一系列用于信息检索的规则。这些规则分为 3 类：

(1)结构规则：在日常浏览网页时，通常是通过观察页面元素的位置以及相互关系来分析和理解整个页面的含义。本文使用下面的结构规则协助数据提取工作：

1)描述信息一般位于控件的左边或上边；

2)当描述信息位于控件内部，即作为元素值域中的一项时通常表示一个默认值；

3)描述信息对应的 HTML 源码一般与输入控件对应的源码左右相邻；

4)表格的某一整行都是描述信息而且数据单元位于描述信息的正下方时，同一列的描述信息与控件及数据单元相互对应；

5)表格的某一整列都是描述信息而且数据单元位于描述信息右侧时，同一行的描述信息与控件及数据单元相互对应。

(2)逻辑规则：是一些人们公认的知识，这些有具体逻辑意义的规则被用来在单元定位和数据抽取时协助验证。

(3)应用规则：因为模型中提到的 1～3 层都是基于应用层构建并为其提供服务的，所以在任务分派、数据提取和决策层引入具有领域特点的规则可以提高这些工作的效率。

在模型中，上述规则被应用于信息检索的各个环节，其中前 2 类规则是在训练阶段由机器学习得到的，应用规则一般由设计者规划和指定。图 2 显示了一个航空机票页面上包含的几种规则。

图 2　Deep Web 中出现的各种规则

3　数据链路层

3.1　任务分派模块

对于一条单独的查询请求来看，并非所有的候选 Deep Web 站点都包含符合条件的结果。在爬虫出发前预测一个可能包含有效结果的目标站点集合能明显减少系统的时间和空间开销，提高爬虫效率。

本文把应用规则和查询接口相似度 2 项指标结合起来进行预测：(1)利用先验的领域知识对候选站点进行第 1 次筛选，用户填写的特定查询条件可以从候选队列中去除一些明显不符的站点。(2)在获得的新候选站点集合中，逐一比较用户查询条件与 Deep Web 站点查询接口之间的相似程度，选择那些相似度最高的候选网站进行查询。

在查询接口中，有意义的信息包括语义描述、数据源控件的类型和值域等。本文定义 2 个接口 A, B 的相似度为

$$S_{AB}=\sum_{k=1}^{\min\{i,j\}}\omega_{Ak}\omega_{Bk}\Big/\sum_{k=1}^{\max\{i,j\}}\omega_k^2 \tag{1}$$

其中，ω 代表接口中不同查询条件的权重，其定义如下：

$$\omega_i=T_i+H_i+\sum^{M}FREQ \tag{2}$$

其中，T_i是A_i的语义描述与领域知识库的匹配度；H_i表示A_i的数据源控件类型；$FREQ$表示各控件取值的频率特性。

另设 Θ_s 为预定义的查询任务阈值，有

$$\lambda_{si}=\begin{cases}1 & S_{0i}>\Theta_s\\ 0 & \text{others}\end{cases} \tag{3}$$

当 $\lambda_{si}=1$ 时，表示访问第 i 个 Deep Web 站点可能查得有效结果。为了获得最好的时间效率，服务器并行访问一组这样的有效站点。

3.2　网络爬虫

本文使用传统的爬虫技术对 Deep Web 站点进行访问，同时考虑了以下 2 种特殊情况：

(1)某些网站为了跟踪用户，设置了 Session 信息并将其放置在服务器地址中，这一串随机数字造成了服务器地址的不确定性。本文对于爬虫的目标地址采取动态构造的方式，由可变信息和固定信息 2 部分组成，固定信息记录训练阶段搜集的站点静态地址，动态信息则在每次爬虫访问某一站点时用实时获得的数据赋值。

(2)某些网站在表单提交之后返回的并不是包含响应信息的结果页面，而是一个中间页面。这种中间页面可能是为了改善查询等待期间的用户体验，也可能是用于执行更新查询参数的操作。由于这些中间页面一般都有特殊的描述信息，因此扩充了应用规则，令爬虫进行 2 步交互：第 1 步检测和获取中间页面；第 2 步从这样的中间页面中提取所需的信息并构造 2 次查询请求，进而获取最终结果页。

4　数据提取层

本文使用 HTML 页面的结构规则提取数据，然后利用逻辑规则和应用规则对获得的结果进行校正。根据训练发现，Deep Web 页面文件中的有效信息大都存放在结构化的区域内。将 HTML 页面转换为一棵结构树后，中间节点一般记录了文档结构，而页节点则存放着数据。因此，本文利用下述树匹配算法从页面中抽取信息，该算法从 2 棵树的叶结点出发进行比对，共分为 5 步，如图 3 所示。

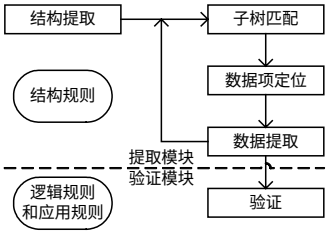

图 3　数据提取层的处理流程

提取模块：

(1)从第 1 个叶结点开始，依次比较 2 棵树的同层兄弟结点的个数，如果匹配成功则取其上层结点进行递归匹配。直至遇到某个节点其类型属性与预定义的 TR 结点一致时，上述过程停止。

(2)若子树比对成功，将所有经过比对的 HTML 表格树中的子树结点标记为匹配结点；否则，标记为为非匹配结点。

(3)对匹配成功的子树重复上述过程，进行子树内信息项定位。

(4)使用训练得到的信息行子树作为模版，从 HTML 表格树中下一个未比对过的叶子结点开始进行比对，直至该表格树中所有的叶子结点均比对完毕。

验证模块：

(5)提取结果页面中的语义信息，验证利用树结构匹配算法得到的数据单元。

该算法是一种模糊匹配算法，具有很高灵活性，一些次要节点导致的子树结构变异也不会造成匹配混乱，算法仍能准确定位出信息行或信息列的结构。

5 决策层和应用层

接收到数据提取层提交的原始数据后，决策层基于应用规则依次完成以下工作以优化查询结果。

数据清洗模块首先去除原始查询结果中主要属性缺失或明显失实的记录。然后合并重复记录，这种重复可能由某些 Deep Web 站点联营或合作产生。最后对来自于异构站点使用不同格式表示的结果进行归一化处理。

排序模块以领域知识为基础，结合用户的查询倾向对结果记录中不同属性的重要性做出评估，然后对结果进行排序。本文设计了一个排序 Agent 来完成上述工作。它的主要职责是搜集用户的查询记录和操作日志，据此分析和预测用户当前的查询期望，最后以领域知识和用户倾向的动态权重为依据实施排序。

6 实验结果

本文在科技文献检索、电子机票定购和工作简历搜索 3 个特定领域应用了上述模型。定义通过率为

$$Cth = Np/Nd \tag{4}$$

其中，Nd 为拥有满足查询条件信息的目标网站数量；Np 为数据链路层实际选择和访问的站点数量。

定义模型的查全率为

$$Cac = Nk/Nf \tag{5}$$

其中，Nf 为符合查询条件的记录总数；Nk 为系统正确搜集到的信息数量。

将模型应用于 3 个领域的实验结果如表 1 所示。

表 1　模型应用于 3 个领域的实验结果

测试集	查询语句	任务分派模块通过率			模型查全率		
		Nd	*Np*	*Cth*/(%)	*Nf*	*Nk*	*Cac*/(%)
科技文献	42	15	15	100.00	928	897	96.7
电子机票	56	37	37	100.00	604	581	96.2
工作简历	32	16	15	93.75	1 305	1288	98.7
合计	130	68	67	98.50	2 837	2 766	97.5

7 结束语

本文设计了一种新的 Deep Web 信息检索四层模型。以规则集为基础，在几个主要环节上改进了传统检索技术。最后将上述模型应用于科技文献检索、电子机票定购和工作简历搜索 3 个领域。实验结果表明，这种模型可以满足实际应用的要求。

参考文献

[1] Chang K C C, He Bin, Li Changkai, et al. Structured Databases on the Web: Observations and Implications[J]. SIGMOD Record, 2004, 33(3): 61-70.

[2] Cope J, Craswell N, Hawking D. Automated Discovery of Search Interfaces on the Web[C]//Proceedings of the 14th Australasian Database Conference. Adelaide, Australia: [s. n.], 2003: 181-189.

[3] Arasu A, Garcia-Molina H. Extracting Structured Data from Web Pages[C]//Proc. of the 22nd International Conf. on Management of Data. San Diego, California, USA: [s. n.], 2003: 337-348.

[4] Liu Ling, Pu Calton, Han Wei. XWRAP: An XML-enabled Wrapper Construction System for Web Information Sources[C]//Proceedings of the 16th International Conference on Data Engineering. San Diego, CA, USA: [s. n.], 2000: 611-621.

[5] Crescenzi V, Mecca G, Merialdo P. Road Runner: Towards Automatic Data Extraction from Large Web Sites[C]//Proceedings of the 27th International Conference on Very Large Data Bases. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001: 109-118.

[6] Song Hui, Giri S, Ma Fanyuan. Data Extraction and Annotation for Dynamic Web Pages[C]//Proceedings of the 2004 IEEE International Conference on E-technology, E-commerce and E-service. Taipei, Taiwan, China: [s. n.], 2004: 499-502.

（上接第 50 页）

参考文献

[1] Flach P A, Lachiche N. 1BC: A First-order Bayesian Classifier[C]//Proceedings of the 9th International Workshop on Inductive Logic Programming. [S. 1.]: Springer-Verlag, 1999.

[2] Liu Hongyan, Yin Xiaoxin, Han Jiawei. An Efficient Multi-relational Naïve Bayesian Classifier Based on Semantic Relationship Graphs[C]//Proc. of 2005 ACM-SIGKDD Workshop on Multi-relational Data Mining. Chicago, IL, USA: [s. n.], 2005-08.

[3] Pompe U, Kononenko I. Naive Bayesian Classifier Within ILP-R[Z]. (1995-05-20). www://httpT citeseer.ist.psu.edu/pompe95naive.html.

[4] Lachiche N, Flach P A. 1BC2: A True First-order Bayesian Classifier[C]//Proceedings of the 12th International Conference on Inductive Logic Programming. [S. 1.]: Springer-Verlag, 2002.

[5] Ceci M, Appice A, Malerba D. Mr-SBC: A Multi-relational Naive Bayes Classifier[C]//Proc. of PKDD'03. Springer, Berlin, Germany: [s. n.], 2003.

[6] Landwehr N, Kersting K, Raedt L D. nFOIL: Integrating Naïve Bayes and FOIL[C]//Proceedings of the 20th National Conference on Artificial Intelligence. Pittsburgh, Pennsylvania, USA: AAAI Press, 2005: 795-800.

[7] Yin Xiaoxin, Han Jiawei, Yang Jiong, et al. Efficient Classification across Multiple Database Relations: A CrossMine Approach[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(6): 770-783.