2015国际计算智能与通信网络会议

分布式网页爬虫框架

库马尔

AIIT，阿米提大学，诺伊达，印度

邮箱：[naresh.dhull@gmail.com](mailto:naresh.dhull@gmail.com)

辛格曼基

基督教青年会科技大学，

法里达巴德,印度

邮箱: mstomer2000@yahoo.com

摘要：从万维网中检索相关信息的快慢取主要决于爬行器的技术和效率。所以爬虫程序必须有足够的能力来理解文本和它们要抓取的链接的上下文。锚文本包含关于非常有用的网页信息，这种页面众所周知，叫作目标网页。由于了解目标网页内容有助于爬行器决定它们特定的首选爬行页面。所以在本文中，我们将会介绍了一种能够实现分布式语义网页爬虫的设计，这个设计使用超文本标记语言和语义网络页面进行爬取,这种语义web页面在书面上叫作“OWL/ RDf”。在我们的爬虫程序中有一个叫做页面分析器的组件，它用于了解网页内容的主题和上下文页面中的锚标记。更具页面分析器的输出来做出爬行决策。我们的方法在链接中提取信息方面有了很大的改进，并且你能够引导爬虫获取更多相关的特定领域信息。

关键词-语义web，RDF, OWL，语义网络爬虫，

分布式系统

# 一、简介

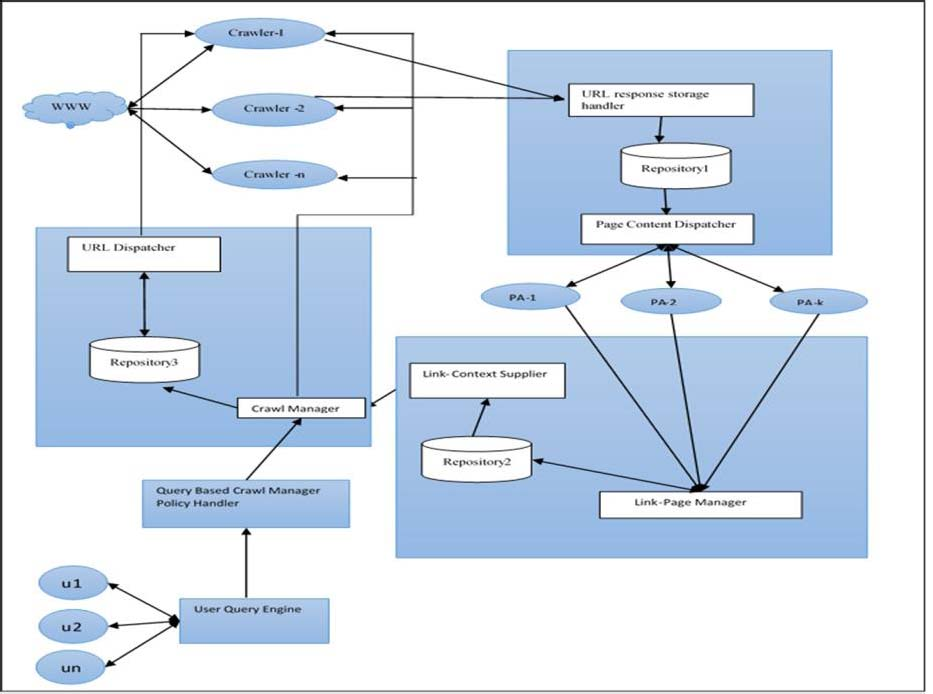
近年来， 随着网络中有价值的信息出现，萃取得到了广泛的应用。语义web是当前web enable机器和人一起工作的扩展。借助RDF和OWL信息等技术，万维网以本体的形式表示。此外语义信息正以较快的速度在万维网上增长。但大多数信息在传统网络中仍然是文本形式。浏览万维网链接（这种连接也叫锚文本）在自然语言中也提供了的传统的web页面文本。所以要做更多的搜索引擎需要作出更多的努力来理解在传统网页中的文本数据。爬虫是一个从万维网中提取信息的程序，如果爬行器可以的话，搜索引擎能够提取更多相关页面为用户查询提供更多相关的结果。为了爬取语义web页面本体，与之相关页面的不同体系结构的爬虫被开发了出来。 另一方面由于与爬取的相关网页来自传统网页，萃取上下文链接的技术也已经被许多研究人员使用[3,4,5]。链接的上下文提供与语义信息有关目标网页链接，这可以让我们从语义信息入手从而决定目标页是否被抓取。在这篇论文中提出并实现了分布式语义web爬虫。我们的爬虫程序能够抓取语义页面以及传统的web页面。在本文中为了理解锚的含义，我们使用了基于斯坦福语法分析器的文本非链接上下文的提取方法。我们也使用Jena框架来理解语义web页面，这个框架为获取的对象创建一个OWL/ RDF模型。jena创建的模型将存储到后面用于制作高级，质量相关的过滤，排名和所收集的语义内容的分析OWL/RDF页面。它与特定的域进行匹配，存储本体来语义并做出爬行web页面的决策。论文其余部分组织如下:相关工作将在第2节中讨论。第三节描述详细介绍了所提出的方案。在第4节实施中。第5节给出了结论和未来的工作。

# 二、工作叙述

相关工作研究人员已经做出了不懈的努力来抓取更多相关页面。基于规则从锚文本结构中提取链接上下文的方法在[3]中，作者使用的是基于规则LR解析器的方法提取链接上下文。在[4]中作者使用有限状态的自动机来识别在给定文本中的命名实体，本系统显示了一个很好的在文本中提取命名实体的改进网页，这样可以理解锚文本的结构。在[5]中万维网上的网页的主要部分包含非描述性锚文本上的链接。作者实现了一个基于Stanford的解析器，即非描述性链接上下文提取方法。这种方法为爬行器做一个盲目搜索的链接并给使用非描述性的锚文本的人提供了很大的帮助。甚至这种方法也是合适的，因为它对描述性锚文本进行了思考和分析。文本中是在在锚文本附近确定链接的上下文。在 [6][7]中提出了一种基于本体的各种技术，它是面向本体的爬虫。在他们的方法中，使用领域特定的本体用户可以先定义爬行目标，限制爬行范围。然后基于本体和爬行范围使用聚焦爬虫检索数据网站。每一次获取的数据之间的相关性是通过计算本体论的概念和用TF-方法抓取的数据IDF算法得到的。在[8]中，通过链接检索web页面以及通过有重点的爬行器将它们转换为主题。也可以把本体用于定义每个主题。局部网页排名算法[6]用于计算之间的术语和文档文本均为全局(整体)数据库)和本地(主题本体)透视图的相关值。一个管道架构的爬行，无论是传统的，以及语义web页面都由[9]给出。他们流水线结构的主要目标是为了获得大量来自web的结构化数据。[10]的作者提出使用web爬虫程序从web中提取元标记和元数据的方法，这种方法将提取的页面存储在知识库中。

# 三、分布式语义框架

图1如下：



我们把系统分成六个主要组件，每一个部分组件在单独的机器上实现。这个完整的系统作为客户端服务器体系结构工作。所有的不同的组件可以在a中采用客户端服务器模式相互通信。当一个组件想要与其他组件通过创建套接字通信时，可以使用该组件中正在运行的机器的ip地址和端口地址。我们提出的体系结构由以下部分组成

如图1所示。

1. 爬控制器

2. 爬行者

3.页面管理

4. 页分析器(PA)

5. 分析页面内容供应商

6. 基于查询相关的爬虫管理器策略处理程序

A.爬控制器

此组件负责维护所有爬行策略。它是一个通过消息传递它们的状态来控制所有运行和维护的爬行器。其中有两个主要部分.

组件包括:

1)爬行管理器

爬行管理器管理运行的爬行器和获取爬行页面的策略。列表爬虫管理器使用的策略包括:

(i)何时何地启动和停止爬行器。

通常爬虫程序将从最初的可见页面开始。然而，根据用户的查询，有时它会启动特定的领域，从某个最中心的地方进入该网站。

(2)并发页面

它是爬行器将同时爬行的页面数，这个数字也是由爬行管理器决定的。

(三)页与页之间停顿

这是爬行器在移动之前必须等待下一个页面的时间。

(iv)保存日志

维护所有爬行器的状态。如果有爬虫进来从同一点开始就进行上下文特定的爬行。

2)URL调度员

这个组件是仓库，存储了接口爬虫程序的各种实例。它提供了一个url列表来响应不同爬虫程序的请求。

B.Crawler

这是分布式语义的第二个组成部分，网络爬虫。独立的爬行器在不同的机器上运行，并创建连接。

管理器和页面管理器

1)万维网

它是一个基于互联网的通过超链接连接页面的全球系统。它是搜索信息的来源引擎。搜索引擎从www下载网页，然后索引它们。

2)爬行者

我们分布式搜索引擎的这些组件(每个组件在单独的机器上执行，从输入url到自动下载相应的网页都能够实现。爬虫程序的单个实例从一组url(例如，http: //www)。雅虎。com)种子开始。它让一个服务器指定一个url请求，然后下载相应的页面或接收错误。下载的页面或错误随后存储在存储库中，然后通过接口显示在仓库 1中的数据结构中。

这个收藏库使用客户端调用URL响应存储处理程序(URSH)以及爬虫程序(作为客户机)和URSH(作为

服务器)。在设计爬虫程序实例时，调度策略由爬行管理器维护。爬虫程序将按照爬虫管理器所决定的策略抓取web页面。

C.页面管理

这个组件负责响应爬行器请求并存储获取的页面，然后将自已存储的页面内容放到页面分析器进行进一步分析。

两个页面管理器的组件包括:

1) URL响应存储处理程序(URSH)

这个子组件充当仓库1之间的接口(表1)和爬虫程序的不同实例。URSH接收包含的URL和相应页面内容相关的请求表单爬虫程序，并将它们存储在存储库中。在这段时间内接收的页也被存储。如果接收到错误，则

存储错误状态，以便以后处理这些错误的url。

2)页面内容调度器

这是页面管理器的另一个子组件,它是一个工作在存储库(表1)和被称为Page Analyzer的组件的实例之间的接口。它分派页面实例上相应的url便于构建页面分析。

D. Page Analyzer (Pa)

页面分析器的一个实例使用基于统计分析和自然语言处理从页面中提取链接内容及其上下文。我们介绍了一种用于链接上下文提取的页面分析器相关的体系结构，它使用了基于斯坦福解析器的方法用于非描述性锚文本的链接上下文提取，由我们在之前的研究中实现。细节算法在下一节给出。页分析器还

提取页面主题。页面的单独实例分析仪在单独的机器上执行。这个通过链接页面的管理器提取信息将存储在仓库2，repository 2显示的数据结构中。

E.分析页面内容供应商

该组件充当页面分析器之间的接口和爬行管理器。它接收页面的主题和页面中所有链接的上下文从而提供上下文来通过链接页抓取。

1)链接页面管理

该组件充当页面分析器和仓库2 Repository 2之间的接口实例，用于存储从页面中提取的url，上下文，以及从页面中提取的主题。

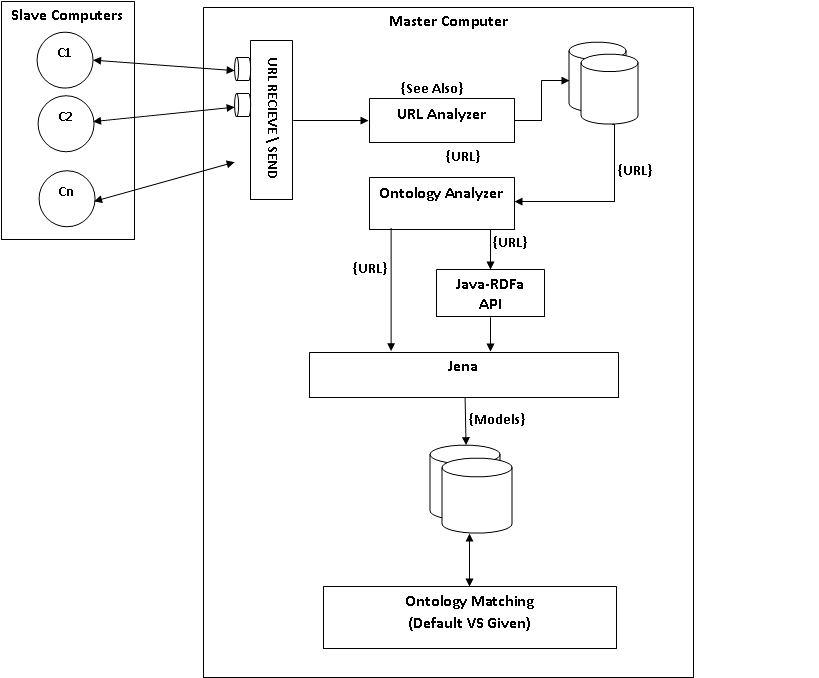
2)Link-Context供应商

它是仓库2 repository 2和爬行管理器之间的接口。这个接口传递url(链接)和url上下文爬的管理器。

F基于查询的爬网管理器策略处理程序

此组件基于用户查询的爬行策略充当爬行管理器之间的接口和用户查询实例，从而辅助爬行管理器做出决策。

# 四、方案提出



图二

爬虫是任何搜索引擎的主要组件，因为它负责从网络世界抓取数据并使其可存储到索引器。如果爬虫有足够的智慧来决定它要抓取的页面是否与其相关，那么它就可用于解决这种索引不相关的问题，因而处理的过程变得更快。在我们提出的方案中，我们试图实现分布式网络爬虫，通过理解链接的概念和上下文做出决策让这个爬虫实现其爬行，使它能够避免无用的爬行页面。在我们的方案中，我们使用了分布式网页爬虫，其中许多是独立的机器用于抓取、下载、提取和索引。

通过网页爬虫爬行实现理论上的爬行。我们开发了一个组件称为页面分析器，如图1所示。这是在多台独立的机器上实现的。它负责分析和提取网页链接并发送给爬行器。如果页面类型为OWL/RDF页面，内容将提供给本体分析器，这个分析器，使用Jena提取语义信息和创建要存储在数据库中的信息，以供进一步使用的模型处理。如果它是一个html网页，所有的链接将提取并存储在mysql数据库中，供链接文本提取器从所有锚文本类型中提取信息。页面分析器的输出，即在页面中上下文和主题的相关的所有链接和文本都将被使用来联系上下文管理器。链接上下文管理器工作有两个层次，它为链接供应商提供链接和上下文链接。链接供应是爬行管理器和链接上下文管理器之间的接口。第二个链接上下文管理器将网页的主题提供给索引器以存储语义信息。我们爬虫框架的主要策略是由爬行管理器处理。这个组件责管理所有运行在不同的机器上的爬虫程序。我们在获取相关页面方面实现了一个很好的体系结构，这是一个巨大的进步。这也是因为它在自然界的分布方面提供最大可靠性的，是一个鲁棒系统。

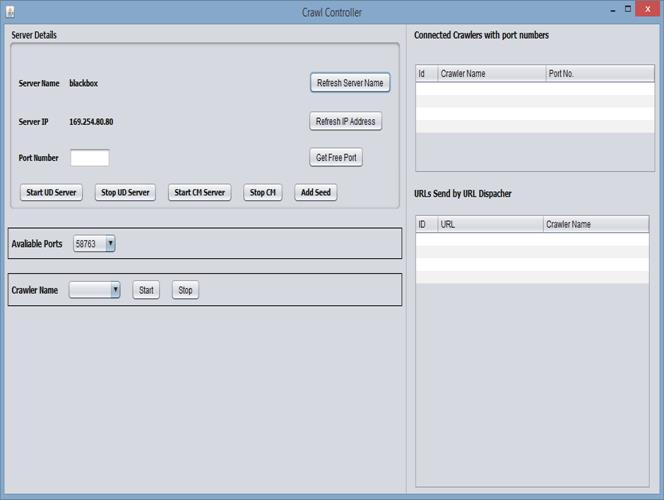
# 五、实现

该系统使用Java作为前端，Oracle11 I 作为后端。我们还使用Jena和Jsoup

模型提取理论上的网页。

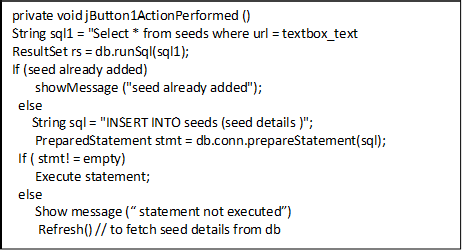
我们使用斯坦福大学的非链接上下文提取的方法让解析器提取网页链接内容。

A:爬虫控制器

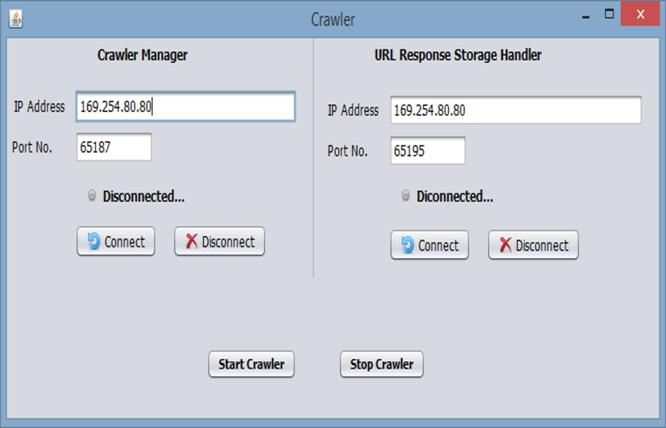


图三

抓取控制器伪代码



图四

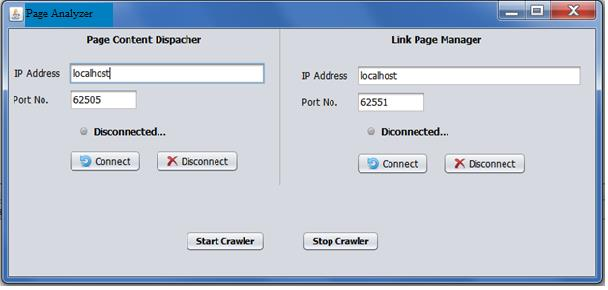


图五

B:爬虫

爬行器客户机连接到爬行启动的管理服务器并获取已经完成了的模型。

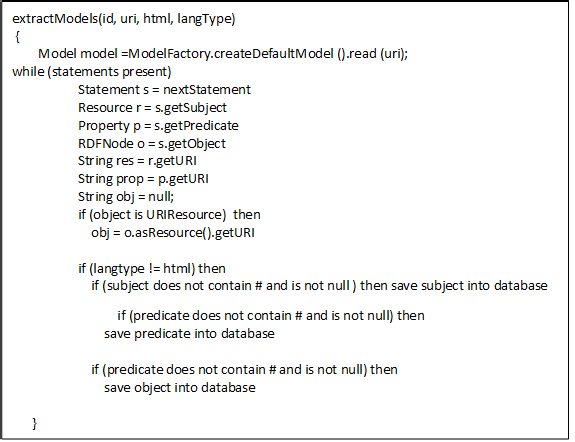
C:页面分析器



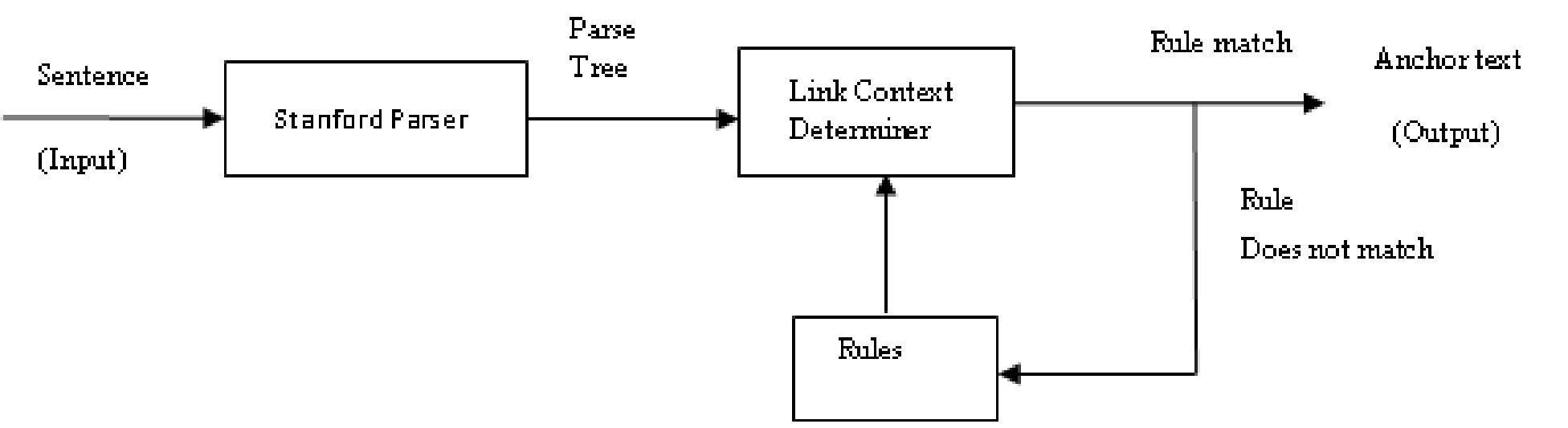
图七

页面分析器充当两个页面内容的客户机，它调度程序和链接页面管理器。它提取超链接RDF或HTML页面，并提取模型，即主题来自给定页面的谓词和对象。提取来自HTML PA的超链接使用JSoup API，查看HREF标记并提取链接。提取超链接RDF和OWL PA使用JENA API。

模型提取伪代码



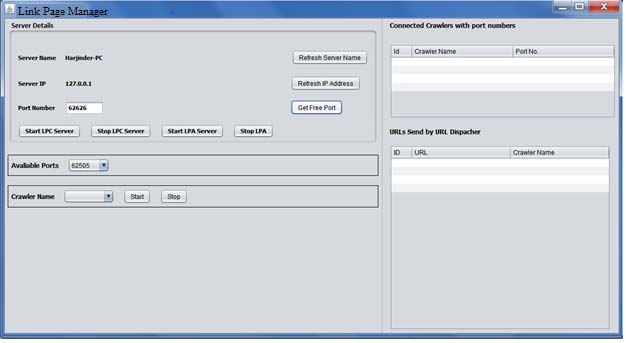
链接页面锚文本上下文提取



对于链接上下文提取，我们使用了给出的技术由美国[5]。页面中的所有链接和页面内容都是

给SPLCA。这个SPLCA的输出被赋给链接页面管理器。

D:分析页面内容供应器



链接页面管理器存储URL或过滤URL都准备好了。这个控制器的工作是只从多个PAs获取数据并过滤URL并将它们存储到数据库中，PA充当客户机，LPM充当客户机服务器。

# **六、**结论与未来展望

在我们分布式语义网页爬虫中，我们设计并实现了一种分布式网络爬行方案，这是能够从传统的爬行页面以及语义网页中爬行。我们尝试通过使用分布式计算架构来让我们的体系结构更加健壮和可靠。我们使用了一个SPLCA方案，这对我们的爬虫获取更多相关信息很有帮助。我们还是用了Jean框架以便RDF/OWL编写的页面能够很容易被我们的系统处理，这个系统可以稍后用于实现基于本体的爬行。

在这篇论文中我们已经使用锚文本的链接上下文实现了分布式语义网页爬行。在未来的工作中，我们也将基于这个爬行原型进一步增强了该体系结构。

# 参考文献：

[1] Slug:语义网爬虫Leigh Dodds, leigh@ldodds.com，

2006年2月

[2] T. Berners-Lee, J. Hendler和O. Lassila。语义网。

《科学美国人》，284(5):34-43,2001。

[3]]。Suresh Kumar, Naresh Kumar, Manjeet Singh, Asok De De A Rule-

基于锚文本的链接上下文提取方法

《结构智能信息学》第182卷系列的研究进展

《智能系统与计算》第261-271页。

[4]。Naresh Kumar, Manjeet Singh“域名独立命名实体”

基于有限状态自动机的识别系统

2013 -8月刊第77-87页。

基于斯坦福解析器的方法

从非描述性锚文本“978-1-”中提取链接上下文

IEEE探索可靠性，Infocom技术

和优化(ICRITO)(趋势和未来方向)，2014年。

以本体论为中心的网路爬行

在ACM应用计算研讨会(SAC 2003)上，

墨尔本,2003。

[7] M. Ehrig, A. Maedche, S. Handschuh, L. Stojanovic, R. Volz，

“基于本体的web文档和基于rdf的元数据的爬行，”

在2002国际语义网会议(ISWC 2002)上，撒丁岛，

2002.

“专注于ALVIS语义搜索引擎的爬行”，in

第二届欧洲语义网会议(ESWC 2005)， Heraklion, 2005。

[9] Anthoniraj Amalanathan, Senthilnathan Muthukumaravel "语义

基于词法数据库的网络爬虫

2012年4月，第2卷第4期，第819-823页

[10]李丁，潘蓉，蒂姆·菲宁，阿努帕姆·乔希，云鹏，普拉纳姆

Kolari“在语义Web上查找和排序知识”

第四届国际语义网会议论文集，高威IE，

2005年11月,气象出版社。

[11] multirawler:用于爬行和索引的流水线架构

语义Web数据

Andreas Harth, Ju¨rgen Umbrich和Stefan Decker

[12] G.S. Tomar, S. Verma & Ashish Jha;“网页分类

"， IEEE TENCON-2006, pp 1-4，

2006年11月14日至17日

[13] Gautam Pant“从HTML标签树派生链接上下文”

DMKD03:第八届ACM SIGMOD数据研究研讨会

《挖掘与知识发现》，2003年。