# Dominant Superstring Algorithmic Trinity Construction for Web Extraction

Prabhu. R
Department of Computer Science and Engineering
Saveetha Engineering College/ PG Scholar
Chennai, Tamil Nadu/India

Mervin. R
Department of Computer Science and Engineering
Saveetha Engineering College/ Associate professor
Chennai, Tamil Nadu/India

***Abstract*— Web mining is the application of data mining techniques to discover the patterns from the web. Most of the end users were searching for an effective system which can provide an optimized comparative solution without any big expenditure. The aim of the paper is to develop more intelligent system to potentially help the user in finding and extracting the valuable information and resources. Thus the developed framework will automatically extract the data from the internet based web applications to process the data in linear tree fashion. An automatic parser will be placed in the backend of the system which will take care of subdividing the web patterns in to smaller pieces of patterns which include prefix, suffix and separators. The exact information about the data located in the web pages is retrieved. The data will be cleaned up and formatted for manipulation which enables an emergent of efficient cost comparative system. A multi perspective crawling mechanism used in fetching the information from admin defined multiple websites and load into the system. Thus the process involves in enhanced level of website content searching with creating centralized network data storage.**

**Keywords— W*eb mining, web data extraction, parser, crawler and trinity.***

## I. INTRODUCTION

World Wide Web (WWW) enriches us with enormous amount of widely dispersed interconnected beneficial and dynamic hypertext information. It has furnished the distinct needs of us in various stages like communication, business, entertainment and so on. The current World Wide Web has been reached the peak of its success with respect to valuable resources of information, Enormous number of users, Multiform and multitude of data, efficient digital commerce. The abundant unstructured or semi-structured information on the web leads a great challenge for users and those who are in need for complete beneficial information. To eliminate these issues data mining techniques must be applied on the World Wide Web. The problem faced in dealing with web data such as the user and provider problem.

### *A. User Problem*

People either browse or use the search service to find specific data on internet. Nowadays search tools have two drawbacks. Low precision due to the irrelevance of various search results and low recall due to the inability to index all the information available on the web as some of the relevant pages is not properly indexed called query generated process. Since it is complex to get specific data, it's very hard to make sense out of it called data generated process.

### *B. The Provider Problem*

Deficient in gathering information about need of the customer to personalize the individual user and lack in effectively using the web data to market products and to service the customer. According to evaluation targets, web mining techniques can be classified in to web content mining, web structure mining, and web usage mining. Web content mining aspects are related to the similar domains in classic data mining includes

- Self-extraction of data from web pages
- Opinion and review extraction
- Knowledge synthesis
- Integration of the information
- Noise detection and segmentation

The aspects listed above is solutions for more or less complicated drawbacks, conjunct to self-extraction of data usage on web which leads increase in several aspects of Internet daily life. Table1 shows various techniques for web content mining.

| Document | Techniques | Process |
|---|---|---|
| Unstructured | Information Extraction | Extracting information from unstructured data and converts into structured data includes Pattern matching |
| Structured | Web crawlers | Traverse the hypertext structure of the web. Internal crawlers go through internal web pages of sites. External web crawlers go to the unknown links or sites. |

| | | |
|---|---|---|
| Semi Structured | Webdata extraction language | Converts web data to structured data and delivers to end users. |

Table1: Techniques for web content mining.

## II. RELATED WORK

Hassan A. Sleiman and Rafael Corchuelo proposed [1] proposed a "Trinity for Unsupervised Web Data Extraction" used to extract data from web documents in order to feed automated processes. The template introduces some shared patterns that do not provide any relevant data and can thus be ignored. Many web data extractors rely on extraction rules which can be classified in to ad hoc rules. The costs involved in handcrafting ad hoc rules motivated to work on automatic techniques. Find a shared pattern and partitions the input documents in to the prefixes, separators and suffixes that they induce and analyses the results recursively, until no more shared patterns are found. Prefix, separators, and suffixes are organized into a trinity tree that is traversed to build a regular expression with capturing groups that represents the template that was used to generate the input documents.

Paolo Tonella and FilippoRicca[2] proposed "Dynamic model extraction for web application". The investigated techniques are carried to support web application and perform analysis and testing. The actual implementation start with the web application on how model is extracted by means of a crawler from the home page of the target Web application. The model can still be considered a useful starting point when trying to model web applications of the future Internet. merging different dynamically generated pages prevent the usage of 2002 model "as is" to analyze and test future Web applications

Donghua Pan, ShaogangQiu and Dawei Yin proposed [3] "Web page extraction method Focus on visual feature of web page". The system applies such visual information to font size, layouts and background colour to divide web page into visual blocks. Simulates how people observe web pages and documents. The complexity of vision feature is that it is hard to find a universal rule set

Mohammad Shafkat Amin and Hasan Jami [4] proposed "Fast wrap from the web". The system can automatically discover table structure by relevant pattern mining from web pages in an efficient way and can generate regular expression for the extraction process. Employs suffix tree based technique to obtain records called tabular data. This tool does not require any prior knowledge of the target page and its content. It requires the domain specific assumption. The wrapper generation process asymptotically takes linear time to progress.

Zhixian Zhang, Kenny Q. Zhu and Haixun[5] proposed "Top k-list from the web Information extraction method" using top-k web pages. The web pages that describe top k instances of a topic which is of general interest. The visual signal method proposed by use vector graph method. The system uses vector describing tag path occurrence patterns. Based on a similarity measure between visual signals they perform clustering of tag paths and rebuild the structure of data in the form of sets of tag paths. A hybrid list extraction approach as it not utilizes the visual alignment on multiple list items difficulties to structural feature.

Lei FU, Yingju XIA, Yao MENG, and HaoYU [6] proposed "CRF model for web data extraction". A suitable category label in 2010 using Conditional Random Fields (CRF). This method applied to the extracted content in their implementation to better solve the drawbacks of the traditional methods. The serialization method based on crude filtering to the DOM node. The system utilizes the linear chain CRFs model to complete the text sequence labelling. To evaluate a ten-fold cross validation method are implemented. The accuracy of the extraction and labelling can achieve in moderate.

## III. ARCHITECTURE

The system focused in designing a multi perspective crawling mechanism in fetching the information from multiple websites. The structure of the website is fetched and an automated stemming process to remove unwanted stuffs surrounding the conceptual data is removed. After fetched data from a website an automatic manipulation is processed and the data will be formatted based on user's requirement and a comparative analysis to suggest an optimized cost effective best solution for the buyers.

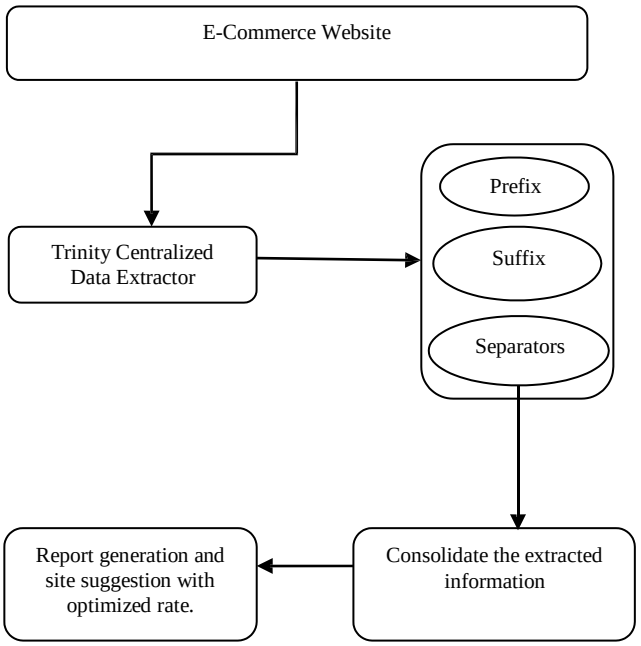



Figure 1: Architecture of Trinity

Figure 1 shows the architecture of trinity. Trinity centralized data extractor is an unsupervised proposal that learns extraction rules from a set of web documents that were generated by the same server-side template. The data extractor

builds on the hypothesis that shared patterns are not likely to provide any relevant data. As it finds a shared pattern, it partitions the input documents in to the prefixes, suffix and separators that they lead and analyses the outcome repeatedly, until no more shared patterns are identified. Prefixes, suffix and separators are organized into a trinity tree that is later traversed to build a regular expression with capturing groups that represents the template that was used to generate the input documents. The extended xml algorithm is used to consolidate website information. The recommendation zone analysis is generated with optimized rate using wrapper model extraction.

*A. Single Website Crawling*

Screen scraping is the process of programmatically accessing and processing information from an anonymous website. For example a rate analysis website might screen scrape a variety of online retailers to build a database of products and what various retailers are selling them to market. The process performs by making an HTTP request from code and then parsing and analyzing the returned HTML. These classes are useful for making an http request to a remote website and pulling down the markup from a particular URL but they offer no assistance in parsing the returned HTML. The following procedure shows the depth restricted crawling function.

**Pseudo code**

```
function depth-restricted-crawl(page q, int f)
   if f == 0
      return
   /* do something with q, store it or so */
   for each (page l in links(q))
      depth-restricted-crawl(linked, f-1)
```

*B. Page Segmentation*

Web Scraping, which has been an important function of search engine crawling - extraction of all links for any given URL. The HTML content from any given URL is downloaded as a string, and all occurrences of hyperlinks are extracted from it

1. List the Meta attribute

Screen scraping usually involves downloading the HTML for a specific web page and picking out particular pieces of information. Html Agility Pack is used to download a remote web page and enumerate the <meta> tags, displaying those <meta> tags that contain both a name and content attribute.The document class represents a complete document and contains a Document Node property which returns an Html Node object that represents the root node of the document. The information about the node such as name attributes and html type can be retrieved by traversing the DOM tree. For Html elements this property returns the name of the tag body for <body> tags, p for a <p> tag and so on. The XPath expression which returns all of the <meta> tags in the document. If there are no <meta> tags in the document then, at this point meta tags will be null. But if there are one or more <meta> tags then meta Tags will be a collection of matching html node objects.

2. List the Link attribute

The hyperlink elements are used for different purposes on the page call JavaScript functions, link to anchors on page and for appropriate usage of hyperlinking to internal and external pages. Among this only hyper linking element must be filtered out. The module starts by crawling the specified web **URL** or any local file resource. All data that map to the match regular expression field will be received as a decision. After the matching process is completed for the respective URL, the crawler will contagiously process next URLs that the appropriate URL links. The whole process is repeated until the final URL has been reached.

*C. Multiple website Crawling*

The following steps used to crawl multiple website information.

**Step 1:** Processing elements in a queue

Web crawling can be regarded as processing elements in a queue. The crawler extracts links to other web pages whenever it visits a webpage. So the crawler puts these URLs at the end of a queue and continues crawling to a URL that it removes from the front of the queue. A set contains all URLs that have so far been collected. The URL is added to queue if it is not found in the respective queue.

**Step 2:** Implementation of the queue

To limit either the number of webpages visit. The process in the queue integration is similar to the desired functionality. To limit the depth level more than one queue is needed. But regardless of the link depth two queues are sufficient. In this module allow the crawler to only fetch URLs from first queue and append links to second queue. After all URLs in first queue are processed, switch the queues.

**Step 3:** Implementation of a thread controller

The thread interface provided to handle the thread methods in java. To add a little more generic functionality makes use of a number of threads that process items of the first queue. When there are elements in the queue need to be processed and total number of threads is less than upper bound the controller is expected to create new threads. The queue can be reloaded with minimum one element then the run processes induce the thread which inherits from the thread class.

**Step 4:** Implementation of a process

The thread in run method fetches new items from the queue and that it ends itself if there are no items left. This is common for all our possible threads therefore implement this in upper class for controllable threads. If there are no more items to

process the controllable thread can terminate it, but has to inform the thread controller.

**Step 5:** Communication

During program execution, it might sometimes be necessary that a working thread tells the main thread what it does at the moment. For this purpose the Message Receiver interface exists. The message receiver class is notified when a thread reacts, a thread is eliminated or all threads are eliminated. Sending message in communication is equal to calling a method in oops concept.

*D. Consolidating website content module*

Essentially, consolidate includes a lot of code to determine which HTML to serve the client's browser. The full page cache stores the emitted HTML the first time each page is requested and resends that response for all subsequent requests. The cache feature takes care to ensure that dynamic content (e.g., cart count, acknowledgement message, etc.) also differs by consumer although the remaining majority of the page is served without reprocessing the code. This module is responsible for merging both JavaScript and CSS content, reducing the number of round trips to the server for each page load and often improving the customer usage. This process might not have an impact on the page response time (as do some of the other optimizations), but the user may experience improved performance with this setting enabled. The flexible ODV (Object Definition Value) gives users the ability to completely customize product attributes but at a performance cost. To condense the ODV attributes into a single table and row reducing the number and complexity of catalog queries being executed and thereby improving the application response times.

*E. Recommendation zone analysis*

Recommendation zone analysis module tends to recommend a systematic assessment of specific website zone content and impacts the extraction of search. It has to go through total four analysis process such as (1) Product Analysis, (2) Category Analysis, (3) Review Analysis, (4) Price Extraction.

1. Product Analysis: Product lists are divided by pages in the leaf node. It goes to all the pages to extract the addresses that linked to the detail information page.

2. Category Analysis**:** Since the link address to move to the next category page is hidden, therefore, code information of the category are extract step by step to analyse the leaf node in which the product is present.

3. Review Analysis: In the detail information page per individual product, review information is divided in several pages. At this time, the part where review list starts or ends can be separated by means of specific HTML tag. However, the target websites of this study have all different review information therefore each one needs to be analysed.

4. Price Extraction: It extracts price contents and save it in the database. Since the price list contents are in HTML type, the html tag needs to be removed to extract the row text. The price list is automatically checked with updated value.

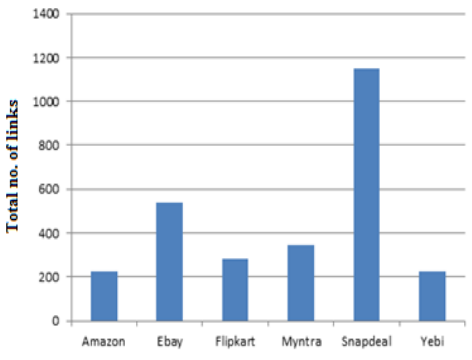
## IV. EXPERIMENTAL RESULT AND ANALYSIS

The Table 2 shows the list of popular on-line shopping websites with the total number of links and keywords available in each websites. The websites are chosen with the popularity in e-commerce application domain knowledge. The websites are listed with the link and keywords attributes to evaluate each websites for estimating and comparing the available online shopping websites. The analysis shows the comparision of the link retrived of the source URL specified.

| Websites | Total links | Sample Keywords | Total Keywords |
|---|---|---|---|
| Amazon | 225 | Amazon, Online Shopping, buy online, buy mobiles , movies. | 25 |
| Ebay | 539 | EBay, clothing, apparel, auction, collectibles. | 14 |
| Flipkart | 285 | Online Shopping, India, Books, Store, Flip kart. | 05 |

Table 2: Website evaluator

Graph 1.1 shows link counter. The link count measures the total number of link attributes available in the source code of the individual websites. The maximum number of link extracted with the help of the crawling are listed in the graph. The crawling depth function performs in this websites and retrives the maximum number of the link in the website listed.




Graph 1.1 Link Counter

## V. CONCLUSION

In this paper, a multi perspective crawling mechanism is implemented to extract effective information from the target website. The crawler that combines search strategy based on content and search strategy based on link structure. It is based on the hypothesis that web documents generated by the same server side template share patterns that do not provide any relevant data but help to delimit them. The recommendation engines are developed to provide users requested product with cost comparison. The future research plan is to perform data mining on user search activities such that user profiles can be learned automatically.

## REFERENCES

[1] Hassan A. Sleiman and Rafael Corchuelo "Trinity: On Using Trinary Trees for Unsupervised Web Data Extraction" IEEETransaction on knowledge and data engineering,Volume. 26, NO. 6, JUNE 2014.

[2] Paolo Tonella and Filippo"Dynamic model extraction and statistical analysis of Web applications" Web Site Evolution, 2002. Proceedings. Fourth International Workshop on 2002.

[3] Donghua Pan , ShaogangQiu and Dawei Yin "Web Page Content Extraction Method Based on Link Density and Statistic" Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on 12-14 Oct. 2008.

[4] Mohammad Shafkat Amin and HasanJami "FastWrap: An efficient wrapper for tabular data extraction from the web" Information Reuse & Integration, 2009. IRI '09. IEEE International Conference on 10-12 Aug. 2009.

[5] Zhixian Zhang, Kenny Q. Zhu and Haixun "Automatic extraction of top-k lists from the web" Data Engineering (ICDE), 2013 IEEE 29th International Conference on 8-12 April 2013.

[6] Lei Fu , YingJuXia,YaoMeng and Hao Yu "Conditional Random Fields Model for Web Content Extraction" Computing in the Global Information Technology (ICCGI), 2010 Fifth International Multi-Conference on 20-25 Sept. 2010.