IJCSC 0973-7391 Volume 6 • Number 2 April - Sep 2015 pp.1-3 available online at www.csjournalss.com

# A Survey of Unsupervised Techniques for Web Data Extraction

Disha Patel, Dr. Ankit Thakkar
Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India
13mcei25@nirmauni.ac.in, ankit.thakkar@nirmauni.ac.in

**Abstract:** World Wide Web contains a large amount of data and to fetch important information from web has become a useful task. There are many web information extraction systems are developed and categorised in manual, supervised, semi-supervised and unsupervised techniques. We will study unsupervised techniques and how they differ from each other. Roadrunner uses match algorithm for generating the wrapper and it does extraction at page level. ExALG uses Large and Frequently occurring equivalence class for extraction. It also does extraction at page level. FivaTech uses tree matching algorithm for generating the template. Trinity uses trinary tree which is divided into prefixes, separators and suffixes. It will be used to generate the regular expression. Trinity has a very less extraction time compared to other techniques, which makes it more efficient.
**Keywords:** Web Information Extraction, Web Mining Wrapper Generation, Unsupervised Learning.

## Introduction

Web Mining refers to extract rich data from World Wide Web(WWW) and generate patterns based on web usage. The World Wide Web is a global information space with billions of web pages that can be accessed via Internet. Web Content Mining consists of text, audio, video or structured records like tables and lists. Several issues can be occurred while generating patterns for web data classification of web documents and extract useful information from images. Web Structure Mining consists of web pages as no des and hyperlinks as edges connecting related pages. By means of Hyperlinks any user can switch to different location within a same web page or different web page. Web pages can be organized by means of several HTML tags and we need to automatically extract Document Object Model (DOM) structure. Web Usage Mining is a technique to discover usage pattern base d on users surf different websites. Logs can be made of IP address and time spent by user on web page. Several E-Commerce websites can have track of products surfed by user and display it again when user comes again. In this paper, we will explore different algorithms used to extract structure of web pages automatically.

## Web Information Extraction System

Web Data Extraction aims at extracting information from the web documents. This information is stored in the database, which can be accessed for retrieving the data. Due to the varied structure of the web pages, automatic extraction of the required information becomes important and tedious task. It needs a system which will automatically extract the information from the web documents.
The goal is to generate a wrapper which will be used to extract the data. We can divide the web information extraction system into four categories: manual, supervised, semi-supervised and unsupervised [9]. In manual, user has to program wrapper by hand using any programming language. In supervised, labeled web pages are taken and it will also specify the examples of the data to be extracted and then it will output the wrapper. In semi-supervised, it accepts rough examples from the examples of the data to be extracted and then it will output the wrapper. In unsupervised, web pages are unlabeled and it will be classified automatically. We will study various unsupervised automatic web data extraction techniques.

A. ROADRUNNER

Roadrunner works on a collection of web documents and it compares the input pages side by side in order to infer a Union free Regular expression(UFRE) \cite{proceeding6}, which will describe its template. This algorithm needs input pages to be well formed, so it uses tools like JTidy for preprocessing. It builds the rule for extraction incrementally by means of string alignment which is tailored to XHTML.

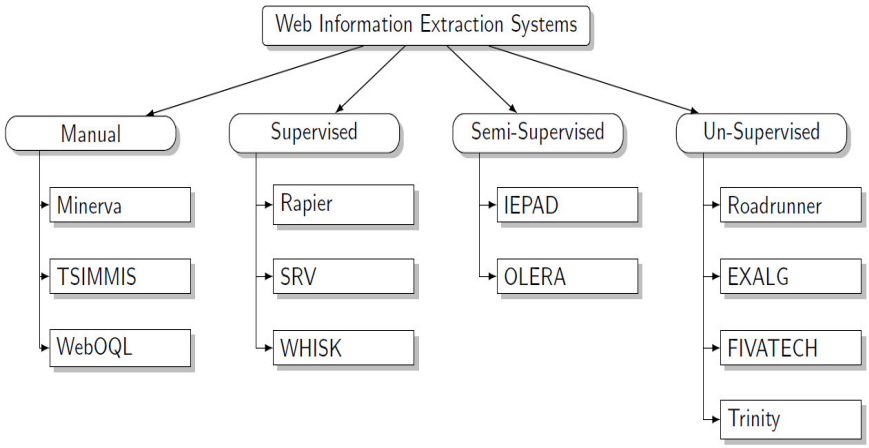


Figure 1. Classification of Web Information Extraction Systems

Roadrunner algorithm can have two inputs pages Wrapper and Sample. The algorithm tries to parse the sample using wrapper and make this wrapper as generalize as possible by means of solving each mismatch occurs while parsing this sample. The algorithm uses AND-OR tree matching which will increase time complexity exponentially with respect to the length of token. A token is either a HTML tag or string value. The algorithm is capable of handling nested structure such as list of books and each book have list of editions.

**Limitations**:

**Union Free Regular Expressions**: This technique is not capable of handling disjunction cases where a web page contains information such as "Red is a color or 6 + 6 = 12". Introducing union operators will increase complexity.

**AND-OR tree matching:** Complexity increases exponentially and in order to limit complexity, this algorithm adapts several pruning techniques such as skip of sub trees which is not appropriate solution.

B. EXALG

ExAlg attempts to search for tokens which are there in input document and which can be added to the template, so that it can generate the extraction rule by means of differentiation and nesting criteria.

ExAlg performs extraction in two stages: First stage is Large and Frequently occurring Equivalence Class (LFEQ)[8]. It is a large maximal subset of tokens which are there in the input document in large numbers. It can guarantee that there exist a unique token using a padding technique which is called the root LFEQ. Now those LFEQs which are not nested within other LFEQs, need to be removed and it is a complex part. If for example, there are two different book titles in <h3>Book Title</h3>, then ExAlg consider it as two separate token. This algorithm will find the initial set of LFEQs and discard invalid and differentiate roles to minimize this set. The refined set of LFEQs will be searched recursively for any uniform pattern. Second stage is analysis stage. In this, analysis module will construct template and extract the data from the pages. The complexity of this algorithm increases linearly but there is no experimental proof. ExAlg works on assumption such as uniqueness of tokens and constructor should be instantiated frequently.

C. FIVATECH

FivaTech is a page level data extraction system which deduces the data schema and templates for the input page generated from the CGI program[7]. This algorithm generate model for dynamic websites by use of tree template technique. To detect a schema of the website, it generates the DOM trees of the input web pages. Now, it will apply tree merging algorithm to the DOM trees. Tree merging includes peer node recognition, multiple string alignment, tandem repeat mining and optional merging. In peer node recognition, each node is represented by a tree and 2-tree matching algorithm and is used to verify whether two nodes with the same tag are similar or not. It calculates the normalized score which is the ratio between the number of parts in the mapping over the maximum size of the two trees. After this, all peer subtrees will be assigned the same symbol. Then in the pattern step, it will detect repetitive

pattern and merge them in the ascending order of length. Detection of the structure includes identifying the schema and generating the template for each type constructor of this schema.

D. TRINITY

Trinity is an unsupervised technique, which learns the rules for the extraction from the set of input web pages which are generated by the same server side template [4]. In this technique, shared patterns are not likely to provide any relevant data. Whenever it finds shared pattern, it will divide it into three parts: prefixes, separators and suffixes that will be generated and will analyse the result until no more shared patterns are found.

Trinary tree is organized into prefixes, separators and suffixes that will be later traversed to build a regular expression with capturing groups which represents the template that was used to generate the input web documents. From similar documents, data can be extracted by using this regular expression. Trinity learns UFRE(Union Free Regular Expression).

It takes a collection of input web documents, it need to be tokenized. It will first create a root node with the input web documents, then it will loop and search for a shared pattern. If such a pattern is found, it will use to create three child nodes with the prefixes, the separators and the suffixes. These nodes will be analyzed recursively. If no shared pattern is found, the tree is not expanded. One the trinary tree is built, it will learn regular expression which will represent the template used for generating the input web documents.

## Conclusion

Trinity is more efficient compared to other techniques. Roadrunner needs well-formed documents while Trinity doesn't have any such criteria. Roadrunner was proven to be polynomial in time but no results of space complexity were given. ExAlg cannot work on malformed input pages and it does not align the input pages. While Trinity aligns the input web pages. FivaTech needs to correct the malformed web pages while Trinity works without correcting the documents. Trinity is polynomial in time and space. It has almost negligible extraction time. So, we can conclude that Trinity iss more efficient and effective than other techniques and further research can be done for improving the extraction time.

## References

[1]. Singh, B. and Singh, H.K.: Web Data Mining research: A survey In: Computational Intelligence and Computing Research (ICCIC), pp. 1-10. IEEE InternationalConference (2010)

[2]. Wang Bin and Liu Zhijing: Web mining research In: Computational Intelligence andMultimedia Applications, pp. 84-89. ICCIMA (2003)

[3]. Baeza-Yates, R.A.: Searching the Web: challenges and partial solutions In: StringProcessing and Information Retrieval, pp. 23-31. A South American Symposium(1998)

[4]. Sleiman, H.A and Corchuelo, R.: Trinity: On Using Trinary Trees for UnsupervisedWeb Data Extraction In: Knowledge and Data Engineering, pp. 1544-1556. IEEETransactions (2014)

[5]. Cooley, R. and Mobasher, B. and Srivastava, J.: Web mining: information and pattern discovery on the World Wide Web In: Tools with Articial Intelligence, pp.558-567. IEEE International Conference (1997)

[6]. Crescenzi, Valter and Mecca, Giansalvatore and Merialdo, Paolo: RoadRunner: Towards Automatic Data Extraction from Large Web Sites In: RoadRunner: TowardsAutomatic Data Extraction from Large Web Sites, pp. 109{118. ACM (2001)

[7]. Kayed, Mohammed and Chia Hui Chang and Shaalan, K. and Girgis, M.R.: FiVaTech: Page-Level Web Data Extraction from Template Pages In: Data MiningWorkshops, pp. 15-20. IEEE International Conference (2007)

[8]. Arvind Arasu and Garcia-Molina, H.: Extracting structured data from Web pages(Poster) In: Data Engineering, pp. 698-710. IEEE International Conference (2003)

[9]. Chia Hui Chang and Kayed, Mohammed and Girgis, M.R. and Shaalan, K.F.: ASurvey of Web Information Extraction Systems In: Knowledge and Data Engineering, pp. 1411-1428. IEEE International Conference (2006)