Volume 4, Issue 5, May 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Automatic Template Extraction from Heterogeneous Web Databases

Ashish Pabitwar

Computer Engineering,

D.Y Patil College Pune, India

ABSTRACT: ***To achieve high productivity in publishing the web pages are automatically methods such as clustering and classification and badly impact the performance and re-sources of tools that processes the web pages. Thus, the template detection techniques have received a lot of attention to improve the performance of search engines, clustering and classification of web documents. Here, project is presenting the approach to detect and extract the templates from heterogeneous web documents and cluster them into different group. The pages belong to each group should possess the same structure evaluated using common templates with contents. The templates are considered harmful because they compromise the relevance judgment of many web information retrieval and web mining.***

Keywords: Clustering, Minhash, Minimum Description Length Principle, Template Extraction.

1. Introduction

As the amount of information on the web grows, there is an increasing demand for software that can automatically process and extract information from web pages. Many pages on the web today are dynamically generated, whereby data from a database fill templates to generate HTML. Even though the underlying data on such web pages is structured, the resulting transformation to HTML adds formatting and layout noise making the original data much harder to access by automated programs.

Web information extraction is the field of discovering automated methods for extracting information from web pages. One of the grand goals of web information extraction is to develop methods that can automatically convert human-readable formatting into machine understandable data structures in a completely unsupervised manner. The state of the art information extraction technologies can learn from human labeled examples to automatically extract data specific to a single type of web page. This is done by learning surface features of the page and inducing an extraction pattern specific to that page type. However, there are still no systems robust enough to generalize web data extraction to an entire gamut of web pages (thus opening the possibility of completely unsupervised extraction from new types of pages with no human labeled examples). The aim of this thesis is to demonstrate a system that strives to realize that goal.

2. Proposed System

In order to alleviate the limitations of the state-of-the-art technologies, we investigate the problem of detecting the templates from heterogeneous web documents and present novel algorithms called TEXT (auTomatic tEmplate eXTraction). We propose to represent a web document and a template as a set of paths in a DOM tree. Our goal is to manage an unknown number of templates and to improve the efficiency and scalability of template detection and extraction algorithms. To deal with the unknown number of templates and select good partitioning from all possible partitions of web documents, we employ Rissanen's Minimum Description Length (MDL) principle. In order to improve efficiency and scalability to handle a large number of web documents for clustering, we extend MinHash. While the traditional MinHash is used to estimate the Jaccard coefficient between sets, we propose an extended MinHash to estimate our MDL cost measure with partial information of documents.

3. Module Description

3.1 Document Collection and DOM: We collect the HTML documents as input. The DOM defines a standard for accessing documents, like HTML and XML. The DOM presents an HTML document as a tree structure. The entire document is a document node, every HTML element is an element node, the texts in the HTML elements are text nodes, every HTML attribute is an attribute node, and comments are comment nodes.

3.2 Essential Paths and Matrix: Given a web document collection $D = \{d_1, d_2, \ldots, d_n\}$, we define a path set P_D as the set of all paths in D. Note that, since the document node is a virtual node shared by every document, we do not consider the path of

the document node in P_D. The support of a path is defined as the number of documents in D, which contain the path. For each document d_i, we provide a minimum support threshold t_{di} .

3.3 Agglomerative with MINHASH (TEXT-HASH): In our problem, although we take only essential paths, the dimension of E_i is still high and the number of documents is large. Thus, the $O(n^2s)$ complexity of TEXT-MDL is still expensive. In order to alleviate this situation, we will present how we can estimate the MDL cost of a clustering by MinHash not only to reduce the dimensions of documents but also to find quickly the best pair to be merged in the MinHash signature space. It is the agglomerative clustering algorithm with MinHash signatures. To compute the MDL cost of each clustering quickly, we would like to estimate the probability that a path appears in a certain number of documents in a cluster. However, the traditional MinHash was proposed to estimate the Jaccard's coefficient. Thus, given a collection of sets $X = \{S_1, \ldots S_k\}$, we extend MinHash to estimate the probabilities needed to compute the MDL cost.

3.4 Agglomerative with Extended MINHASH (TEXT-MAX): When we merge clusters hierarchically, we select two clusters which maximize the reduction of the MDL cost by merging them. Given a cluster c_i, if a cluster c_j maximizes the reduction of the MDL cost, we call c_j the nearest cluster of c_i. In order to efficiently find the nearest cluster of c_i, we use the heuristic 1. By using Heuristic 1, we can reduce the search space to find the nearest cluster of a cluster c_i. The previous search space to find the nearest cluster of c_i was the same as the number of current clusters. But, using Heuristic 1, the search space becomes the number of clusters whose Jaccard's coefficient with c_i is maximal. The Jaccard's coefficient can be estimated with the signatures of MinHash and clusters whose Jaccard's coefficient with c_i is maximal can be directly accessed in the signature space.

Heuristic1:

$$\frac{\cap\, d_k \in (D_i \cup D_j) E(d_k)}{\cup\, d_k \in (D_i \cup D_j) E(d_k)}$$

3.5 MINHASH with Dice's coefficient: When we merge clusters hierarchically, we select two clusters which maximize the reduction of the MDL cost by merging them. Given a cluster c_i, if a cluster c_j maximizes the reduction of the MDL cost, we call c_j the nearest cluster of c_i. In order to efficiently find the nearest cluster of c_i, we use the heuristic 1. By using Heuristic 1, we can reduce the search space to find the nearest cluster of a cluster c_i. The previous search space to find the nearest cluster of c_i was the same as the number of current clusters. But, using Heuristic 1, the search space becomes the number of clusters whose DICE coefficient with c_i is maximal. The DICE coefficient can be estimated with the signatures of MinHash and clusters whose DICE coefficient with c_i is maximal can be directly accessed in the signature space.

Heuristic 1 (DICE Coefficient):

$$s = \frac{2|S1 \cap S2|}{|S1| + |S2|}$$

3.6 Estimating & Comparing COST: Estimate the final cluster cost in each Algorithm. In TEXT-HASH & TEXT-MAX using approximate MDL cost model. Compare the MDLCOST of all models.

3. Figures And Tables

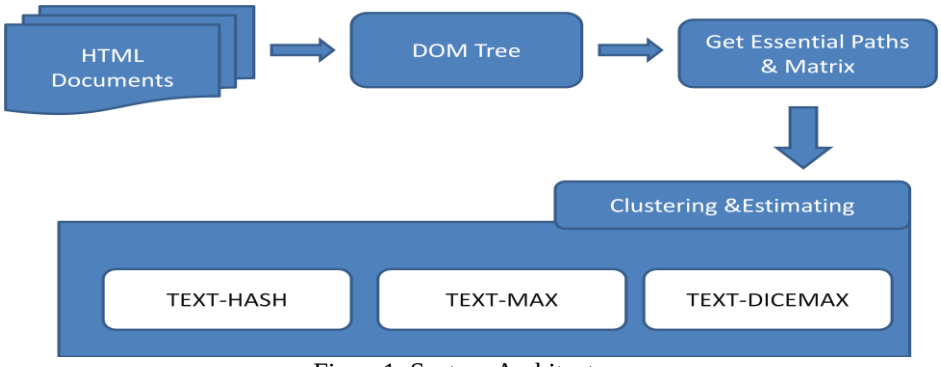

Figure1: System Architecture

5. Conclusion and Future Work

We used template detection and extraction in heterogeneous web pages. We cluster the documents based on the template used in the web documents. We also extract the data used in the web pages. By using this web pages are fully studied and there contents are compared and extracted.

Although extracting templates from heterogeneous web pages needs large time to extract and detect, so we have to reduce the time cost .Even though efficiency is increased in our project we not attain the full efficiency or near to full efficiency. So we have to concentrate to increase the efficiency and scalability.

The high cost of the manual creation of metadata for a large collection implies a great demand on tools for automatically extracting metadata from a collection. However, existing automatic metadata extraction approaches have limitations on working with a large heterogeneous collection. This dissertation has proposed a template-based approach to automate the task of extracting metadata from a large legacy collection.

The template-based approach first classifies documents into groups, and then creates a template for each group. In this way, a heterogeneous collection is converted to a set of homogeneous sub-collections. Templates are written in a designed language, which can be understood by the metadata extraction code. As such, the template-based approach should be able to work with different collections. Ideally, by creating new templates, the template-based approach should work with new kinds of documents that are added to a collection over time or be adapted to a different collection without changing the metadata extraction code.

One possible enhancement is to integrate metadata from different kinds of pages. A document may have more than one page containing metadata.

Another possible development is to extend our metadata extraction code to work with a hierarchy document structure instead of working on the line level only. The feature set and rule language could be also improved.

6. Result set Data

	Data set 1 (D1)				Data set 2 (D2)			
	#	P	R	Sec.	#	P	R	Sec.
RTDM(0.5)	32.3	1.0	0.29	149.6	477.5	1.0	0.21	276.8
RTDM(1.0)	17	1.0	0.53	561.9	181	1.0	0.55	751.5
TEXT-MDL	10	1.0	0.9	4.3	110	1.0	0.91	41.1
TEXT-HASH	11.0	1.0	0.82	1.4	112.2	1.0	0.89	4.5
TEXT-MAX	11.0	1.0	0.82	1.1	109.8	1.0	0.92	4.4

(a)

	Car		Baseball		Artist	
	P	R	P	R	P	R
RTDM(0.5)	0.42	1.0	0.41	0.80	0.53	0.88
RTDM(1.0)	0.42	1.0	0.32	1.0	0.67	1.0
TEXT-MDL	0.96	1.0	1.0	0.88	1.0	1.0
TEXT-HASH	0.96	1.0	1.0	0.88	1.0	1.0
TEXT-MAX	0.96	1.0	1.0	0.88	1.0	1.0

(b)

7. Acknoweledgement

I would like to place on record my deep sense of gratitude to Prof. Santosh Biradar Project Guide Dept. of Information Technology, D.Y.Patil College Ambi Pune for his generous guidance, help and useful suggestions.

I express my sincere gratitude to ICAET 2014 International Conference for his stimulating guidance, continuous encouragement and supervision throughout the course of present work.

I also wish to extend my thanks to all my well wishers and supporters for attending my seminars and for their insightful comments and constructive suggestions to improve the quality of this project work.

References

[1] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*. World Scientific, 1989.

[2] J. Rissanen, *"Modeling by Shortest Data Description,"* Automatica, vol. 14, pp. 465-471, 1978.

[3] T.M. Cover and J.A. Thomas, *Elements of Information Theory*. Wiley Interscience, 1991.

[4] H. Zhao, W. Meng, Z. Wu, V. Raghavan, and C. Yu, "Fully Automatic Wrapper Generation for Search Engines," *Proc. 14th Int'l Conf. World Wide Web (WWW)*, 2005.

