Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 13, Number 2 (2017), pp. 719-732

http://www.ripublication.com

Optimized Template Detection and Extraction Algorithm for Web Scraping of Dynamic Web Pages

Gaurav Gupta[1] and Indu Chhabra[2]

[1] CSIR-CSIO, Sector 30C, Chandigarh, India.

[2]DCSA, PU, Chandigarh, India.

Abstract

There are a huge number of dynamic websites spanning across the World Wide Web and containing manifold text information that may be helpful for data analysis. Although these dynamic websites generates thousands to millions of web pages for a certain search criteria by user, however, still they represent few unique web templates. Thus identifying the underlying web templates may result in efficient and optimized extraction of information from these web pages. Different techniques such as text_MDL, Minhash jaccard coefficient and Minhash dice coefficient have been used for web template detection and extraction. These path based techniques are complex in nature and not feasible when the number of web documents are very large. A feature based web data extraction algorithm is presented that can identify the underlying template of the web pages represented as DOM tree by clustering the similar web pages together based on feature similarity of web pages and comparing other dynamic web pages with the identified template. It enables processing and extraction of large data sets from heterogeneous web pages in reliable time. We have applied the proposed algorithm on live dynamic web pages of patent portal to compare the result against existing web extraction algorithms and found to be more efficient in terms of throughput.

Keywords: Clustering, Document Object Model (DOM) tree, Web Extraction, Template Detection

INTRODUCTION

Dynamic websites returns large number of web pages when a user searches for a particular data in it. Though, these web pages provide significant information to the users, there are certain obstacles that may make it difficult for the users to access the information efficiently and frequently. These web pages keep undergoing dynamic changes and it becomes a very challenging task to visualize the relations between all these web pages. Extraction of data from such websites becomes very tedious. It is needed to generate an optimized algorithm for the extraction of data from such websites.

A lot of attention is being paid to template detection and extraction these days as it is useful in improving the performance of many web applications like search engine optimization, data integration from multiple sources, controlling website phishing attacks (Arasu and H. Garcia-Molina 2003; Garofalakis et al. 2000; Wanawe et al. 2014). The web pages are axiomatically populated with the content by using the common underlying templates.

These templates make it easy for the users to access the contents because of the consistency provided by the template in the structure of the web pages. The distance measure (Thakare and Bagal 2015) is used to find out the similarity between DOM trees with respect to the template extracted in the first step. The proposed algorithm process enables the optimized extraction of data from the heterogeneous web pages.

The algorithm should be able to extract the data from the web pages irrespective of the dynamic changes going on in the underlying structure of the web pages. The extraction algorithm consists of mainly two steps- Template extraction and template matching. A web template represents the generic structure of the web pages it represents. An HTML page can be represented by a DOM tree with the nodes representing the HTML tags and the edges between the nodes representing the hierarchy of the relation between various nodes. We have proposed feature based extraction algorithm for the detection of web template and extraction of data values from corresponding web pages. Section 2 presents a survey of related work. Section 3 describes the proposed work. In section 4, the experiments and results are discussed and section 5 states the conclusion.

RELATED WORK

There have been many different approaches used for extracting the data from the web pages. Template detection and extraction leads to enhancement in the performance of many web applications. Arasu et al. (2003) presented an algorithm that takes a certain set of web pages as input and provides an output by extracting the values encoded in the pages. Crescenzi et al. (2001; 2005) discussed the problem of data extraction in

detail. Chowdury et al. (2008) proposed a technique for automatically extracting the data from heterogeneous web pages without considering the dynamic features of web pages. Dhillon et al. (2003) presented an approach, in which a small number of documents were taken initially and clustered, and then, the other documents were classified according to the closest clusters. Emilio Ferrara et al. (2014) provided a classification framework for grouping the web data extraction applications at two levels i.e. Enterprise level and Social web level. Chulyun Kim et al. (2011) carried out an in-depth analysis of the use of Minimum Description Length(MDL) principle for the purpose of clustering the web documents. Kento Ikeda et al. (2011) evaluated the similarities between tree structured data using the concept of tree edit distance as well as characteristics of DOM tree such as number of nodes, depth etc. Gondse et al. (2014) illustrated the concept that web pages can be split into block structures. Tools like html parser may be used to separate the informatory content from non-informatory content. Teena Merin Thomas et al. (2012) suggested a template extraction technique by performing clustering of underlying templates. Rashmi D Thakare et al. (2015) studied various clustering techniques for clustering the web documents. Loet Leydesdorff et al. (2012) developed a technique for generating the google maps from the patent information available at the USPTO website. Kaizhong Zhang et al. (1996) illustrated the problem of comparing connected, acyclic, undirected graphs having labelled nodes. Davi de Castro Reis et al. (2004) presented an approach for automatically extracting the data from the websites using tree structures for the web pages.

PROPOSED WORK

A feature based web page template detection and classification algorithm is proposed by implementing K Means algorithm using Cityblock distance. The United States Patent and Trademark Office (USPTO) website is being considered as a case study for implementing the algorithm. The website consists of a large number of dynamic web pages. It may return number of pages corresponding to a single search query. In this paper, the refined search takes place by searching for the patents granted during a particular time period. Figure 1 illustrates the main page depicting the patents granted between the dates 7/3/1979 and 8/3/1979.

The main page consists of two fields-patent number and the title corresponding to that number. Each hyperlink under the title directs the user to the corresponding page providing all necessary information regarding that particular patent.

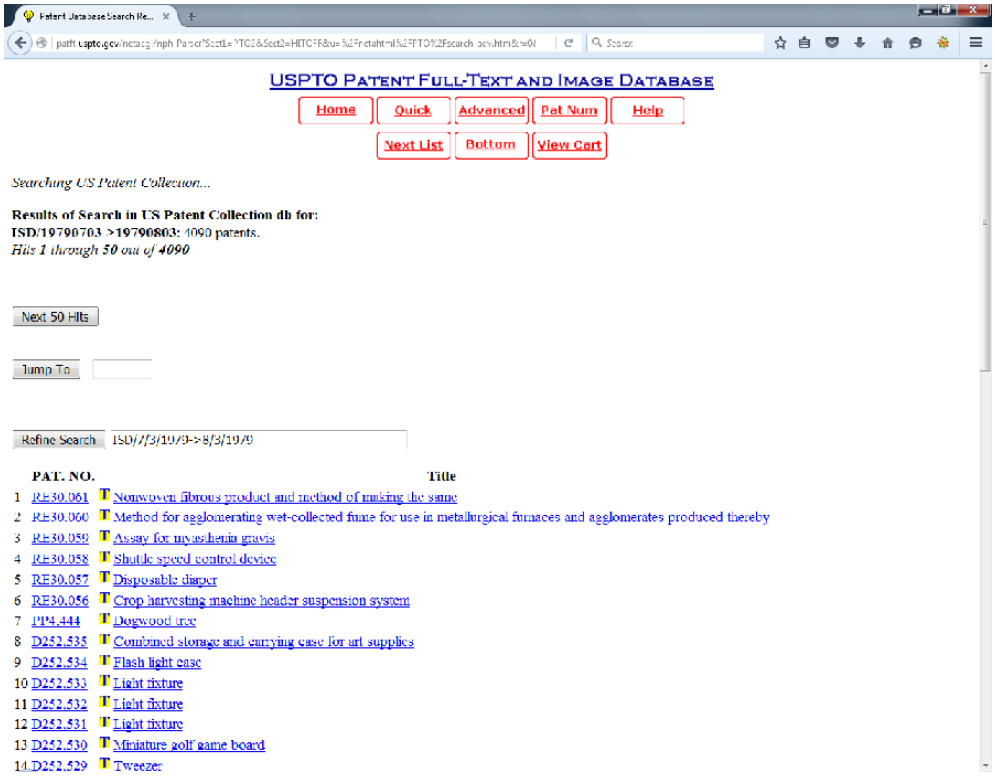


Figure 1. Main page layout of USPTO website

Feature Evaluation :

There are five features that have been chosen for the purpose of extracting the template from the web pages (Kento Ikeda et al. 2011) by performing regression analysis of the features in a pair-wise manner considering one feature as a dependent variable and another as an independent variable at a time-

(i) Number of nodes
(ii) Number of internal nodes
(iii) Number of edges
(iv) Depth of tree
(v) Number of branch nodes

Extraction Algorithm :

1) Generate DOM tree for each web page.
2) Calculate the value of five features for each web page.
3) Generate a matrix representing the values of five features in each of the web documents.
4) Apply k-means clustering technique to the matrix.

5) Calculate the values of five features for a new web page being used as a test web page. Calculate the distance of the new test point from each of the generated clusters using a distance measure.
6) Assign new test point to the cluster from which distance is minimum.

The stepwise description is given below-

Generation of DOM trees

Each web page can be represented by a DOM tree. The nodes in the DOM tree represent the tags used in HTML pages. Figure 2 represents a web page of USPTO site. A DOM tree is generated for each web page with the nodes representing the tags of the html page and the edges representing the hierarchical relations between the nodes.

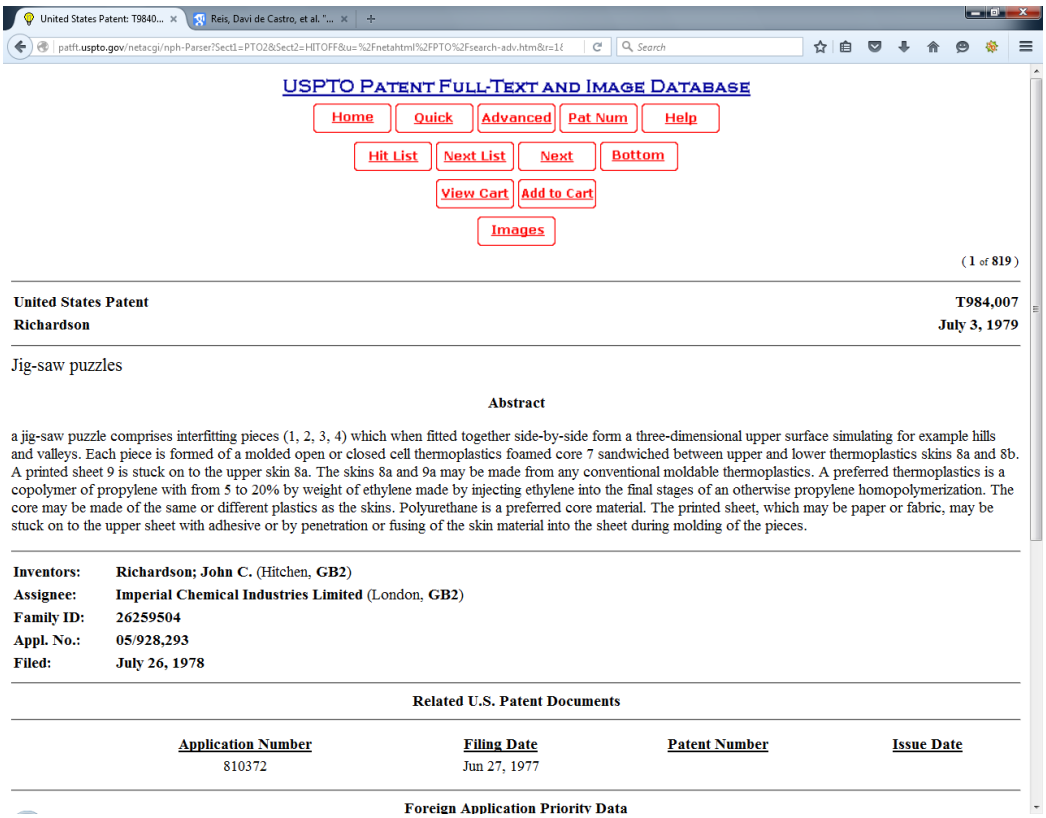


Figure 2. A web page of USPTO site

Template detection and extraction

The web pages confirm to a common template. This template is to be detected and extracted from the web pages. Each of the web pages confirms to a common template. The template is detected and extracted by applying another technique to the web documents in order to find out the similar and dissimilar documents.

Feature Calculation

The values of the five features are computed for all the web pages corresponding to the search query. Afterwards, a matrix is generated with dimensions m*n where m is the number of web pages resulting from the query and n is the number of features.

Clustering of web documents

The clustering of the web documents is performed using k-means clustering to find out the pages that are similar and dissimilar to one another.

Tree Matching

The template extracted as a result of clustering is used to find out the cluster to which a new page belongs. The values of the five features are calculated for that particular web page. Afterwards, the distance of that particular test matrix is computed from each of the clusters using cityblock distance measure. The web page belongs to the cluster which is at minimum distance from the test matrix. The formula for cityblock distance is given by-

$$d(x,c)=\sum_{j=1}^{p}|xj-cj|$$

i.e., the sum of absolute differences. Each centroid is the component-wise median of the points in that cluster.

EXPERIMENTS AND RESULTS

Experimental Setup :

The implementation is carried out on USPTO Patent portal using cluster analysis soft computing technique in a PC with an Intel Core2 Duo CPU 2.33 GHz processor with 4 GB RAM. Eclipse 3.8 IDE, Jsoup-1.8.3.jar in java and MATLAB R2013a are used to perform the extraction in an optimized manner.

For the pilot study, we have extracted 819 pages from USPTO website by searching all the patents granted on 3 July 1979 by specifying the date in the search query. To develop an optimized and general template, 819 randomly selected web pages are considered to extract their features in form of DOM trees. For the search query, a page opens consisting of various hyperlinks starting from patent title "Jig-saw puzzles" and ending at "Tie holder". On clicking each hyperlink, a page opens consisting of all the information regarding the patent. The values of the features extracted from each of the web pages are calculated as shown in Table 1.

Table 1. Values of extracted features from web pages

	Total number of nodes	Number of internal nodes	Number of edges	Depth of tree	Number of branch nodes
1	210	149	209	9	150
2	177	126	176	9	127
3	194	139	193	9	140
4	194	139	193	9	140
5	158	115	157	9	116
6	197	140	196	9	141
7	157	114	156	9	115
8	409	200	408	10	201
9	674	242	673	11	243
10	228	159	227	11	160
11	196	139	195	11	140
12	214	143	213	11	144
13	200	137	199	11	138
14	203	135	202	11	136
15	213	145	212	11	146
16	205	141	204	11	142
17	219	160	218	11	161
18	197	140	196	11	141
19	197	140	196	11	141
20	197	140	196	11	141
21	240	179	239	11	180
22	237	163	236	11	164
23	216	151	215	11	152
24	208	147	207	11	148
25	206	147	205	11	148
26	222	146	221	11	147
27	200	136	199	11	137
28	212	147	211	11	148
29	228	155	227	11	156
30	220	155	219	11	156
31	220	149	219	11	150
32	210	142	209	11	143
33	217	150	216	11	151
34	247	169	246	11	170
35	198	135	197	11	136
36	207	140	206	11	141
37	207	140	206	11	141
38	240	165	239	11	166

Results and Discussion :

Once the general features like number of nodes, number of internal nodes, number of edges, depth of tree and number of branch nodes are obtained, k-means clustering technique is applied with a starting cluster size of 3 in order to find out the major different kinds of templates that are supported by the input data.

Figure 3(a) depicts the clustering results when number of clusters is 3.

When number of clusters is 3, 4 or 6, the results are not very appropriate because certain dissimilar web pages appear to be merged together into one cluster. When number of clusters is 5, there is more accuracy in getting similar web pages. Hence, five clusters have been taken because after introducing 6th cluster, though six different clusters are formed but the centroid are not as optimally located as for 5 clusters as observed in figure 3(c) and figure 3(d). Afterwards, the clustering is applied with size 4, 5 and 6. Figure 3(b), 3(c) and 3(d) depict the clustering results when number of clusters is 4, 5 and 6 respectively.

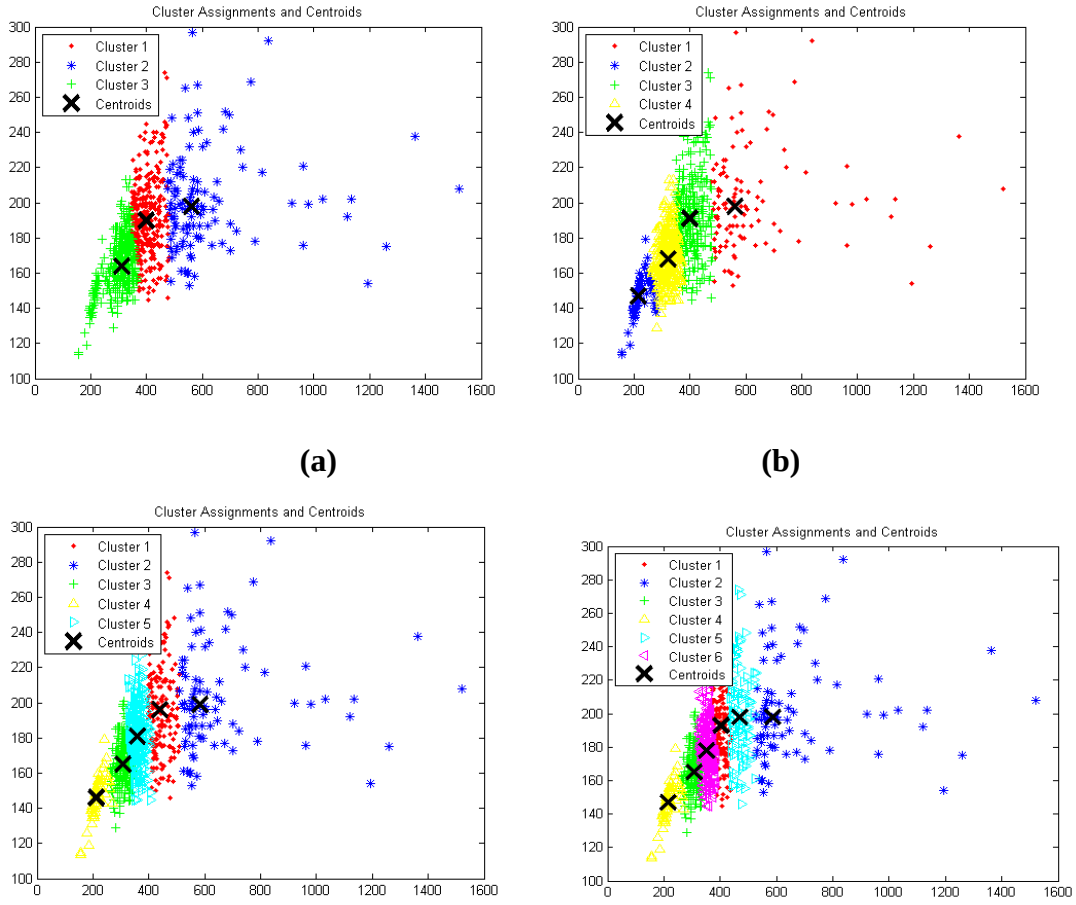
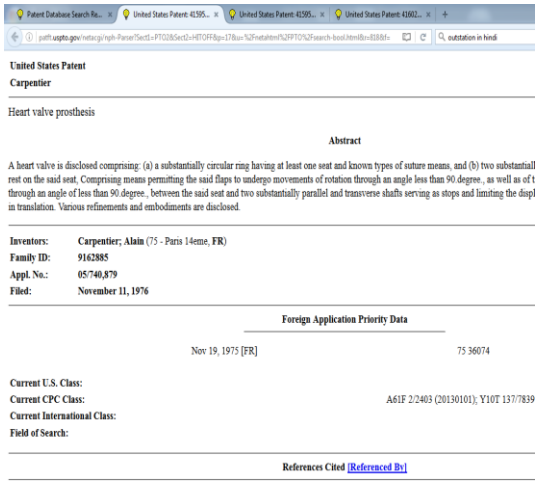


Figure 3 Clustering results for varying cluster for optimum size

a Cluster size=3, **b** Cluster size=4, **c** Cluster size=5, **d** Cluster size=6

In figure 3(d), the introduction of new 6th cluster has replaced the portions of 1st and 2nd clusters, so has found subcategory within already existing data. But as templates must be general, different and minimum in frequency to represent whole document as much as possible, hence cluster size of 5 is providing the optimal value having sufficient number of web documents possessing common features of tags such as paragraphs, tables, hyperlinks, image etc. examples shown in Figure 4(a), 4(b), and 4(c).



United States Patent
Carpentier

Heart valve prosthesis

Abstract

A heart valve is disclosed comprising: (a) a substantially circular ring having at least one seat and known types of suture means, and (b) two substantiall rest on the said seat, Comprising means permitting the said flaps to undergo movements of rotation through an angle less than 90 degree., as well as of t through an angle of less than 90 degree., between the said seat and two substantially parallel and transverse shafts serving as stops and limiting the displ in translation. Various refinements and embodiments are disclosed.

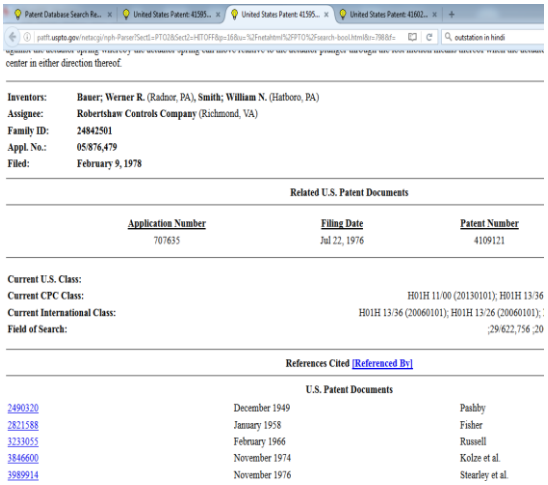
Inventors: Carpentier; Alain (75 - Paris 14eme, FR)
Family ID: 9162885
Appl. No.: 05/740,879
Filed: November 11, 1976

Foreign Application Priority Data

Nov 19, 1975 [FR] 75 36074

Current U.S. Class:
Current CPC Class: A61F 2/2403 (20130101); Y10T 137/7839
Current International Class:
Field of Search:

References Cited [Referenced By]



center in either direction thereof.

Inventors: Bauer; Werner R. (Radnor, PA), Smith; William N. (Hatboro, PA)
Assignee: Robertshaw Controls Company (Richmond, VA)
Family ID: 24842501
Appl. No.: 05/876,479
Filed: February 9, 1978

Related U.S. Patent Documents

Application Number	Filing Date	Patent Number
707635	Jul 22, 1976	4109121

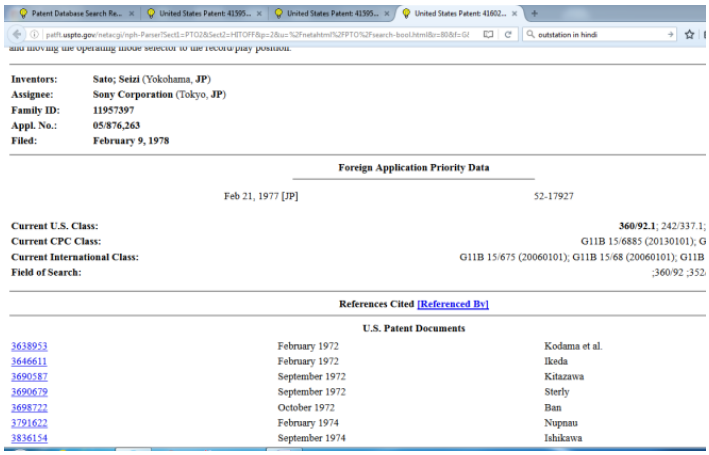
Current U.S. Class:
Current CPC Class: H01H 11/00 (20130101); H01H 13/36
Current International Class: H01H 13/36 (20060101); H01H 13/26 (20060101);
Field of Search: ;29/622,756 ;20

References Cited [Referenced By]

U.S. Patent Documents

2490320	December 1949	Pashby
2821588	January 1958	Fisher
3233055	February 1966	Russell
3846600	November 1974	Kolze et al.
3989914	November 1976	Stearley et al.

(a) **b)**



Inventors: Sato; Seizi (Yokohama, JP)
Assignee: Sony Corporation (Tokyo, JP)
Family ID: 11957397
Appl. No.: 05/876,263
Filed: February 9, 1978

Foreign Application Priority Data

Feb 21, 1977 [JP] 52-17927

Current U.S. Class: 360/92.1; 242/337.1;
Current CPC Class: G11B 15/6885 (20130101); G
Current International Class: G11B 15/675 (20060101); G11B 15/68 (20060101); G11B
Field of Search: ;360/92 ;352

References Cited [Referenced By]

U.S. Patent Documents

3638953	February 1972	Kodama et al.
3646611	February 1972	Ikeda
3690587	September 1972	Kitazawa
3690679	September 1972	Sterly
3698722	October 1972	Ban
3791622	February 1974	Nupnau
3836154	September 1974	Ishikawa

(c)

Figure 4. Web pages representing web template

a WebPage 1, **b** WebPage 2, **c** WebPage3

As indicated by optimized cluster size of 5, there are general 5 templates to cover the complete structural layout of input data of 819 randomly selected pages. Table 2 represents the centroids calculated for each of the five clusters formed.

Table 2 derived result showing five common templates

	No. of nodes (f1)	No. of internal nodes (f2)	Number of edges (f3)	Depth of tree (f4)	Number of branch nodes (f5)
Template 1	584	199	583	12	200
Template 2	214	147	213	11	148
Template 3	442	196	441	12	197
Template 4	361	181	360	12	182
Template 5	309	165	308	12	166

Now, certain test pages are to be taken to find out the cluster to which the test pages belong. Seven random test pages are taken as samples and compared with the 5 extracted templates by comparing the features of test pages with the centroids of selected clusters. The deviation results are given in table 3.

Table 3: Derived Result Showing Deviation of Sample Data From Templates

Web Page (WP)	Distance from cluster 1	Distance from cluster 2	Distance from cluster 3	Distance from cluster 4	Distance from cluster 5
WP1	564	281	274	82	54
WP2	332	513	54	150	286
WP3	114	731	176	368	504
WP4	734	147	444	252	116
WP5	1027	182	737	545	409
WP6	1310	1975	1588	1720	1792
WP7	522	323	232	40	96

The distance of the matrix formed for each of the test web pages is calculated from each of the clusters using the cityblock distance as a measure. The formula for cityblock distance is given by-

$$d(x,c)=\sum_{j=1}^{p}|xj-cj|$$

i.e., the sum of absolute differences. Each centroid is the component-wise median of the points in that cluster. The distances are depicted in Table 3.

The web page belongs to the cluster (template) from which its cityblock distance is minimum. A total of five optimal clusters are identified to distinguish group of extracted HTML pages. As the minimum distance in each row represents the web page deviation from that particular template, so, web page1 belongs to the cluster 5 because its distance from cluster 5 is minimum and thus webpage1 belongs to template 5. As also can be seen in figure 4a, webpage1 is missing "Assignee" row tag in HTML table while it has "Foreign Application Priority Data" HTML table tag with features such as number of nodes, internal nodes, edges, depth and branch nodes represented in table 3 having centroid value i.e. 54 which is most nearer to cluster 5 as shown in table 2. Similarly, webpage2 in figure 4(b) is having "Assignee" row HTML tag while "Foreign Application Priority Data" HTML table tag is missing. In case of webpage3 as shown in figure 4(c) is most nearer to cluster 3 having both "Assignee" row tag and "Foreign Application Priority Data" HTML table tag. The same process is performed subsequently for other web pages.

Hence, the structural representation of that test web page in form of number of nodes, number of internal nodes, number of edges, depth of tree and number of branch nodes can be easily connected to other links for the further analysis of transformation and loading process.

The performance metrics of the implementation of the algorithm are given in Table 4.

Table 4: Performance metrics comparative analysis

Template extraction techniques	Text_MDL	Minhash Jaccard coefficient	Minhash Dice coefficient	Proposed Feature based
Execution Time (milliseconds)	17.939	0.968	0.935	0.0025365

We have compared the three techniques, i.e., text_MDL, Minhash jaccard coefficient and Minhash dice coefficient (Chulyun Kim and Kyuseok Shim 2011; Trupti B. Mane and Girish P. Potdar 2013) for clustering the web pages based on their templates are compared. These path based techniques are complex in nature and not feasible when the number of web documents are very large. These also results in a lot of calculative work. This drawback is overcome by feature based technique of cityblock distance. The feature based technique for extraction is applied using Cityblock distance on clusters. The size of the clusters can be varied based on optimal template characterisation of data sets. This feature based technique considers five

relevant features and centroid is calculated by applying cityblock distance. Now, for every incoming webpage, its corresponding features are calculated and minima of cityblock distance is calculated for every cluster. This results in most relevant features of webpage grouped to its template cluster. The earlier techniques i.e. text_MDL, minhash jaccard coefficient and minhash dice coefficient are based on path based method where unique path for every node is to be calculated and matrix calculation is performed which limits the scope of webpage extraction and throughput. The proposed technique helps to reduce the calculative work as well as makes it easier to detect and extract the template. The time taken by feature based technique is approximately 0.0025365 milliseconds that is much less compared to the three techniques i.e. text_MDL, minhash jaccard coefficient and minhash dice coefficient that have execution times of 17.939 milliseconds, 0.968 milliseconds and 0.935 milliseconds.

CONCLUSION

This paper has proposed feature based data extraction technique in contrast to existing path based extraction techniques for web scraping. This has made the extraction process from heterogeneous web pages more efficient and fast as can be observed from table 4 wherein the execution time is much less in comparison to other path based techniques. The throughput is much higher for generation of templates and classifying web pages. This enables the data extraction from a huge number of heterogeneous web pages unless the earlier techniques wherein efficiency is inversely proportional to the population of web pages. The use of features for data extraction makes the process quite useful and optimized.

We proposed extraction algorithm with template generation and detection based on features of DOM tree. Extensive experiments for evaluating the efficiency and effectiveness of the proposed algorithm are performed. The result shows that efficiency of web page scrapping can be increased with the feature based technique as it is more robust and controlled.

ACKNOWLEDGMENTS

The author would like to express United States Patent and Trademark Office, www.uspto.gov web pages source for providing access for extracting data from web pages. He further wishes to acknowledge the reviewers for their valuable and helpful comments.

REFERENCES

[1] Abdur Chowdury, Ling Ma & Nazli Goharian (2008) Automatic Data Extraction from Template Generated Web Pages. Journal of Software, vol.19, pp.209-223

[2] Arasu & H. Garcia-Molina (2003) Extracting Structured Data from Web Pages. Proc. ACM SIGMOD.

[3] M.N. Garofalakis, A. Gionis, R. Rastogi, S. Seshadri & K. Shim (2000) Xtract: A System for Extracting Document Type Descriptors from Xml Documents. Proc. ACM SIGMOD.

[4] Davi de Castro Reis, Paulo B. Golgher, Altigran S. da Silva & Alberto H. F. Laender (2004) Automatic web news extraction using tree edit distance. Proceedings of the 13th international conference on World Wide Web. ACM, pp. 502-511.

[5] I.S. Dhillon, S. Mallela & D.S. Modha (2003) Information Theoretic CO-Clustering. Proc. ACM SIGKDD.

[6] Emilio Ferrara and Giacomo Fiumara (2014) Web data extraction, applications and techniques: A survey Knowledge-based systems, 301-323.

[7] Kento Ikeda, Takashi Kobayashi, Kenji Hatano & Daiji Fukagawa (2011) Calculating Similarities between Tree Data Based on Structural Analysis. Intelligent Decision Technologies. Proceedings of the 3rd International Conference on Intelligent Decision Technologies (IDT 2011), pp.719-729, Springer-Verlag Berlin Heidelberg.

[8] Kim Chulyun & Kyuseok Shim (2011) Text: Automatic template extraction from heterogeneous web pages. Knowledge and Data Engineering, IEEE Transactions on, vol. 23 no. 4, pp. 612-626.

[9] Gondse, Ms Pranjali G. & A. Raut, (2014) Main Content Extraction From Web Page Using DOM. International Journal of Advanced Research in Computer and Communication Engineering.

[10] Teena Merin Thomas and V. Vidhya (2012) A Novel Approach for Automatic Data Extraction from Heterogeneous Web Pages ," International Conference on Emerging Technology Trends on Advanced Engineering Research (ICETT'12) Proceedings published by International Journal of Computer Applications (IJCA).

[11] Thakare, Rashmi D. & Manisha R. Patil (2015) Extraction of Template using Clustering from Heterogeneous Web Documents, International Journal of Computer Applications, vol. 119.11 pp. 23-31.

[12] Leydesdorff, Loet & Lutz Bornmann (2012) Mapping (USPTO) patent data using overlays to Google Maps. Journal of the American Society for Information Science and Technology, vol. 63.7, pp. 1442-1458.

[13] Mane B. Trupti & Potdar P. Girish (2013) An Approach for Template Extraction from Heterogeneous Web Pages, Proc. of Int. Conf. on Advances in Computer Science, AETACS, Elsevier

[14] Thakare, Y. S. & S. B. Bagal (2015) Performance Evaluation of K-means Clustering Algorithm with Various Distance Metrics. International Journal of Computer Applications, vol. 110.11.

[15] V. Crescenzi, G. Mecca & P. Merialdo (2001), Roadrunner: Towards Automatic Data Extraction from Large Web Sites. Proc. 27th Intl Conf. Very Large Data Bases (VLDB).

[16] V. Crescenzi, P. Merialdo & P. Missier (2005) Clustering Web Pages Based on their Structure. Data and Knowledge Eng., vol. 54, pp. 279- 299.

[17] Wanawe, Kranti, Supriya Awasare & NV Puri (2014) An Efficient Approach to Detecting Phishing A Web Using K-Means and Naïve-Bayes Algorithms. International Journal of Research in Advent Technology, vol.2, no. 3.

[18] Zhang, Kaizhong, Jason TL Wang & Dennis Shasha (1996) On the editing distance between undirected acyclic graphs. International Journal of Foundations of Computer Science 7.01, pp: 43-57.