International Journal of Engineering Sciences & Emerging Technologies, June 2013.
ISSN: 2231 – 6604 Volume 5, Issue 2, pp: 119-126 ©IJESET

HAULING TEMPLATES FROM WEB PAGES USING CLUSTERING TECHNIQUES

R.Manjula[1] and A.Chilambuchelvan[2]
[1] Research Scholar, Department of CSE, R.M.K Engineering College Chennai, India
manjula2702@gmail.com
[2] Professor, Department of CSE, R.M.K Engineering College, Chennai, India
chill97@gmail.com

ABSTRACT

In today's world, World Wide Web is the most popular information providers. A website is a collection of web pages and Web pages usually include information for the users. The web sites are designed with common templates and content. The template is used to access the content easily by consistent structures even the templates are not explicitly announced. The current Template extraction techniques are degrading the performance of web applications such as search engine due to irrelevant terms in templates.In this work, we present new method for extracting templates from a large number of web documents which are generated from heterogeneous templates. This paper cluster the web documents based on the similarity of underlying template structures in the documents so that the template for each cluster is extracted simultaneously.

KEYWORDS: *Document Object Model, Minimum Description Length, Template Extraction, VIPS.*

I. INTRODUCTION

To tremendous growth of World Wide Web, lot of information is published on Web through Web pages. Many Web pages are generated online for Web site maintenance, flexibility, and scalability purposes. The Web pages consist of information which is published and designed by Templates [1, 5] and they are usually generated by putting page content stored in back-end databases into predefined templates. Templates are otherwise called as non-content blocks. It is used mostly in dynamically generated Web pages. E.g. Commercial Websites such as online shopping. Web pages make use of these templates to display their day to day new products in Web sites. There exists much redundant and irrelevant information in these Web pages, such as navigation panels, advertisements, catalogs of services, and announcements of copyright and privacy policies which are distributed over almost all pages of a systematic Web site. Such information is still crawled and indexed by search engines and information agents, thus significantly increasing corresponding storage and computing overhead. Lot of research has been done in extracting templates [2, 3, 4, 6, and 10] from web pages. But the existing methods are failed to improve the performance of the web applications. In this paper our new Approach is used to extract templates from web pages with better performance. This work consists of the following steps: Initially the web pages developed from multiple templates are downloaded and stored in database. The downloaded web pages are parsed using an html parser and DOM tree is constructed. Then the constructed DOM is divided into number of blocks using vision-based page segmentation (VIPS) algorithm. Support Vector (SV) based classifier is used to identify the non-content blocks. Third step Applying Hierarchical algorithm for the Web pages in order to find the similarities between Web Pages in order to eliminate irrelevant non content blocks which degrade the performance of the search engines.

The rest of the paper is organized as follows: In Section II we have discussed the related work. We define the proposed system in Section III and a few related terms in Section IV, V, VI, VII, VIII and XI. We indicate our conclusion in Section X.

II. RELATED WORKS

There are many methods which have proposed for hauling Templates from web pages. [3] Vieira *et al* says templates are detected by finding identical nodes in Document Object Model (DOM) trees and subtrees by performing mapping between the tree structures of web pages.
Liu *et al* [7] proposed DEPTA from extracting structured data from web pages. It relies on DOM trees and performs mining from single web pages. It compares only adjacent substrings with starting tags having the same parent in the HTML tag tree.
Crescenzi *et al* [17] also assumes that every HTML tag is generated by the template. It extracts template by analyzing a pair of web pages based on Matching technique called ACME (Align, Collapse under Mismatch, Extraction). RoadRunner does not rely on any priori knowledge about the about page contents.
EXALG [8] similarly does not make full use of the tree structure (DOM). EXALG first discovers the unknown template that generated the pages and uses the discovered template to extract the data from the input pages. Roadrunner and EXALG analyzed number of web pages.
R. Henzinger *et al* [10] says the templates present in web pages are considered to be great challenge in search engine since, it degrades the performance of the search engines. Yi and Liu [9], have proposed an algorithm for eliminating noisy blocks from Web-pages. They took multiple web pages from single Web site as input. They considered the similar style information in multiple web pages as noise.
Song et.al. [12] used VIPS algorithm to identify blocks in Web-pages. Bar-Yossef and Rajagopalan [13] proposed local template detection algorithm and global template detection algorithm for identifying templates which violate hypertext IR principle web pages. Kushmerick [14] identified and removed Internet advertisements in web page and generated wrappers.

III. PROPOSED SYSTEM

This section describes the proposed system that extracts templates from web pages using Template Extraction from web pages (TEW) algorithm. Fig.1 shows the overall architecture of the proposed system.

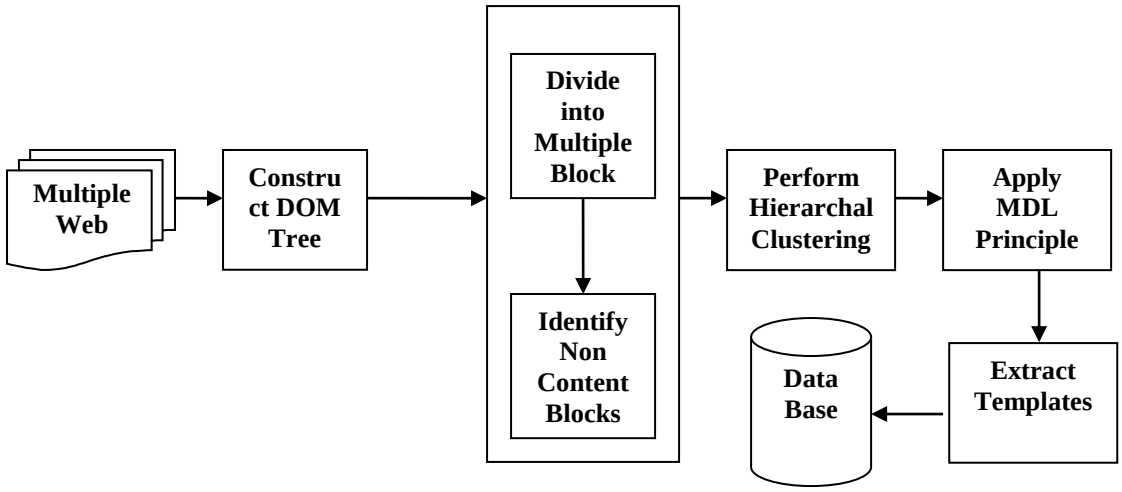


Fig.1 Proposed System Architecture

Multiple WebPages which have different templates are downloaded. Downloaded WebPages are parsed and DOM tree is constructed in order to reduce into number of small blocks. These Blocks are composed of both content and non content blocks. Templates are present in non content blocks, Support vector machine is used to separate the non content blocks from the blocks of web pages. Some items in a non content block will degrade the performance of search engine. Hence hierarchical agglomerative clustering is performed to find the common non content blocks among the web pages

downloaded. This clustering technique is used to group the common items in the each non content blocks into a cluster. Cost efficiency of this algorithm is reduced by applying Minimum description Length principle by eliminating the clusters which uses large number of bits to define itself. Finally the Templates are extracted and stores in database. The templates stored in database can be further used by the web designer to develop WebPages which does not corrupt the performance of the search engines. It also enables the web designer to develop web pages fast and easy.

The proposed TEW algorithm is as follows

Algorithm TEW

Begin

1. Download Web pages which are to be composed.
2. Parse the Web pages using DOM parser and construct a DOM (Document Object Model) tree.
3. Convert the DOM tree into visual Blocks using VIPS (Vision-Based Page Segmentation) algorithm.
4. Identify the non- content blocks using SVM classifier.
5. Cluster the non-content blocks using Hierarchical Clustering method.
6. Store extracted templates in database.

End

IV. DOM TREE CONSTRUCTION

Document Object Model (DOM) is used to represent a web page in tree structure [5]. DOM splits a tree into many small sub trees and nodes, which belong to the DOM, have descending relationship with each other. Fig 2 shows the DOM tree structure of the Web pages.

The sub elements of the <table> shows some of the partitioning tags such as <td><tr> which further partition the web pages into distinct structure. The leaf node consists of a contents present in Web pages it is denoted by c1,c2,…c6 in fig 2, Or it may consists of hyperlink of a document denoted by d1,d2,..d4.

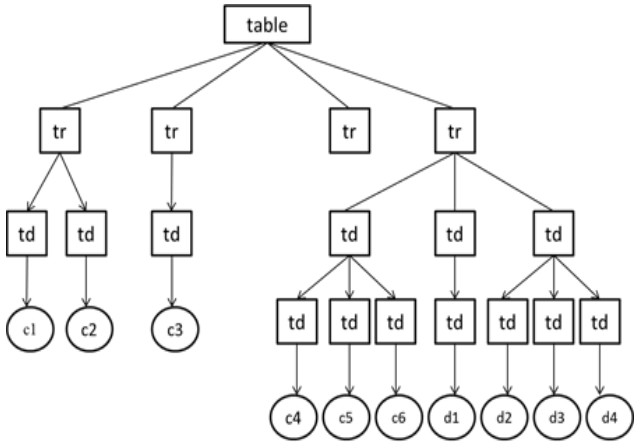


Fig. 2 DOM Tree Structure

V. VISION-BASED PAGE SEGMENTATION (VIPS) ALGORITHM

Web page usually contains various contents such as navigation, decoration, interaction and contact information, which are not related to the topic of the web-page. Furthermore, a web page often contains multiple topics that are not necessarily relevant to each other. Therefore, detecting the semantic content structure of a web page could potentially improve the performance of web information retrieval.

Vision-based Page Segmentation [1, 11], is used to find all of the relevant regions of which a document is to be composed. The different web designers provide visual cues that help people to recognize the different regions of a document. We use VIPS algorithm to separate the web pages into

number of blocks based on visual features for e.g, horizontal or vertical rules, boxes, colored panels, special fonts, or background images [11].

Algorithm VIPS

1. For a given input document, the DOM tree is constructed, and it has information about visual features, such as
   font, color, or background image.
2. Traverse the DOM tree from root node and identify the sub regions.
3. Separators is detected by calculating weight using visual difference between the regions
4. After identifying all the regions, they are organized into a tree based on visual features is constructed.

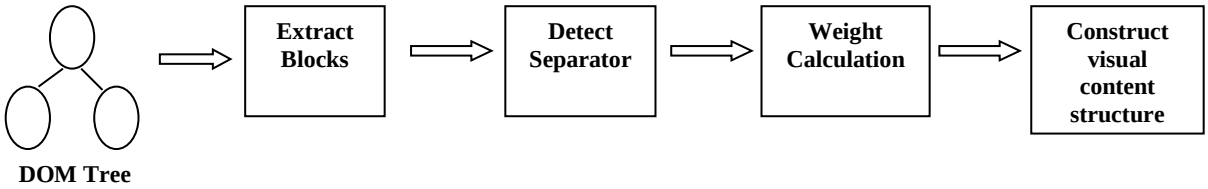


Fig 3 Diagrammatic Representation of VIPS Algorithm

VIPS algorithm is illustrated in Fig 3. First the blocks are extracted from the DOM tree. It consists of different subtree for both content and non content blocks. It is used in web pages such as yahoo which are developed based on multiple semantics. Fig 4 shows the block representation of Yahoo page. Yahoo page consists of News, Advertisement, search box, footers, related links, and images. Each and Everything in yahoo page is considered as blocks.

Then it tries to find the separators between these extracted blocks. Here, separators denote the horizontal or vertical lines in a web page that visually cross with no blocks. Finally, based on these separators, the semantic structure for the web page is constructed. VIPS algorithm employs a top-down approach, which is very effective.

VI. SUPPORT VECTOR MACHINE

Vapnik [15] introduced a new learning method called SVM. It is used for classifying non content blocks. It separates the data belonging to different classes using a region. It transforms original training data into high dimensional data by performing nonlinear mapping. It is used for both prediction and classification. Here it is used for classifying the non-content block of data from the content block. SVM based algorithm achieves good performance in identifying the non-content blocks. The blocks are used as vectors in Support Vector classifier in order to identify the non-content blocks among the web pages. SVM effectively classifies the non contents blocks of each among the blocks of web page and its hyperlinks.

Fig 4 Block representation of Yahoo Home Page

VII. HIERARCHICAL CLUSTERING

Web page clustering Methods used similarity measures based on calculating distance between DOM trees [16]. Buttler proposed algorithm for Document structure similarity .However, In order to manage large amount of web pages in web, clustering of web pages are performed. Hierarchical clustering is the most popular clustering technique used in clustering web documents**.** Agglomerative clustering is considered to be monotonic since the similarities between the clusters get decreases with the level of clustering. Each non-content block as an individual cluster and then successively merges pairs of clusters. It performs single link clustering .It repeats the clustering process until all clusters are merged into a single cluster by merging two closest groups of clusters.

MDL is an agglomerative hierarchical clustering algorithm which starts with each input document as an individual cluster. When a pair of clusters is merged, the MDL cost of the clustering model can be reduced or increased. For each new cluster MDL cost is calculated. The final output cluster contains the template material which is common to the multiple web pages.

The proposed agglomerative algorithm with MDL principle is as follows:

Algorithm for clustering

Input: NC multiple non content blocks nc_1,nc_2,nc_3....nc_m

Output: Single cluster C having common template

Begin

1. Consider an each non content block as single cluster
2. Then Merge the pair of most similar cluster
   For each cluster
   Calculate MDLcost
   do
   If (MDL_{new} < MDL_{old})
   Accept new pair as cluster
   else
   Accept old pair as cluster
3. Repeat until a single cluster C is formed

End

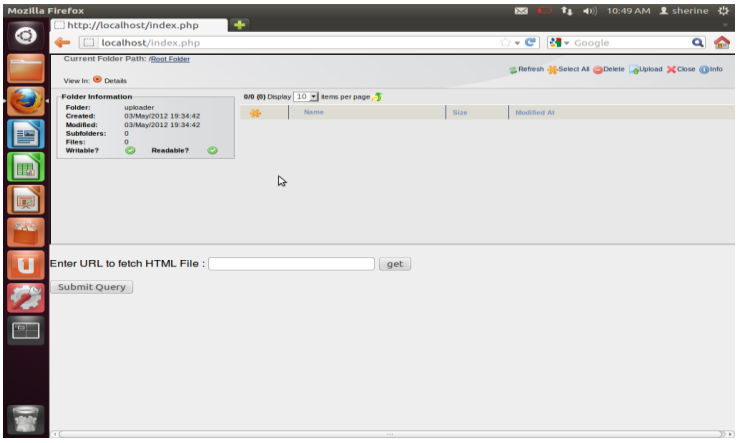
VIII. MINIMUM DESCRIPTION LENGTH

MDL [13] (Minimum Description Length) principle states that the best model to describe a set of data is the one which minimizes the sum of: (1) the length of a string of bits which encodes the model, and (2) The length of a string of bits which encodes the data with the help of the model: the shorter the description, the better the model. We use this principle along with the agglomerative clustering in order to reduce the cost and efficiency of clustering. MDL cost for the each cluster is calculated. This principle finds the pair of clusters whose MDL cost is low in each step of merging and the pair is repeatedly merged until any reduction is not possible. In order to calculate the MDL cost when each possible pair of clusters is merged, Cluster having low MDL cost is accepted as the final output of agglomerative clustering.

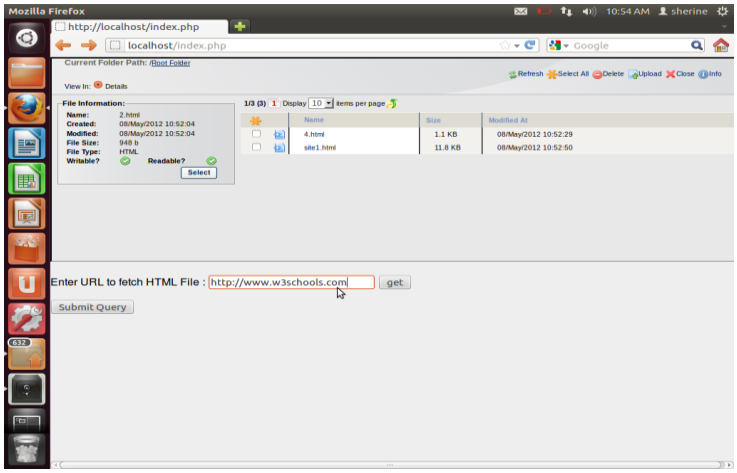
IX. EXPERIMENT RESULTS

The WebPages with different templates were downloaded for testing this work. Then the web pages were parsed using HTML parser and DOM tree was constructed. The content and Non content blocks were identified using Support vector machine. This clustering technique is used to group the common items in the each non content blocks into a cluster. Minimum Description Length principle was used to reduce the cost by eliminating the clusters. Finally the extracted templates were stored in the database.

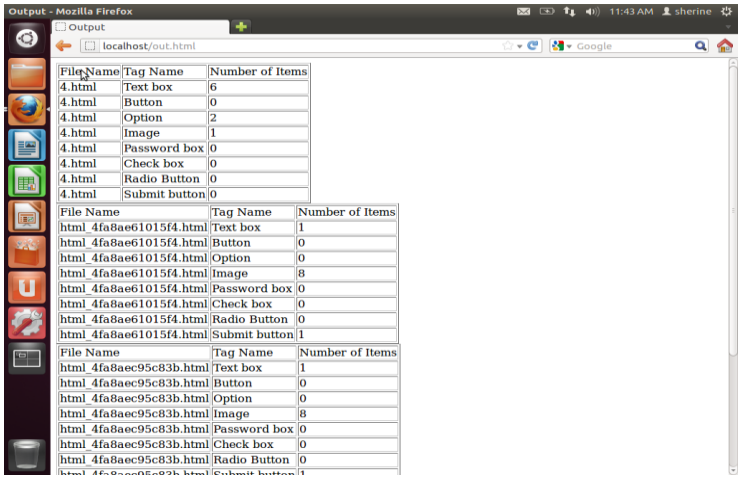
HOME Page



Template Extraction Page



Report Page



File Name	Tag Name	Number of Items
4.html	Text box	6
4.html	Button	0
4.html	Option	2
4.html	Image	1
4.html	Password box	0
4.html	Check box	0
4.html	Radio Button	0
4.html	Submit button	0

File Name	Tag Name	Number of Items
html_4fa8ae61015f4.html	Text box	1
html_4fa8ae61015f4.html	Button	0
html_4fa8ae61015f4.html	Option	0
html_4fa8ae61015f4.html	Image	8
html_4fa8ae61015f4.html	Password box	0
html_4fa8ae61015f4.html	Check box	0
html_4fa8ae61015f4.html	Radio Button	0
html_4fa8ae61015f4.html	Submit button	1

File Name	Tag Name	Number of Items
html_4fa8aec95c83b.html	Text box	1
html_4fa8aec95c83b.html	Button	0
html_4fa8aec95c83b.html	Option	0
html_4fa8aec95c83b.html	Image	8
html_4fa8aec95c83b.html	Password box	0
html_4fa8aec95c83b.html	Check box	0
html_4fa8aec95c83b.html	Radio Button	0

The Performance analysis of our work is given in Fig 5 using line Chart and bar chart.

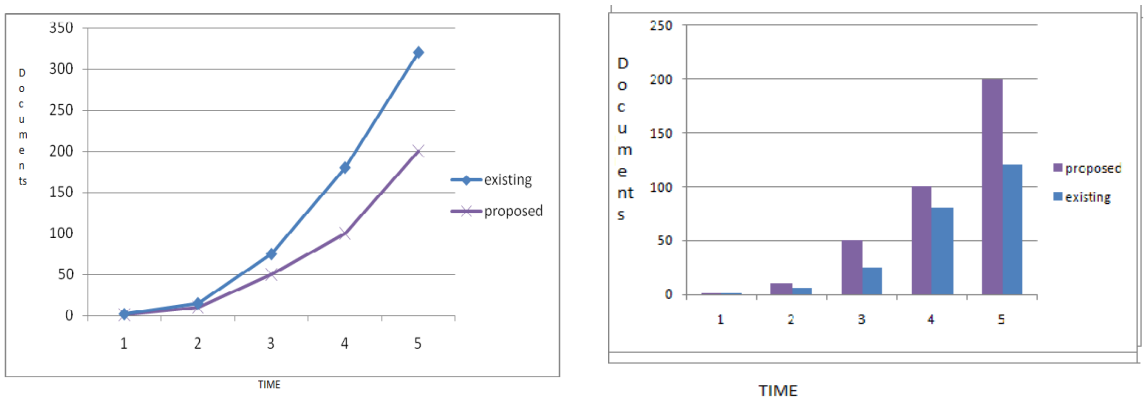


Fig 5. Performance Comparison chart

X. CONCLUSION

Templates present in web sites degrade the performance of search engine retrieval, especially in content relevance and link analysis. Template detection is an important technique since templates could heavily cripple the performance of other modules such as page classification modules or index builders In order to manage the unknown number of Templates present in the web document; this project proposed a novel approach for Extracting Templates form web documents which were developed from multiple templates. Agglomerative clustering is performed in order to reduce the size of the web documents. The resulting clusters are merged together and MDL principle is employed to manage the unknown number of clusters.

REFERENCES

[1] Hassan A. Sleiman and Rafael Corchuelo," A Survey on Region Extractors From Web Documents" , IEEE Transactions on Knowledge and Data Engineering, 27 June 2012.
[2] Chulyun Kim and Kyuseok Shim "TEXT: Automatic Template Extraction from Heterogeneous Web Pages", Data and knowledge engineering, Vol. 23, No. 4, April 2011
[3]Mohammed Kayed and Chia-Hui Chang, " FiVaTech: Page-Level Web Data Extraction from Template Pages" IEEE TRA NSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 22, NO. 2, FEBRUARY 2010.
[4] K. Vieira, A.S. da Silva, N. Pinto, E.S. de Moura, J.M.B. Cavalcanti, and J. Freire, "A Fast and Robust Method for Web Page Template Detection and Removal," Proc. 15th ACM Int'l Conf. Information and Knowledge Management , 2006.
[5] Hung-Yu Kao, Jan-Ming Ho, and Ming-Syan Chen,"WISDOM: Web Intrapage Informative Structure Mining Based on Document Object Model". IEEE Transactions on Knowledge and Data Engineering, Vol 17,No 5, MAY 2005
[6] Ling Ma,Nali Goharian, Abdur Chowdhury."Extracting unstructured Data from Template Generated Web Document". In Proc. of the CIKM'03 Conf., pages 516-519, 2003
[7] Y. Zhai and B. Liu, "Web Data Extraction Based on Partial Tree Alignment," Proc. 14th Int'l Conf. World Wide Web (WWW), pp. 76- 85, 2005.
[8] A. Arasu and H. Garcia-Molina, "Extracting Structured Data from Web Pages," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 337-348, 2003.
[9] Lan Yi, Bing Liu, Xiaoli Li. "Eliminating Noisy Information in Web Pages for Data Mining". In Proc. of the SIGKDD'03 Conf., pages 296-305, 2003
[10] Monika R. Henzinger Rajeev MotwaniCraig Silverstein, "Challenges in Web Search Engines" ACM SIGIR Forum, Volume 36 Issue 2, Fall 2002.
[11] Deng Cai, Shipeng Yu , Ji-Rong Wen and Wei-Ying Ma "VIPS: a Vision-based Page Segmentation Algorithm." November 2003
[12] Ruihua Song, Haifeng Liu, Ji-Rong Wen, and Wei-Ying Ma." Learning block importance models for web pages". In Thirteenth World Wide Web conference, pages 203–211, 2004.
[13] Ziv Bar-Yossef and Sridhar Rajagopalan. "Template detection via data mining and its applications.'' In Proceedings of WWW2002, pages 580–591, 2002.

[14] Nicholas Kushmerick. "Wrapper induction: Efficiency and expressiveness". Artificial Intelligence,118(1-2):15–68, 2000.
[15] V. Vapnik. Support-vector networks. Machine Learning, 20:273-297, November 1995
[16] D. C. Reis, P. B. Golgher, A. S. Silva, and A. F. Laender." Automatic web news extraction using tree edit distance". In WWW, 2004.
[17] V. Crescenzi, G. Mecca, and P.Merialdo, "Roadrunner: Towards Automatic Data Extraction from Large Web Sites," Proc. 27th Int'l Conf. Very Large Data Bases (VLDB), 2001.

AUTHORS

R. MANJULA received B.E Degree in 2002 from Bharathidasan University and M.E degree in Computer Science and Engineering from Anna University in 2008. She is currently research scholar under the guidance of Dr .A. Chilambuchelvan, Professor in the Department of Computer Science and Engineering in R.M.K Engineering College. Her research interests include various aspects of Knowledge and Data Discovery.

A. CHILAMBUCHELVAN obtained B.E. Degree in 1989 from Mepco Schlenk Engineering College, Sivakasi and M.E. Degree in 1994 from Coimbatore Institute of Technology, Coimbatore. He did his PhD from College of Engineering, Guindy, Anna University, Chennai in 2008. He is in the teaching profession for the past 22 years and his areas of interest are Knowledge and Data Discovery, soft computing and bio medical engineering. He published 24 papers in International journals and conferences.