

# An Efficient Web Search Engine for Noisy Free Information Retrieval

Pradeep Sahoo<sup>1</sup> and Rajagopalan Parthasarthy<sup>2</sup>

<sup>1</sup>Department of Computer Science and Engineering, Anna University, India

<sup>2</sup>Department of Computer Science and Engineering, GKM College of Engineering and Technology, India

**Abstract:** *The vast growth, various dynamic and low quality of the world wide web makes it very difficult to retrieve relevant information from internet during query search. To resolve this issue, various web mining techniques are being used. The biggest challenge in web mining is to remove noisy data information or unwanted information from the webpage such as banner, video, audio, images, hyperlinks etc. which are not associated to a user query. To overcome these issues, a novel custom search engine is proposed with efficient algorithm in this paper. The proposed Uniform Resource Locator (URL) pattern extractor algorithm will extract the all relevance index pages from the web and ranking the indexes based on user query. Then, Noisy Data Cleaner (NDC) algorithm is applied to remove the unwanted content from the retrieved web pages. The results show that the proposed URL Pattern Extractor (UPE)+NDC algorithm provides very promising results for different datasets with high precision and recall rate in comparison with the existing algorithms.*

**Keywords:** *Web content extraction, relevant information, noise data elimination, noisy data cleaner algorithm, URL pattern extractor algorithm.*

*Received November 27, 2014; accepted June 1, 2015*

## 1. Introduction

Now-a-days, search engines have become one of the biggest important applications on the internet that extract relevant information based on the user query. The internet continues to grow its capacity with various types of modern data and infiltrates into every aspect of our daily life. Hence, the development of more robust and efficient search engine is a constant requirement. The first objective of current primary search engines is to approximately rank the retrieved pages at the data level which is an old mechanism for information gathering. However, efficient information on the internet is often accompanied by a large volume of noisy information such as poster, advertisements, navigation bars, etc., [12]. Although such information types are typically useful for user and as well for the website holders, they frequently hamper the automated generation of information retrieval and web mining, e.g., web page filtering, classification, information gathering and information extraction [4]. The noisy information in the web content can be classified into two categories according to their granularities. They are global noises and local noises. Global noises on the web are having large granularity and they are usually not smaller than individual pages. For example, Illegal Web pages, older version of website. Local noises involve in regions/items based within a web page. They are usually incoherent with the original contents of the web pages. Such a noisy web page contains poster advertisements, flush news, marquee pictures, etc., [18].

This paper implements a novel custom search engine to extract the noisy free information from the internet. The proposed URL pattern extractor algorithm will extract all relevant index pages from the web and thereby ranking the indexes based on the user query. Then, NDC algorithm is applied to remove the unwanted content from the retrieved web pages. The proposed custom search engine provides the following features:

1. Provides scalability for large volume of data as well small data.
2. Classifies the retrieved information into different categories; web, images and news .
3. Removes all noisy information from the retrieved web pages.

For example, when a user enters the desired query, the proposed system extracts all related indexes pages using URL algorithm and subsequently NDC algorithm is applied to filter the noisy information from the extracted index pages. Finally, it displays the retrieved information to the user.

The rest of this paper is organized as follows: in section 2, the related works which are very close to the proposed mechanism are reviewed. Section 3 introduces the proposed system architecture. The results obtained by the proposed system are discussed in section 4 briefly. Finally, the conclusion of the proposed system with future enhancement is given in chapter 5.

## 2. Related Works

A framework that employs an easily extensible set of techniques for web mining is developed in [8]. The key insight is to work with DOM trees, a W3C specified interface that allows programs to dynamically access document structure rather than with raw HTML markup. This proxy can be used both centrally administered groups of users, as well as by individuals for personal browsers. The use of latent semantic analysis to uncover semantic structures or concepts hidden in entity contexts towards improving named entity recognition on the web is described in [22]. The vast analysis of such relations helps to build implicit concepts around entity types, making the entity contexts more discriminative and avoiding data sparsity for a better classification. In order to rank a relevant index pages, three semantic similarity measures are used in [1]. The higher order association mining and clustering approaches are used to compute semantic similarity between words and sentences.

A public web intelligence tool called MySpiders, a threaded multi-agent system is designed [19] for information discovery. The performance of the system is evaluated by comparing its effectiveness in locating recent, relevant documents with that of existing search engines. Similarity based mechanisms [25]; Web Data Extraction using Similarity (WDES), to extract desired Search Engine Result Pages (SERPs) and store them in the local depository for offline browsing and Web Data Integration using Cosine Similarity (WDICS), to integrate the requested contents and enable the user to perform the intended analysis and extract the desired information. A method which utilizes click through data for training, namely the query-log of the search engine in connection with the log of links the users clicked on in the presented ranking [11]. Such click through data is available in abundance and can be recorded at very low cost. Support Vector Machine (SVM) approach is used for learning retrieval functions.

A Semantic Web Search Engine (SWSE) is developed in [9]. It consists of crawling, data enhancing, indexing and a user interface for search, browsing and retrieval of information; unlike traditional search engines, SWSE operates over Resource Description Framework (RDF) web data, also known as linked data which implies unique challenges for the system design, architecture, algorithms, implementation and user interface. Formal methods for filtration of noises from web pages are developed in [24]. It uses DOM structures of web documents to efficiently remove the irrelevant data. Document Entity And Resolution (DEAR) system [23] combines semantic similarity matching as provided by the open source Word Net database with the ability to recognize named entities through the Open Calais system. When used in concert, it provides a novel way

for users to quickly find the relevant content and detect and identify the uniquely named entities within that content.

A web extraction and storage tool [10] is implemented in Java that automates the downloading task from a given user query and also reduces the time complexity. A method to filter web pages and retrieve the actual content of a web page is implemented in [16]. It removes the noises from a given web page in order to improve the performance of web content mining.

Case-Based Reasoning (CBR) approach is used to identify the multiple noise models present in the web pages and remove the same from any web sites [14]. It uses a back transmission neural network algorithm to categorize the various noise models by corresponding noise data in current web page. In order to improve the performance of mining, the noises are initially identified and then removed [6]. The irrelevant information (i.e., web page noise or local noise) in web pages that can seriously harm web mining task such as clustering and classification etc are focused [3].

Web data analysis for sessionization is implemented in [7]. Sessionization is the determination of the number of visitors to a web site. The user session identification is very important for the traffic characterization purpose. The content mining [5] to extract specific information of interest to the user by removing unnecessary things like, navigation panels, advertisement, hyperlinks, date and time formatted data, noise and redundant data. The web pages resulting from search engines are divided into frames and each frame is then sub divided into blocks which are processed for removal of all unwanted information.

The main objective of the proposed system is to design custom search engine which provides noisy free information during query search. It integrates the search engine implementation along with noise removal. There is no need to work separately. It does not rank the web pages based on its popularity of web but its ranking is based on user query whose index have relevant information about user query.

## 3. Proposed System Model

The proposed system extracts the noisy free information and then classifies the information into different categories. The input to the system is only the user query. Hence, the proposed consists of three sequential stages; user interface, pre-processing and classification. The user can enter the query in the user interface. The proposed system does require the exact keyword and it supports multi-keyword search. In the preprocessing stage, the proposed URL pattern extraction and NDC algorithms are applied. Both algorithms are implemented in Java and utilizes Google API key as web environment which provides interface to user to search the query. In the

classification stage, the retrieved information's are classified into one of the predefined classes; web, news and images. Before pre-processing, the proposed system initially validates the user query from web database based on user behavior or interest. Figure 1 shows the flow diagram of the proposed web search engine for noisy free information retrieval.

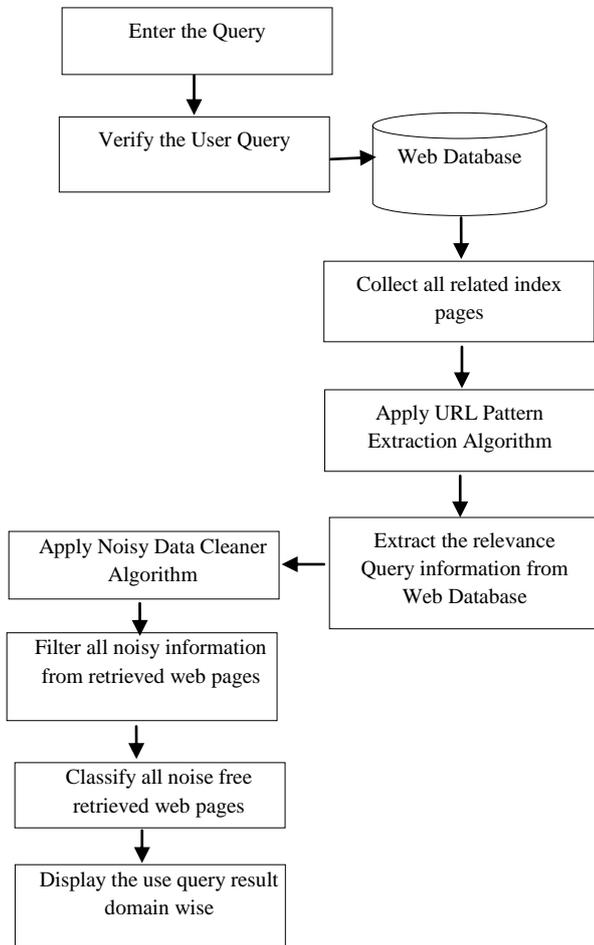


Figure 1. Flow diagram of the proposed customized search engine.

### 3.1. URL Pattern Extractor Algorithm

In this subsection, the proposed URL pattern extractor algorithm is discussed. The algorithm flow is as follows:

1. Verify the user query from web database.
2. Extract all pages which contain user’s query information, and,
3. Assist to build index ranking according to the user query.

The proposed mechanism does not provide unwanted large number of index pages which increases the retrieval time.

URL pattern extractor algorithm extracts all relevant information of the user query. The relevant information is the core information on the web page that the user wants to see. For example, the main content in the web page of a news article is the core information. These documents are arranged in a directory and

preprocessed for content extraction [2, 9, 13, 14, 21]. The proposed algorithm extracts the web pages based on the prefix and suffix content of the user query. It does not extract the famous or most visited pages only but also it considers all pages that have relevant information based on the user query. It also verifies the user query from web database such as spell-checking and as well for multi-keyword search supports. The pseudo code for the proposed URL pattern algorithm is given below:

Algorithm 1: Pseudo code of URL pattern extractor algorithm.

```

    Int InitLists (void);
    // initializes the data structure and resource
    Define max_url_length=60;
    // the max length of url path.
    Define interval_per_retrieval=1;
    //sleep interval of the crawler.
    Set url_prefix=http://localhost:8080/noisy/custom_search.jsp;
    // url we extract should be start with given prefix.
    Define content length =300;
    //make hash in large in comparison to number of url
    //extracted. For example depth of 3 on
    //http://localhost:8080/noisy/custom_search.jsp //approximate
    300.
    Define max_hash_function=300;
    //unlikely we have more than 1000 url for given query
    Define max url_per_page=300;
    //
    Char*getpage (char *URL, int depth, char *path);
    // get html into a string and return and return as page with
    //correct format
    Char **extracturl (char *html_buffer, cha *current);
    //extract url, given pointer to a string of html and return
    //extracted url
    Void updateListLink (char **url_list, int depth);
    //update link list visited for all url
    Void seturlvisited (char *url);
    // given pointer to a string of html and return current url
    Char *getAddressFromlinktobeVisited (int depth);
    //get the net url to visit from the extracted list
  
```

### 3.2. NDC Algorithm

The proposed NDC algorithm is discussed in this section. The basic idea of NDC algorithm is as follows:

1. Find the entire unstructured URL pattern with global as well local noisy content. A set of data cleaner patterns are used to detect the URL and index pages which are affected with global noise.
2. Remove all irrelevant data such as hyperlink, images and video [17].

The proposed NDC algorithm uses derive in size-3 noisy data cleaner patterns at a given *ndc*-cleaner threshold *ndc* from the transaction set *T'*, where *T'* is the transpose of the original transaction data *T*, since we are interested in clustering objects. The noisy data objects are simply those which are not relevant to the search query patterns [15]. In other words, for identification of noisy data object, it cannot find other objects which have pair-wise similar content with this

object above the noisy data-cleaner threshold *ndc*. Indeed, the noisy data cleaner threshold specifies the fraction of noise data objects. NDC is very efficient in removing irrelevant information from the extracted pages. The main advantage of the proposed NDC algorithm is that it is scalable for large volume of data set as well small dataset. The proposed NDC algorithm proficiently removes all kinds of noise. The pseudo code for the proposed NDC algorithm is given below:

Algorithm 2: Pseudo code of NDC algorithm.

```

Data: Transaction set T
Result: Set of noisy data points N, Set of non-noisy data points P,
Noisy data fraction £
NDC ← Noisy Data Cleaner (T) //NDC: the Noisy Data Cleaner set;
T[1...ntrans].covered ← false;
num ndc ← size(NDC);
for i = 1 to num ndc do
    for j = 1 to ntrans do
        if ((!T[j].covered)&&contains(T[j],NDC[i])) then
            T[j].covered ← true;
        end
    end
end
N ← {};
P ← {};
for i = 1 to ntrans do
    if T[i].covered then
        P ← P ∪ T[i];
    end
    else
        N ← N ∪ T[i];
    end
end
£ ←  $\frac{|N|}{ntrans}$ ;
Return N, P, £;
    
```

## 4. Results and Discussions

### 4.1. Experimental Setup

The proposed method is implemented in JAVA using Net Beans 8.0 with Google API key, HTML parser and W3c java library with real time Web database. Also it utilizes the open source Weka tool 3.7.2 version for performance evaluation. The experimental setup uses a system with Intel Dual Core processor (1.836 Hz), 2GB memory under Window 7 Ultimate system. The proposed method searches various types of query given by the user and collects the information based on the respective dataset types. Three different types of data sets; web, news and images are used in this study.

As the proposed system requires labelled training data, 10% of the pages from each dataset are used as a training data set and the remaining pages as an evaluation data set. The pages for training data set are chosen from various website.

### 4.2. Simulation Results

In order to select the best learning algorithm for the classification process, 10-fold cross validation is performed. The classifiers used in the proposed system are Decision Tree (DT), Radom Forest (RF), Sequential Minimal Optimization (SMO), and Multi-Layer Perceptron (MLP). WEKA tool is used for the classification with default parameter settings. Table 1 shows the performance of the proposed search engine.

Table 1. Performance of the proposed customized search engine.

Dataset Type	No. of Query	Relevance Pages	Total Pages
Web	25	122	125
Image	30	145	150
News	80	397	400

The performance measures used to analyze the proposed system for both URL Pattern Extractor and NDC are precision, recall, and F1 measure. For every web page in the evaluation data set, the extracted content is compared with the gold standard [20]. In order to compute these measures, the proposed system finds the overlap between the gold standard and the extracted content. The computation of recall (*r*), precision (*p*), and F1 measures are defined in Equation (1).

$$p = \frac{|c|}{|g|}, r = \frac{|c|}{|e|}, f = \frac{2 \times p \times r}{p + r} \tag{1}$$

Where |c| is the string length of the URL pattern extractor of the closest index pages. |g| is the length of the gold standard string. |e| is the length of the extracted content. Tables 2 and 3 show the performance of the proposed URL Pattern Extractor and NDC algorithm respectively. Figures 2, 3, and 4 shows the comparative analysis of the proposed system with existing algorithms in the literature.

Table 2. Performance of the proposed URL pattern extractor algorithm.

Learning Algorithms	Web			Images			News		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
DT	0.964	0.962	0.963	0.952	0.975	0.963	0.988	0.988	0.988
RF	0.966	0.976	0.971	0.963	0.988	0.975	0.992	0.993	0.992
SMO	0.906	0.902	0.904	0.907	0.99	0.947	0.926	0.99	0.957
MLP	0.942	0.969	0.955	0.948	0.956	0.957	0.988	0.983	0.986

Table 3. Performance of the proposed NDC algorithm.

Learning Algorithms	Web			Images			News		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
DT	0.984	0.961	0.972	0.922	0.94	0.931	0.958	0.98	0.969
RF	1	0.969	0.984	0.939	0.97	0.954	0.973	0.98	0.993
SMO	0.939	0.969	0.953	0.828	0.95	0.885	0.749	0.921	0.826
MLP	0.992	0.984	0.988	0.929	0.938	0.934	0.95	0.98	0.965

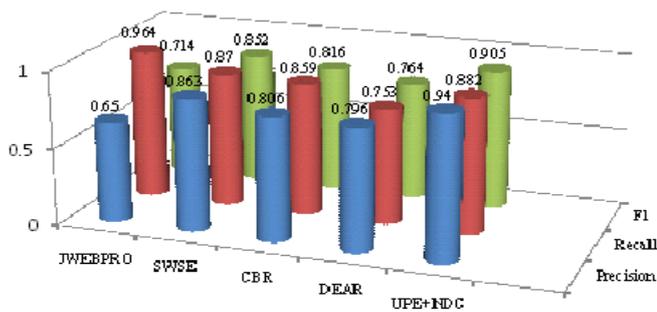


Figure 2. Performance of the proposed customized search engine for Web dataset.

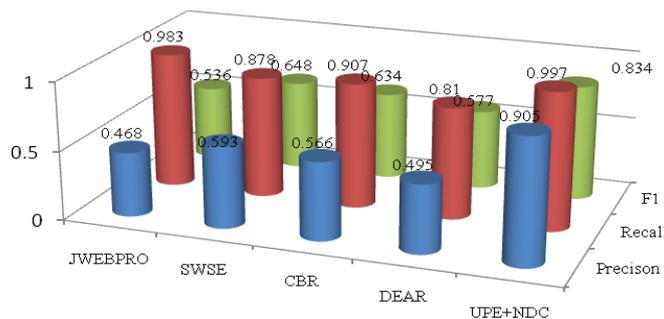


Figure 3. Performance of the proposed customized search engine for news dataset.

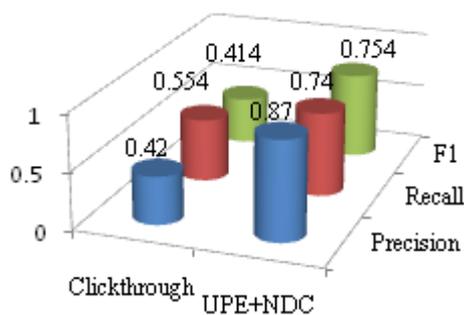


Figure 4. Performance of the proposed customized search engine for Image dataset.

It is observed from the Tables 2 and 3 that all the classifiers used in this study achieve promising results with more than 90% accuracy. Among the different classifiers, RF and DT produce better results for all dataset types.

The computed performance measures such as precision, recall and F1 score of the proposed system for Web, News and Image datasets are shown in Figures 2, 3, and 4 respectively. The performance of the proposed system is compared with Java Web Processing (JWEBPRO) [22], SWSE [9], CBR [14], DEAR [23] and Clickthrough [11]. It is observed from Figure 2 that the proposed system has a precision 94% for web dataset. In the literature, only a maximum of 86.3% precision rate is achieved by SWSE which means that the proposed system increases the precision rate by a minimum of 7.7%. It is noted that the recall rate of the proposed system is only 88.2% where JWEBPRO has a high recall rate. However it has a very low precision rate of 65% means that it returned more irrelevant web pages. The proposed system has a

high F1 score of 0.905 in comparison with existing systems.

The performance on News dataset is given in Figure 3. The proposed system outperforms all existing systems in terms of all measures such as precision (90.5%), recall (99.7%) and F1 score (0.834) used in this study. Similar to Web dataset, JWEBPRO has a good recall rate with a very low precision comparison with all other techniques. In order to analyze the performance of the proposed system for Image dataset, a comparison is made with Clickthrough. It is noted from the Figure 4 that the proposed system outperforms Clickthrough with an increase in precision of about 45% with recall of 31% and F1 score of 19%.

It is concluded that the proposed URL Pattern Extractor (UPE) with NDC is preferable based on the performance measures in the above Tables and Figures. The proposed method is very efficient to produce URL content extraction and noisy content removal. All existing methods work separately; some of them are efficient either for content extraction or to remove irrelevant content. However, the proposed system achieves better result for both content extractions without noisy content.

## 5. Conclusions

A novel custom search engine is implemented to search the user query efficiently in this paper. The proposed URL pattern extractor algorithm will extract all relevance index pages from the web database and ranking the indexes based on the user query. Then, noisy data cleaner algorithm is applied to remove the unwanted content from the retrieved web pages. Experimental results show that the proposed custom search engine outperforms other existing technique in the literature. The proposed UPE+NDC method have the highest precision of 0.940 for web and 0.905 for news dataset and 0.870 for image dataset. Hence, the proposed approach is preferable since it has better precision and recall rate. Also it gives reasonably high F1 score. In near future, the proposed work can be extended to eliminate noisy information from the audio and video data.

## References

- [1] Adhikesavan K., "An Integrated Approach for Measuring Semantic Similarity between Words and Sentences Using Web Search Engine," *The International Arab Journal of Information Technology*, vol. 12, no. 6, pp. 589-596, 2015.
- [2] Anita R., Bharani V., Nityanandam N., and Sahoo P., "Deep iCrawl: An Intelligent Vision Based Deep Web Crawler," *International Journal of Computer and Information Engineering*, vol. 5, no. 2, pp. 128-133, 2011.
- [3] Bhamare S. and Pawar B., "Survey on Web Page

- Noise Cleaning for Web Mining,” *International Journal of Computer Science and Information Technologies*, vol. 4, no. 6, pp. 766-770, 2013.
- [4] Bhawsar S., Pathak K., Mariya S., and Parihar S., “Extraction of Business Rules from Web Logs to Improve Web Usage Mining,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 8, pp. 333-340, 2012.
- [5] Boddu S., “Eliminate the Noisy Data from Web Pages Using Data Mining Techniques,” *Computer Science and Telecommunications*, vol. 2, no. 38, pp. 39-46, 2013.
- [6] Das S., Vijayaraghavan P., and Mathew M., “Eliminating Noisy Information in Web Pages using Featured DOM Tree,” *International Journal of Applied Information Systems*, vol. 2, no. 2, pp. 27-34, 2012.
- [7] Garhwal R., “Improving Privacy in Web Mining by Eliminating Noisy Data and Sessionization,” *International Journal of Latest Trends in Engineering and Technology*, vol. 3, no. 3, pp. 373-378, 2014.
- [8] Gupta S., Kaiser G., Grimm P., Chiang M., and Starren J., “Automating Content Extraction of Html Documents,” *Journal of World Wide Web*, vol. 8, no. 2, pp. 179-224, 2005.
- [9] Hogan A., Harth A., Umbrich J., Kinsella S., Polleres A., and Decker S., “Searching and Browsing Linked Data With SWSE: The Semantic Web Search Engine,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, no. 4, pp. 365-401, 2011.
- [10] Jeyalatha S., Vijayakumar B., and Firoz M., “Design and Implementation of a Tool for Web Data Extraction and Storage Using Java and Uniform Interface,” *International Journal of Computer Applications*, vol. 22, no. 4, pp. 1-6, 2011.
- [11] Joachims T., “Optimizing Search Engines using Clickthrough Data,” in *Proceedings of the 8<sup>th</sup> ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, pp. 133-142, 2002.
- [12] Jusoh S. and Alfawareh H., “Techniques, Applications and Challenging Issue in Text Mining,” *International Journal of Computer Science Issues*, vol. 9, no. 2, pp. 431-436, 2012.
- [13] Keole R., Karde P., and Thakare V., “Clustering Web Search Engine Results for Improving Information Retrieval: A Survey,” *International Journal of Computer Science*, vol. 2, no. 3, pp. 24-33, 2014.
- [14] Kumar P. and Parvathi R., “Neural Networking Using Multiple Web Page Noise Removing Method,” *International Journal of Computer Science and Technology*, vol. 3, no. 1, pp. 336-339, 2012.
- [15] Lingwal S., “Noise Reduction and Content Retrieval from Web Pages,” *International Journal of Computer Applications*, vol. 73, no. 4, pp. 24-30, 2013.
- [16] Lingwal S. and Gupta B., “A Comparative Study of Different Approaches for Improving Search Engine Performance,” *International Journal of Emerging Trends and Technology in Computer Science*, vol. 1, no. 3, pp. 123-132, 2012.
- [17] Market Brew White paper, The Noisy Query Layer: How Brands Can Avoid Chasing Their Tails, available at <http://cdn.marketbrew.com/wpcontent/uploads/2014/09/The-Noisy-Query-Layer.pdf>, Last Visited, 2015.
- [18] Mary S. and Baburaj E., “An Efficient Approach to Perform Pre-Processing,” *Indian Journal of Computer Science and Engineering*, vol. 4, no. 5, pp. 404-410, 2013.
- [19] Menczer F., “Complementing Search Engines with Online Web Mining Agents,” *Decision Support System*, vol. 35, no. 2, pp. 195-212, 2003.
- [20] Nie Z., Ma Y., Wen J., and Ma W., “Object-Level Web Information Retrieval,” Technical Report of Microsoft Research, 2005.
- [21] Oza A. and Mishra S., “Elimination of Noisy Information from Web Pages,” *International Journal of Recent Technology and Engineering*, vol. 2, no. 1, pp. 115-117, 2013.
- [22] Phan X., Horiguchi S., Nguyen L., and Nguyen C., “Semantic Analysis of Entity Contexts towards Open Named Entity Classification on the Web,” in *Proceedings of the 10<sup>th</sup> Conference of the Pacific Association for Computational Linguistics*, China, pp. 137-144, 2007.
- [23] Sedbrook T. and Lightfoot J., “DEAR: A New Technique for Information Extraction and Context Dependent Text Mining,” *Communications of the IIMA*, vol. 10, no. 3, pp. 33-47, 2010.
- [24] Singh A., “Web Content Extraction to Facilitate Web Mining,” *International Journal of Electronics and Computer Science Engineering*, vol. 1, no. 3, pp. 1292-1299, 2010.
- [25] Srikantaiah K., Suraj M., Venugopal K., and Patnaik L., “Similarity Based Dynamic Web Data Extraction and Integration System from Search Engine Result Pages for Web Content Mining,” *ACEEE International Journal on Information Technology*, vol. 3, no. 1, pp. 42-49, 2013.



**Pradeep Sahoo** received his B.Tech from The Institution of Engineer India, Calcutta, India in 2000. He completed his M.Tech in Computer Science and Engineering from Anna University Chennai, Tamilnadu (India). He is pursuing his Doctorate

degree from Anna University Chennai, Tamilnadu (India). Currently, he is serving as Associate Professor at Computer Science & Engineering Department in Sai Ram Engineering College, Chennai, Tamilnadu India. He participated in total 8 National Conferences, Workshop & Seminar in various institutions in India. He published 4 papers in International Journal and 3 in National Journal. His research area is Data Mining in Pattern Recognition and Content Extraction and Software Engineering. He is holding the following membership: ISTE, CSI, and IAENG & IACSIT.



**Rajagopalagn Parthasarthy** received his Master degree in Applied Mathematics from IIT Madras and has obtained PhD in Computer Science from the University of Madras. He has 40 years of teaching experience in

various institutions in India. He is recognized as research supervisor for Anna University, Dr. MGR University, Vels University, Mother Terasa University and University of Madras Tamilnadu India. He has successfully led around 25 scholars for obtaining their PhD and more than 169 scholars to obtain their M.Phil degree. He served as Faculty, Visiting Professor, Project Co-ordinator, Resource Personal and Panel Member for various academic institution and government organization in India. Currently, he is serving as a veteran at Research and Development Cell in Department of CSE at GKM College Engineering of Technology Chennai, India. He served as many administrative positions like Chairman, Principal, President, Director, Chief Guest, Dean, Head of Department, Advisor, Member, Convener and Subject Expert in various organizations in India. He wrote 13 books and published 84 International journal and national journal in his related area. He awarded as Life Time Achievement Award, Seva Ratna Award, Distinguished Educationist and A Person of Eminence, Seer Seyai Maamani Award, A Living Legend, Educationalist Born-Noble and Best Teacher Award from various government & private organization and institutions. His field of interests and specialization are Quantitative Techniques, Data Processing and Project Management, Management Information System, Programming Languages, Simulation, Text Generation, Cryptography and Data Mining.