Change detection in Migrating Parallel Web Crawler: A Neural Network Based Approach

Md. Faizan Farooqui[1], Dr. Md. Rizwan Beg[2] and Dr. Md. Qasim Rafiq[3]

[1]Research Scholar, Department of Computer Application, Integral University, Lucknow, Uttar Pradesh 226026, India

[2]Professor, Department of Computer Science and Engineering, Integral University, Lucknow, Uttar Pradesh 226026, India

[3]Professor, Department of Computer Engineering, Aligarh Muslim University, Aligarh, Uttar Pradesh, India

ABSTRACT

Search engines are the tools for Web site navigation and search. Search engines maintain indices for web documents and provide search facilities by continuously downloading Web pages for processing. This process of downloading web pages is known as web crawling. In this paper we propose A neural network based change detection method in migrating parallel web crawler. This method for Effective Migrating Parallel Web Crawling approach will detect changes in the content and structure using neural network. This crawling strategy makes web crawling system more effective and efficient. The major advantages of migrating parallel web crawler are that the analysis portion of the crawling process is done locally at the residence of data rather than inside the Web search engine repository. This significantly reduces network load and traffic which in turn improves the performance, effectiveness and efficiency of the crawling process. The another advantage of migrating parallel crawler is that as the size of the Web grows, it becomes necessary to parallelize a crawling process, in order to finish downloading web pages in a comparatively shorter time. Neural network based change detection method in migrating parallel web crawler will yield high quality pages and detect for changes will always download fresh pages.

Keywords: *Web crawling, parallel migrating web crawler, search engine, neural network*

1. Introduction

The Internet is a system of interconnected computer networks. The searching and indexing tasks for the web are handled from applications called search engines. The search engines are divided into three parts they are search engine,

the database and the web crawling system. A web crawler is a program that browses the Web in a systematic manner. The process of traversing the web is called Web crawling. Web crawler starts with a queue of known URLs to visit. Migrating parallel web crawlers are special kind of web crawlers. The Migrating Parallel Crawler system consists of Central Crawler, Crawl Frontiers, and Local Database of each Crawl Frontier and Centralized Database. It is responsibility of central crawler to receiving the URL input from the applications and forwards the URLs to the available migrating crawling process. Crawling process migrated to different machines to increase the system performance. Local database of each crawl frontier are buffers that locally collect the data. This data is transferred to the central crawler after compression and filtering which reduces the network bandwidth overhead. The central crawler has a centralized database which will contain the documents collected by the crawl frontiers independently. The main advantages of the migrating parallel crawler approach are Localized Data Access, Remote Page Selection, Remote Page Filtering, Remote Page Compression, Scalability, Network-load dispersion, Network-load reduction.

2. Literature survey

In [13], the author demonstrated an efficient approach to the "download-first process-later" strategy of existing search engines by using mobile crawlers. In [14] author has implemented UbiCrawler, a scalable distributed and fault-tolerant web crawler. In [15] author presented the architecture of PARCAHYD which is an ongoing project aimed at designing of a Parallel

Crawler based on Augmented Hypertext Documents. In [16] the author studied how an effective parallel crawler is designed. As the size of the Web grows, it becomes imperative to parallelize a crawling process. In [17] the author proposes Mercator, which is a scalable, extensible crawler. [18] Presented Google, a prototype of a large scale search engine which makes heavy use of the structure present in hypertext. [19] Aims at designing and implementing a parallel migrating crawler in which the work of a crawler is divided amongst a number of independent. In [20] the author has reviewed the various crawling techniques. [21] is an extended model for effective migrating parallel web crawlers with domain specific and incremental crawling.

3. Issues in Migrating parallel Web Crawlers

According to [1], Web crawlers of big commercial search engines crawl up to 10 million pages per day. Assuming an average page size of 6K [2], the crawling activities of a single commercial search engine adds a daily load of 60GB to the Web. One of the first Web search engines, the World Wide Web Worm [3], was introduced in 1994 and used an index of 110,000 Web pages. The Web is expected to grow further at an exponential speed, doubling its size (in terms of number of pages) in less than a year [4]. Current Web crawlers download all these irrelevant pages because traditional crawling techniques cannot analyze the page content prior to page download [13].

The following issues are important in the study of a migrating parallel crawler interesting:

Quality: Prime objective of migrating parallel crawler is that to download the "important" pages first to improve the "quality" of the downloaded pages.

Communication bandwidth: Communication is really important so as to prevent overlap, or to improve the quality of the downloaded content, crawling processes need to communicate and coordinate with each other to improve quality thus consuming valuable communication bandwidth.

Overlap: When multiple processes run in parallel to download pages, it is possible that different processes download the same page multiple times. One process may download the same page that other process may have already downloaded, one process may not be aware that another process has downloaded the same page.

4. Background on Neural Networks

Artificial neural networks (ANNs) resemble the processing capabilities of the human brain. The components of neural network are computational units similar to the neurons of the brain. A neural network is formed from one or more layers of neurons. Each neuron in the network performs calculations that contribute to the overall learning process of the network. The neural network is parallel information processing system that learn and store knowledge about its environment. The two factors that influence the performance of the neural network are its parallel distributed design and its capability to extrapolate the learned information.

Data mining, pattern recognition, and function approximation are tasks that can be performed by neural networks. In our technique, the neural network is used to produce a output when an input values are given. The neurons of the network are composed of: a set of n synaptic weights, an adder and an activation function.

If the input signal to neuron i is x_j , the synaptic weight associated with the interconnection between the input signal and the neuron is denoted w_{ij} . A multi-layer feed forward neural network consists of an input layer of non computational units (one for each input), one or more hidden layers of computational units, and an output layer of computational units. Backpropagation is the standard training method that is applied to multi-layer feedforward networks.

The algorithm consists of passing the input signal forward and the error signal backward through the network. Multi-layer feed forward neural networks are capable of solving complex problems, and the backpropagation technique is efficient as a training method.

5. Proposed Work: Neural network based model for Change detection of web pages

Migrating parallel crawlers move to resource and take the advantages of localized access of data. Migrating parallel crawler after accessing a resource migrates to the next host or server or to their home system. Each migrating parallel crawling process performs the tasks of single crawler that it downloads pages from the Web server or host, stores the pages in the local database, extracts URLs from the content of downloaded pages and follows the extracted URLs or links. When Crawling process run on the same local area network and communicate through a high speed interconnection network then it is known as intra-site migrating parallel web crawler. When Crawling process run at geographically distant locations connected by the Internet or a wide area network then it is known as distributed migrating parallel web crawler. The architecture consists of central coordinator system and crawling process.

Web content is dynamic in nature [22] and 14% of the links in search engines are broken [25]. At regular interval of time it is very necessary to refresh the data base of web pages. The primary objective is to update those documents of the database which has actually changed. Web pages are updated, rearranged or modified every other second on the World Wide Web. Modifications take place very frequently.

This changing nature of the World Wide Web is one of the major concerns in designing the Migrating Parallel web crawler. The rate of change varies from site to site. Therefore, managing the local collection afresh becomes a challenging task. The changes that can occur in web pages can be classified into following 4 major categories [23] are Structural Changes, Content level changes, Cosmetic changes and Behavioral changes. Some mechanism [24,26], used for change detection, does use the policy of retaining cached copies of web pages which are later compared with downloaded web document to determine if there has been any change. Here neural network is used to detect change in web pages. Proposed Algorithm is implemented in MATLAB. The nntool of MATLAB has been used to implement the proposed algorithm.

Algorithm: Neural network based model for Change detection of web pages

Change in structure of web pages

a) Initially the graph is derived from web pages.

b) Adjacency matrix A is obtained at any time t and Adjacency matrix B is obtained at any time t' form graph.

c) Using MATLAB for detecting change in the structure.

d) Using nntool for creating neural network

e) Assign input vector (v) values

f) Assign target values (t)
g) Create neural network
h) Training the neural network
i) The result of training
j) Simulating the neural network
k) Analysis of result obtained.

Change in contents of web pages

a) Change in Content vector V is obtained .

b) Using MATLAB for detecting change in the content.

c) Using nntool for creating neural network

d) Assign input vector (v) values
e) Assign target values (t)
f) Create neural network
g) Training the neural network
h) The result of training
i) Simulating the neural network
j) Analysis of result obtained.

Description of Algorithm:

Change in structure of web pages

Neural network based model for Change detection of web pages The following method is followed for detecting change in the structure.

1. Initially the graph G1 and G2 are derived from web pages.

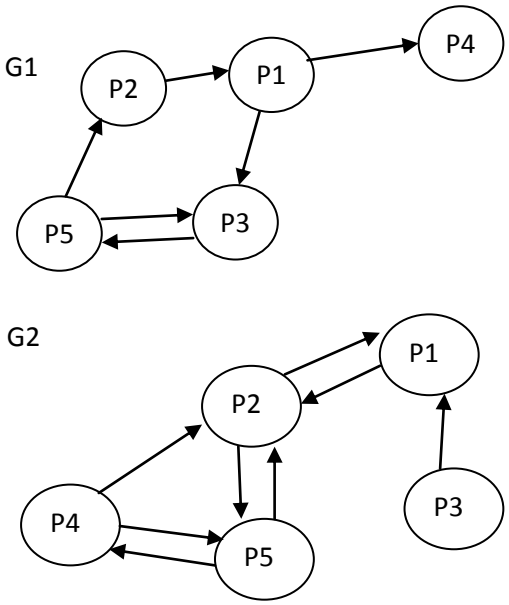

Fig. 1 Graph G1 and Graph G2derived from web pages.

2. Adjacency matrix A is obtained from graph G1 at any time t and Adjacency matrix B is obtained from graph G2. at any time t'

$$A=\begin{bmatrix}0&0&1&1&0\\1&0&0&0&0\\0&0&0&0&1\\0&0&0&0&0\\0&1&1&0&0\end{bmatrix}$$

$$B=\begin{bmatrix}0&1&0&0&0\\1&0&0&0&1\\1&0&0&0&0\\0&1&0&0&1\\0&1&0&1&0\end{bmatrix}$$

MATLAB is used for detecting change in the structure. The algorithm is implemented in nntool box. The nntool box is used for creating neural network. Assign input vector (v) values V={0 0 1 1 0;0 1 0 0 0} and assign target values (t) t={0 1 1 10}.
The neural network is created with name Changedetection along with the following Network properties. The Network Type is Feed forward Back propagation with Input Range [0,1;01] and TRAINLM as the Training Function. The
Adaptation Learning Function is LEARNGDM and Performance function as MSE with No of Layers is 2.

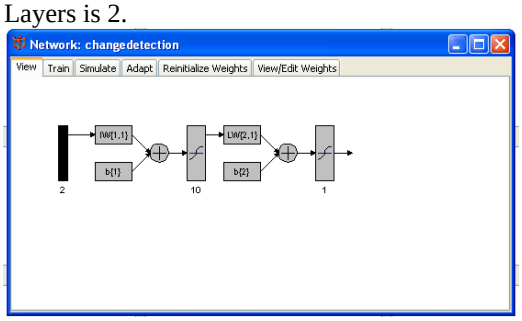

Fig. 2 Neural Network for Detecting Change in Structure

Training, Simulation and Analysis of Result:

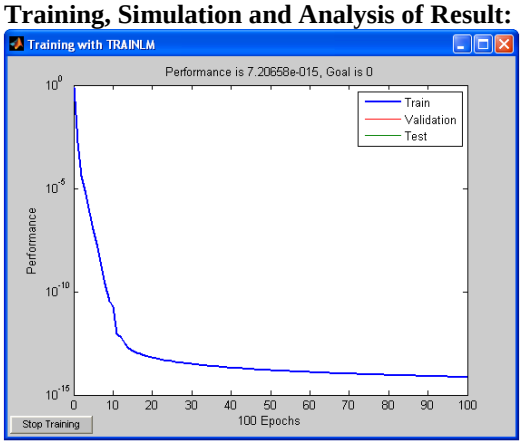

Fig. 3 Training of Neural Network.

The neural network is trained and the result of training is obtained. Then the neural network is simulated. Then the result is analyzed as '1' in the output vector will indicate the change in content and '0' in the output vector will corresponds to no change in content.

$$R=\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Change in contents of web pages

Method followed for detecting change in the content: Initially the web graph is derived from web pages.

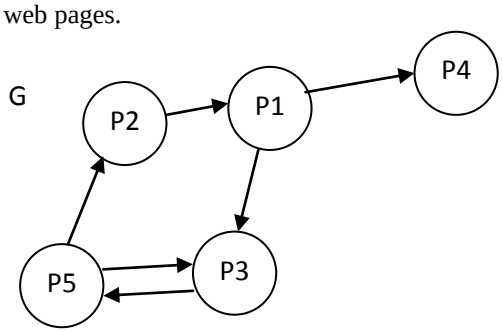

Fig. 4 Web Graph derived from web page.

Obtaining content vector V1 for graph G1 at any time t and content vector V2 for the same graph G1 at any time t + t1:
V1= (0, 0, 1, 1, 0)
V2= (0, 1, 0, 0, 0)

MATLAB is used for detecting change in the Content. The nntool box is used for creating neural network. Assign input vector (v) values v={0,0,1,1,0;0,1,0,0,0} and Assign target values (t) t= {0,1,1,1,0}
The neural network is created with the name Changedetection and with the following Network properties they are the Network Type is Feed forward Back propagation with Input Range [0,1;01] and TRAINLM as the training Function.
The Adaptation Learning Function is LEARNGDM and MSE as Performance function with number of layer as 2.Neural Network is shown in figure given below.

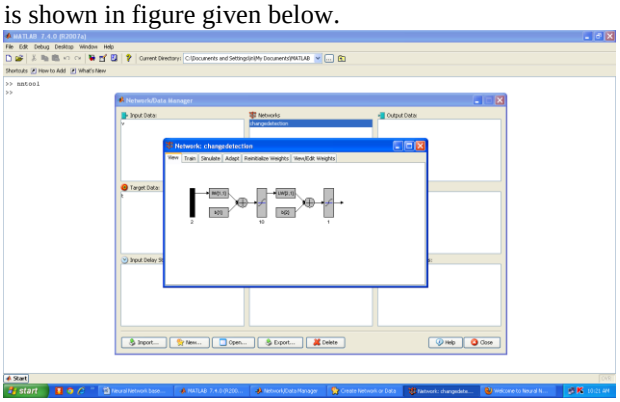

Fig. 5 Neural Network For Detecting Change in Content.

Training, Simulation and Analysis of Result:

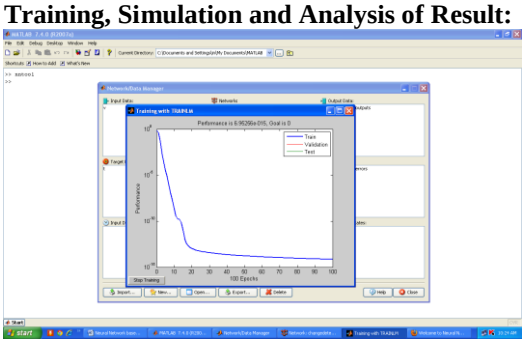

Fig. 6 Training the Neural Network.

The neural network is trained and the result of training is obtained. Then the neural network is simulated. Then the result is analyzed as '1' in the output vector will indicate the change in structure and '0' in the output vector will corresponds to no change in structure.

Fig. 7 Result Obtained.

Neural network based change detection method in migrating parallel web crawler will yield high quality pages and detect for changes will always download fresh pages.

6. Conclusions

In this paper we proposed a Neural network based change detection method in migrating parallel web crawler will yield high quality pages and detect for changes will always download fresh pages.
The research directions in migrating parallel crawler include:
• Security could be introduced in migrating parallel crawlers
• Migrating parallel crawler could be made polite
• Location awareness could be introduced in migrating parallel crawlers
This future work will deal with the problem of quick searching and downloading the data. The data will be collected and analyzed with the help of tables and graphs.

Acknowledgments

First and foremost, our sincere thanks goes to Prof. Syed Wasim Akhtar, Honorable Vice Chancellor, Integral University, Lucknow. Prof Akhtar has given us unconditional support in many aspects, which enabled us to work on the exciting and challenging field of Software Engineering. We would also like to give our special thanks to Prof. T. Usmani, Pro Vice Chancellor, Integral University. His encouragement, help, and care were remarkable during the past few years. We are also grateful to Prof S. M. Iqbal Chief Academic Consultant, Integral University. Prof Iqbal provided us with valuable thoughts for our research work. My gratitude also goes to Dr. Irfan Ali Khan, Registrar, Integral University for his constructive comments and shared experience.

References

[1] D. Sullivan, "Search Engine Watch," Mecklermedia, 1998.
[2] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," Stanford University, Stanford, CA, Technical Report, 1997.
[3] O. A. McBryan, "GENVL and WWW: Tools for Taming the Web," in Proceedings of the First International Conference on the World Wide Web, Geneva, Switzerland, 1994.
[4] B. Kahle, "Archiving the Internet," Scientific American, 1996.
[5] J. Gosling and H. McGilton, "The Java Language Environment," Sun Microsystems, Mountain View, CA, White Paper, April 1996.
[6] J. E. White, Mobile Agents, MIT Press, Cambridge, MA, 1996.
[7] C. G. Harrison, D. M. Chess, and A. Kershenbaum, "Mobile Agents: Are they a good idea?," IBM Research Division, T.J. Watson Research Center, White Plains, NY, Research Report, September 1996.
[8] H. S. Nwana, "Software Agents: An Overview," Knowledge Engineering Review, Cambridge University Press, 11:3, pp. , 1996.
[9] M. Wooldridge, "Intelligent Agents: Theory and Practice," Knowledge Engineering Review, Cambridge University Press, 10:2, pp. , 1995.
[10] P. Maes, "Modeling Adaptive Autonomous Agents," MIT Media Laboratory, Cambridge, MA, Research Report, May 1994.
[11] P. Maes, "Intelligent Software," Scientific American, 273:3, pp. , 1995.
[12] T. Finin, Y. Labrou, and J. Mayfield, "KQML as an agent communication language," University of Maryland Baltimore County, Baltimore, MD, September 1994.
[13] Joachim Hammer , Jan Fiedler "Using Mobile Crawlers to Search the Web Efficiently" in 2000
[14] Paolo Boldi, Bruno Codenotti, Massimo Santini, Sebastiano Vigna, "UbiCrawler: A Scalable Fully Distributed Web Crawler" in 2002
[15] A.K. Sharma, J.P. Gupta, D. P. Aggarwal, "PARCAHYDE: An Architecture of a Parallel Crawler based on Augmented HypertextDocuments" in 2010
[16] J. Cho and H.Garcia-Molina, "Parallel crawlers". In Proceedings of the Eleventh International World Wide Web Conference, 2002, pp. 124 – 135
[17] A. Heydon and M. Najork, "Mercator: A scalable, extensible web crawler". World Wide Web, vol. 2, no. 4, pp. 219 -229, 1999.
[18] Akansha Singh , Krishna Kant Singh , "Faster and Efficient Web Crawling with Parallel Migrating Web Crawler" in 2010
[19] Min Wu, Junliang Lai, "The Research and Implementation of parallel web crawler in cluster" in International Conference on Computational and Information Sciences 2010
[20] Md. Faizan Farooqui, Md. Rizwan Beg, Md. Qasim Rafiq, "A Critical Review of Migrating Parallel Web Crawler", Advances in Computing and Information Technology,Advances in Intelligent Systems and Computing Volume 177, 2013, pp 631-637
[21] Md. Faizan Farooqui, Dr. Md. Rizwan Beg and Dr. Md. Qasim Rafiq, "An Extended Model For Effective Migrating Parallel Web Crawling With Domain Specific And Incremental Crawling", International Journal on Web Service Computing (IJWSC), Vol.3, No.3, September 2012, DOI : 10.5121/ijwsc.2012.3308 85
[22] Cho, Junghoo, Angeles, Los, and Garcia-Molina, Hector, "Effective Page Refresh Policies for Web Crawlers", ACM Transactions on Database Systems, Volume 28, Issue 4, pp. 390 – 426, December 2003.
[23] Francisco-Revilla, L., Shipman, F., Furuta, R., Karadkar, U. and Arora, A., "Managing

Change on the Web", In Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, pp. 67 – 76, 2001.

[24] Manku, Gurmeet Singh, Jain, Arvind, and Sarma, Anish Das, "Detecting near-duplicates for web crawling", In Proceedings of the 16th international conference on World Wide Web, pp. 141 - 150 , May 2007.

[25] Lawrence, S., and Giles, C. L., "Accessibility of information on the web", Nature, 400:107-109, 1999.

[26] Schleimer, S., Wilkerson D. S., and Aiken, A., "Winnowing: Local algorithms for document fingerprinting", Proceedings of the ACM SIGMOD international conference on Management of data, pp. 76-85, June 2003.

First Author Mohammed Faizan Farooqui obtained his Bachelor's degree in Computer Application from the University of Lucknow in 2000 and his M.C.A. degree from Uttar Pradesh Technical University, Lucknow in 2003. Md Faizan Farooqui works since 2005 at Integral University, Lucknow as a full time Associate Professor (Jr Scale). Currently is giving courses on programming Computer Graphics and animation in MCA program at IU, Lucknow

Second Author Dr. Md. Rizwan Beg is Professor in the Department of Computer Science & Engineering in Integral University, Lucknow. He has published over 70 research paper in the International Journal of Repute. He has published many papers in International Conferences. He is the editor and Chief Editor of many reputed International Journals. He is the member of program committee of International conferences.

Third Author Dr. Md. Qasim Rafeeq was the chairman of Computer Engineering Department in Aligarh Muslim University.