Automation of Manual Seed URLs Cull Approach for Web Crawlers

Suvarna Sharma, Amit Bhagat

Abstract*: Web mining has become a more emerging topic these days and is speedily increasing with the growth of data on web. It is playing an essential role in our life as it helps us providing quicker information by using new trends and technologies to improve. Hyperlink structure analysis and web crawling provide scope for more advanced research topics. If a system coverers various most relevant web pages in search engine environment, then it can improve the result of search engine. This URL's set may be useful for extracting more relevant information or improving on existing and may also be useful to manage crawling infrastructure to offer quicker responses. Today, web crawling is an emerging issue in search engine which considers search quality, accessing pages at various servers to extract features. In the current scenario, the user may only be interested in the best result with some specific constraints. The constraint may define to the domain of search or importance of relevant pages. Here, we consider important or useful pages for particular user in searching environment. We proposed a framework, namely BUDG (Base URL's Set for Directed Graph) which deals with URL's hyperlink structure and generates a min set of 'K' URLs and then discover the covered graph for directed graph. Experimental results show that the proposed framework is working properly for different domain.

Index Terms*: Information Retrieval, Seed URLs, Web crawler, Web graph analysis, Web Mining.

I. INTRODUCTION

Today web is playing vital role in our life. Web services are useful and easily accessible by many types of gadgets i.e. smart phones, desktop, laptops etc. Some useful web services are searching, Social Networking, News etc. These web services are useful for different perspectives. By these services, web became very useful, easier, and faster. These services are frequently accessed by users for different perspectives. In web environment, search engine has arisen as an emerging issue in web mining. Search engine performance based on the crawling technique. Earlier crawlers were used to collecting statics about the web and indexing for search engine [1][2]. In these times crawlers are more efficient and can also used to perform accessibility and vulnerability check on the web.

Nowadays because of technology expansion web crawling is very big challenge. Over the past years, many approaches were introduced to reduce the time and cost for crawling. From early days up to the recent days many crawlers were introduced, at broad level these divided into three categories such as Traditional Crawlers, Deep Web Crawlers and RIA (Rich Internet Application) [3]. Some recent studies are going to mine Domain specific web crawling. As per literature web crawlers performance depends on the seed URLs. Researchers have also studied the different issues and challenges that web crawlers face [3][4] and seed selections [5][6][7][8].

Figure 1. Simple Directed Graph

Fig.1 shows the simple directed graph of pages which are randomly connected by hyperlinks. Here P_1, P_2, P_3….P_n shows different pages and e_1, e_2, e_3. . .e_n represents directed links.In this paper, we proposed a novel approach for selecting seed URLs of the web focused crawler based on the hyperlink structure and DFS. The main ideal of our proposed approach is described in Fig.2. It mainly includes seven parts as follows:

- **Select a *Graph***
  First we select a simple directed graph for special domain as shown in Fig.3.

Fig.3 represent a simple directed graph where represent pages related to a particular domain and represent pages those are not relevant to that domain.

- **Computing *Indegree***
  All web pages are considered to construct indegree set through given adjacency matrix of graph. Count number of pages pointed to a page.
- **Computing *Outdegree***
  All web pages are considered to construct outdegree set through given adjacency matrix of graph. Count number of pages pointed from it.
- **Computing Pagerank**
  Calculate pagerank for all pages, i.e. initial ranking of any page. Assign rank 1 to page with highest pagerank value.

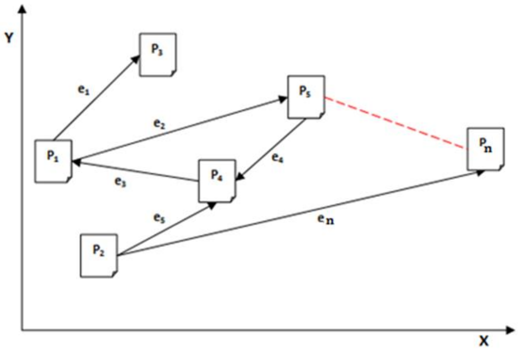

Figure 1. Simple Directed Graph

Manuscript published on 28 February 2019.
***Correspondence Author(s)**

Suvarna Sharma, Department of Mathematics and Computer Appications, Maulana Azad National Institute of Technology, Bhopal, India.
Amit Bhagat, Department of Mathematics and Computer Appications, Maulana Azad National Institute of Technology, Bhopal, India.

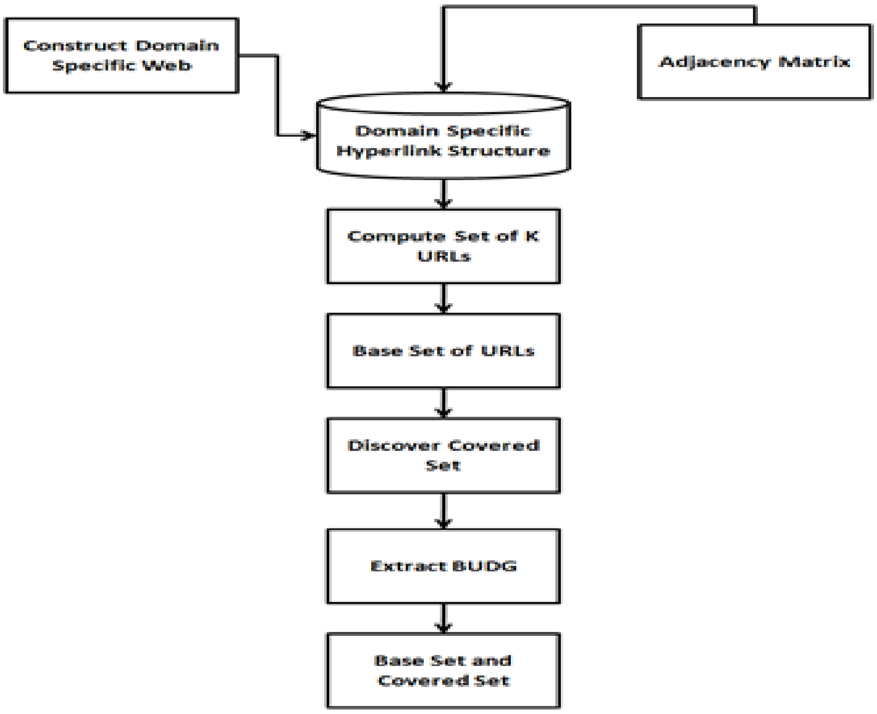

Figure 2. Proposed Approach

- **Initialize *Seed Urls Set* and *Covered Set***
  First we choose size of Seed Urls set than anyone of given approach. This set of nodes fetched other nodes, so these nodes must be included in the C*overed Set* first.
- **Collect Covered Nodes**
  From initialized *Cover Set* find out nodes those are connected through hyperlinks, add them to Cover Set. Repeat this step until all the nodes of the Cover set are not visited.
- **Expand Graph**
  Perhaps Covered set will modify very often so it have new components. So we have repeat last steps for all components.

II. RELATED WORK

At present, there are obscure researches on the seed URL cull. Cull of seed URLs has a paramount effect on the search engine results quality. For a large dataset, like web data, it is difficult to collect seed URL set that represent the entire domain.Mirtaheri et al. [3] described the comparative study on web crawlers, from early days to the recent days, based on performance and objectives of crawlers. This paper briefly discussed the techniques and algorithms used by crawlers. In this paper at broad level crawler classified into three categories such as Traditional Crawlers, Deep Web Crawlers, and RIA (Rich Internet Application).

Generally in domain-specific search engine collection of seed URLs done manually. But, In 2009 Zheng et al. [5] first introduced a graph based Crawler Seed Selection approach. In this they present an effective algorithm for seed selection based on Maximum K-Coverage Problem. This approach does not focus on the Domain Specific Seed URLs Selection problem.

In 2008 an approach was introduced i.e. Host Based Seed Selection Algorithm for Web Crawlers [6]. This algorithm based on features of quality importance and potential yield for seed URLs selection of the host.

In 2014 Priyatam et al. [7] proposed an algorithm for diverse seed URLs selection. They used Twitter URL graph to automate the seed URLs selection process. They proposed various methods for computing similarity between URLs such as Content similarity, URL N-Grams similarity, User Similarity and Zero similarity. They compare their algorithms performance with baseline zero similarity seed selection.

In 2014 another approach introduced for seed URLs selection for focused crawler by Du et al. [8]. This approach based on user-interest ontology. They used user log profile for applying formal concept analysis to construct user-interest concept lattice.

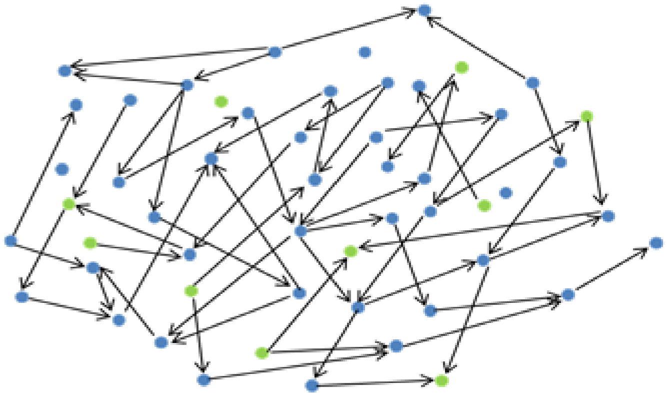

Figure 3. Web as a directed graph

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

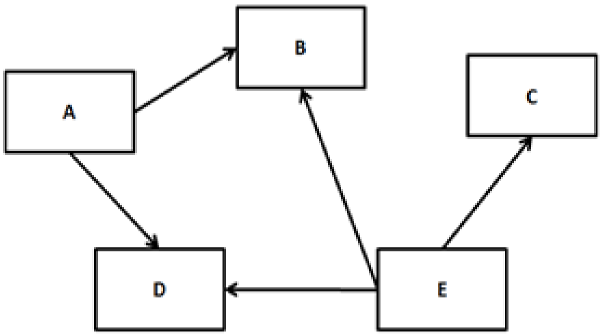

Figure 4.

III. PRELIMINARIES AND PROBLEM DEFINITION

Definition 1. (URLs set) $U = \{u_1, u_2, u_3, \dots \dots u_n\}$ is the set of URLs. Each URL is associated with a domain specific web page.

Definition 2. (Simple Directed Graph) A simple directed graph is a directed graph having no multiple edges or graph loops. In terms of web *G = (V, E)* is a simple directed graph in which the web documents can be viewed as vertices and the hyperlinks as directed edges.

Fig.4 shows an example of simple directed graph [9].

Definition 3. (Indegree) The in-degree of a page is the number of nodes that have links to it. Table I shows indegree of nodes of graph represented in Fig.4.

Table I

Nodes	Indegree
A	0
B	2
C	1
D	2
E	0

Definition 4. (Outdegree) The out-degree of a page is the number of nodes that have links from it. Table II shows outdegree of nodes of graph represented in Fig.4.

Table II

Nodes	Outdegree
A	2
B	0
C	0
D	0
E	3

Definition 5. (Page Rank value). Page rank value use the link structure of the web to determine the importance of web pages. Page with higher page rank value is most important. A simplified version [1] of PageRank is defined in Eq. 3:

$$\mathrm{PR}(u) = (1 - d) + d \sum_{v \in B(u)} \frac{\mathrm{PR}(v)}{N_v} \qquad (1)$$

Where u represents a web page, *B(u)* is the set of pages point to *u*, *PR(u)* and *PR(v)* are rank scores of page *u* and *v* respectively, *NV* denotes the number of outgoing links of page *v* and *d* is the damping factor that denotes the probability of deriving a particular node through a random traverse. A standard value of d is 0:85. Conversely, *(1 - d)* is the probability to stay on the current web page.

Table III shows Ranking of nodes of graph represented in Fig. 4.

Table III

Page	Rankvalue	Index
A	0.15	4
B	0.21	3
C	0.21	2
D	0.28	1
E	0.15	5

Definition 6. (Seed Set) Seed URLs set $M = \{u_1, u_2, u_3, \dots \dots u_k\}$ is the set of pages s.t. $k \le n$ and for selection of these URLs we used different methods.

Definition 7. (Visited Set) Visited URLs set $V = \{v_1, v_2, v_3, \dots \dots v_k\}$ is the set of pages that we can traverse through the URLs of set S.

Problem statement Because of rapidly growing data on Web, different challenges are arise for search engine. Crawling is an important phase of a search engine. So many crawling techniques are available at present and all of them are working in different ways. All crawling techniques starts with a given set of URLs i.e. referred as Seed URL. The problem of finding a minimal URLs set for a simple directed graph so through this set we can access all nodes of that graph.

IV. PROPOSED APPROACH

We pose the problem of seed selection as a tree traverse problem. As mentioned in section 3 we form a graph of *N* vertices where each vertex is a URL from a specific domain (s.t. abortion, movie etc.). Two URLs are connected if there is at least one hyperlink from one to another. Algorithm 1 shows how the URLs traverse to seed URLs is done. We used various methods of selecting seed URLs between these URLs which are explained in the next section.

Dataset

The Nodes file: Format of Node file is given below. First there is an entry that gives the number of pages in the graph. Then there is a list of the page entries. An example of a page entry is the following [10][11]:

34(67)
http://www.ece.wpi.edu/~jinlee/events/wave/sld024.htm
Accuracy&ComputationalComplexity
0 1

The first number is a unique identifier for each page i.e. page id. When a page entered in the system an id assigned to the page that written followed by the page id (this can be ignored). The following line is the http addresses of the page and the next the title of the page. The last line has the two numbers that are in and out degree of the page. The **Adjacency Matrix:** This file contain adjacency matrix for the nodes of Nodes file.

Section 6 shows the working of the algorithm with an example. In the example we have twelve vertices and size of seed URLs set is predefined *(K = 3)*. As we can see that the final set of URLs returned by the algorithm are a set of URLs those are reachable from seed URLs.

Having explained our selection algorithm, we now propose different ways of selecting URLs between given URLs of a particular domain. Later, we compare the performance of each of these on the basis of visited URLs, Iteration, process timing.

V. SEED SELECTION

In the previous section we have seen how to extract *‘K’* URLs given a graph of connected URLs. Thus far, we have assumed that we already have a graph where URLs related to a domain are connected. This section explains various ways in which such a graph can be constructed using hyperlinks.

A. Random Selection

Number We select *‘K’* URLs randomly then apply traversing algorithm to find URLs that connected to these URLs. This is based on the random selection of URLs that relevant to domain.

Eq. 4 shows the formula to select URLs randomly, where *N and S* are the Number of total URLs related to domain and random selected *‘K’* URLs from *0* to *N-1* indices.

$$M = Random\{0\ to\ N-1\}$$

(4)

If such a graph is given as an input to Algorithm *1*, it would merely return any random *‘K’* URLs from the graph, this is our baseline system.

B. 0-Indegree Selection

We select *‘K’* URLs with in-degree is *0* then apply traversing algorithm to find URLs that connected to these URLs. This selection based on indegree and select URLs that relevant to domain and having *0-indegree*. Eq 5 is nothing but a case is used to calculate Seed URL set from URLs those indegree is *0*.

$$Seed\ Set\ M = \{s | s \in U\ and\ indegree(s) = 0\} \qquad (5)$$

C. Highest Outdegree Selection

In this approach we use outdegree of pages, pages with higher outlinks may fetch more pages. This is based on the assumption that, if one page have higher outdegree able to fetch more page.

We select *K* URLs with highest outdegree then same traversing algorithm to find URLs that connected to these URLs. This approach is meant to collect more pages.

D. Highest Pagerank Selection

In this approach we use a simple ranking assign to a page while it first entered in the system. The evaluation of ranking is used the *PageRanking*[1] formula:

$$\mathrm{PR(u)} = (1-\mathrm{d}) + \mathrm{d}\sum_{\mathrm{v}\in\mathrm{B(u)}}\frac{\mathrm{PR(v)}}{\mathrm{N_v}}$$

Where *u* represents a web page, *B(u)* is the set of pages point to *u*, *PR(u)* and *PR(v)* are rank scores of page *u* and *v* respectively, N_v denotes the number of outgoing links of page *v* and *d* is the damping factor that denotes the probability of deriving a particular node through a random traverse. A standard value of d is 0.85. Conversely, *(1 - d)* is the probability to stay on the current web page.

Seed URL set contains those URLs have highest page rank. It would be noteworthy to mention that the ranking can also be used to construct a graph. The ranking of a page consists of the importance of that. The rationale behind this approach is that important page may point to other important pages about the same event/entity. The purpose here is to get a set of important URLs.

Algorithm 1. Extracting the BUDG from DG

```
1: $G = (V, E)$ and define Seed URL size K.
2: Initialized Seed URLs set $M = \{\phi\}$ and Covered nodes set $C = \{\phi\}$.
3: Sequential Selection $K$ URLs as Seed URLs Set $M$
4: $C = M$ .
5: $U = V - M$ .
6: for each $u \in C$ do
7: extract URL $v$ .
8: If $v \notin C$
9: $C = v \bigcup C$ .
10: end if.
12: end for.
13: return $M$ and $C$ .
```

The algorithm is used for discovering Seed URLs Set that is used to traverse a graph of pages. Step 2 selects *K* nodes. We propose and evaluate four approximation procedures for performing Step 2:

M1: Choose a page p randomly

M2: Choose a page p with indegree-0

M3: Choose a page p with the highest outdegree

M4: Choose a page p with the highest pagerank

In our approach we find visited nodes, evaluation time and percentage of iteration. For calculating percentage of iteration we used Eq. 6

$$I_p = \frac{T_i}{N \times C} \times 100 \qquad (2)$$

Where I_P is percentage of Iterations, T_i is total iteration, *N* is number of nodes in domain and *C* is covered nodes.

VI. EXPERIMENT AND RESULTS

With the help of Fig. 5 we can easily interpret the all selection approaches:

$$V = \{A, B, C, D, E, F, G, H, I, J, K, L\}$$

Random Selection Approach

Fig.6. shows the Random selection approach on given example:

Visited Set $C = \{A, J, F, B, E, D, G, H\}$

0-Indegree Selection Approach

Fig.7. shows the Indegree-0 selection approach on given example:

Visited Set $C = \{A, J, C, B, E, D, L, F, H, G\}$

Highest Outdegree Selection Approach

Fig. 8. shows the Highest Outdegree selection approach on given example:

Visited Set $C = \{A, D, C, B, E, H, F, L, G\}$

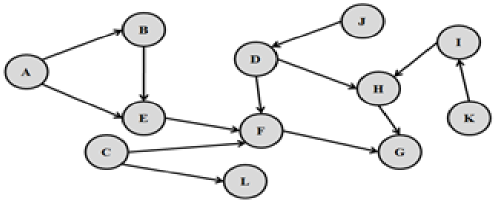

Figure 5.

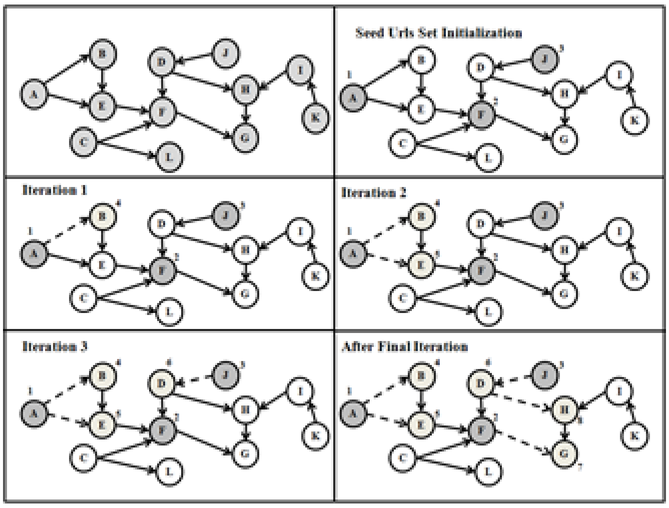

Figure 6. Random Selection Approach

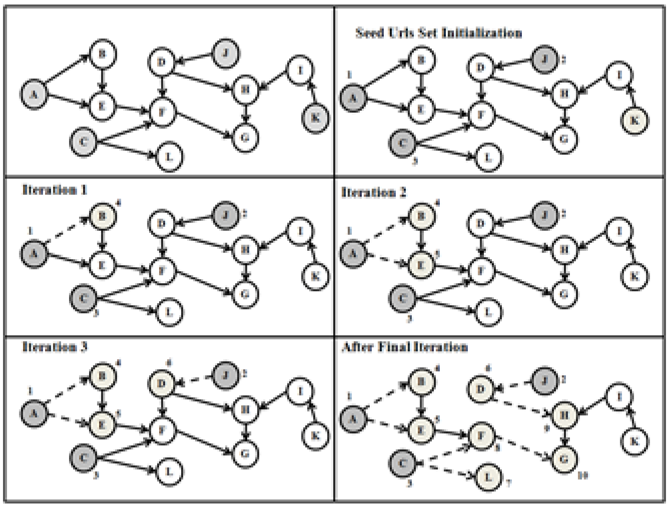

Figure 7. 0-Indegree Selection Approach

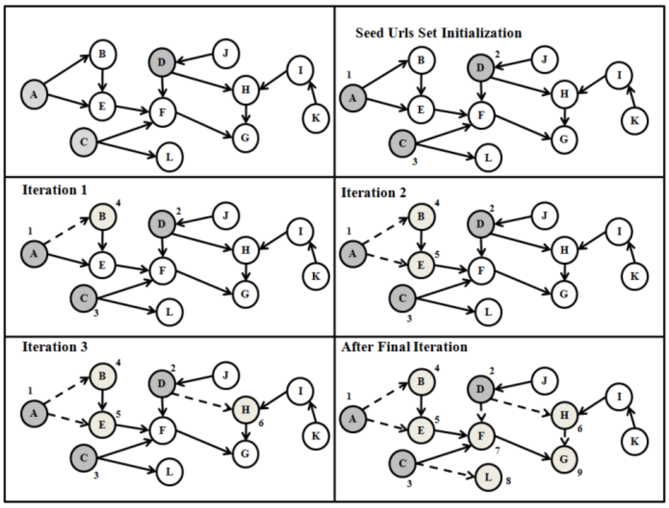

Figure 8. Highest Outdegree Selection Approach

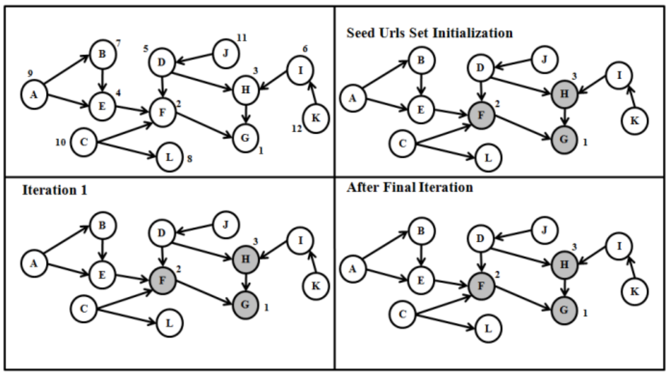

Figure 9. Highest Pagerank Selection Approach

Highest Pagerank Selection Approach

Table.4 indicates the pagerank value and rank of nodes for the given example:

Fig. 9 shows the Highest Pagerank selection approach on given example:

Visited Set $C = \{G, F, H\}$

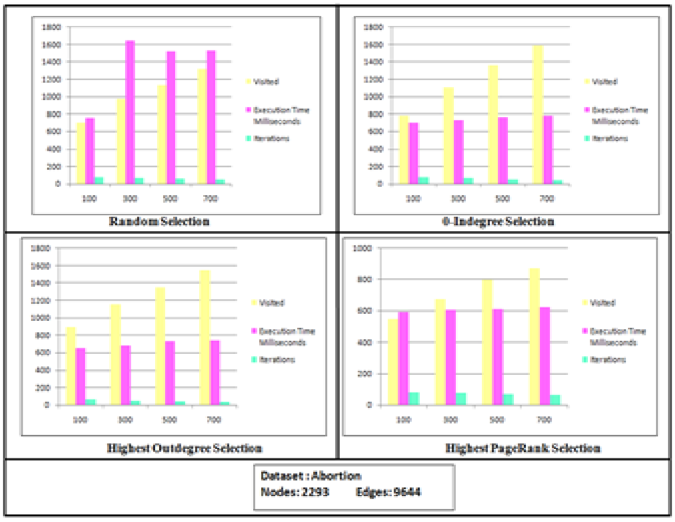

Figure 10. Results of all approaches on the sample 'Abortion' domain

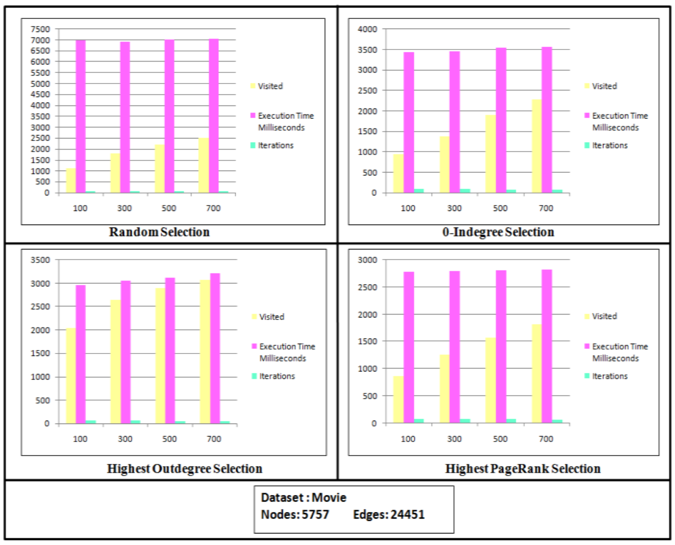

Figure 11. Results of all approaches on the sample 'Movie' domain

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

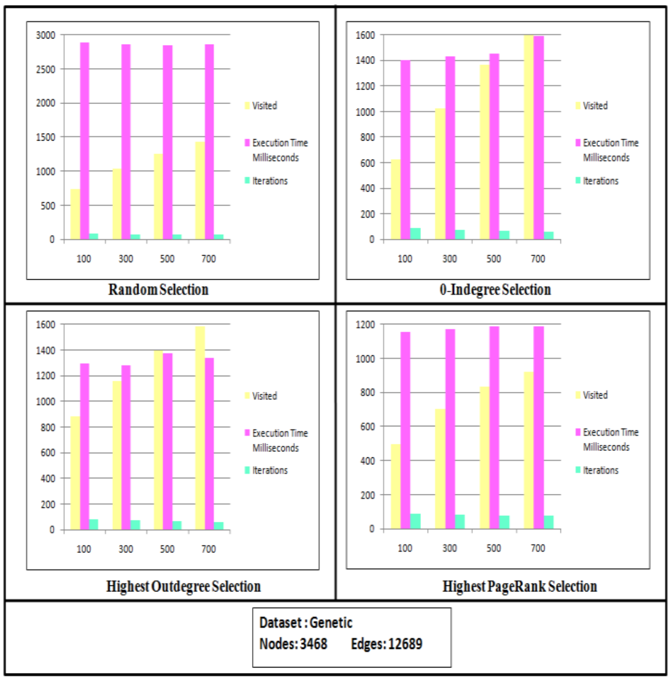

Figure 12. Results of all approaches on the sample ‘Genetic’ domain

Fig. 10, Fig. 11, Fig. 12, Fig. 13 and Fig. 14 shows the histogram of Visited Nodes, Iterations and Evaluation Time for five sample dataset respectively ‘Abortion’, ‘Movie’, ‘Genetic’, ‘Gun Control’, ‘Net Censorship’ with Seed set of different sizes.

VII. CONCLUSION

The seed URLs selection for the domain specific web crawler is an important research in search engine; most of researches focus on coverage, freshness, politeness etc. features of crawler. In this paper we addressed the problem of seed URLs selection for crawling. We experimentally evaluated seed URLs for further process based using the hyperlinks available on Web pages.

In general, our results show no clear winner.

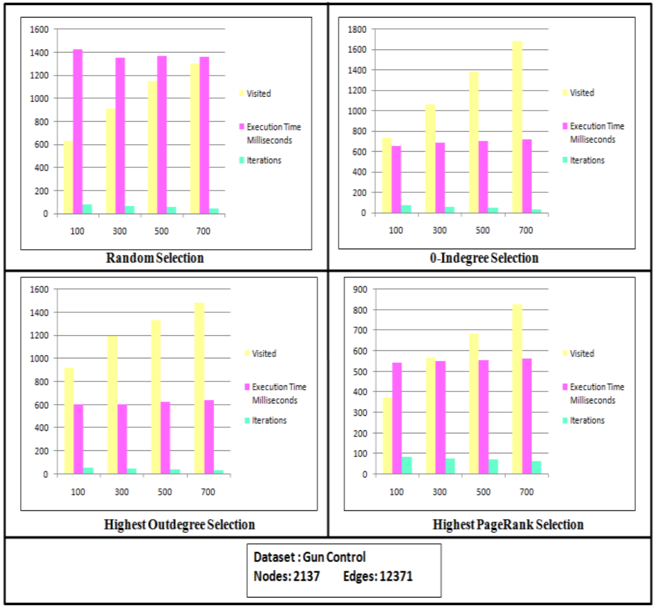

Figure 13. Results of all approaches on the sample ‘Gun Control’ domain

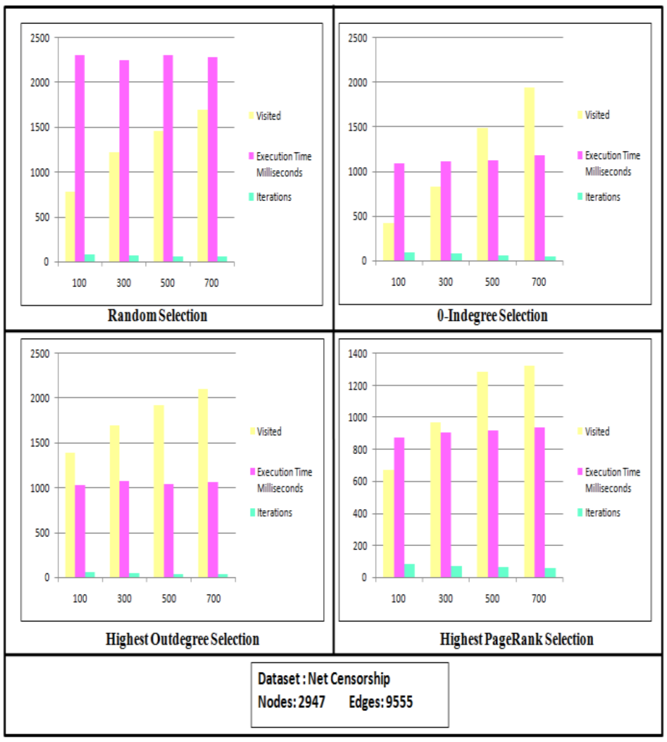

Figure 14. Results of all approaches on the sample ‘Net Censorship’ domain

In the light of these results, it seems approach based on the highest outlinks is better among all strategies. URL’s Set, S, is a Seed Set of URLs that used to covers graph through hyperlinks that set of covered nodes referred as visited nodes. In addition, if the page is already visited it skips the page, it is useful to save seed URLs set rather than random URL set:

- *Random selection* is not trustworthy sometimes it give good result.
- *Indegree-0 degree selection* traverses more nodes but those may be important or may not be.
- *Highest Outdegree selection* traverses more nodes and covered set contains more important pages rather than other approaches.
- *Highest Pagerank selection* traverse important pages first but size of covered nodes is less than *Highest Outdegere selection*.

With a seed URLs selection, we can build crawlers that can use URLs set of the pages for extracting other pages. This property can be extremely useful when we are trying to crawl a fraction of the Web, when we have limited resources, or when we need to visit pages links to those pages.

Nevertheless, it will be useful to discover in the future for other dataset of Web pages to analyze best seed URLs and the generated seed URLs are provided to be accessible by a crawler.

REFERENCES

1. S. Brin, and L. Page , “The anatomy of a large-scale hypertextual web search engine,” Computer networks and ISDN systems, vol. 30, no. 1,pp.107-117,Apr. 1998.
2. S. Sharma, A. Bhagat, “Research on Ranking Algorithms in Web Structure Mining,” International Journal of Knowledge Based Computer Systems, vol. 3, no. 2, pp.13-20, Dec. 2015.

3. S. Mirtaheri, M. E. Dincturk, S. Hooshmand, G. Bochmann and G.-V. Jourdan, “A Brief History of Web Crawlers,” Proc. of the 2013 Conf. of the Center for Advanced Studies on Collaborative Research. IBM Corp, pp.40-54, Nov. 2013.
4. C. Olston and M. Najork, “Web crawling,” Foundations and Trends in Information Retrieval, vol. 4,no. 3, pp.175-246, Feb. 2010.
5. S. Zheng, P. Dmitriev, and C. Giles, “Graph based crawler seed selection,” In Proc. of the 18th ACM international Conf. on Information and knowledge management, ACM, pp.1089-1090, Nov. 2009.
6. P. Dmitriev, “Host-based seed selection algorithm for web crawlers,” US Patent App. 12/259,164, Oct. 2008.
7. P. N. Priyatam, A. Dubey, K. Perumal, S. Praneeth, D. Kakadia, and V. Varma, “Seed Selection for Domain-Specific Search,” In Proc. of the 23rd International Conf. on World Wide Web, ACM, pp.923-928, April 2014.
8. Y. J. Du, Y. F. Hai, C. Z. Xie, and X. M. Wang, “An approach for selecting seed URLs of focused crawler based on user-interest ontology,” Applied Soft Computing , (Elsevier) , vol.14, pp.663–676, Jan. 2014.
9. Weisstein and W. E., “Website of the Simple Directed Graph – from Wolfram Math world,” 1996
10. J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” Journal of the ACM, vol.46,no.5, pp.604-632, Sep. 1999.
11. TORONTO.EDU, “Website of the Datasets for Experiments on Link Analysis Ranking Algorithms,” http://www.cs.toronto.edu/tsap/experiments/datasets/index.html, 1986.

AUTHORS PROFILE

Suvarna Sharma has received her B.Sc. degree in Computer Science from Jiwaji University, Gwalior, Madhya Pradesh in 2005. She has received M.Sc. degree in 2007 from Jiwaji University, Gwalior, Madhya Pradesh and M.Tech. Degree from Davv, Indore, Madhya Pradesh in year 2013. She is currently pursuing her Ph. D. degree from the Department of Mathematics and Computer Applications, Maulana Azad National Institute of Technology, Bhopal, Madhya Pradesh, India. Her research interests include Web Mining, Web Structure Mining and Web Crawling from Web Data.

Amit Bhagat has received his B.C.A and MCA degree in Computer Applications from Makhanlal Chaturvedi National University of Journal- ism, Madhya Pradesh in the year 2000 and 2003. He has done PhD from MANIT Bhopal in the year 2013. He is currently working as Assistant Professor in the Department of Mathematics and Computer Applications, Maulana Azad National Institute of Technology, Bhopal, Madhya Pradesh, India. His research interests include Data Mining, Neural Networks, Sentiment Analysis, Web Mining and Big Data.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)