A Hybrid Revisit Policy For Web Search

[1]Vipul Sharma , [2]Mukesh Kumar, [3]Renu Vig
UIET, Panjab University Chandigarh, INDIA
[1]vipul_85cse@yahoo.co.in
[2]{mukesh_rai9@yahoo.com,mukesh_rai9@pu.ac.in}
[3]renuvig@hotmail.com

***Abstract* – A crawler is a program that retrieves and stores pages from the Web, commonly for a Web search engine. A crawler often has to download hundreds of millions of pages in a short period of time and has to constantly monitor and refresh the downloaded pages. Once the crawler has downloaded a significant number of pages, it has to start *revisiting* the downloaded pages in order to refresh the downloaded collection. Due to resource constraints, search engines usually have difficulties keeping the entire local repository synchronized with the web. Given the size of web today and inherent resource constraints: re-crawling too frequently leads to wasted bandwidth, re-crawling too infrequently brings down the quality of the search engine. In this paper a hybrid approach is build on the basis of which a web crawler maintains the retrieved pages "fresh" in the local collection. Towards this goal the concept of Page rank and Age of a web page is used. As higher page rank means that more number of users are visiting that very web page and that page has higher link popularity. Age of web page is a measure that indicates how outdated the local copy is. Using these two parameters a hybrid approach is proposed that can identify important pages at the early stage of a crawl, and the crawler re-visit these important pages with higher priority.**

Index Terms – Revisit Policy, Search Engines, Web Crawler

I. INTRODUCTION

A Web crawler [1] is a program that downloads Web pages, commonly for a Web search engine or a Web cache. Roughly, a crawler [1] starts off with an initial set of URLs S_0. It first places S_0 in a queue, where all URLs to be retrieved are kept and prioritized. From this queue, the crawler gets a URL (in some order), downloads the page, extracts any URLs in the downloaded page, and puts the new URLs in the queue. This process is repeated until the crawler decides to stop, for any one of various reasons. Every page that is retrieved is given to a client that saves the pages, creates an index for the pages, or analyzes the content of the pages. Crawlers are widely used today. Crawlers for the major search engines (e.g., Google, AltaVista, and Excite) attempt to visit a significant portion of textual Web pages, in order to build content indexes. Other crawlers may also visit many pages, but may look only for certain types of information (e.g., email addresses). At the other end of the spectrum, we have personal crawlers that scan for pages of interest to a particular user, in order to build a fast access cache.

By Web crawling we mean, a process by which we collect web pages, index them and support a search engine. The main objective of crawling is to quickly and efficiently gather useful web pages along with the link structure, which are of more concern to the user. That is why web crawlers are also called as robots and spiders. Web pages are frequently updated by the content providers, freshness of the search engine's index is always endangered and crawling is never ending process. It's very hard to synchronize the local repository and the live web pages because of the resources constraints and the size of the web. However if the crawler have more information about the update schedule of the content providers, the crawling decision will become easier and the process will be more efficient. For example, a normal homepage websites periodically refreshes their content. Most probably they refresh it within 24 hours or less. Sports website updates at least their content after every match or game and the news website updates most frequently, which depend on the happenings around. Also commercial websites have their routine update schedule. All these updates can be collaborated with the search engine to increase their efficiency and the effectiveness of the crawling.

The Web has a very dynamic nature, and crawling a fraction of the Web can take a really long time, usually measured in weeks or months. By the time a Web crawler has finished its crawl, many events could have happened. These events can include creations, updates and deletions. From the search engine's point of view, there is a cost associated with not detecting an event, and thus having an outdated copy of a resource. The objective of the crawler is to keep the average freshness [4] of pages in its collection as high as possible, or to keep the average age of pages as low as possible. These objectives are not equivalent: in the first case, the crawler is just concerned with how many pages are out-dated, while in the second case, the crawler is concerned with how old the local copies of pages are. A crawler needs to revisit Web pages in order to maintain the local collection up-to-date. From the search engine's point of view, there is a cost associated with not detecting an event, and thus having an outdated copy of a resource.

II. PROBLEM STATEMENT

How should the crawler refresh the pages stored in local collection?

Web pages are frequently updated by the content providers, freshness of the search engine's index is always endangered and crawling is never ending process. It's very hard to synchronize the local repository and the

doi:10.4304/jait.3.1.36-47

live web pages because of the resources constraints and the size of the web. Due to resource constraints, search engines usually have difficulties keeping the entire local repository synchronized [4] with the web. Given the size of web today and inherent resource constraints: re-crawling too frequently leads to wasted bandwidth, re-crawling too infrequently brings down the quality of the search engine. There have been several studies of web crawling in its relatively short history. Cho and Garcia-Molina [5] concluded that Pages in .com were the shortest-lived, with a half of all .com pages changing within 11 days, while those in .gov and .edu experienced change at a far slower rate. Cho et al.[2] [4] , introduce some new design and performance improvement. This property can be useful when we are trying to crawl a fraction of the Web with some limited resources. Hadrien Bullot and S.K. Gupta [7] introduce data mining approach for optimizing performance of an Incremental Crawler. With the method presented here, it is the user who chooses which pages the crawler must update. K.S. Kuppusamy, G. Aghila [13] The approach provided in this paper involves user participation in larger extent in order to get the focused and more relevant information.

A common drawback of all these approaches is that they involve user participation in larger extent in order to get the focused and more relevant information. It is the user who chooses which pages the crawler must update. The pages, which are not very popular, are not updated frequently by the crawler.

Moreover, the major difficulty is to stimulate the different behaviour of users. It is obvious that a novice user will not have the same behaviour than a person who has advanced computer skills.

As web pages are changing at very different rates, the crawler needs to carefully decide which pages to revisit and which pages to skip in order to achieve high “freshness” of pages.

A. Related Work

Cho and Garcia-Molina [5] collected data from 720,000 pages on 270 Web sites over a period of four months in 1999. The pages were downloaded daily and compared to a recorded checksum to determine whether a page had changed. Their primary focus in this effort was to model proposed estimators for the frequency of change against real-world data and to derive some idea of the lifespan of a Web page. Although their data does not provide much measure of the degree of change, they do find a number of interesting results related to the domain of sites and its effect upon the frequency of updates. Notably, more than 70% of the pages across all domains were unchanged for at least one month and 50% of pages in the .gov and .edu domains lasted for more than 4 months (the duration of their study). *Pages in .com were the shortest-lived, with a half of all .com pages changing within 11 days, while those in .gov and .edu experienced change at a far slower rate.*

In a series of papers, Cho [2] [4], introduce some new design and performance improvement. In [2] they examine different crawling strategies using the Stanford University intranet. Their approach it to visit more important pages first. They showed that a crawler with a good ordering scheme can obtain important pages significantly faster than one without. *This property can be useful when we are trying to crawl a fraction of the Web with some limited resources.*

A.K. Sharma and Ashutosh Dixit [12], proposed an efficient approach for building an effective incremental web crawler [5]. It selectively updates its database and/or local collection of web pages instead of periodically refreshing the collection in batch mode there by improving the freshness of the collection significantly and bringing new pages.

Rahul Choudhari and Ajay Choudhari [11], address the scheduling problem and solution for the web crawlers with the objective of the optimizing the resources like freshness of repository and the quality of the index. Towards this, they divided the web content providers into two parts: 1) active 2) inactive. For inactive content providers they use agents who continuously crawls the content providers and collect the update pattern of the content providers.

Hadrien Bullot and S.K. Gupta [7] introduce data mining approach for optimizing performance of an Incremental Crawler. The information collected from the users can help the crawler to know which the popular pages are and to revisit them as soon as possible. *With the method presented here, it is the user who chooses which pages the crawler must update.* The pages, which are not very popular, are not updated frequently by the crawler.

K.S. Kuppusamy, G. Aghila[13] A multi-step feedback centric web search engine ensuring the retrieval of relevant fresh live results instead of those existing in the indexes. The methodology is based on the new concept called "Micro Search" which in turn creates the "Micro Indexes". These micro-indexes are the key factors utilized in re-ranking the selected documents. *The approach provided in this paper involves user participation in larger extent in order to get the focused and more relevant information.*

In the literature, a Poisson process is often used to model the change of a web page. We believe that it is a good model because a Poisson process models a sequence of random events that happens independently with fixed rate over time. Also, we make the assumption that web page change at a uniform rate. Thus, the average rate of change λ is uniform.

From literature survey it is concluded that:

- Most of the approaches involve user participation in larger extent in order to get the fresh and more relevant information. It is the user who chooses which pages the crawler must update. The pages, which are not very popular, are not updated frequently by the crawler.
- Moreover, the major difficulty is to stimulate the different behaviour of users. It is obvious that a novice user will not have the same behaviour than a person who has advanced computer skills.

- As web pages are changing at very different rates, the crawler needs to carefully decide which pages to revisit and which pages to skip in order to achieve high "freshness" of pages.
- Revisiting the frequently changing web pages cannot obviously improve the effect of search engine. We should focus the resources on the web pages changing not so quickly. However a problem is still existent. Page update frequency has been modeled with statistical functions such as Poisson distributions. Actually these models are not exact. There are numerous of unchanged web pages during the updating period. And re-crawling unmodified web pages implies a cost in terms of network bandwidth and resource usage. Consuming little for a single page, it becomes considerable on large scale. Under the typical refreshing strategy the crawler revisits all pages at the same frequency regardless of how often they change. In fact a large number of pages change very slowly.

III. PROPOSED APPROACH

In proposed approach, a crawler revisits a web page on the basis of Page Rank & Age of web page. Basically Page Rank is a link analysis algorithm that defines the link popularity [3] of web page. Higher Page Rank means that more number of users are visiting that web page. Age [4] of web page is a measure that indicates how out-dated the local copy is. In the previous research most of the approaches involve user participation in larger extent in order to get the fresh and more relevant information. It is the user who chooses which pages the crawler must update. So taking into account the user preference Page Rank [3] is taken as a standard. As Page rank [3] measures the relative importance of a web page. In addition to Page Rank, Age of web page is also used as a measure for revisiting the web page. Age of web page is defined as time difference between the Present Date time & Last modified date time. Present Date time is the time when we are calculating the Age and Last modified Date time is the Time when the web page is last modified. The last modified date of web page is taken form the HTTP Header information. We send HTTP Web Request to Server and server respond us back by the last modified date. Thus the age of web page is calculated. In order to get the Page Rank of web Page, we send a Query to one of the local Google Data Centers. After getting the Page Rank and Age of web page, refresh score is calculated. Using this refresh score two collections are made from the initial collection on the basis of Pareto's Principle. One is named as 20% Collection that contains 20% of web pages that have higher score and other is named as 80% Collection that contains all the remaining web pages. Generally the 20% collection contains the web pages that have less age & higher Page rank value. Thus the crawler revisits the web pages in the 20% collection with higher priority, giving to user fresh results. The rate at which crawler re-crawls the 20% collection is high as compared to the rate at which crawler re-crawls the 80% collection, as 80% collection contains more number of web pages.

A. Metrics for Solving the Problem

Page Rank (PR) :- PageRank[3] is a link analysis algorithm, named after Larry Page, used by the Google Internet search engine that assigns a numerical weighting to each element of a hyperlinked set of documents, such as the World Wide Web, with the purpose of "measuring" its relative importance within the set. Higher Page Rank means that more number of users are visiting that very web page. The numerical weight that it assigns to any given element *E* is also called the *PageRank of E* and denoted by *PR* (*E*).

Age of the page (Ag):- Age [4] of a page is a measure that indicates how outdated the local copy is. It is defined as the time difference between the Present Date Time and Last Modified Date Time.

Page Score (PS):- Let a element e_i with Page Rank ($PR(e_i)$) and Age of Page($Ag(e_i)$). Page Score a quantity which is dependent on Page Rank and inversely dependent on Age of page of a particular element.

Page Score is Directly Proportional to the Page Rank.
& Page Score is Inversely Proportional to the Age.
i.e. Page Score =K*Page Rank/Age Where K is Constant & K=1.
So, Page Score = Page Rank/Age

Pareto's Principle: The Pareto's principle[16] also known as 80-20 rule, the law of vital few states that, for many events, roughly 80% of effects comes from 20% of the causes. In computer science the Pareto's principle can be applied to optimization efforts.

B. Algorithm Representation

Table: 3.1 Algorithm Representation

Step 1: Pick URL from Initial Collection of URLs.
Step 2: Send Request to Google Data Centers to find Page Rank of Web Page.
Step 3: Get the Last modified date of the Web Page from HTTP header info.
Step 4: Calculate Age of Web Page as:
F(ag)= Pdt-Ldt where Pdt is Present Date Time
Ldt is Last Modified Date Time
Validate
If (Days=0)
Return Hours
Else
Return Days * 24 + Hours
Step 5: Compute the Score using page rank and age.
f (PS) = (PR)/(Ag)
where PR is Page Rank of web Page
Ag Is Age of web Page
If Age = 0 Then
Return Page Rank
Else

PS = (PR) / (Ag)

Step 6: Distribution according to Pareto's principle.
i.e. In the ratio of 80:20(A:B)
B Collection -> 20% Webpages with highest Score value.
A Collection -> 80% Remaining Webpages.

Step 7: Generate Crawler Initiation Policy.

Step 8: Activate Crawler as per Initiation Policy.

Step 9: Log the statistics for the verification purpose.

C. Decision Architecture

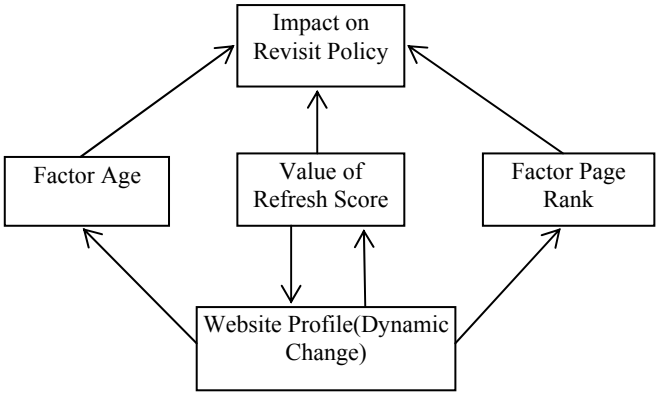

Figure 3.1: Decision Architecture

Figure 3.1 shows the decision architecture. Web server builds and maintains the meta-data files for websites. These files are dynamic in nature. As website is updated by the content provider the meta-data associated with that very website is also changed. So website profile is dynamic in nature. A hybrid approach is based on the Pareto's distribution in which two collections are formed. The decision that which web page goes in which collection largely depends upon the Refresh score. This score is not static. After re-crawling the web pages in 20% collection the score is again calculated.

IV. IMPLEMENTATION

A. Creating Initial Collection

In initial collection 26 websites are taken from different sources. Each website is associated with some parameters like Page Rank, Score value, Submitted by & Submission Date. Initially each website has Page rank and Score value as 0. Submission Date is the date when the website is stored in the collection. Below is the snapshot of creating the initial collection.

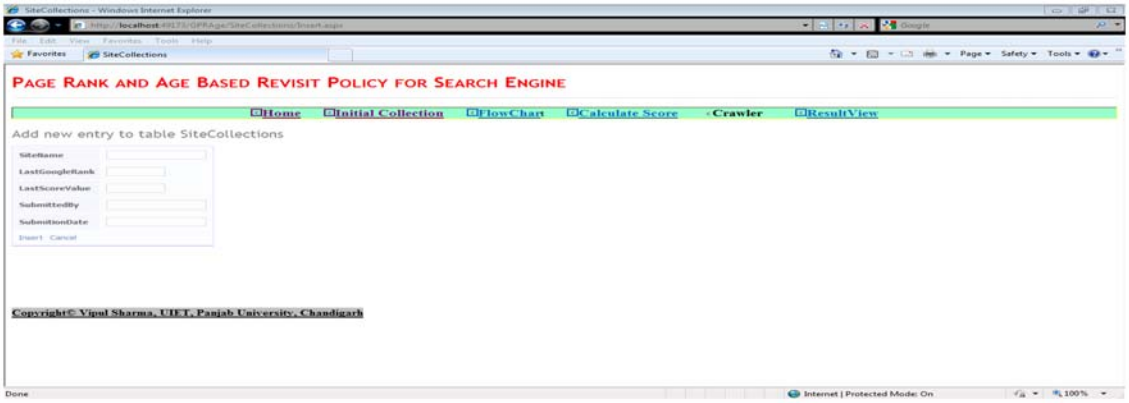

Figure 4.1: Creating Initial Collection

It is also possible to update the information about any web page stored in the initial collection.

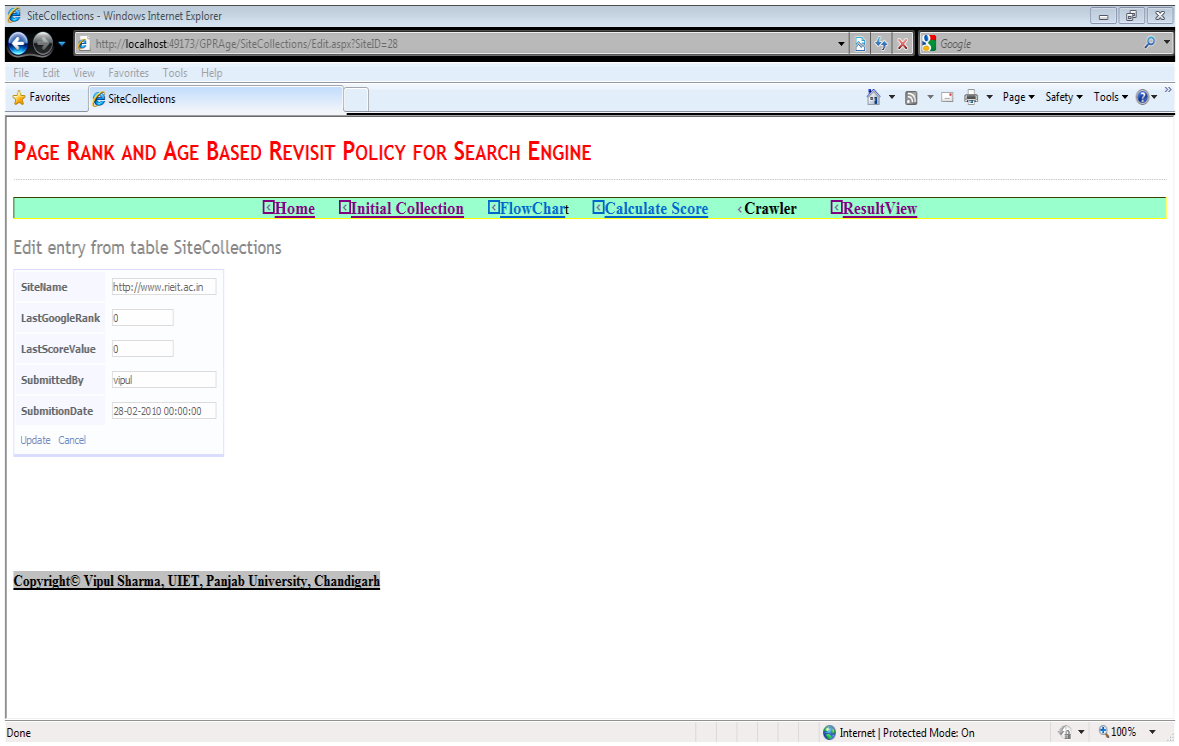

Figure 4.2: Edit entry from initial collection

Some of the websites in the initial collection.

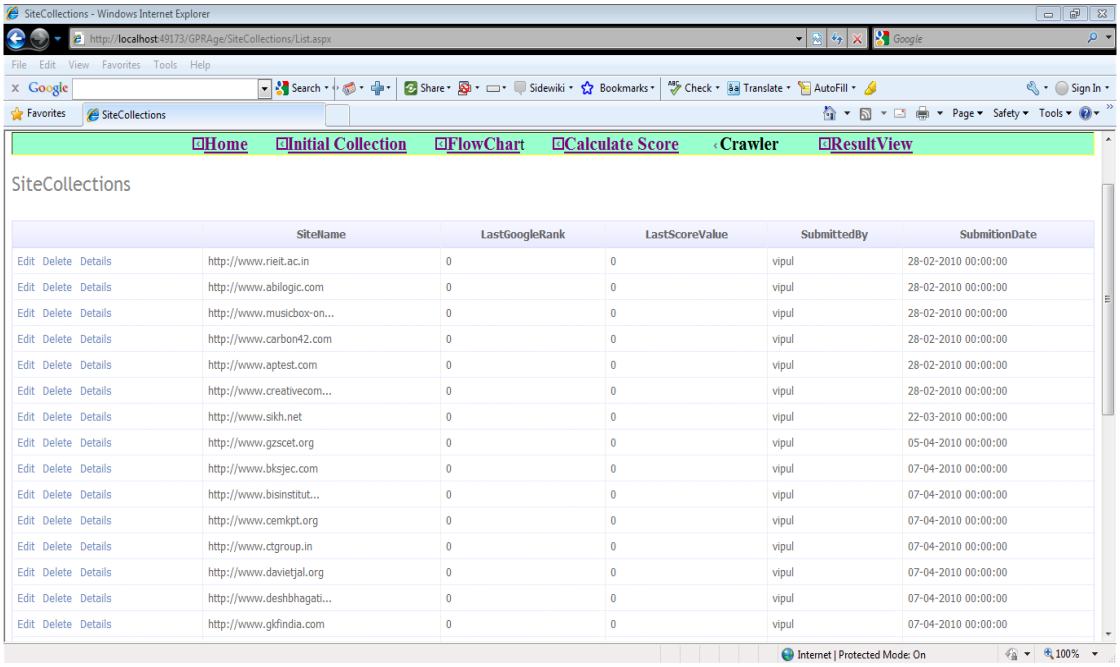

Figure 4.3: Websites in the Initial Collection

B. Finding Score

The refresh score of each website stored in the initial collection is calculated using the algorithm as mentioned in section 3.2. The score is calculated using Page Rank & Age of web page. Three collections are made i.e. Initial collection (Sorted according to score value), 20% collection (20% websites with higher score), 80% collection (Remaining Websites). The 20% collection and 80% collection are made from the Initial collection. Figure 4.4 shows the Page Rank, Age, Last Modified Date and Score of each website.

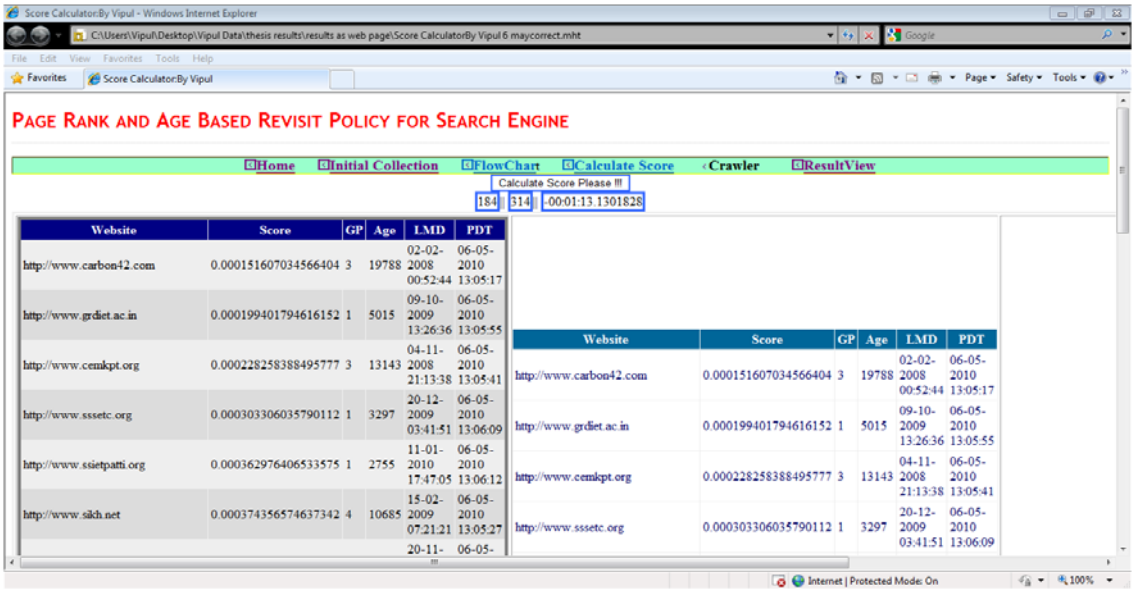

Figure 4.4: Calculation of score using Page Rank & Age.

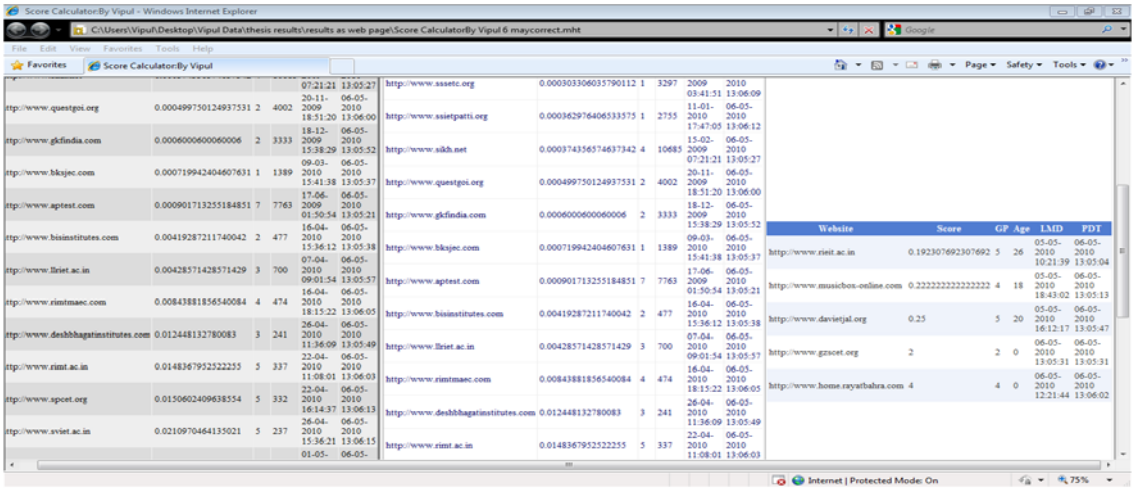

Figure 4.5: Three Collections after calculating score.

V. EXPERIMENTS, RESULTS AND DISCUSSIONS

For the experimentation purpose 26 websites are taken in the initial collection from different sources. Table 5.1 shows the Score, Page Rank, Age and Last modified date of each website taken in the initial collection.

Table 5.1: Initial collection (Sorted according to score value)

Website	Score	GP	Age	LMD	PDT
http://www.carbon42.com	0.00015	3	19670	02-02-2008 00:52:44	01-05-2010 15:10:14
http://www.grdiet.ac.in	0.00020	1	4897	09-10-2009 13:26:36	01-05-2010 15:10:46
http://www.cemkpt.org	0.00023	3	13025	04-11-2008 21:13:38	01-05-2010 15:10:36
http://www.sssetc.org	0.00031	1	3179	20-12-2009 03:41:51	01-05-2010 15:11:45
http://www.sikh.net	0.00038	4	10567	15-02-2009 07:21:21	01-05-2010 15:10:24
http://www.ssietpatti.org	0.00038	1	2637	11-01-2010 17:47:05	01-05-2010 15:11:49
http://www.questgoi.org	0.00051	2	3884	20-11-2009 18:51:20	01-05-2010 15:10:50
http://www.gkfindia.com	0.00062	2	3215	18-12-2009 15:38:29	01-05-2010 15:10:43
http://www.bksjec.com	0.00079	1	1271	09-03-2010 15:41:38	01-05-2010 15:10:31
http://www.aptest.com	0.00092	7	7645	17-06-2009 01:50:54	01-05-2010 15:10:17
http://www.ctgroup.in	0.00380	4	1054	18-03-2010 16:42:46	01-05-2010 15:10:38
http://www.llriet.ac.in	0.00515	3	582	07-04-2010 09:01:54	01-05-2010 15:10:47
http://www.bisinstitutes.com	0.00557	2	359	16-04-2010 15:36:12	01-05-2010 15:10:34
http://www.rimtmaec.com	0.01124	4	356	16-04-2010 18:15:22	01-05-2010 15:10:56
http://www.rimt.ac.in	0.02273	5	220	22-04-2010 11:08:01	01-05-2010 15:10:54

http://www.spcet.org	0.02336	5	214	22-04-2010 16:14:37	01-05-2010 15:11:50
http://www.deshbhagatinstitutes.com	0.02439	3	123	26-04-2010 11:36:09	01-05-2010 15:10:41
http://www.sbscet.ac.in	0.03896	3	77	28-04-2010 09:52:15	01-05-2010 15:11:46
http://www.rieit.ac.in	0.04065	5	123	26-04-2010 11:28:36	01-05-2010 15:10:06
http://www.sviet.ac.in	0.04202	5	119	26-04-2010 15:36:21	01-05-2010 15:11:53
http://www.abilogic.com	0.08772	5	57	29-04-2010 05:30:00	01-05-2010 15:10:09
http://www.davietjal.org	0.10638	5	47	29-04-2010 15:34:13	01-05-2010 15:10:40
http://www.home.rayatbahra.com	0.16000	4	25	30-04-2010 14:04:47	01-05-2010 15:10:52
http://www.musicbox-online.com	0.20000	4	20	30-04-2010 18:11:06	01-05-2010 15:10:12
http://www.creativecommons.org	0.69231	9	13	01-05-2010 01:34:32	01-05-2010 15:10:22
http://www.gzscet.org	2.00000	2	0	01-05-2010 15:02:43	01-05-2010 15:10:27

As shown in Table 5.2, the 20% collection contains the 20% websites from the initial collection that have higher score value means these websites have higher updation rate. These websites have less age and higher page rank value.

Table 5.2: 20% Collection (Collection with high score)

Website	Score	GP	Age	LMD	PDT
http://www.davietjal.org	0.10638	5	47	29-04-2010 15:34:13	01-05-2010 15:10:40
http://www.home.rayatbahra.com	0.16000	4	25	30-04-2010 14:04:47	01-05-2010 15:10:52
http://www.musicbox-online.com	0.20000	4	20	30-04-2010 18:11:06	01-05-2010 15:10:12
http://www.creativecommons.org	0.69231	9	13	01-05-2010 01:34:32	01-05-2010 15:10:22
http://www.gzscet.org	2.00000	2	0	01-05-2010 15:02:43	01-05-2010 15:10:27

All the remaining websites other than the 20% collection from the initial collection are placed in the 80% collection. As shown in the Table 5.3, these websites have less score value as compared to the 20% collection.

Table 5.3: 80% Collection (Remaining Collection)

Website	Score	GP	Age	LMD	PDT
http://www.carbon42.com	0.00015	3	19670	02-02-2008 00:52:44	01-05-2010 15:10:14
http://www.grdiet.ac.in	0.00020	1	4897	09-10-2009 13:26:36	01-05-2010 15:10:46
http://www.cemkpt.org	0.00023	3	13025	04-11-2008 21:13:38	01-05-2010 15:10:36
http://www.sssetc.org	0.00031	1	3179	20-12-2009 03:41:51	01-05-2010 15:11:45
http://www.sikh.net	0.00038	4	10567	15-02-2009 07:21:21	01-05-2010 15:10:24
http://www.ssietpatti.org	0.00038	1	2637	11-01-2010 17:47:05	01-05-2010 15:11:49
http://www.questgoi.org	0.00051	2	3884	20-11-2009 18:51:20	01-05-2010 15:10:50
http://www.gkfindia.com	0.00062	2	3215	18-12-2009 15:38:29	01-05-2010 15:10:43
http://www.bksjec.com	0.00079	1	1271	09-03-2010 15:41:38	01-05-2010 15:10:31
http://www.aptest.com	0.00092	7	7645	17-06-2009 01:50:54	01-05-2010 15:10:17
http://www.ctgroup.in	0.00380	4	1054	18-03-2010 16:42:46	01-05-2010 15:10:38
http://www.llriet.ac.in	0.00515	3	582	07-04-2010 09:01:54	01-05-2010 15:10:47
http://www.bisinstitutes.com	0.00557	2	359	16-04-2010 15:36:12	01-05-2010 15:10:34
http://www.rimtmaec.com	0.01124	4	356	16-04-2010 18:15:22	01-05-2010 15:10:56
http://www.rimt.ac.in	0.02273	5	220	22-04-2010 11:08:01	01-05-2010 15:10:54
http://www.spcet.org	0.02336	5	214	22-04-2010 16:14:37	01-05-2010 15:11:50
http://www.deshbhagatinstitutes.com	0.02439	3	123	26-04-2010 11:36:09	01-05-2010 15:10:41
http://www.sbscet.ac.in	0.03896	3	77	28-04-2010 09:52:15	01-05-2010 15:11:46
http://www.rieit.ac.in	0.04065	5	123	26-04-2010 11:28:36	01-05-2010 15:10:06
http://www.sviet.ac.in	0.04202	5	119	26-04-2010 15:36:21	01-05-2010 15:11:53
http://www.abilogic.com	0.08772	5	57	29-04-2010 05:30:00	01-05-2010 15:10:09

Analysis of 20% Collection

These are the results of 20% collection that are stored from 7 April, 2010 to 10 May, 2010 for the analysis purpose.

7 April					
Website	Score	GP	Age	LMD	PDT
http://www.rieit.ac.in	0.09259	5	54	05-04-2010 14:50:24	07-04-2010 21:33:29
http://www.sbscet.ac.in	0.10345	3	29	06-04-2010 16:13:00	07-04-2010 21:35:56
http://www.llriet.ac.in	0.25000	3	12	07-04-2010 09:01:54	07-04-2010 21:34:16

http://www.creativecommons.org	0.50000	9	18	07-04-2010 03:33:18	07-04-2010 21:33:43
http://www.musicbox-online.com	2.00000	4	2	07-04-2010 18:44:23	07-04-2010 21:33:35
8 April					
Website	Score	GP	Age	LMD	PDT
http://www.abilogic.com	0.07692	3	39	07-04-2010 05:30:00	08-04-2010 21:27:59
http://www.llriet.ac.in	0.08333	3	36	07-04-2010 09:01:54	08-04-2010 21:28:34
http://www.musicbox-online.com	0.15385	4	26	07-04-2010 18:44:23	08-04-2010 21:28:02
http://www.creativecommons.org	0.60000	9	15	08-04-2010 06:22:14	08-04-2010 21:28:10
http://www.sbscet.ac.in	0.75000	3	4	08-04-2010 16:44:46	08-04-2010 21:28:56
12 April					
Website	Score	GP	Age	LMD	PDT
http://www.abilogic.com	0.07895	3	38	11-04-2010 05:30:00	12-04-2010 20:20:29
http://www.davietjal.org	0.08929	5	56	10-04-2010 11:49:23	12-04-2010 20:22:20
http://www.creativecommons.org	0.13235	9	68	10-04-2010 00:05:12	12-04-2010 20:21:01
http://www.sbscet.ac.in	0.50000	3	6	12-04-2010 13:23:40	12-04-2010 20:23:06
http://www.musicbox-online.com	2.00000	4	2	12-04-2010 17:59:26	12-04-2010 20:20:38
15 April					
Website	Score	GP	Age	LMD	PDT
http://www.davietjal.org	0.03846	5	130	10-04-2010 11:49:23	15-04-2010 22:00:24
http://www.abilogic.com	0.04688	3	64	13-04-2010 05:30:00	15-04-2010 21:59:48
http://www.creativecommons.org	0.14286	9	63	13-04-2010 06:11:24	15-04-2010 22:00:03
http://www.musicbox-online.com	0.14815	4	27	14-04-2010 18:15:53	15-04-2010 21:59:53
http://www.sbscet.ac.in	0.37500	3	8	15-04-2010 13:25:37	15-04-2010 22:00:53
19 April					
Website	Score	GP	Age	LMD	PDT
http://www.home.rayatbahra.com	0.11111	4	36	18-04-2010 09:11:33	19-04-2010 21:30:27
http://www.sbscet.ac.in	0.12500	3	24		19-04-2010 21:32:43
http://www.creativecommons.org	0.13235	9	68	17-04-2010 01:06:53	19-04-2010 21:29:35
http://www.sviet.ac.in	0.62500	5	8	19-04-2010 13:29:05	19-04-2010 21:32:50
http://www.musicbox-online.com	1.33333	4	3	19-04-2010 18:01:38	19-04-2010 21:29:09
20 April (Morning)					
Website	Score	GP	Age	LMD	PDT
http://www.creativecommons.org	0.11392	9	79	17-04-2010 01:06:53	20-04-2010 08:39:09
http://www.cemkpt.org	0.12500	3	24	04-11-2008 21:13:38	20-04-2010 08:39:37
http://www.spcet.org	0.20833	5	24	19-11-2009 12:58:13	20-04-2010 08:42:19
http://www.sviet.ac.in	0.26316	5	19	19-04-2010 13:29:05	20-04-2010 08:42:22
http://www.musicbox-online.com	0.50000	4	8	20-04-2010 00:25:07	20-04-2010 08:38:50

20 April (Night)					
Website	Score	GP	Age	LMD	PDT
http://www.sbscet.ac.in	0.37500	3	8	20-04-2010 12:39:39	20-04-2010 21:29:38
http://www.rieit.ac.in	0.45455	5	11	20-04-2010 09:58:52	20-04-2010 21:27:55
http://www.spcet.org	0.55556	5	9	20-04-2010 12:04:40	20-04-2010 21:29:42
http://www.sviet.ac.in	0.83333	5	6	20-04-2010 14:54:44	20-04-2010 21:29:45
http://www.musicbox-online.com	1.33333	4	3	20-04-2010 18:24:17	20-04-2010 21:28:03
21 April (Morning)					
Website	Score	GP	Age	LMD	PDT
http://www.sbscet.ac.in	0.15789	3	19	20-04-2010 12:39:39	21-04-2010 07:53:24
http://www.rieit.ac.in	0.23810	5	21	20-04-2010 09:58:52	21-04-2010 07:52:25
http://www.spcet.org	0.26316	5	19	20-04-2010 12:04:40	21-04-2010 07:53:28
http://www.musicbox-online.com	0.30769	4	13	20-04-2010 18:24:17	21-04-2010 07:52:31
http://www.sviet.ac.in	0.31250	5	16	20-04-2010 14:54:44	21-04-2010 07:53:30
21 April (Evening)					
Website	Score	GP	Age	LMD	PDT
http://www.sbscet.ac.in	0.11111	3	27	20-04-2010 12:39:39	21-04-2010 15:48:06
http://www.rieit.ac.in	0.17241	5	29	20-04-2010 09:58:52	21-04-2010 15:44:13
http://www.musicbox-online.com	0.19048	4	21	20-04-2010 18:24:17	21-04-2010 15:44:31
http://www.sviet.ac.in	0.20833	5	24	20-04-2010 14:54:44	21-04-2010 15:48:24

http://www.spcet.org	5.00000	5	1	21-04-2010 14:00:04	21-04-2010 15:48:19
23 April (Night)					
Website	Score	GP	Age	LMD	PDT
http://www.spcet.org	0.17857	5	28	22-04-2010 16:14:37	23-04-2010 21:12:11
http://www.creativecommons.org	0.40909	9	22	22-04-2010 22:41:33	23-04-2010 21:10:23
http://www.sbscet.ac.in	0.42857	3	7	23-04-2010 13:51:31	23-04-2010 21:12:08
http://www.sviet.ac.in	1.25000	5	4	23-04-2010 16:16:15	23-04-2010 21:12:13
http://www.musicbox-online.com	2.00000	4	2	23-04-2010 18:27:20	23-04-2010 21:10:17
27 April (Night)					
Website	Score	GP	Age	LMD	PDT
http://www.home.rayatbahra.com	0.12500	4	32	26-04-2010 12:37:42	27-04-2010 20:49:11
http://www.rieit.ac.in	0.15152	5	33	26-04-2010 11:28:36	27-04-2010 20:48:37
http://www.musicbox-online.com	0.15385	4	26	26-04-2010 18:11:08	27-04-2010 20:48:41
http://www.sviet.ac.in	0.17241	5	29	26-04-2010 15:36:21	27-04-2010 20:49:23
http://www.sbscet.ac.in	1.00000	3	3	27-04-2010 16:54:02	27-04-2010 20:49:15
1May					
Website	Score	GP	Age	LMD	PDT
http://www.davietjal.org	0.10638	5	47	29-04-2010 15:34:13	01-05-2010 15:10:40
http://www.home.rayatbahra.com	0.16000	4	25	30-04-2010 14:04:47	01-05-2010 15:10:52
http://www.musicbox-online.com	0.20000	4	20	30-04-2010 18:11:06	01-05-2010 15:10:12
http://www.creativecommons.org	0.69231	9	13	01-05-2010 01:34:32	01-05-2010 15:10:22
http://www.gzscet.org	2.00000	2	0	01-05-2010 15:02:43	01-05-2010 15:10:27
10 May					
Website	Score	GP	Age	LMD	PDT
http://www.musicbox-online.com	0.11111	4	36	08-05-2010 21:53:11	10-05-2010 10:34:06
http://www.creativecommons.org	0.16071	9	56	08-05-2010 02:02:34	10-05-2010 10:34:34
http://www.abilogic.com	0.17241	5	29	09-05-2010 05:30:00	10-05-2010 10:33:54
http://www.rieit.ac.in	0.20833	5	24	05-05-2010 10:21:39	10-05-2010 10:33:50
http://www.home.rayatbahra.com	4.00000	4	0	10-05-2010 10:28:30	10-05-2010 10:35:44

From the analysis conducted for a period of 1 month it is concluded that the 20% of websites in initial collection have age less than 2 or 3 days i.e. these websites have high updation rate, so crawler should revisit these websites with high priority than the other websites. Earlier crawler revisits all the websites with same frequency regardless of how often they change. In fact, a large number of websites changes very slowly. So, these websites should be given less priority in terms of re-crawling. But with this hybrid approach, only the websites that have less age and higher page rank value are given more importance and crawler revisits these websites with higher priority. As the crawler has to maintain the fresh copy of web page in the local collection, with this hybrid approach the crawler always maintain the fresh copy of web page in local collection. Thus freshness of local collection is always maintained.

The 30 websites stored in the initial collection are analyzed for the results purpose. The Pareto's principle 80:20 rule is applied and verified on web corpus and two collections were created from the initial collection. One is named as "20% collection" that contains 20% web pages from the initial collection that have high Score value means these web pages have less age and high Page Rank value & other is named as "80% Collection" which contains remaining web pages from the initial collection. It is also found that detecting changes using HTTP Meta data can successfully reduce network traffic.

A. Graph for Initial Collection

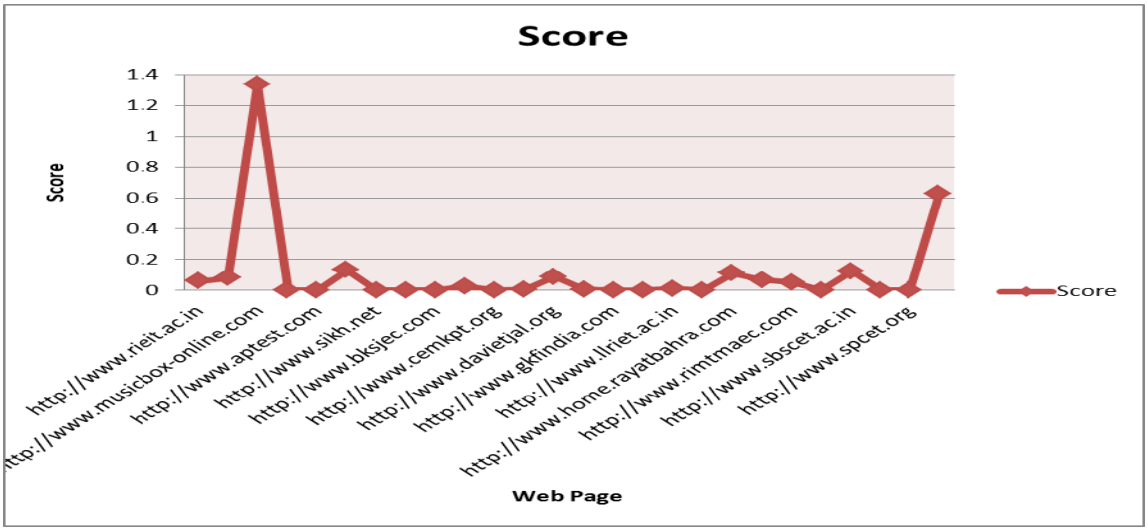

Figure 5.1: Graph for initial collection

Figure 8.1 shows the graph for initial collection. The score of websites in the initial collection is calculated in the same order, as they are indexed in the database. As it is clearly shown in the graph, that the score of initial collection increases or decreases for websites. The crawler revisits all the pages with same frequency regardless of how often they change. In fact, a large number of pages changes very slowly.

Note: Graphs are based on the results of day 19 April.

B. Graph for 20% Collection

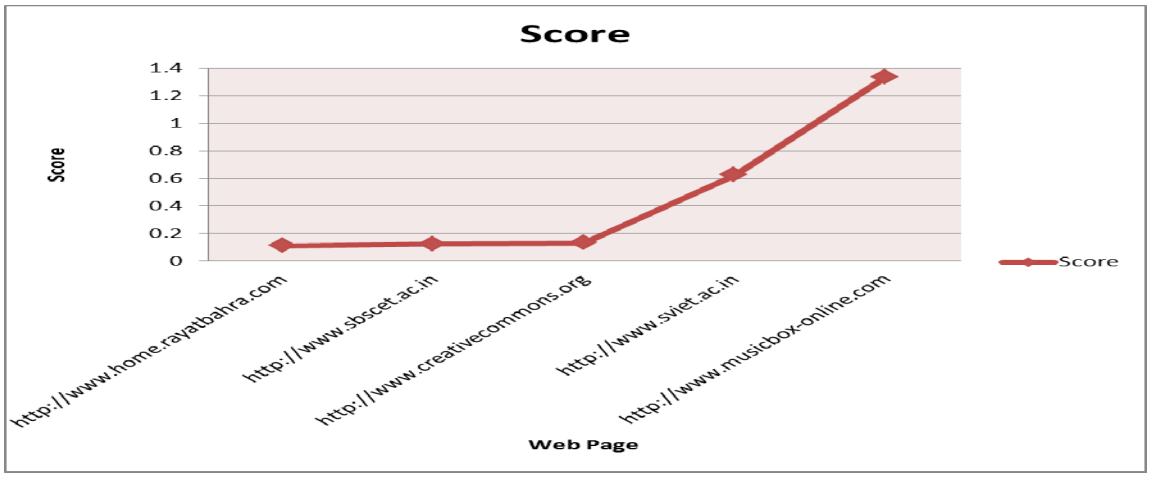

Figure 5.2: Graph for 20% collection

It is clearly shown in the graph that the score of 20% collection is increasing. Website http://www.musicboxonline.com has higher score means the age of this website is less. So, crawler refreshes this website first. Then it refreshes the website http://www.sviet.ac.in and so on. Thus only the websites that have higher score i.e. less age are revisited first as compared to other websites.

C. Graph for 80% Collection

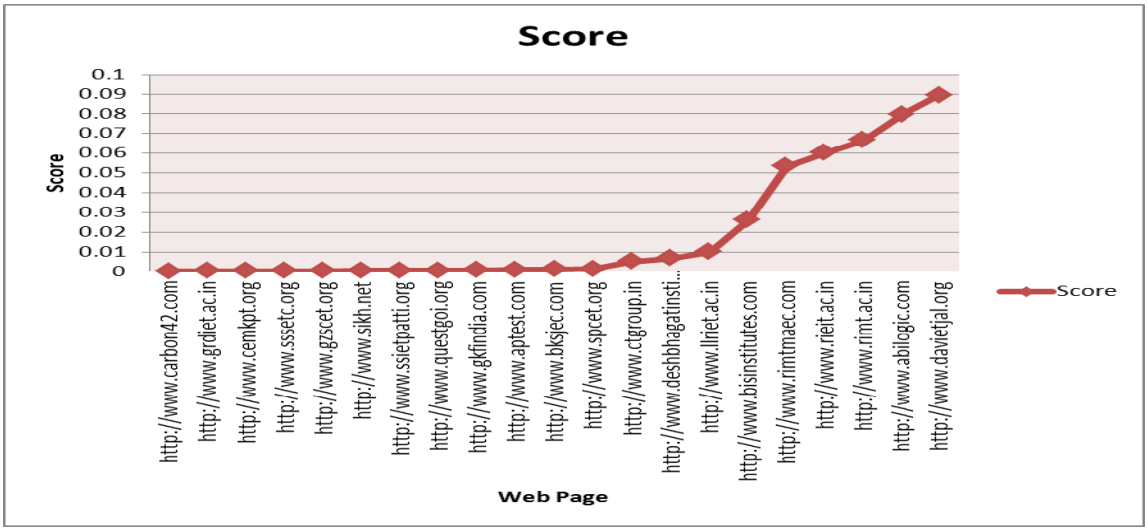

Figure 5.3: Graph for 80% collection

These are the remaining websites other than the 20% collection. These websites have score less than the websites in the 20% collection. As shown in graph, the score of 80% collection is also increasing. As the number of websites in the 80% collection are more so the rate at which crawler crawls the 80% collection is less than the rate at which crawler crawls the 20% collection. It is clearly shown in the graph that http://www.davietjal.org has high score, so crawler crawl this web page first.

D. Comparison of three scores

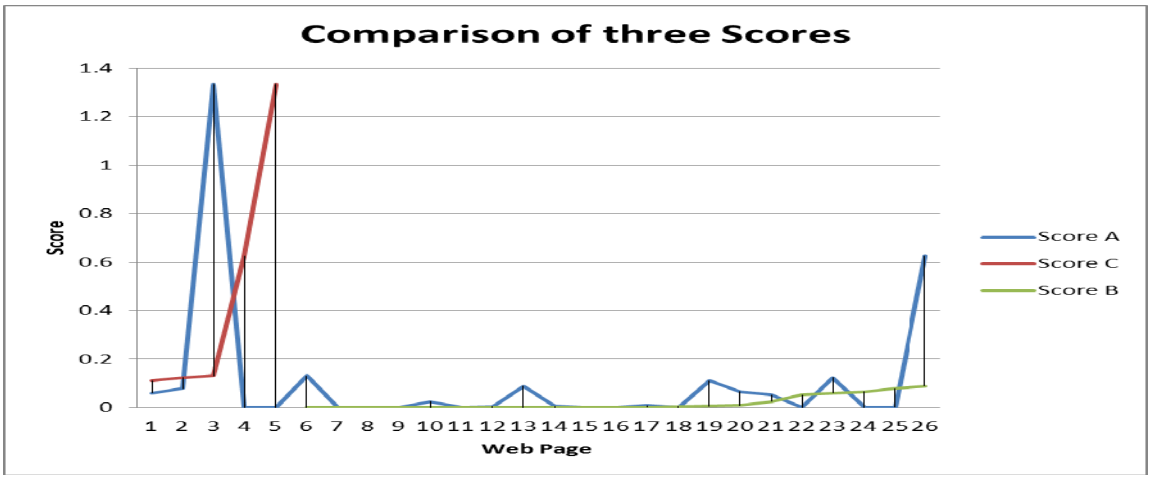

Figure 5.4: Graph of comparison between three scores

Score A defines initial collection; Score B defines 80% collection; Score C defines 20% collection

In the graph, the score of initial collection (Score A), increases or decreases for websites. The crawler revisits all the websites with same frequency regardless of how often they change. Thus providing users unsatisfactory results.

The score of 20% collection (Score C) is increasing, shows that the crawler will revisit a web page with higher score first. As the number of websites in the 20% collection is less, so the re-crawl rate is high in this case.

The score of 80% collection (Score B) is also increasing but the re-crawl rate is less in this case as the number of websites are more in this case.

VI. Conclusion & Future Scope

From literature review, we study how a crawler can effectively refresh downloaded pages to maximize their "freshness". Most of the papers include the approaches that involve user participation in larger extent in order to get the fresh results. We have developed a hybrid approach on the basis of which a web crawler maintains the retrieved pages "fresh" in the local collection. Towards this goal, we use the concept of Page rank and Age of a web page, using these two parameters we propose a simple approach that can identify important pages (i.e. Pages with less age and higher page rank), and the crawler re-visit these important pages with higher priority. We applied and verified the Pareto's principle 80:20 rule on our web corpus. The conclusion that has been derived through the experiments conducted on our

approach is that only the webpages that have less age are in the 20% collection. The average age of web pages in 20% Collection is less than 2 or 3 days. Cho [15] proposed the concept of page quality, which is closely, related to the popularity metrics of web pages. In our prospect ion, if such quality metric can be used to evaluate the updates of web pages, then people may be able to develop solutions to improve the quality, as well as freshness, of web index for retrieval. We leave it as a good future direction that how web change detection on the basis of change frequency, quality, popularity, etc. can be used in a unified framework for web index synchronization problem.

REFERENCES

[1] Christopher D.Manning and Prabhakar Raghavan. An introduction to Information Retrieval. Preliminary draft© 2008 Cambridge UP.

[2] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through url ordering. In *Proceedings of 7th World Wide Web Conference (WWW7)*, 1998

[3] Sergey Brin and Lawrence Page. "The anatomy of a large scale hypertextual web search engine". *In proceedings of the seventh international world wide web conference, Bristbane, Australia, April 1998.*

[4] J. Cho and H. Garcia-Molina. Synchronizing a database to improve freshness. In *Proceedings of 2000 ACM International Conference on Management of Data (SIGMOD)*, 1999.

[5] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *Proceedings of 26th International Conference on Very Large Databases (VLDB)*, pages 136 – 147, 2000.

[6] Junghoo Cho and Hector Garcia Molina. "Effective page refresh policies for web crawler" . *ACM Transactions on Database Systems, December 2003*.

[7] Hadrien Bullot and S K Gupta. A Data-Mining Approach for Optimizing Performance of an Incremental Crawler. *Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03)*

[8] Christopher Olston and Sandeep Pandey. User centric Web crawling. *In Proceedings of WWW'05, pages 401–411, New York, NY, USA,2005. ACM Press.*

[9] Wen-Kun Mie, LU Zeng Dhing. "A cooperative schema between Web sever and search engine for improving freshness of Web repository". *Wuhan University Journal of natural sciences, Vol. 11, No.1, 2006.*

[10] Divakar Yadav, J.P.Gupta. "Change Detection in Web Pages*". In proceedings of 10th International Conference on Information Technology, 2007.*

[11] Rahul choudhari and Ajay choudhari. "Increasing Search Engine Efficiency using Cooperative Web". *In proceedings of International Conference on Computer Science and Software Engineering*,2008.

[12] A.K. Sharma and Ashutosh Dixit. Self adjusting Refresh Time based Architecture for incremental web crawler. *IJCSNS International Journal of Computer Science and network security*, Vol.8 No.12 ,December2008.

[13] K.S. Kuppusamy and G. Aghila. "FEAST - A Multistep, Feedback Centric, Freshness Oriented Search Engine". *In proceedings of 2009 IEEE International Advance Computing Conference (IACC 2009).*

[14] Ravita Chahar , Komal Hooda, and Annu Dhankahar . "Management Of Volatile Information In Incremental Web Crawler". *IJCSI International Journal of Computer Science Issues, Vol. 4, No. 1, 2009.*

[15] Junghoo Cho. "Crawling the web: Discovery and Maintenance of Large Scale Web Data". A Thesis Nov 2001.

[16] Scott J. Simon. "Network Theory: 80/20 Rule and Small Worlds Theory".

[17] W3 Header Field Definitions.http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html

[18] List of HTTP Headers. en.wikipedia.org/wiki/List_of_HTTP_headers

[19] Robots exclusion. http://info.webcrawler.com/mak/projects/robotsexclusion.html

[20] Carlos Castillo, "Effective Web Crawling", PhD. thesis University of Chile November 2004.

[21] B. E. Brewington and G. Cybenko "How dynamic is the web?" In Proceedings of 9th World Wide Web Conference (WWW9), January 2000.