International Journal of Engineering Research and General Science Volume 3, Issue 3, Part-2 , May-June, 2015
ISSN 2091-2730

A Novel Approach For Web Pre-fetching and caching

Varun Kumar[1] Ms.Nidhi Seth[2]
[1]Research Scholar [2]Assistant Professor
[1 2] Department of Computer Science & Engineering [1,2]JMIT, Radaur, Haryana, India
varunsaini727@gmail.com er.nidhi25@gmail.com

Abstract- Due to the fast development of internet services and a huge amount of network traffic web caching and prefetching are the most popular techniques that play a key role in improving the Web performance by keeping web url that are likely to be visited in the near future closer to the client. Web caching technique work integrated or independently with the web prefetching. The Web caching and prefetching techniques are complement each other since the web caching exploits the temporal locality for predicting revisiting requested url, while the web prefetching utilizes the spatial locality for predicting next related web object of the requested Web url. In this paper proposed work represent the working of a novel approach for web caching and prefetching.This technique enhance the performance with help of using user priority approach. In this paper explain how the response time of hit taken from the user cache is less as compare to the data taken directly from the log file.In this approach cache size and which replacement policy is used for replacement in cache play an important role .For this in this paper explain the comparision of three replacement technique – LRU,FIFO,LFU on basis of hit rate on cache .For improving the performance and response time in web caching and prefetching technique use the best replacement policy after comparision .In this paper also explain enhance the performance of caching and prefetching using user based approach and analyze the prefetching hit ratio b/w priority and user based approach in which clustering used.

Keywords— Web Caching, Web pre-fetching, Response Time, Proxy server, replacement policy, Hit,clustering

Introduction

Web caching is a well-known strategy for improving the performance of Web- based system by keeping Web objects that are likely to be used in the near future in location close to user. The Web caching mechanisms are applied at three levels: client level, proxy level and original server level[5,6] . Significantly, proxy servers play the key roles between users and web sites in lessening of the response time of user requests and saving of network bandwidth.Thus, for achieving better response time, an efficient caching technique can be built in a proxy server

Web caching is a well-known strategy for improving the performance of Web- based system by keeping Web objects that are likely to be used in the near future in location close to user. The Web caching mechanisms are applied at three levels: client level, proxy level and original server level[5,6] . Significantly, proxy servers play the key roles between users and web sites in lessening of the response time of user requests and saving of network bandwidth.Thus, for achieving better response time, an efficient caching technique can be built in a proxy server

The cache replacement is the core or heart of the web caching; consquently, the design of efficient cache replacement algorithms is crucial for caching mechanism achievement. so, cache replacement algorithms are also called web caching algorithms[7]. Because of limited space of cache, an intutive mechanism is required to manage the Web cache content properly. The conventional caching policies are not efficient in the Web caching since they consider just one factor and ignore other factors that have impact on the efficiency of the Web caching . In these caching policies, most popular objects get the most requests, while a large segment of objects, which are stored in the cache, are never requested again. This is called cache pollution problem. Therefore, many Web cache replacement policies have been proposed attempting to get good performance. Hence, the difficulty in determining which ideal web objects will re-accessed is still a big challenge faced by the existing Web caching techniques. In other words, what Web objects should be cached and what Web objects should be replaced to make the best use of available cache space, better hit rates, decrease network traffic, and reduce loads on the original server[3,4] .

Unfortunately, the cache hit ratio is not improved much with caching schemes. despite with a cache of infinite size, the hit ratio are still limited only at the range from 40% to about 50%, regardless of the caching scheme [8,9,10]. This is because most people browse and explore the new web pages trying to find new information. In order to improve the hit ratio of cache, Web pre-fetching technique is integrated with web caching to overcome these limitations

Web prefetching is fetching web pages in advance by proxy server/client before a request is send by a client/proxy server. The major advantage of using web prefetching is reduced latency. When a client makes a request for web object, rather than sending request to the web server, it may be fetched from a pre-fetch area. The main factor for selecting a web prefetching algorithm is that its ability to predict the web object to be pre-fetched in order to reduce latency. Web prefetching exploits the spatial locality of web pages, ie. pages that are linked with current page will be accessed with higher probability than other pages. Web prefetching can be implemented in a web environment as between clients and web server, between proxy server and web server and between clients and proxy server [11]. If it is implemented between web server and client , it is helpful in decreasing user perceived latency, but the problem is that it will increases network traffic. If it is implemented between web server and proxy server, can reduce the bandwidth usage by prefetching only a specific number of hyper links. If it is implemented between clients and proxy server, the proxy starts feeds pre-fetched web objects from its cache to the clients so there won't be extra internet traffic.

Web Pre-Fetching Techniques

(i) Domain Top

In Domain Top approach for web prefetching, combination of knowledge of most popular domains andmost popular documents is done by proxy server. In this approach proxy is responsible for calculating the most popular domains and most popular documents in those domains, and then prepares a rank list for prefetching.

(ii) Top 10 Approach

Evangelos P. Markatos et al. proposes a top 10 approach to prefetching on the web, in which the server calculates the list of most popular documents. This approach is easy to implement in client server architecture. It considers frequency of access for predicting the web object, not the client characteristics on the web.

(iii) A Keyword based semantic prefetching approach in internet news services

This proposes a key word based semantic pre-fetching, in which prediction of future requests are based on semantic preferences of past retrieved web documents. This technique is applied to internet news service; it finds out semantic preferences by analyzing keywords in URL anchor text of previously accessed documents in different news categories.The semantic representation is represented in an open self learning capable model which collects the knowledge about the client preferences. Client future request predictions are based on this knowledge. The pre-fetched documents are stored in internal cache. When client makes a request the web document is fetched from cache if it is available otherwise it fetches self learning.

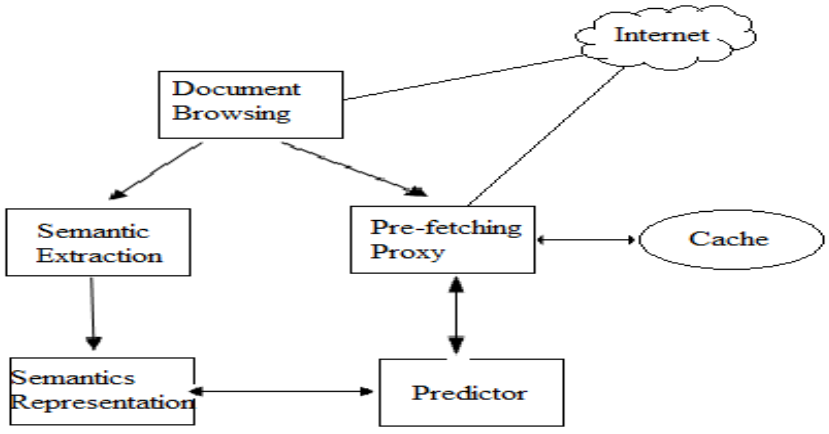

Fig (a) semantic prefetching approach

(iv) Dynamic web prefetching

In dynamic web pre-fetching technique [12], each user can keep a list of sites to access immediately called user's preference list. The preference list is stored in proxy server's database. Intelligent agents are used for parsing the web page, monitoring the bandwidth usage and maintaining hash table, preference list and cache consistency. It controls the

web traffic by reducing pre-fetching at heavy traffic and increasing pre-fetching at light traffic. Thus it reduces the idle time of the existing network and makes the traffic almost constant. A hash table is maintained for storing theist of accessed URLs and itsweight information [12], [13]. Depending upon the bandwidth usage and weights in the hash table, the prediction engine decides the number of URLs to be pre-fetched and gives the list to pre-fetch engine for pre-fetching the predicted web pages. After prefetching, the proxy server keeps the pre-fetched web pages in a separate area called pre-fetch area.

(v) Link Pre-fetching

A web page provides a set of pre-fetching hints to the browser and after the browserfinishes loading the page, it starts pre-fetching specified documents and stores them in its cache. When the user visits one of the pre-fetched documents, it can be served up quickly out of thebrowser's cache. Fisher et. Al proposed a server drive approach for ink prefetching [14].In thisapproach browser follows special directives from the web server or proxy server that instructs itto pre-fetch specific documents. This mechanism allows servers to control the contents to be perfectedby the browser. The browser looks for either HTML <link> tag or an HTTP Link: headerTag to pre-fetch the subsequent links. The Link: header can also be specified within the HTMLdocument itself by using a HTML <meta>tag[16].When the browser is idle, it observes thesehints and queues up each unique request to be pre-fetched.

(vi) Adaptive pre-fetching Scheme

Adaptive pre-fetch scheme are developed to adapt user's browsing history and habits [15]. Jiangand et al. proposed an adaptive pre-fetch scheme, in which the number of files to be pre-fetcheddepends on user access history and network conditions . This scheme consists of two modules: prediction module and threshold module. The prediction module updates the history and computesthe access probability of each file. Files whose access probabilities greater than or equal to thepre-fetch threshold are onlypre-fetched. Chen and et. al[7] proposed an adaptive pre- fetchscheme, in which dynamically adjust the pre¬fetch aggressiveness in web servers and uses athreshold to adjust the aggressiveness of pre¬fetching. Fagni and et. al[proposed usesanapproach for boosting the performance of search engine by exploiting the spatial and temporallocality present in the stream of processed queries.They do not consider real semantics of document, however. As semantic prefetching we understand prefetching based on preferences of past retrieved documents in semantics, rather than on the chronological relationships between URL accesses. Semantically based prefetching tries to extract a semantic description of a document and asks server to provide pages with similar semantics, with the same so called "semantic locality". Based on the document semantics, this approach is capable of prefetching documents whose URLs have neverbeen accessed

Related work

(i) A Survey of Web Caching and Prefetching"(Waleed Ali , Siti Mariyam Shamsuddin, and Abdul Samad Ismail)(2011)
Web caching and prefetching are the most popular techniques that play a key role in improving the Web performance by keeping web objects that are likely to be visited in the near future closer to the client. Web caching can work independently or integrated with the web prefetching. The Web caching and prefetching can complement each other since the web caching exploits the temporal locality for predicting revisiting requested objects, while the web prefetching utilizes the spatial locality for predicting next related web objects of the requested Web objects. This paper reviews principles and some existing web caching and prefetching approaches. The conventional and intelligent web caching techniques are investigated and discussed. Moreover, Web prefetching techniques are summarized and classified with comparison limitations of these approaches. This paper also presents and discusses some studies that take into consideration impact of integrating both web caching and web prefetching together.

(ii) A Survey On Web Pre-Fetching and Web Caching Techniques in a Mobile Environment"(Greeshma G. Vijayan1 and Jayasudha J. S.) (2012)
As the Internet continues to grow in size and popularity, web traffic and network bottlenecks are major issues in the network world. The continued increase in demand for objects on the Internet causes severe overloading in many sites and network links. Many users have no patience in waiting more than few seconds for downloading a web page. Web traffic reduction techniques are necessary for accessing the web sites efficiently with the facility of existing network. Web pre-fetching techniques and web caching reduces the web latency that we face on the internet today. This paper describes about the various prefetching and caching techniques, how they

predict the web object to be pre-fetched and what are the issues challenges involved when these techniques are applied to a mobile environment

(iii) Survey on Improving the Performance of Web by Evaluation of Web Prefetching and Caching Algorithms" (Arun Pasrija) (2013)

Web caching and prefetching have been studied in the past separately. In this paper, present an integrated architecture for Web object caching and prefetching. Our goal is to design a prefetching system that can work with an existing Web caching system in a seamless manner. In this integrated architecture, a certain amount of caching space is reserved for prefetching. To empower the prefetching engine, a Web-object prediction model is built by mining the frequent paths from past Web log data.We show that the integrated architecture improves the performance over Web caching alone, and present our analysis on the tradeoff between the reduced latency and the potential increase in network load.

(iv) Survey of Recent Web Prefetching Techniques" (Sonia Setia, Dr. Jyoti, Dr. Neelam Duhan) (2013)

Web caching and web prefetching are the two major areas of research focused at reducing the user perceived latency. Both if used well can greatly help in reducing this latency as web caching helps in exploiting temporal latency while web prefetching helps in exploiting spatial latency. However if prefetched pages are not visited by the users in their future accesses, they can increase the network traffic and overload the web server. This paper aims at surveying various research papers who have worked in this direction.

(v) Study of Web Pre-Fetching With Web Caching Based On Machine Learning Technique " (K R Baskaran, Dr. C.Kalarasan, A Sasi Nachimuthu) (2013)

High bandwidth utilization, reduced load on the origin server, high access speed are possible by combining Web caching and pre-fetching techniques. Pre-fetching is the process of fetching few Web pages in advance which will be assumed to be needed by the user in near future and those pages are cached in the memory. Lots of work has been reported for caching and pre-fetching of Web pages in the literature. In this paper, pre-fetching using clustering technique is combined with SVM (Support Vector Machine) - LFU algorithm, a machine learning technique for Web proxy caching .By using real dataset it will be shown that the SVM technique will be better than clustering based prefetching technique using caching policy like LFU considering bandwidth utilization and access latency

(vi) Hybrid Approach for Performance of Web Page Response through Web Usage Mining ” (Ravinder Singh,Bhumika garg) (2014)

In this paper present the web caching and prefetching together using Dynamic technique into Domain Top approach. . Optimized top domain approach will consist of preference list along with the rank list. In this approach proxy is responsible for calculating the most popular domains and most popular documents in those domains, and then prepares a rank list for pre-fetching. In Dynamic web pre-fetching technique, each user can keep a list of sites to access immediately called user's preference list.

PROBLEM DEFINITION

Caching is an significant technique for enhancing the performance of web based applications with help of web caching techniques. Web caching provides great features like traffic reduction, less load on servers, user-end retrieval delays by replicating popular content on proxy caches that are strategically placed within the network. Web pre-fetching schemes have also been widely discussed where web pages and web objects are pre-fetched into the proxy server cache. In our research we will work on integration of web caching and web pre-fetching approach to improve the performance of proxy server’s cache. In Domain Top approach for web pre-fetching, combination of knowledge of most popular domains and most popular documents is done by proxy server. In Dynamic web pre-fetching technique, each user can keep a list of sites to access immediately called user’s preference list. In recent research concept of preference list from Dynamic technique into Domain Top approach is used but there is no graphical representation about performance and there is not mentioned about which replacement policy used in this technique. The main focus of the research is to improve accuracy in caching and prefetching technique. Our research is started with information fetching of pre-fetching and caching techniques. The major targets and objectives for our research is given as below:

- Develop an significant technique for optimizing the web caching and web pre-fetching processes with user based new technique in which we use clustering .
- Enhance the performance of existing technique using priority concept and using the best replacement policy.
- .Analyze FIFO,LRU,LFU replacement policy on the basis of hit rate on cache using various size of cache.
- .Analyze the existing prefetching approach and user based prefetching technique on the basis of hit rate . .
- For find loopholes and issues in new approach and to highlight the benefits for new approach.

Proposed Work

(i). WEB LOG FILE

A **log file** is a recording of everything that goes in and out of a particular server.Web user visits many web sites time to time and spent random quantity of time among various visits.To deal with the user browsing behavior, we should analyze the proxy server log file. In fussy,

the web proxy access log is an in order file with one user access data per line. Web proxy log files make available information about actions performed by a user from the moment the user logs.

(ii). Extraction & preprocessing

In extraction phase web log file is extacted which maintain the all record of the users. In our proposed work during preprocessing phase we carried out the cleaning task to filter out all the unwanted entries from the proxy log data.suppose in log file it contain the link http://www.sscnwr.org/notice_down_file/Advt.%20No.%20ER2015%20English%20Version-FINAL%20FOR%20ADVT.pdf.
So after the preprocessing we get only " www.sscnwr.org"

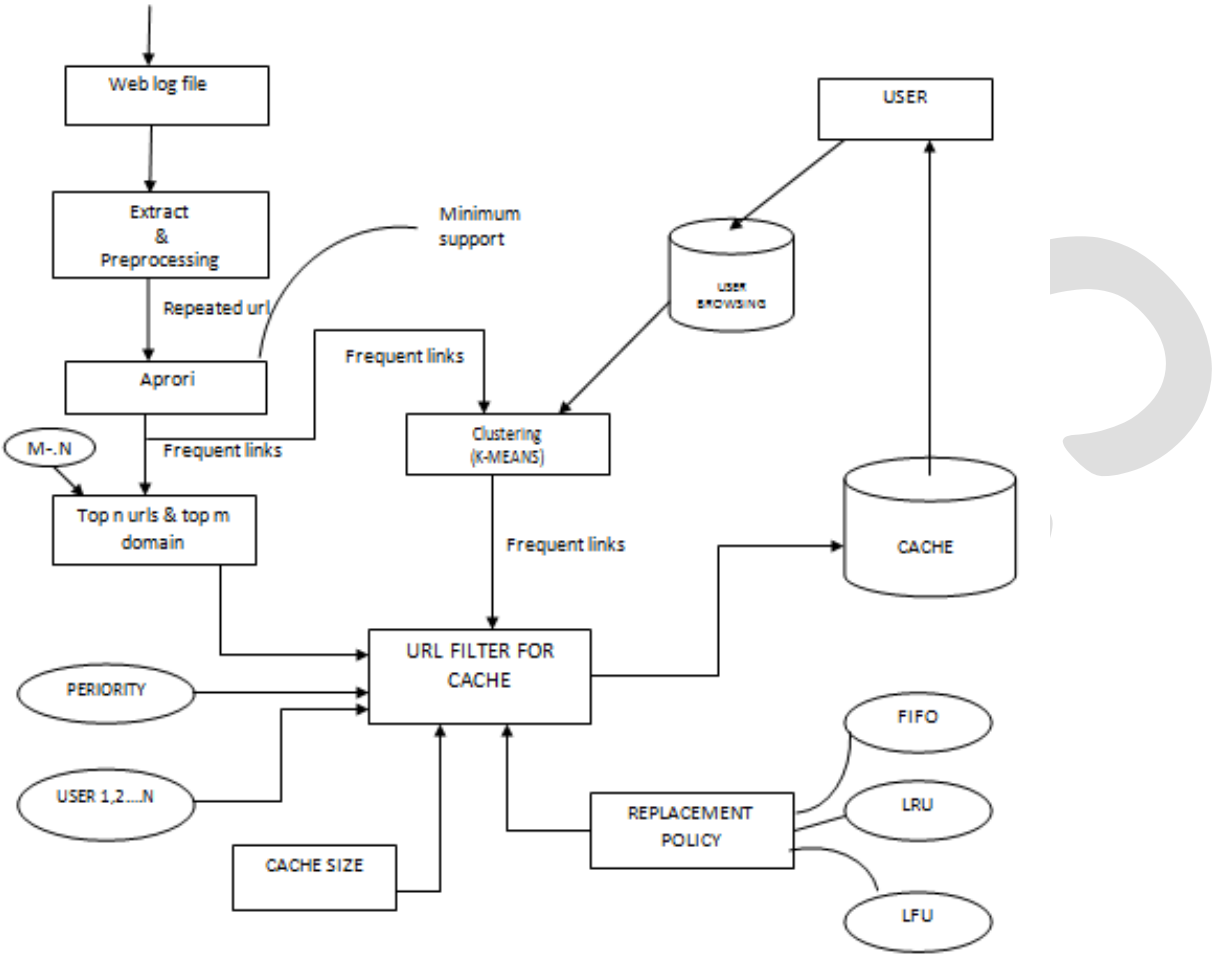

Fig.(b) Proposed work flow chart

(iii). Aprori After the preprocessing step we get all type of domain which are accessed by user.After this we use Aprori for getting unique url and their frequency how many time they used by user.with help of this we get the frequent links In this algorithm we will use the minimum support.minimum support is the confidence level which we decide p% and the term that appears less than p% will be remove and the more combination is applied to take the proper frequent set of the given data.after this we get top m url of top n domain.

(iv). Clustering:

To cluster users we use the KMeans clustering which is used to gather different users into clusters on the basis of their usage behavior and searching pattern. The *K-Means* is the simplest clustering algorithm widely used for web proxy server .The algorithm is used to cluster users data based onattributes into *K* clusters. Each cluster has its center (known as centroid) at point *Cj*. The centroid is calculated from mean distance of all records in the cluster [26]. In this study, we make an assumption that users in the same cluster should have same surfing habits and patterns. Users surfing habits can be determined by several factors such as the time of day of their access, and their most frequently visited websites. In our proposed work we apply this on user browsing history and frequent link which we get after aprori .

(v). URL Filter

url filter work for cache.it's working is that how the url arrange or which url contain by the cache suppose cache size is 20 and our url is 50 then its work is that after filtering we get those url for cache which are more relevant and used in future by user .For filtering we used priority concept,diff-2 replacement policy,user based approach with help of clustering.

Replacement policy The cache replacement is the core or heart of the web caching; consquently, the design of efficient cache replacement algorithms is crucial for caching mechanism achievement. so, cache replacement algorithms are also called web caching algorithm.there are following replacement policy is used in our proposed work and there comparision

a. LFU-least frequent used
b. LRU-least recently used
c. FIFO-first in first out

Priority concept without priority most frequent url is displayed on cache if we use the priority concept in this the one of the users is having highest priority other will have less priority.suppose there are n user and suppose One of the user has highest priority than other.suppose cache size is 20 and in this already 19 url placed if all the user browse the diff url at same time then url which have highest priority is added on cache.

User base concept

in this url displayed on cache on the basis of user . with this concept url in cache accrding to the user.in this approach we use the clustering on user browsing history and frequent link which we get after aprori.in this url in cache according to the user .in our proposed work there is 1,2,3……10 user .with this concept we see the top 5,10,15,20 frequent link on cache according to user1,2……..10.in this if we want to cache maintain top 15 url of user 5 then it show the top 15 url of user 5.it increase the hit ratio . if we want to cache maintain top 5 url of user 3 then it show the top 5 url of user 3.this is user based approach .it maintain the cache according to user interest.

RESULT ANALYSIS

In this section, result analysis has discussed on the basis of experimental work in which we have tested .with help of our experimental work we draw two graph

1. In our work we used three replacement policy for cache i.e FIFO ,LRU,LFU and compare each of them on the basis of hit rate at different -2 cache size.

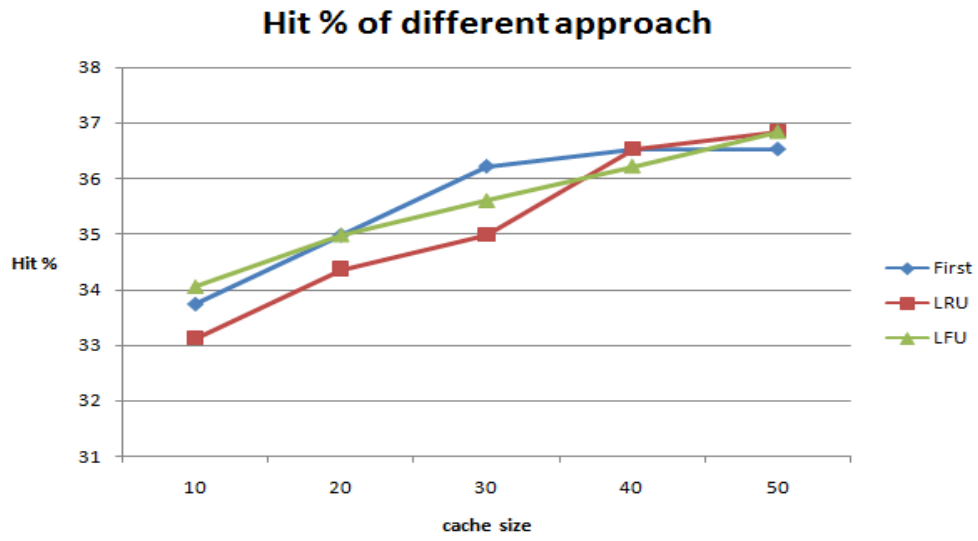

Fig c Hit ratio comparison between FIFO,LRU & LFU with prefetching

Fig. shows effect of hit ratio on different values of cache size used in all the cases FIFO,LRU and LFU. In all the cases, as the cache size increases, the hit % also increases accordingly. But from the Fig. it is clear that when the size of cache is small than LFU approach works better than LRU & FIFO.But if the cache size is large then LRU approach works better than LRU & FIFO.After a certain size of cache the result is same for FIFO.so it show if we use small size cache then we should use LFU or if we use large size cache then we use LRU replacement policy.

2) with help of our experimental work we compare the existing system and our proposed sysem on the basisiof hit ratio at different -2 size cache.

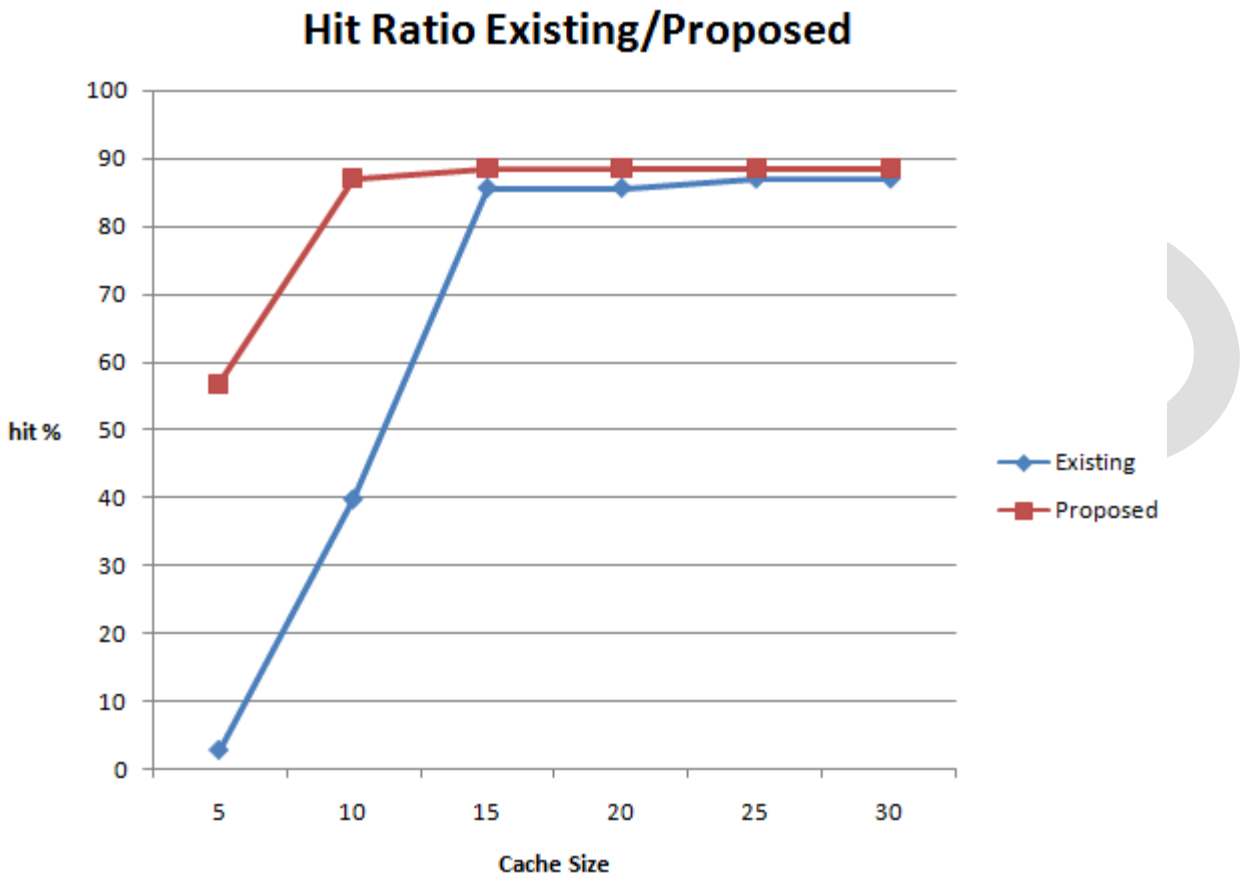

Fig (d) Hit ratio comparison between existing and proposed system

Fig. shows effect of hit ratio on different values of cache size used in both the cases existing(top 10 domain +dynamic approach) and proposed(user based using clustering) system. In both the cases, as the cache size increases, the hit ratio % also increases accordingly. But from the Fig. it is clear that our approach works better than existing approach

CONCLUSION AND FUTURE WORK

In this paper, we have proposed a system for prediction of web requests of users and accordingly,prefetching the content from the server. The dataset is used for the experimental work which has collected from the user history. The proposed framework improves performance of web proxy server using user based approach for caching and prefetching scheme which is clear in the result section. By using this proposed framework hit ratio is improved as shown in result section. In this paper also compare replacement policy.Designing of proxy server and implementation of the proposed framework are under the future scope.To find other technique which help in increase the hit ratio and improve performance uder future scope.

Acknowledgement

I would express my deep sense of regard to our Head of Department Mr. Vivek Sharma and to Mrs. Nidhi Seth, AP CSE Department for regular inputs and for constantly guiding. I am obliged to the staff members of CSE department, JMIT for the valuable information provided by them as and when necessary

REFERENCES:

[1]. Toufiq Hossain Kazi, Wenying Feng and Gongzhu Hu "Web Object Prefetching: Approaches and a New Algorithm", IEEE 2010, pp 115-120.

[2]. J. B. Patil and B. V. Pawar, "Improving Performance on WWW using Intelligent Predictive Caching for Web Proxy Servers," IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011.

[3]. C. Kumar and J.B. Norris, "A new approach for a proxy-level Web caching mechanism", *Decision Support Systems, Elsevier*, 46(1), (2008), pp.52-60.

[4]. R. Chiang, P. B. Goes, and Z. Zhang, "Periodic cache replacement policy for dynamic content at application server", *Decision Support Systems, Elsevier*, 43 (2), (2007), pp. 336- 348.

[5]. H.T. Chen, *Pre-fetching and Re-fetching in Web caching systems: Algorithms and Simulation*, Master Thesis, TRENT UNIVESITY, Peterborough, Ontario, Canada(2008).

[6]. T.Chen, "Obtaining the optimal cache document replacement policy for the caching system of an EC Website", *European Journal of Operational Research.181(* 2),(2007), pp. 828. Amsterdam.

[7]. Koskela, J. Heikkonen, ,and K. Kaski, (2003). "Web cache optimization with nonlinear model using object feature", *Computer Networks journal, elsevier* , 43(6), (2003), pp. 805-817

[8]. H.k. Lee, B.S. An, and E.J. Kim, "Adaptive Prefetching Scheme Using Web Log Mining in Cluster-Based Web Systems", *2009 IEEE International Conference on Web Services (ICWS)*, (2009), pp.903-910.

[9]. L. Jianhui, X. Tianshu, Y. Chao. "Research on WEB Cache Prediction Recommend Mechanism Based on Usage Pattern", *First International Workshop on Knowledge Discovery and Data Mining(WKDD)*, (2008), pp.473-476.

[10]. Abhari, S. P. Dandamudi, and S. Majumdar , "Web Object-Based Storage Management in Proxy Caches", *Future Generation Computer Systems Journal* , 22(1-2), (2006). pp. 16-33.

[11]. *Seung Won Shin, Byeong Hag Seong & Daeyeon Park, (2000)"Improving World-Wide- Web Performance Using Domain-Top Approach to Prefetching", Fourth International Conference on High- Performance Computing in the Asia-Pacific Region vol. 2, pp. 738-746.*

[12]. Sandhaya Gawade and Hitesh Gupta, "Review of algorithms for prefetching and caching" 2012 IJARCCE

[13]. [13]HeunKi Lee, Baik Song an, and Eun Jung Kim, "AdaptivePrefetching Scheme Using Web Log MininginCluster-basedWeb", InternationalConference on Web Services, IEEE, 2009, pp. 903-910.

[14]. Wenying Feng and Hua Chen ,"A Matrix Algorithm for Web Cache Prefetching", International Conference IEEE/ACIS, 2007, pp. 788-794

[15]. J. B. Patil and B. V. Pawar, "Improving Performance on WWW using Intelligent Predictive Caching for Web Proxy Servers," IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 1, January 2011

[16].]MukeshDawar, Charanjit Singh, "A Review on Web CachingTechniques",International Journal of Advanced Research inComputer Science and Software Engineering, ISSN: 2277 128X,Volume 4, Issue 3, March 2014

[17]. B.lalithadevi, A.Merry Ida, W.AncyBreen "A new Approach for improving world wide web techniques in data mining".IJARCSSE,2013.

[18]. Suvarnatemgire, poonam Gupta, "Review on web prefetchingtechniques", IJTEEE,2013

[19]. Anuradha,kiran jain, "Survey on an Integrated Approach for Web Pre-Fetching and Caching," Vol. 2, Issue 02, 2014 | ISSN (online): 2321-0613

[20]. Bhumika garg,Ravinder singh , "Hybrid Approach for Performance of Web Page Response through Web Usage Mining " Vol. 4, Issue 07, july 2014 | ISSN (online): 2277-128X