

A Comprehensive Survey on Importance of Web Caching and Pre-Fetching

Rituparna Chandrakar¹, Shivani Varshney²

¹M. Tech. (CT) Scholar, Department of CSE, RCET, Bhilai, Chhattisgarh, India

²Assistant Professor, Department of CS, RCET, Bhilai, Chhattisgarh, India

Abstract: *This paper provide survey on some principles, standards and some current and existing web caching and prefetching approaches, discusses some studies that take into consideration of integrating both web caching and web prefetching together. This paper consists of a survey of the web page prefetching mechanism which is provided by some authors. Two essential methodologies used to decrease the noticeable response time by users are Web caching and pre-fetching. A perfect pre-getting and caching scheme is a system that able to predict the next set of requests and previously load those into the cache.*

Keywords: prefetching mechanism, web page

1. Introduction

Web caching is a well-known strategy for enhance the performance and execution of Web-based system by keeping Web objects in location closer to user that are likely to be used in the near future. The objects which need to pre-fetched are kept and stored in a local cache so that the latency time can be reduced. This is paper presents survey for handling a web caching and pre-fetching are provided.

The Worldwide web has been examining as considerably the largest distributed information systems which provide access to shared data. Mass exploration and research has done to enhance the response time and latency of web based s/s as the information is distributed over a large geographical location. Pre-fetching and Web caching are two essential approaches used to reduce the observable response time perceived by users. There are so much documents of data available in net that keep to a wide range of interests, for example education, sports, scientific research, news, travel, entertainment, shopping, stock market growth, weather and maps. Predictive web prefetching refer to the strategy of deducting the upcoming page accesses of a client based on it's previous accesses. Web prefetching is the procedure of deriving user's further future request for web document and getting that requested document in to the cache, before an explicit demand of request is made for them. The Web caching intends the aims to enhance the performance and execution of web-based systems by putting, storing and reusing web objects that are likely going to be used in the near future for upcoming access. The main fundamental advantage of applying prefetching is that it prevents underutilization of band-width and hides part of latency . Web prefetching acts complementary to caching; it can essentially and significantly improve. Web caching has concentrated on the use of historic information of previous searches about web objects.

1.1 Prefetching Definition

Web-Page prefetching is a very popular and new technique for reducing the web latency. Prefetching overcomes the

problem and limitations which regular web caching faces, It is also considered as "Active Caching" It is about to getting and guessing what can be the next requested page will be It attempts to assume about what the upcoming requested page will be. For standard HTML file accesses, pre-fetching mechanism try to predict the next set of pages or files that will be requested, and use this information to pre-fetch the files/pages into the server cache. This significantly speeds up access greatly to those records and files, and enhances the client's experience. To be effective however, the pre-fetching strategies, techniques and methods must be able to predict sensibly and reasonably with at-least minimum computational overheads consequent web accesses.

Pre-fetching mechanism and procedures can only be valuable only if they can predict accesses with reasonable and sensible accuracy and else represent a significant computational load at the server if they do not.

1.2 Prefetching Mechanism Steps

Web prefetching associate basically two main steps. At First, forecasts are made based on past experience about client's accesses and preferences, and the corresponding hints are provided. Than at Second step, the prefetching engine determines and chooses what objects should be going to be perfected. The prefetching engine can be situated at an intermediate web proxy server or at the web browser.

1.3 Basic Principle of Prefetching

Web prefetching is a technique for reducing web latency based on predicting the next future web objects to be accessed by the user and prefetching them during times. If finally the user requests any of these objects, it will be already on the client cache. This technique takes advantage of the spatial locality shown by the web objects [1,2].

The prefetching strategy has two main components: The prefetching engine and the prediction engine. The prediction engine which forecast and runs a prediction algorithm to forecast next client's request. The Prefetching engine

Volume 5 Issue 6, June 2016

www.ijsr.net

Licensed Under Creative Commons Attribution CC BY

chooses and decide to prefetch them or not depending on some situation like available bandwidth .Each engine can work at any component of the web architecture engineering [3].

The predictions are the quantity and number of objects. It can be predicted by the trained prediction engine. The PR (prefetch request) represents the quantity or number of component prefetched. The number of component prefetched which are requested later by the client is the prefetch hit. The inverse of the prefetch hit is the prefetch miss, which shows the number and quantity of prefetched component that were never requested by the user. finally, user request assign to the aggregate volume of items requested by the client, and the client request not prefetched, shows the number of objects demanded by the client that were not prefetched [1].

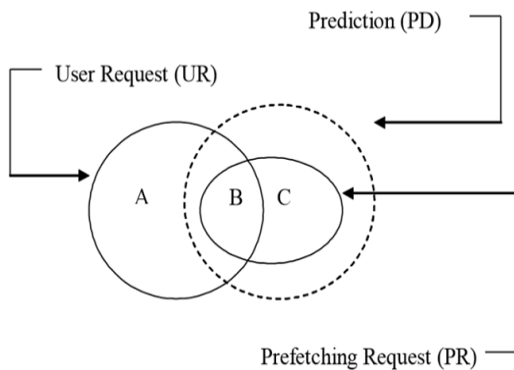


Figure 1: Web prefetching type of request

As appeared in fig. 1, the arrangement of prefetch request that is PR is a subset of the forecast set that is PD. The result that is after-effect of interaction and crossway in between the client demand or user request set that is (UR) and Prefetch ask for set is the prefetch hit subset that is (PH). This subset is the principle element to lessen the latency. In Fig. 1, A speaks to a client demand not prefetched that is shown as user request not prefetched (URnP), which is a client demand neither anticipated nor prefetched. B is a prefetch demand made by the prefetching engine that is asked for later by the client, therefore turning into a prefetch hit. C is a prefetch miss (PM) coming about because of an unsuccessful forecast that was prefetched yet never requested by the client. This solicitation or request turns out to be additional movement and additional server burden or load.

A. Web Prefetching Samples and examples

It is easy to visualize the following three prefetching instances in fig. 2 prefetching between web servers and web clients, prefetching between web clients and proxy caches, and prefetching between proxy caches and web servers [3].

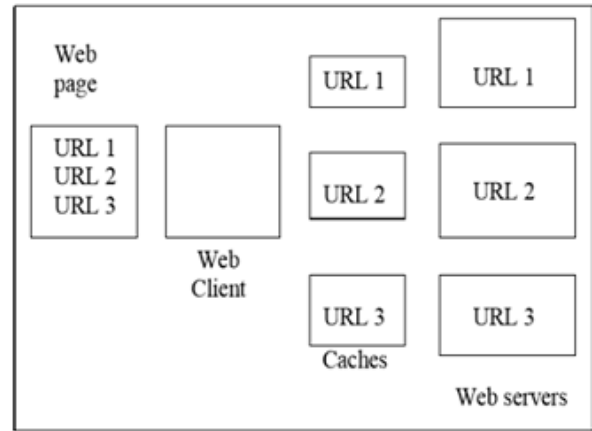


Figure 2: Prefetching possibilities

Reduce and Lessen latency: A Proxy server saves the aftereffects of all the requests from various clients for a specific measure of time. in case in a point, consider a situation where both users y and x get access to the www through a proxy server. Let's give a chance and assume that user x demands for a specific page of website assume page 1. Sometime later, than next user y also demands the same page. Rather than forwarding the request to the web server where page 1 really resides, this can be a time-consuming and a period devouring operation, proxy server just simply returns back this page from its storage cache where all the downloaded pages are held before being over written by fresh new arrivals. Since proxy server which is used is most often on the same network as the user, this is a much speedier and faster operation, thereby reducing the perceived latency to some extent [4].

2. Literature Review

A. A Study of Strategies for Prefetching

This segment corresponds to the study of various methodologies and strategies required till now for prefetching web pages.

Sr. No.	Author Name	Conclusion
1	Jose Borges and Mark Levens	They have proposed a dynamic clustering-based method to increase a Markov models accuracy in representing a collection of user web navigation sessions. The method makes use of the state cloning concept to duplicate states in a way that separates in-links whose corresponding second-order probabilities diverge. In addition, the new method incorporates a clustering technique which determines ancient way to assign in-links with similar second-order probabilities to the same clone.
2	Sriporn Chimphee	They have presented a rough set clustering to cluster web transactions from web access logs and using Markov model for next access prediction. Using this approach, users can effectively mine web log records to discover and predict access patterns. He performs experiments using real web trace logs collected from .the servers. In order to improve its prediction ratio, the model includes a rough sets scheme in which search similarity measure to compute the similarity between two sequences using upper approximation.
3	Silky Makkar and R.K. Bhaty	They have proposed a bracing approach for increasing web server performance by analyzing user behavior, in this pre-fetching and prediction is done by pre-processing the user access log and integrating the three techniques i.e. Clustering, Markov model and association rules which achieves better web page access prediction accuracy; This work also overcomes the limitation of path completion i.e. by extracting web site structure paths are completed, which helps in better prediction, decreasing access time of user and improving web performance.

Figure 3: Literature Review

B. Study of Prefetching Models

This area relates to the investigation of different calculation and algorithms which has been utilized as a part of the different phases of site page prediction process i.e. V. Padbanabham and J. Big shot [6] use N-hop Markov models which predicted the next page of website which users or client will most likely access by matching the client's ongoing current access sequence with the user's historical web access successions for enhancing prefetching procedures for web reserves for web caches. R.R. Sarukkai [7] utilized first-order Markov models to display and model the sequence arrangement of pages asked by a user for foreseeing the next page accessed. A "personalized" Markov model is prepared for every individual and utilized for expectations in client's future request sessions. By and by In practice, however, it is exceptionally costly to develop a special model for every client separately, and the issue gets even worse when there exist many diverse users within a major Web site.

F. Khalil[7] presents the Integration Prediction Model (IPM) by joining Markov model, Association standards and clustering calculation together. At that point, the expectation is performed on the cluster sets instead of the real sessions. The IPM incorporation model depends on the distinctive requirement. The web client sessions first are separated into various bunches utilizing k-mean clustering algorithm and cosine distance measure.

At that point, a coordination model figures Markov model forecast on the subsequent resulting cluster. This calculation enhances the state space complication nature since Markov model forecast did on the specific clusters instead of the entire information set. In the case of state nonappearance or absence in the preparation information or training data or where, the state prediction expectation probability likelihood is not marginal, Association principles are analyzed more

states than Markov model taking a gander at more history. At last, if another page is exhibited, the cosine separation is figured and distinguishes a suitable cluster that another website page belong to have a place with. The coordination model has been demonstrated through the trials that enhance the forecast exactness. In addition, executing the forecast model on the cluster achives preferable results over on the non-clustered information. Despite the fact that, a page access forecast execution was enhanced, in any case, it can be seen that their coordinated algorithm has a complex strategies and should over and again utilize so as to enhance their prediction execution.

S. Chimphee [8] presented a Hybrid Markov Fuzzy Models (HyMFM) that are obtained by integrating the advantages of all three prediction models: Markov model, Association rules and Fuzzy Adaptive Resonance Theory (Fuzzy ART). HyMFM algorithm was developed for the web user sessions clustering by proposing the new sequence representations and the new similarity measures in incremental learning of Fuzzy ART control structure. A web user session was represented into the transition matrix representation, referred to as session matrix, which is constructed based on a transition matrix of a first Hybrid Markov model. Both elements fit well into the design of this thesis and the clustering task which the web user sessions are treated as order sets of accesses. Consequently, the new similarity measures were developed to enable the application of Fuzzy ART clustering. This study defined two new similarity measures: Matrix norm similarity and Matrix distance similarity. These measures alleviate the overestimation problem in Fuzzy ART algorithm which use the city-block distance metric as the similarity between input and prototypes. Thus, the web user sessions were clustered into groups with similar patterns in during the training phase and when it is confronted by a new input, it produces a response

that indicates which cluster the pattern belongs to and then HyMM applied to each cluster.

Problem Identification

Web users are facing the problems of information overload and drowning due to the significant and rapid growth in the amount of data information and the quantity of users. Prefetching strategies try to attempt and predict the next set of records, files, pages that will be requested, and utilize this data to pre get or pre-fetch the documents, files, pages into the server cache. This significantly accelerates access to those documents, and enhances the clients' experience. In this paper, a study of the website page prefetching is provided.

Types of Web Prefetching

The prefetching procedures and techniques can be implemented on server, proxy or client side. The client-based prefetching focuses on the exploration patterns of a single client user across distinguish Web servers. On other side, the sever-based prefetching focuses on the exploration patterns of all users retrieving a single website. The proxy-based intermediary prefetching focuses on the exploration patterns of a group of users across distinguish Web servers. Subsequently, this method can reflect a common interest for user's community. In other words, the prefetching contents can be shared by several users. Table 4 outlines and summarizes types of prefetching as per location.

Table 4: Types of prefetching based on location

Prefetching Location	Data for Prediction Model	Advantages	Disadvantages
Client	Historical and current user requests	Easy to partition user session and realize personalized prefetching.	<ul style="list-style-type: none"> •not share prefetching content among users. •needs a lot of network bandwidth.
Proxy	Proxy log and current user requests	<ul style="list-style-type: none"> •reflects common interests for a group of users. •shares prefetching content from different servers among users. 	<ul style="list-style-type: none"> •not reflect common interests for a single Website from all users.
Server	Server log and current user requests	<ul style="list-style-type: none"> •records a single website access information from all users and better reflect all users' common interests. 	<ul style="list-style-type: none"> •not reflect users' real browsing behavior . •difficult to partition user session. •needs additional communications between clients and servers for deciding

Significantly in recent years, huge attention and consideration has been paid to intermediary proxy-based prefetching since it is more successful and more exact in foreseeing and predicting the correlated pages of distinguish Websites in the similar interests for more homogeneous users [10 , 34 ,35, 36].

Web Prefetching Approaches

The exiting prefetching calculations and algorithms can be categorized into two main categories according to the information taken for expected prediction: content-based prefetching and history-based prefetching. The content-based prefetching classification predicts future user requests

demand relying upon the investigation of website page content information to find HTML links that are likely to be trailed by the clients. [9] Utilize ANN to predict upcoming requests depending upon words in anchor text of page URL. The keywords retrieved from web page documents were given as inputs to ANN which than that predict whether the URL needs to get prefetched or not. The content-based prefetching strategies are not suggested to be practically implemented at server side because the high load for parsing each web page served can adversely influence the server administration and service time[10].The history-based prefetching classification predicts upcoming user requests depending on analyzed web page access behaviors in the past. Numerous studies were interested in improving the techniques of this category. The algorithms of this classification can be categorized into four approaches:
 1)Dependency graph based approach
 2)Markov model based approach
 3)Cost function based approach
 4)Data mining based approach [11].

A. Prefetching Approach Based on Cost Function

This methodology prefetches some web data objects into the store cache depending on some facts such as :

- 1) Popularity
- 2) Lifetime of web pages.

Many Web prefetching techniques based on cost function approach are presented in the literature. The algorithms and calculations under this approach are summarized as follows.

- By Popularity Prefetching: [12] In this Top-10 prefetching approaches are recommended, which keeps in cache the ten most popular objects from each web server.
- By Lifetime Prefetching: [13] Recommended prefetching approach called Prefetch by Lifetime. This approach chooses n data objects that have the longest lifetime, and thus intends to minimize the data transmission bandwidth.
- Good Fetch: Approaches for prefetching such as Good-Fetch[14] and APL [13] attempted to balance data object popularity and update rate of data object, thereby they can get nice amount of hit ratio improvement at a modest expense of bandwidth.
- Objective-Greedy Algorithms: [15] It proposed a family of prefetching algorithms, Objective-Greedy prefetching, wherein each algorithm calculation greedily prefetches those web data objects that give the highest performance as per the metric that it goals to improve.

B. Prefetching Approach Based on Data Mining

The prefetching approach based on data mining can be classified into

- 1) Based Prefetching Approach Based on Association Rule
- 2) Prefetching Approach Based on Clustering.

1. Prefetching Approach Based on Association Rules

Association rule discovers gatherings of homogeneous pages that are generally retrieve together in same user session. The user session can be explained as the sequence of pages made

by a individual user during a visit to a particular site. In association rules, confidence and support are two essential measures for benchmarking the quality and strength of any association rule standard. Support can be defined as the analysis of frequent pages and confidence can be defined as the analysis of association rules from those frequent pages. Since a client session may incorporate several pages, it is exceptionally hard to discover matching rule antecedents. So in this way here, sliding window is utilized in association rule algorithms. The size of sliding window is iteratively decreased until an exact match with a rule antecedent. [16] Presented four sort of successive sequential association rules:

- 1) Subsequence rules: The order of the pages has taken into consideration here in association rules.
- 2) Latest subsequence rules: The latest or most recent pages and order of the pages has taken into consideration here in association rules.
- 3) Substring rules: The adjacency and the order of the pages has taken into consideration here in association rules.
- 4) Latest substring rules: The most recent pages, the adjacency and the order of the pages has taken into consideration here in association rules. Many inquiries used association rules techniques in web prefetching. [17] used an N-gram mining strategy and extended the GDSF function to enhance hit ratios of caching and prefetching. [18]

Developed an A priori-based mining method to Reduce a rule table for predicting and prefetching the highest document into a proxy buffer. However, several pointless rules are produced in the rule table. Recently, [19] have built an access sequence arrangement miner to mine popular surfing 2- sequences with their conditional probabilities from the proxy log, and stored them in the rule table. After than, as per buffer keeps the rule table, a prediction-based buffer manager manage to achieved relevant appropriate actions such as data data document prefetching, document caching, and even cache and prefetch buffer measure size adjusting to get better buffer usage. The main complexity involved with association rule mining-based method is that several unnecessary rules and calculations are produced in the rule table of the frequent pages with a high frequency. This causes inconsistent expectations and predictions, especially when the dataset is huge [20]. In other words, present association rule based Web prefetching calculations and algorithms foresee and predict a particular specific page depending on patterns analyzed from all client's references. As a consequence, the prediction based several patterns is inaccurate and inefficient [20, 21].

2. Clustering Based Prefetching Approach

Clustering is used for discovering similarity groups and closeness-ed bunches in data called clusters, such that the data instances information in the same cluster group are similar to each other while data instances in different clusters are very distinguished from each other. It can say, the intra-clusters span is minimized, while the inter-clusters span is maximized. Numerous studies utilized the clustering for enhance the effectiveness and scalability of the real-time personalization tasks [20, 22,23,24,25]. The clustering

approaches can be categorized into following two approaches:

- 1) Distance-Based Approaches
- 2) Model-Based Approaches.

The Model-Based Clustering Approach:

It often determines the model type apriori . Moreover, the model structure can be determined by model selection techniques and parameters estimated using maximum possibility algorithms, like, the Expectation Maximization (EM) [26]. The distance-based clustering bunches the similar user sessions together into clusters based on a distance measure in between pairs of pages.

There are basically two types of distance-based clustering techniques:

- 1) Hierarchical Clustering
- 2) Partitional Clustering.

Partitional Clustering Method: The data is partitioned into K groups or clusters.

The most used partitional clustering method is k-means algorithm. In the k-means algorithm, the user needs to specify the number of cluster group k in advance.

The hierarchical method has a number of desirable characteristics, which make it popular. It builds a hierarchical set of nested clusters. The clustering techniques can be used in either Website page clustering or user client session clustering [20]. The web page clustering is accomplished by gathering group of pages having similar content as suggested in [27, 28]. At other side, the clustering techniques are used for creating group of the similar sessions in clusters. The user client session clustering can be widely utilized in several fields such as web page prediction expectations, prefetching, personalization and web. [33] have introduced an algorithm for clustering inter-site Web pages in proxy, called clust Web. The clust Web algorithm calculation has evolved the clusters by partitioning the Web navigational graph using association rule mining strategies.

This method depended on the content of clusters. If each cluster group contains a large amount of objects, it will bring out result in an aggressive prefetching arrangement policy. Moreover, grouping of IPs as per their domains may fail when IP get change dynamically. [29] presented a prefetching scheme using ART1 network depend on clustering strategy to group hosts, users and then prefetch their requests as per the prototype vector of each group. This strategy may cause a noticeable increase in network traffic since all data objects in the similar cluster are fetched. Moreover, the experiment only concentrated on the prefetching and did not point the interaction between web caching and prefetching.[30] proposed 3D cube model that shows successive sequential user session data and permits distinct data analyses to be comfortably produce, such as summarized statistical observation, clustering and successive sequential association analysis. Then, [30] applied the clustering group results utilizing the data cube model to the complexity of integrated web caching and prefetching. The analyser utilised a variant of the k-means calculation

algorithm for clustering, so the clustering need both prior information about number of clusters and initializing cluster centroids randomly. [31] provided page rank-based prefetching strategy for clustered website page accesses. The pages which linked to any requested page are ranked, and the rank is utilized to figure out the pages to be prefetched. Moreover, page rank is used as a replacement strategy, i.e. those pages with the lowest rank are replaced with new ones. The analyzer heuristically defined any Website directory with 200 or more files under it as a candidate cluster. In current years, the data mining procedure have been utilized effectively in web proxy prefetching area [34, 35, 36].

Performance and Execution Measures

There are various metrics facts utilized to observe the effectiveness and the execution of Web pre-fetching strategies. The following measures are most oftenly utilized for measuring performance of web prefetching [32, 19] :

- **Precision:** It is the ratio of prefetch hits to the total number of data objects prefetched as shown in Eq.

$$P_c = \frac{\# \text{PrefetchHits}}{\# \text{Prefetches}}$$

- **Byte Precision:** Byte precision calculates the percentage of prefetched bytes which are subsequently demanded. It can be measured by replacing the number of prefetched data objects with their size in bytes .
- **Recall (Rc):** Recall is the ratio of measured prefetch hits to the aggregate number of data requested by users as shown

$$R_c = \frac{\# \text{PrefetchHits}}{\# \text{Requests}}$$

- **Byte Recall (RcB):** Byte recall calculates the rate of requested bytes B that were already prefetched and subsequently requested.
- **Traffic Increase (ΔTr):** Traffic Increase is the bytes transferred through the network system when prefetching is employed splited by the bytes moved in the non-prefetching case.
- **Latency per page ratio proportion:** The latency per page ratio proportion is described as the proportion ratio of the latency that prefetching accomplish to the latency with no prefetching. The latency per page is measured by looking the time in between the browser startup process of an HTML page GET and the browser reception of the last byte of the last embedded image or object for that page. This metric shows the advantage perceived by the user, that will be better as lower its value is.

Incorporating Web Prefetching and Web Caching

Web intermediary proxy caching and prefetching are the famous strategies that plays an important role in enhancing the Web execution performance. Since the Web intermediary storing proxy caching exploits the temporary area and the web prefetching uses the spatial area of locality of the Web data objects, prefetching and Web proxy caching both complement each other. Hence, collaboration of the prefetching and caching helps on enhancing hit ratio

proportion and deducing the user-perceived latency. Wherever, if the web intermediary caching prefetching and caching integrated without efficiently, this may cause increase in the network traffic and as well as load of web server. However, the cache area is not utilized optimally [37,35,33,38]. Hence, the prefetching strategies should be created by keeping in mind in order to overcome these constraints of limitations. Fundamentally, the web prefetching includes two steps:

First assuming future pages of users

Second prior loading them into a cache . This means the web prefetching includes also the caching. However, the prefetching and web caching are addressed disgustingly by many analyzers in the past. It is essential to think about the effect of these two techniques combined together. Few researches were discussed integration of web caching and web prefetching together. [39] Researched effect of a combination of caching and prefetching on end user latency. They concluded that the collaboration of web reserving caching and prefetching can potentially enhance latency up to 60%, whereas web caching singly enhance the latency up to 26%. [17] Recommend an application of web log mining to get web-document access patterns and utilized these patterns arrangement to expand the well-known GDSF caching protocols and prefetching protocols. [40] proposed cache replacement calculation algorithm called IWCP for coordinating Web caching and Web prefetching in client-side proxies. They formulated, detailed a normalized benefit function to figure out the benefits from caching an object either a non implied object or an implied object according to some prefetching principles. Likewise [9], [37] utilized ANN in both prefetching rules and Web cache deduction decision. This method based on the keywords of URL anchor text to forecast the user's future requests. The most noteworthy factors that is recency and frequency were avoided in web cache replacement decision. In addition, since the keywords retrieved from huge web documents were provided as inputs to ANN, applying ANN in this way may cause additional overhead on the server. [41] given a concise set of calculation algorithms for incorporating web caching and prefetching for wireless remote local area network, including successive sequence mining based prediction expectation algorithm, context-aware prefetching algorithm and profit-driven caching substitution policy. [42] Given a framework system for combining Web caching and prefetching on cell phone environment. They provide hybrid technique that is Rough Neuro-PSO based on collaboration of ANN and PSO for categorization of Web object. Then, rules from log data information are generated by Rough Set strategies on the intermediary proxy server. In prefetching side, prefetching methods based on XML is recommended to be executed on cell phone to handle communication in between client and server. In summarized way, the past works coordinated the web prefetching with caching; However, these methodologies are still not sufficiently effective [34]. Most past works utilized association rules for prefetching methodologies, which are not exact and inefficient since these works predict a particular page based on patterns analyzed from all users' references [20, 21]. Moreover, these strategy employ the conventional replacement policies that are not efficiently productive in web.

3. Conclusion

Prefetching and Web caching are well known strategies for improving the performance of internet systems. The combination of the prefetching and the web web caching doubles the performance compared to single caching. I have given glance into principles standards and existing works of web-Caching and prefetching. Furthermore, categories and types of prefetching have presented, reviews have been discussed with concentrated approaches. This paper surveys standards and some current web reserving and prefetching approaches. Furthermore, sorts and classifications of prefetching have exhibited

References

[1] "An intelligent technique for controlling web prefetching costs at the server side", by Jos'e A. Johann M'arquez, Josep Dom'enech, Gil and Ana Pont International Conference IEEE/WIC/ACM, 2008, pp. 669-676

[2] "A Web Caching and Prefetching Simulator", by Johann M'arquez, Josep Dom'enech, Jos'e A. Gil and Ana Pont IEEE, 2008, pp. 346-350.

[3] "Web Caching and Its Applications", by S. V. Nagaraj Kluwer Academic Publishers, 2004

[4] "A Novel Approach for Determining Next Page Access", by Payal Gulati, A. K. Sharma, Amit Goel, Jyoti Pandey, IEEE, 2008, pp. 1109-1113.

[5] "A clustering based approach for modeling user navigation with increased accuracy" by J. Borges and M. Levene Proceedings of the Second International Workshop on Knowledge Discovery from Data Streams (WKDD) in conjunction with PKDD 2005, Porto, Portugal, Outubro, 2005

[6] "Using Predictive prefetching to improve World Wide Web latency", by V. Padmanabhan, J. Mogul and ACM SIGCOMM Computer Comm. Rev., Vol. 26, no. 3, July 1996

[7] "Link prediction and path analysis using Markov by R. Sarukkai

[8] Siriporn Chimphlee, Naomie Salim, Mohd Salihin Bin Ngadiman, Witcha Chimphlee, Surat Srinoy, "Rough Sets Clustering and Markov model for Web Access Prediction", 2006 From Web Prefetching Approaches

[9] "A Keyword-Based Semantic Pre-fetching Approach in Internet News Services", by T.I. Ibrahim, and C-Z. Xu IEEE Trans. Knowl. Data Eng., 16 (5), (2004). 601-611

[10] "Using current web page structure to improve prefetching performance", by J. Sahuquillo, J. A. Gil, J. Domenech, and A. Pont Computer Network Journal, 54(9), (2010), 1404-1417.

[11] "A Survey of Web Prefetching", by J. Yu Journal and G. Zhimin Zhijie, of computer research and development, 46(2), (2009), pp. 202-210.

[12] "A Top-10 approach to prefetching on the Web", by P. Markatos and C. E. Chronaki Proceedings of INET'98 Geneva, Switzerland, (1998), pp. 276-290.

[13] Web prefetching: Costs, benefits and performance", by W. Shu Proceedings and Y. Jiang of the 11th

International World Wide Web Conference, New York, ACM, (2002).

[14] "The potential costs and benefits of long-term prefetching for content distribution", by P. Yalagandula, Venkataramani and R. Kokku Computer Communications, 25(4), (2002). pp. 367-375.

[15] "Objective-greedy algorithms for long-term Web prefetching", by W. Bin, and A. D. Kshemkalyani Proceedings of Third IEEE International Symposium on Network Computing and Applications (NCA 2004), (2004).

[16] "Building association-rule based sequential classifiers for web-document prediction", by Q. Yang, T. Li, and K. Wang Journal of Data Mining and Knowledge Discovery. 8(3), (2004), 253-273.

[17] Q. Yang, H. Zhang, and T. Li, "Mining web logs for prediction models in WWW caching and prefetching", Proceedings of the 7th ACM International Conference on Knowledge Discovery and Data Mining, (2001), pp. 473-478

[18] "Rule-assisted prefetching in web-server caching", by Lan, S. Bressan, B. C. Ooi, and K. L. Tan Proceedings of the 9th ACM International Conference on Information and Knowledge Management, (2002), pp. 504-511

[19] "Mining web logs to improve hit ratios of prefetching and caching". By J. Y. f. Huang and J. M. Hsu Knowledge-Based Systems, 21(1), (2008), pp. 62-69

[20] "Integrated model for next page access prediction", by Khalil, J. Li An, and H. Wang Int. J. Knowledge and Web Intelligence, 1(1-2), (2009), pp. 48-80 (33).

[21] "Measuring similarity of interests for clustering Web-users", by J. Xiao, Y. Zhang, X. Jia, and T. Li 12th Australasian Database Conference (ADC), (2001), pp. 107-114

[22] "A method for personalized clustering in data intensive web applications", by M. Rigou, S. Sirmakessis, and G. Tzimas APS'06, Denmark, (2006), pp. 35-40.

[23] "STAVIES: A system for information extraction from unknown web data sources through automatic web warppergeneration using clustering techniques", by N. K. Papadakis, and D. Skoutas IEEE Transactions on Knowledge and Data Engineering, 17(12), (2005), 1638-1652.

[24] "Model-based clustering and visualization of navigation patterns on a web site", by I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White, Data Mining and Knowledge Discovery. 7(4), (2003), 399-424.

[25] "Clustering documents in a web directory", by Adami, P. Avesani, and D. Sona WIDM'03, USA, (2003), pp. 66-73.

[26] "A unified framework for model-based clustering", by S. Zhong, and J. Ghosh Machine Learning Research, 4, (2003), pp. 1001-1037.

[27] "An artificial immune system approach to document clustering", by N. Tang and R. Vemuri Proceedings of the Twentieth ACM Symposium on Applied Computing. Santa Fe, New Mexico, USA, (2005), 918-922.

[28] "Document Clustering Based on Modified Artificial Immune Network", by L. Xu, H. Mo, K. Wang, and N. Tang Rough Sets and Knowledge Technology. G. Wang, J. Peters, A. Skowron and Y. Yao, Springer Berlin / Heidelberg, 4062, (2006), 516-521.

- [29] S. K. Rangarajan, V. V. Phoha, K. Balagani, R. R. Selmic, and S.S. Iyengar, "Web user clustering and its application to prefetching using ART neural networks", IEEE Computer, (2004).
- [30] "A data cube model for prediction-based web prefetching", by Q. Yang, J.Z. Huang, and M. Ng Journal of Intelligent Information Systems, 20(1), (2003), pp.11-30.
- [31] "Optimizing Web servers using page rank prefetching for clustered accesses", by V. Safronov, and M. Parashar Information Sciences, 150(3-4), (2003), pp.165-176.
- [32] "Evaluation, Analysis and Adaptation of Web Prefetching Techniques in Current Web", by J. Domenech, A. Pont-Sanjuán, J. Sahuquillo, and J. A. Gil, Web-based Support Systems, Springer, London, (2010). 239-271.
- [33] "A clustering-based prefetching scheme on a Web cache environment", by Pallis, A. Vakali, and J. Pokorny Computers and Electrical Engineering, 34(4), (2008). pp.309-323
- [34] Web Latency Reduction With Prefetching, by Q. Liu PhD thesis, University of Western Ontario, London (2009).
- [35] "Mining web logs to improve hit ratios of prefetching and caching". By Y.f. Huang and J.M. Hsu Knowledge-Based Systems, 21(1), (2008), pp. 62-69
- [36] "A clustering-based prefetching scheme on a Web cache environment", by Pallis, A. Vakali, and J. Pokorny Computers and Electrical Engineering, 34(4), (2008). pp.309-323
- [37] Personalized and Artificial Intelligence Web Caching and Prefetching. Master thesis, by U. Acharjee University of Ottawa, Canada (2006).
- [38] "Markov Tree Prediction on Web Cache Prefetching", by W. Feng, S. Man, and G. Hu Software Engineering, Artificial Intelligence (SCI), Springer-Verlag Berlin Heidelberg, 209, (2009). pp. 105-120.
- [39] Exploring the bounds of web latency reduction from caching and prefetching", T. M. Kroeger, D. D. E. Long, and J. C. Mogul, Proceedings of the USENDC Symposium on Internet Technology and Systems, (1997), pp. 13-22.
- [40] "Integrating Web Caching and Web Prefetching in Client-Side Proxies", by W. Teng, C. Chang, and M. Chen IEEE Transaction on Parallel and Distributed Systems, 16(5), (2005), pp 444-455.
- [41] "An Integrated Prefetching and Caching Scheme for Mobile Web Caching System". By Jin, T. Sihua, C. Lin, X. Ren, and Yu. Huang Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD), (2007).
- [42] "Rough Neuro-PSO Webcaching and XML prefetching for accessing Facebook from mobile environment", by S. Sulaiman, S. M Shamsuddin, and A. Abraham World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), (2009).