ISSN:0975-9646

A Survey on Web Pre-Fetching Techniques

Priyansha bangar,
M.tech scholar,
TIT science,Bhopal

Kedar Nath Singh,
Asst. professor,
TIT science,Bhopal

Abstract— **web is a rich domain of opportunities, where different kind of data and knowledge is available. To get advantage of these data a huge amount of internet users access the web. The content distribution over internet is a complex task, in this process a number of intermediate proxy servers, ISP servers are participating. In order to get advantage of the distributable contents on client end web pre-fetching techniques are used to deliver data efficiently. In this paper, we learn about the process of web pre-fetching and evaluating different web pre-fetching techniques that are recently developed. Web prefetching and web caching both are the two major areas of research to reduce the latency. Web caching is used to reduce temporal latency and Web prefetching is used to reduce spatial latency. Finally based on basis these studies, a more optimum new web pre-fetching technique is proposed using data mining technique.**

Keywords— **web prefetching, web caching, proxy server, k-means, cluster, outlier.**

I. INTRODUCTION

Web is a rich source of data and every day a large number of internet users access the web to find data. To make efficient data delivery for these internet users various web caching and pre-fetching techniques are employed. These techniques works on the basis of data, which is required to distribute. Web servers provides two different types of data, first static data stored in files at a server and dynamic data which are constructed by programs at runtime. That execution of program is taken place when a user makes request for particular content. The presence of dynamic data, often slows down Web sites, because of data is prepared on demand basis. On the other hand, high performance Web servers typically delivers several hundred static files per second, In contrast the rate at which dynamic pages are delivered is often one or two order of magnitudes slower.

A web cache is a mechanism for the temporary storage (caching) of web documents, such as HTML pages and images, to reduce bandwidth usage, server load, and perceived lag through faster access and in less time. A web cache stores copies of documents passing through it. subsequent requests may be satisfied from the cache if certain conditions are met.[1] for example Google's cache link in its search results provides a way of retrieving information from websites that have recently gone down and a way of retrieving data more quickly than by clicking the direct link.

People use WWW to access information from remote sites. But they do not like to wait long for their results. The latency in retrieving a Web document depends on several factors like.

- Speed of Servers
- Speed of Clients
- Network Bandwidth and Propagation Delay

Some of these delays, such as client or server slowness or transmission time, can be reduced by using faster computers or higher bandwidth links. However, propagation delay which is basically determined by the physical distance traversed cannot be reduced beyond a point.

Prefetching systems are usually based on the basic web architecture. The basic web architecture is characterized by web clients, which is, the software employed by users to access the web and web servers, which contain the information that users demand.

The framework for Client Server model is:-

- **Client Side:** Association rules are defined in prefetching engine. Prefetching engine is at client side. Prefetching engine stores web objects that are prefetched by the server side prediction engine by pre-processing web log. For a client's request, pages predicted by the prediction engine are stored in prefetching engine at the client side. A cache based queue contains at prefetching engine which keeps the predicted pages for the user.
- **Server Side:** There are two modules at server side namely the Prediction Engine and the Web log. When a client sends a request for a web page/web object, prediction engine will predict extra pages stored in the prefetching engine.
- **Prediction Engine:** The prediction engine contains an algorithm that will predict the pages for the user. In the proposed system, we have taken sequential rank based selection algorithm**.**

Web logging: For replying the client's request, the server maintains a log of this request. From the web log, using our steps of mining the logs, we predicted the pages in the real environment and made pages local to the users. The predicted page for the client is brought in the predicting a page.

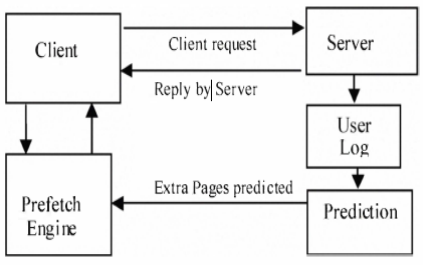

Figure 1 Framework for Web Prefetching[4]

When a client requests for a web page, before accessing the web page a prediction is made for accessing that web page. All the web objects are brought from server to the client. The access to the web objects are on the basis of the data prefetched from the server. The data of the web logs from servers will be tested on both existing algorithm and the proposed model. In the real environment, the results show that our proposed mechanism performed is better than the existing algorithm for web page prediction.

Major advantage of the Sequential Rank Selection algorithm is that, It selects only one web page of a website for prefetching purposes of user; hence consumes much less memory space of users and utilizes much less bandwidth of the network. The proposed architecture reduces the user's latency due to the efficient prediction of web pages by the Sequential Rank Based Selection algorithm from the cluster.

Web usage mining is used to discover interesting user navigation pattern and can be applied to many real world problems, such as improving websites/pages, making additional topic or product recommendations, user/customer behaviour studies, etc. This article provides a survey and analysis of current web usage mining system and technologies. A web usage mining system performs 5 major tasks: data gathering, data preparation, navigation pattern discovery, pattern analysis and visualization and pattern application. Each task is explained in detail and its related technologies are introduced. A list of major research system and projects concerning web usage mining is also presented [2].

Therefore Dynamic web cache is better suited to dynamic web pages than most proxy caches because it allows the applications program to explicitly cache, invalidate and update objects.

The cache replacement is the core or heart of the web caching; hence, the design of efficient cache replacement algorithms is crucial for caching mechanisms achievement. Thus, cache replacement algorithms are also called web caching algorithms. Due to the limitation of cache space, an intelligent mechanism is required to manage the Web cache content efficiently. The traditional 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 portion 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. However, combination of the factors that can influence the replacement process to get wise replacement decision is not an easy task because one factor in a particular situation or environment is more important than other environments. 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 the available cache available cache space, improve hit rates, reduce network traffic, and alleviate loads on the original server.

We explore the limitations of existing caching mechanisms in slow networks and propose a new model of web caching designed for developing regions called interactive caching. Unlike conventional caching, interactive caching makes interacting with the cache the focus of web browsing when the connection is bad. Interactive caching achieves this by organizing the cache into topics for presentation to the user, optimizing for latency, and unaliasing cached content [3].In this paper we proposed a novel approach to caching in developing regions with slow networks, which we call interactive caching. In our interactive caching model we maximize the utility of web caches in three general ways: organize the cache into topics and present these topics, optimize for latency, and alias content.

Proxy servers can reduce the response time in both Web and multimedia services. In this paper, we propose hierarchical proxy architecture for multimedia services. To analyse the impact of the proxy deployment in an operator network, we define an analytic model to compare proxy architectures in terms of the response time [4].

In this paper, we investigate the roles and operations of proxy servers in multimedia services. Then we propose hierarchical proxy architecture for multimedia services with two assumptions. In the first scenario, we assume that the proxy servers know the content cached with peers only. In the second scenario, we assume that the proxy servers know the content cached with others in the network. To analyse the impact of proxy deployment, we define an analytic model to compare proxy architectures in terms of the response time. Using this model, we analyse the service response time with and without proxy servers under different architectures for both multimedia and Web-based traffic. We show that the addition of more tiers to the proxy hierarchy reduce the response time. The hierarchical architecture with the distributed content list among all proxy servers records the best response time compared to other architectures. Our numerical results indicate that locating proxy servers close to the clients reduce the response time. The network without proxy servers performs well when the arrival rate is low. Operators always try to achieve the desired QoS to minimize their cost function. While the cost function could be used to make specific tradeoffs in a particular network deployment, here we show that there are some results that can be widely applicable in particular at the level in the network where caching should take place with proxy server

Hypertext transfer protocol (HTTP) defines three basic mechanisms for controlling caches: freshness, validation, and invalidation.

Freshness

That allows a response to be used without re-checking it on the origin server, and can be controlled by both the server and the client. For example, the Expires response header gives a date when the document becomes stale, and the Cache-Control: max-age directive tells the cache how many seconds the response is fresh for.

Validation

Can be used to check whether a cached response is still good after it becomes stale, For example, if the response

has a Last-Modified header, a cache can make a conditional request using the If-Modified-Since header to see if it has changed. The E-Tag (entity tag) mechanism also allows for both strong and weak validation.

Invalidation

Is usually a side effect of another request that passes through the cache, For example, if a URL associated with a cached response subsequently gets a POST, PUT or DELETE request, the cached response will be invalidated.

Due to the rapid growth in web a demand of networking resource and web services is increased. Users often experience long and unpredictable delays when retrieving web pages from remote sites. Therefore, a clear solution to improve the class of web services would be the increase of bandwidth, but such type of solution increases the cost of system. Hence we need to improve the performance of the cache and reduce the web traffic for improving the latency.

In this paper [5] author discuss a comprehensive approach to analyse web access pattern for user by using information present in the proxy server log files. Using this approach, we identified frequently requested web pages by users and integrate the prefetching scheme with the web caching to achieve performance improvement for the proxy server cache.

CATEGORIES OF PREFETCHING APPROACHES

Web prefetching has been extensively studied. Roughly speaking, major approaches fall in the following three categories: probability based, clustering based and using weight-functions.

Probability Based Prefetching

Web object prefetching approaches according to Probability, The central problem for Web prefetching are the prediction algorithm. When a request comes, a decision needs to be made on which page would mostly likely to be requested next time. Probability based prediction is a natural approach. Probabilities are calculated using the history access data. This method assumes that the request sequence follows a pattern (is not random) and the probabilities are trying to follow this pattern. One of the advantages for this approach is the number of pages prefetched can be controlled. Some data structure need for record probability for this purpose tree structure is use.

Clustering Based Prefetching

Web object prefetching approaches according to Clustering based prefetching methods make decisions using the information about the clusters containing pages that have been previous fetched, anticipating that pages that are "close" to those previously fetched pages are more likely to be requested in the near future. Support vector machine (SVM) is a data mining based classification algorithm. The method is to use hyper planes to separate data in different classes. This idea is adopted in to develop a SVM-based online learning algorithm to deal with the web prediction problem. This online learning algorithm is based on incremental chunk for LS-SVM (Least Square Support Vector Machines) classifier. The training of the LSSVM can be placed in a way of incremental chunk avoiding large scale matrix inverse but maintaining the precision when training and testing data. The online algorithm is especially useful for the large data set and practical applications where the data come in sequentially. The clustering based prefetching presented in effectively integrates caching and prefetching.

Weight-Function Based Prefetching

This model presented Probability Based Prefetching and Clustering Based Prefetching has been shown to be efficient, they only consider the request patterns and mainly the request probabilities. The cost of the network traffic and server workload as the overhead of the programs was not considered. To consider factors other than just the probabilities (such as size, priority), a function that involves multiple factors is needed. Several approaches in this direction have been proposed include the following:

(1) Web page size consideration
(2) Prediction by partial match model (PPM)
(3) Mining Web logs

The contribution of understanding the pattern and behavior of internet is beneficial to predict and prefetch web objects in caching. These web objects stand for new comer requests is used to decrease response latency and increase efficiency of source management. Integrating web caching and prefetching to provide the proxy with opulent information, a Web server may purposely send all possible prefetching clues with various levels of assurances to the proxy. Without any control, a proxy will prefetch every implied object into its cache, despite that the confidences of some prefetching rules may be low. In this case, a significant portion of the cache content will be replaced because a proxy may concurrently serve a large amount of client requests and each of these requests may trigger certain prefetching rules. As a result, the state of the cache content will become insecure and the cache hit ratio will drop sharply. On the contrary, if the prefetching control is over authoritarian, a proxy will tend to discard some beneficial hints provided by the Web server, thus shaping down the advantage of Web prefetching

In this paper, we present an application of web log mining to obtain web-document access patterns of closely related pages based on the analysis of the request from the proxy server log files. If we use the prefetching with caching then the performance of cache is improved. Prefetching fetches objects that are likely to be accessed in the near future and store them in advance thus the response time of the user request is reduce. Prefetching is based on the prediction of the caching and while predictive caching can optimize the WWW in many respects and improve the performance [6]. Therefore, in this paper web prefetching technique is evaluated and optimum methodology is suggested for developing new technique. In this paper, we present an application of web log mining to obtain web-document access patterns of closely related pages based on the analysis of the request from the proxy server log files.Using the proposed method, we identify frequently requested web pages by the users and integrate the pre-fetching scheme with the web caching to achieve performance improvement for the proxy server cache. In above discussed experiment, we could observe that there was performance improvement in terms of the Hit Ratio and the Byte Hit Ratio.

II. LITERATURE REVIEW

In this section of paper introduces various different recent efforts on web prefetching techniques. That provides us guidelines for developing a new and efficient system for web prefetching.

In this paper [7] author report an approach for prefetching the web prefetching the web pages for a user on the basis of his history of browsing by using sequential data mining technique. Proxy servers are the popular applications which are used to facilitate client for accessing web. These server help in reducing the network traffic and perceived lag by storing copies of web objects accessed in local temporary memory storage area, known as cache and to provide to other users the same page on demand. The cache of proxy server os limited therefore it needs replacement of pages to increase throughput. The overall performance of the cache using this approach for both page replacement namely LRU and LFU using prefetching has improved.

Sequential mining approach to mine frequent web access pattern from the raw log of Web server data. In this approach the pre-processed web logs are arranged in access sequence of individual user sequence resulting in access sequence database known as Web Access Sequence Database (WASD), so that sequential mining can be done on it. A Data structure known as WAP (Web Access Pattern) is devised to register access sequences and corresponding counts compactly for eliminating the expansive support count. WAP tree and also maintains linkages for traversing prefix with respect to the same suffix pattern efficiently. Then a graph known as WAP (Web Access Pattern) tree is constructed to mine the frequent subsequence of web access pattern for each user (Client IP) recursively, by scanning the WASD twice only. In first pass, it determines the set of frequent events and next scan WAP-mine builds a data structure, called WAP tree, using frequent events, to register all count events for further mining. Then WAP-mine recursively mines the WAP –tree using conditional search to find all the frequent Web Access Pattern.

Web caching is used to moderate the network traffic by caching (temporarily storing) web pages at the proxy server level. Nowadays caching alone is not satisfactory, because of World Wide Web has evolved rapidly simple information-sharing mechanism. This technique offers only static text and images to a rich variety of dynamic and collaborative services, such as e-commerce, video/audio conferencing, and distance learning programs. Therefore, Web is facing difficulties and need to improve the cache presentation. If we use the prefetching methodologies with caching schemes then the performance of cache may improved. Prefetching raises objects that are possible to be accessed or in the near future may be accessed. These schemes store them in advance by which the response time for user request to access a web page is reduced. In this paper, author's main aim is to provide a new framework for improving performance of web proxy server, using web usage mining and prefetching scheme. Further, they cluster the user data according to their access pattern and usage behaviour with the help of K-Means algorithm and then Apriori algorithm. The data mining algorithms are applied to generate rules for prefetching pages. This cluster based approach is applied on proxy server web access log data. To test the results using LRU and LFU prefetching schemes are compared with given system [8].

In this paper [9] Author presented an overview of k-means initialization methods with an emphasis on their computational efficiency. He then compared eight commonly used linear time initialization methods on a large and diverse collection of real and synthetic data sets using various performance criteria. Finally, then he analyzed the experimental results using non-parametric statistical tests and provided recommendations for practitioners. K-means is undoubtedly the most widely used partitioned clustering algorithm. Unfortunately, due to its gradient descent nature, this algorithm is highly sensitive to the initial placement of the cluster centers. Numerous initialization methods have been proposed to address this problem. Primary goals of clustering include gaining insight into data (detecting anomalies, identifying salient features, etc.), classifying data, and compressing data.

Web Usage Mining is the application of data mining techniques to discover interesting usage patterns from Web data, in order to understand and better serve the needs of Web-based applications.[10]Usage data captures the identity or origin of Web users along with their browsing behaviour at a Web site. Web usage mining itself can be classified further depending on the kind of usage data considered. They are web server data, application server data and application level data. Web server data correspond to the user logs that are collected at Web server. Some of the typical data collected at a Web server include IP addresses, page references, and access time of the users and is the main input to the present Research. This Research work concentrates on web usage mining and in particular focuses on discovering the web usage patterns of websites from the server log files.

The main goal of the proposed system [11] is to identify usage pattern from web log files of a website. Apriori and FP Growth Algorithm is used for this purpose. The main goal of the proposed system is to identify usage pattern from web log files of a website. For this purpose, the usage of apriori and FP Growth algorithms are proposed. Both are influential algorithms for mining frequent item sets for boolean association rules The present Research work is designed to operate on log files.

The aprori algorithm attempts to find subsets which are common to at least a minimum number C (the cut off, or confidence threshold) of the item sets. The system operates in the following three modules. Pre-processing module

- Apriori or FP Growth Algorithm Module
- Association Rule Generation
- Results

The FP Growth algorithm operates in the following four modules.

- Preprocessing module
- FP Tree an FP Growth Module
- Association Rule Generation
- Results

Web caching and web prefetching are the two major areas of research. These researches intensive to reduce the

user observed latency. Web caching helps in manipulating temporal latency however web prefetching supports in take advantage of spatial latency. Conversely, if perfected pages are not accessed by the users in their future accesses. This can increase the network traffic and overload the web server. This paper aims at surveying various researches, which have make effort in this direction. Prefetching techniques are applied to control the network traffic and server load, in order to improve the user satisfaction [12]. A large number of researches have been performed in this area of web prefetching in different directions. Finally the article is concluded and found that the Markov model and prediction by partial matching based approaches consumes memory proficiently. Additionally, Double dependency graph based techniques mainly emphases on the network traffic and server load. And finally, Sequential mining, clustering and content based approaches principally attention on the precision and Compression algorithm based approach considers both the precision and memory efficiency

There are two communicating parties client and server. At client side browser to access the web and cache to store the web document are available. At server side Markov model and prediction algorithm are available. At the server side the Markov model will be constructed with the help of web log. The prediction algorithm will predict the most probable next access web page.

Caching is a temporary memory location where web page data is stored. Two types of cache is used here Regular cache in which lot of number of web pages can be stored and the prefetch cache in which only one web page can be stored which is predicted one. With the help of the Markov the next accessed web page will be predicted and prefetched for caching. If predicted page is same as the user's requirement then it is provided to the web user from cache for providing the fast access of web page and to reduce the server load.

Web page access prediction gained its importance from the ever increasing number of e-commerce Web information systems and e-businesses. Web page prediction, that involves personalising the Web users' browsing experiences, assists Web masters in the improvement of the Web site structure and helps Web users in navigating the site and accessing the information they need. The most widely used approach for this purpose is the pattern discovery process of Web usage mining that entails many techniques like Markov model, association rules and clustering. Implementing pattern discovery techniques as such helps predict the next page to be accessed by the Web user based on the user's previous browsing patterns. However, each of the aforementioned techniques has its own limitations, especially when it comes to accuracy and space complexity. This dissertation achieves better accuracy as well as less state space complexity and rules generated by performing the following combinations.

We explore the limitations of existing caching mechanisms in slow networks and propose a new model of web caching designed for developing regions called interactive caching. Unlike conventional caching, interactive caching makes interacting with the cache the focus of web browsing when the connection is bad. Interactive caching achieves this by organizing the cache into topics for presentation to the user, optimizing for latency, and unaliasing cached content. In this paper we implement a prototypical version of interactive caching that includes: topic identification and presentation, a latency aware value function, DNS caching, and missing hyperlink suggestions. We evaluate our system based on a system implementation and web traces from multiple web cache deployments across different geographic locations in developing regions.

This paper [13] presents the novel architecture will provide an effective solution to the inherent limitation (initial request to any content is served lately) of traditional proxy caching approach with the effective usage of web prefetching techniques. With the fast development of internet and associated technologies, amount of regular internet users and amount of data being accessed are increasing exponentially. Internet users now access internet to find HD and 3D videos rather than static web pages. Due to the reasonably slow growth in bandwidth, the end-user seeming latency emerges as a significant drawback in internet. Methods like web caching and web prefetching have been employed successfully in order to reduce this latency. Content Delivery Networks (CDNs) provide the advantage of better performance and high availability through a network of distributed servers hosting a large portion of the internet content by off loading traffic directly. These contents served from the content provider's infrastructure. In this paper, the authors attempt to combine these three different techniques namely; prefetching, caching and the CDN approaches to reduce the user perceived latency [13].

CDNs have emerged as another technique often used by the content provider to host the content in a globally distributed set of servers known as CDN surrogates, so that the end user will automatically be redirected to the closest server for faster service. The CDNs have evolved as a service provided by a third party to the content provider, in order to improve scalability and performance in delivering the content. However, a CDN node chosen by the content provider as closest or best suited for a particular client may not always deliver content as fast as expected. This is because all CDN nodes exist outside the LAN of any particular user. Therefore, the internet connection speed often becomes the bottleneck, even for content hosted on CDNs.

This paper [14] focuses on improving the runtime performance by applying data mining models to make predictions, specifically using gradient-boosted regression trees. That technique is basically used for learning to rank. Even though conceptually most implementations of tree-based models do not efficiently utilize modern superscalar processors. This survey identifies over 50 algorithms including approaches that are direct adaptations of accuracy based methods, use genetic algorithms, use anytime methods and utilize boosting and bagging. The survey brings together these different studies and novel approaches to cost-sensitive decision tree learning, provides a useful taxonomy, a historical timeline of how the field has developed and should provide a useful reference point for

future research in this field. Decision trees are a natural way of presenting a decision-making process, because they are simple and easy for anyone to understand. Learning decision trees from data however is more complex, with most methods based on an algorithm, known as ID3.

By laying out data structures in memory in a more cache- conscious fashion, removing branches from the execution flow using a technique called predication, and micro-batching predictions using a technique called vectorization. Author here strongly argued that the given techniques able to better achieve with modern processor architectures. Experiments on mock data and on three standard learning-to-rank datasets demonstrate that given approach is effectively faster than standard techniques [15].

In this paper, machine learning techniques are used to improve the performances of traditional Web proxy caching schemes like Least-Recently-Used, Greedy-Dual-Size and Greedy-Dual-Size- Frequency. A support vector machine and C4.5decision tree are intelligently combined with traditional Web proxy caching method to create intelligent caching technique known as SVM–LRU, SVM–GDSF and C4.5–GDS. The proposed intelligent methods are calculated by trace-driven simulation and compared with the most relevant Web proxy caching polices. Investigational results have discovered that the given SVM–LRU, SVM–GDSF and C4.5–GDS significantly enhance the performances of LRU, GDSF and GDS correspondingly [16].

Web proxy caching plays a key role in improving Web performance by keeping Web objects that are likely to be visited again in the proxy server close to the user. This Web proxy caching helps in reducing user perceived latency, i.e. delay from the time a request is issued until response is received, reducing network bandwidth utilization, and alleviating loads on the original servers. Since the space apportioned to a cache is limited, the space must be utilized effectively. Therefore, an intelligent mechanism is required to manage Web cache content efficiently. The cache replacement is the core or heart of Web caching. Thus, the design of efficient cache replacement algorithms is extremely important and crucial for caching mechanism achievement. Cache pollution means that a cache contains objects that are not frequently visited. This reduces the effective cache size and affects the performance of the Web proxy caching negatively.

In a Web proxy server, Web proxy log files record the activities of the users and can be considered to contain complete and prior knowledge of future accesses. The availability of Web proxy log files that can be exploited as training data is the main motivation for utilizing machine learning techniques in adopting intelligent Web caching approaches. The second motivation is that an efficient and adaptive scheme is required for the Web environment, which changes and updates rapidly and continuously. The machine learning techniques can adapt to some important changes throughout the training phase.

Recent studies have proposed exploiting machine learning techniques to cope with the above problem. Most of these studies utilize an artificial neural network (ANN) in Web proxy caching, although ANN training may consume considerable amounts of time and require extra computational overheads. More importantly, the integration of intelligent techniques in Web cache replacement is still being researched.

Support vector machine (SVM) and decision tree (C4.5) are two popular supervised learning algorithms that perform classifications more accurately and faster than other algorithms. These machine learning algorithms have a wide range of applications such as text classification, Web page classification and bioinformatics applications. Hence, SVM and C4.5 can be utilized to produce promising solutions for Web proxy caching.

Rapid growth of web application has increased the researcher's interests in this era. There is a very useful application called web applications are used for the communication and data transfer. An application that is accessed via a web browser over a network is called the web application. Web caching is a well-known approach for improving the performance of Web based system. By keeping Web objects that are likely to be used in the near future in location closer to user. The Web caching mechanisms are implemented at three levels: client level, proxy level and original server level. Proxy servers play the main roles between users and web sites in reducing the response time of user requests and preserving of network bandwidth. Therefore, to obtaining better response time, an efficient caching technique should be created over the proxy server. This paper uses FP growth, weighted rule mining concept and Markov model for fast and frequent web pre fetching. The obtained results indicate the improved the hit ratio of the web page and accelerates users visiting speed [17].

Association Rule Mining is a data mining technique that has been used to discover related transactions. ARM finds relationships among item based on their co-occurrence in the transactions. Specifically, ARM focuses on associations among frequent item sets. For example, in a supermarket store, ARM helps uncover items purchased together which can be utilized for helving and ordering processes. In the following, we briefly present how we apply ARM in WPP. For more details and background about ARM, In WPP, prediction is conducted according to the association rules that satisfy certain support and confidence as follows. For each rule, $Z = X \rightarrow Y$, of the implication, X denotes user session and Y is the target destination page. Prediction is resolved as follows:

$$\text{prediction } (X \rightarrow Y) = arg\,max \frac{supp(X \cup Y)}{supp(X)}, X \cap Y = \phi$$

Note that the cardinality of Y can be greater than one, i.e. prediction can resolve to more than one page. Moreover, setting the minimum support plays an important role in deciding a prediction. In order to mitigate the problem of no support for $X \cup Y$, we can compute prediction $(X \rightarrow Y)$, where X is the item set of the original session after trimming the first page in the session. This process is very similar to the all-K^{th} Markov model. However, unlike in the all-K^{th} Markov model, in ARM, we do not generate several models for each separate N-gram. In the following sections, we will refer to this processes all-K^{th} ARM model.

III. WORK REVIEW

In this section presents review on the studied models, in addition of that issues that are observed and solution that are appropriate for optimize the system is also discussed. The observed facts of different research papers and articles are concluded as.

1. To improve performance of any kind of service required to add new resources.
2. Resource implementation is a complex and expensive task.
3. Discovery of new techniques that optimize resource consumption and resource management.
4. Design new techniques to improve performance of current domain, using more than one method.
5. Use of optimized and hybrid methods to improve the performance of current system

In this paper [18], a novel method Closed Sequence-Sequence Generator Mining (CSGM) is proposed to generate closed sequences and sequence generators for no redundant sequential rule mining. By applying the proposed method on web logs, we can extract sequential associations among products which reflect users' preference on products. This method could extract non redundant sequential rules for user's initial query expansion. This work reduces the redundant sequential rules in the way of mining sequential rules from closed sequential patterns and sequential generators instead of frequent sequential patterns. The method performs significant redundancy and time-cost reduction.

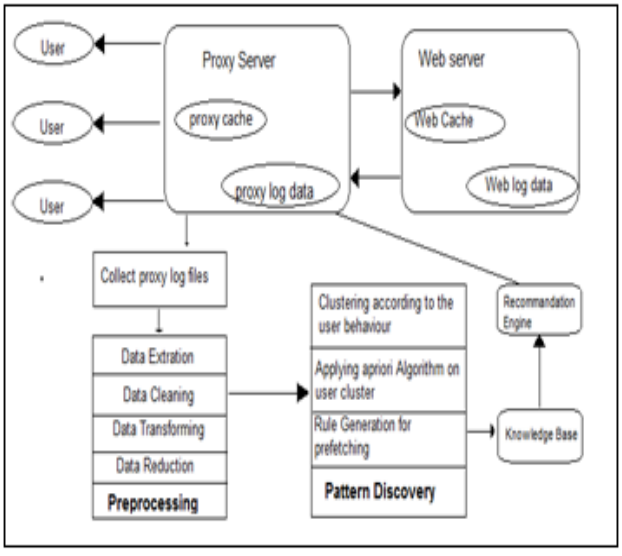

Figure 2 Given model in [2]

In order to find more effective and promising technique search for enhancing the web pre-fetching performance various research papers are studied and an efficient and promising model [18] is found in. According to the given results, this model is more appropriate for enhancing performance of web prefetching schemes. The given model is demonstrated using figure 1. Where only one deficiency is observed that is K-mean clustering implementation in this model.

Clustering is an interesting domain, that group similar data according to their property. The proposed study is based on previously made effort and improvements over traditional K-mean clustering scheme. The selected algorithm K-mean has some deficiencies i.e. [19]

1. **Handling Empty Clusters:** One of the key issues with the basic K-means algorithm given previously is empty clusters. The empty cluster problem can be found if no points are assigned to a cluster during the assignment step of cluster computing. If this situation occurred, then an approach is required to find a replacement of centroid, otherwise, the mean squared error will be larger than its requirement.
2. **Outliers:** When outliers are exist in training data, the resulting cluster centroids (prototypes) may not be as demonstrative as they are otherwise would be and thus, the Sum of Squares for Error will be higher as well.
3. **Reducing the SSE with Post processing:** In k-means to get better clustering. Therefore required to reduce the Sum of Squares for Error (SSE) that is most difficult and computationally complex task. There are various types of clustering methods available which reduces the SSE.

To overcome the above discussed problems with k-mean clustering schemes. An appropriate and efficient methods are suggested in [8] and in [19]. To obtain high efficient results here solution steps are provided in further sections.

IV. CLUSTERING STUDY

In this paper [20], author concluded that partitioning based clustering methods are suitable for spherical shaped clusters in small to medium sized data sets. K-means and k-medoids – both the methods find out clusters from the given database. Both the methods require to specify k, no of desired clusters, in advance. Result and runtime depends upon initial partition for both of these methods. The advantage of k-means is its low computation cost, while drawback is sensitivity to noisy data and outliers. Compared to this, k-medoid is not sensitive to noisy data and outliers, but it has high computation cost.

Clustering is an essential task in Data Mining process which is used for the purpose to make groups or clusters of the given data set based on the similarity between them. K-Means clustering is a clustering method in which the given data set is divided into K number of clusters. This paper is intended to give the introduction about K-means clustering and its algorithm. The experimental results of K-means clustering and its performance in case of execution time are discussed here. But there are certain limitations in K-means clustering algorithm such as it takes more time for execution. [19]This paper conclude that increasing efficiency of k mean algorithm and Users find better results corresponding to queries and Execution time also decreased.

This section involves the classical K-mean clustering algorithm and in addition of that an enhanced algorithm that used to improve performance of the clustering.

An improved k-means clustering algorithm based on K-MEANS algorithm is proposed. This paper [22] gives an improved traditional algorithm by analyzing the statistical data. After a comparison between the actual data and the simulation data, this paper safely shows that the improved algorithm significantly reduce classification error on the simulation data set and the quality of the improved algorithm is much better than K-MEANS algorithm. Such comparative results confirm that the improved algorithm is a powerful for this problem.

A. K-MEANS CLUSTERING ALGORITHM

The K-Means clustering algorithm is a partition-based cluster analysis method [21]. According to the algorithm required to select k objects as initial cluster centres, then after the distance between each object and each cluster center are calculated. According to the calculated distance centers are assign it to the nearest cluster, update the averages of all clusters. This process is repeated until the criterion function converged.

Input: N objects to be cluster (x_1, x_2 … x_n), the number of clusters k;
Output: k clusters and the sum of dissimilarity between each object and its nearest cluster center is the smallest;
1. Arbitrarily select k objects as initial cluster centers$(y_1, y_2, \dots, y_k)$; 2. Calculate the distance between each object Xi and each cluster center, after that assign each object to the nearest cluster, formula for calculating distance is given as: $d(x_i, y_i) = \sqrt{\sum_{j=1}^{d} (x_i - y_{j1})^2}, i = 1 \dots N, j = 1 \dots k$ $d(x_i, y_i)$is the distance between data i and cluster j. 3. Calculate the mean of objects in each cluster as the new cluster centers, $y_i = \frac{1}{N} \sum_{j-1}^{n_i} x_{ij}, i = 1,2, \dots, K$ N_i is the number of samples of current cluster i; 4. Repeat step 2 and 3 until the criterion function E converged, 5. Return$(y_1, y_2, \dots, y_k)$ Algorithm terminates.

The calculation of Square error criterion for clustering

$$E = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left\| x_{ij} - y_i \right\|^2$$

Where $x_{ij}$is the sample j[th] of i[th]-class, and $y_i$is the cluster center of i[th]-class, n_i is the number of samples of i[th]-class. K-means clustering algorithm is simply described as

The traditional K-mean clustering is given in this section, in addition of that the enhanced and improved K-mean clustering approach that promises to produce optimum clusters.

The strengths and weaknesses of this algorithm are mentioned as below:-

Strengths:

- More robust than k-means in the presence of noise and outliers; because a medoid is less influenced by outliers or other extreme values than a mean.

Weaknesses:

- Relatively more costly; complexity is O(i k (n-k)2), where i is the total number of iterations, is the total number of clusters, and n is the total number of objects.
- Relatively not so much efficient.
- Need to specify k, the total number of clusters in advance.
- Result and total run time depends upon initial partition.

LIMITATIONS OF K-MEANS ALGORITHM

Even after being so popular algorithm for clustering, K-Means Algorithm suffers from two major limitations

1. It is computationally very expensive as it involves several distance calculations of each data point from all the centroids in each iteration.
2. The final cluster results heavily depend on the selection of initial centroids which causes it to converge at local optimum.

B. DENSITY -BASED OUTLIER DETECTION

The K-Means algorithm is very sensitive to select the initial cluster centres [23]. Therefore, the clustering results very different outcomes from initial cluster centers. If the data point's presents in separation, mean to say a small amount of data points are far away from data-intensive areas. The calculation of the mean point would be affected and the new cluster center may deviate from the true data-intensive areas. That ultimately leads to a clustering output result of large deviation, therefore first required to remove isolation point in Data set before initiating data clustering.

The abnormal degree of each object in data set is estimated by local outlier factor (LOF). Local outlier factor first generates k-Neighbourhood and k-nearest neighbour distance of all objects, and then calculates the distance between each object. The objects that are in its k-Neighbourhood at last LOF identifies local outlier according to the local outlier factor of each object [25]. The technique of outlier detection is briefly described as follows:

1. Calculate k-nearest neighbourhood distance named distance $(P, i)(i \in N_k(P))$ of each object p, distance (p, i) is defined as the direct connection distance between object p and i,

$$distance = \sqrt{(x^1 - y^1)^2 + (x^2 - y^2)^2 + \cdots + (x^n - y^n)^2}$$

Where n is dimension of dataset

2. Calculate the density of each object p. The density of object p which reflects the distribution of the data near is defined by the reciprocal of k-nearest neighbour mean. It is described as follows:

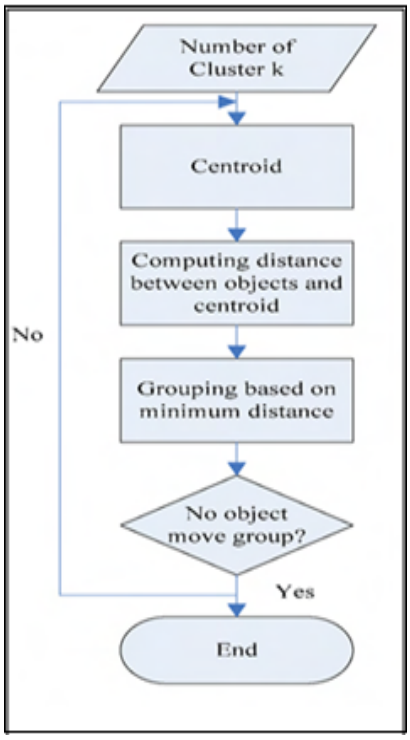

Figure 2 The k-mean algorithm process

$$lrd(p) = \frac{1}{\frac{1}{K}\sum_{i=1}^{K} distance(p,i)}$$

3. Calculate the local outlier factor of p.

$$lof(p) = \frac{\sum_{i=1}^{k} \frac{lrd(i)}{lrd(p)}}{k}$$

lrd (i) is the local density of k-nearest neighbour of p, lrd (p) is the local density of p. lof (p) reflects the extent of p as a outlier. The value of outlier factor is about I in the data set of density distribution. As the local density of the outliers in the data set is much less than the local density of its neighbor hood, outlier factor by which outlier can be distinguished is larger than others.

4. If lof (p) is much larger than 1, P is an isolated point, delete the object p, return the first step until the number of data sets remains unchanged. The new data sets generated is the data set to be clustered.

The K-Means Algorithm suffers from the two major limitations of being computationally very expensive as it involves several distance calculations of each data point from all the centroids in each iteration and secondly the final cluster results heavily depends on the selection of initial centroids which causes it to converge at local optimum. This paper [26] presents the way to find the initial centres for the k-means so that every time the K-Means Algorithm produces same result for the same dataset and remove the second limitation of K-Means of producing different clustering results with different initial centroids.

The proposed method in this paper [27] ensures the entire process of clustering in O(nk) time without sacrificing the accuracy of clusters. Experimental results show the improved algorithm can improve the execution time of k-means algorithm. So the proposed k-means method is feasible. This paper elaborates k-means algorithm and analyses the short comings of the standard k-means clustering algorithm. Because the computational complexity of the standard k-means algorithm is objectionably high owing to the need to reassign the data points a number of times during every iteration, which maks the efficiency of standard k-means clustering is not high. This paper presents a simple and efficient way for assigning data points to clusters.

V. CONCLUSIONS

This paper presents study about web pre-fetching and caching technique. In order to improve the working of cache performance various prefetching models are recently developed. Some of them are employed at intermediate proxy servers and some of them are works at the client end. These web pre-fetching and caching techniques are help to improve the content delivery. In such direction an essential contribution is given in a web pre-fetching model. The given pre-fetching technique is much promising and efficient according to obtained results. But, the given pre-fetching system contains some limitations. To overcome the listed issues in previous model, K-mean clustering algorithm is helpful to improve. An improved version of k-means algorithm in also introduces in this paper. In near future, it is suitable and efficient to use model for web prefetching on the proxy servers. Additionally, comparative performance is produces.

REFERENCES

[1] MukeshDawar, Charanjit Singh, "A Review on Web Caching Techniques",International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 4, Issue 3, March 2014

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

[3] Jay Chen, LakshmiSubramaniam "Interactive Web Caching for Slow or Intermittent Networks" ,ACM DEV 4, December , 2013

[4] NamalKarunarathna, GyuMyoung Lee, Anbin Kim, Seong-Ho Jeong "Performance Evaluation of Hierarchical Proxy Servers for Multimedia Services"

[5] Suvarnatemgire, poonam Gupta, "Review on web prefetching techniques", IJTEEE,2013

[6] Mahesh Manchanda, Dr. Neena Gupta," Make web page instant: by integrating Web-cache and Web-prefetching",CAC2S 2013.

[7] Abhaysingh, anilkumarsingh, Web prefetching at proxy server using Sequential data mining", 3rd ICCCT,2012.

[8] Nanhay Singh, ArvindPanwar, and Ram Shringar Raw, "Enhancing the Performance of Web Proxy Server through Cluster Based Prefetching Techniques ", 978-1-4673-6217-7/13/$31.00c 1158, IEEE 2013

[9] M. EmreCelebi, Hassan A. Kingravi, Patricio A. Vela," A Comparative Study of Efficient Initialization Methods for the K-Means Clustering Algorithm" Expert Systems with Applications, 40(1): 200–210, 2013

[10] Sonia Setia, Dr. Jyoti, Dr. NeelamDuhan, "Survey of Recent Web Prefetching Techniques", International Journal of Research in

Computer and Communication Technology, Vol 2, Issue 12, December- 2013

[11] B.Santhosh Kumar , K.V.Rukmani, Implementation of Web Usage Mining Using APRIORI and FP Growth Algorithms

[12] MahendrapratapYadav, MhdFeroz, Vinod Kumar Yadav, " Mining customer behavior using web usage mining in e-commerce, IEEE 20180, ICCCNT 2012

[13] L.R.Ariyasinghe, C.Wickramasinghe, P.M.A.B Samarakoon, U.B.P Perera, R.A.PrabhathBuddhika, M.N.Wijesundara, "Distributed Local Area Content Delivery Approach with Heuristic based Web Prefetching", Computer Science & Education (ICCSE), IEEE 2013 8th International Conference onColombo

[14] SUSAN LOMAX, SUNIL VADERA "A Survey of Cost-Sensitive Decision Tree Induction Algorithms"2012

[15] NimaAsadi, Jimmy Lin, Arjen, P. de Vries, "Runtime Optimizations for Tree-Based Machine Learning Models", This is the preprint of an article that has been accepted for publication April, 2013

[16] Waleed Ali, SitiMariyamShamsuddin, Abdul Samad Ismail, "Intelligent Web proxy caching approaches based on machine learning techniques",Published by Elsevier B.V. All rights reserved. doi:10.1016/j.dss.2012.04.011,0167-9236/$ – see front matter. Crown Copyright © 2012

[17] Ms. Veena Singh Bhadauriya, Dr. BhupeshGour, Dr. AsifUllah Khan, "A weighted Markov model for web prefetching to improve user interface over internet",ISSN : 0975-0290, Int. J. Advanced Networking and Applications Volume: 05, Issue: 03, Pages:1962-1967 2013

[18] RaheleBehbahani, Ali MahaniNon-Redundant "Sequential Association Rule Mining and Application in Recommender Systems".

[19] An improved K-Means clustering algorithm,JuntaoWang,Xiaolong Su, 978-1-61284-486-2/111$26.00 ©2011 IEEE

[20] Shalini S Singh, N C Chauhan, "K-means v/s K-medoids: A Comparative Study", National Conference on Recent Trends in Engineering & Technology,2011.

[21] ManpreetKaur, UsvirKaur"A Survey on Clustering Principles with K-means Clustering Algorithm Using Different Methods in Detail", IJCSM, 2013.

[22] Huang Xiuchang, SU Wei,"An Improved K-means Clustering Algorithm"JOURNAL OF NETWORKS, VOL. 9, NO. 1, JANUARY 2014

[23] Prasanta Gogoi1, D K Bhattacharyya1, B Borah1 and, Jugal K Kalita, "A Survey of Outlier Detection Methods in Network Anomaly Identification",

[24] JaideepVaidya, Chris Clifton,"PrivacyPreservingKMeans Clustering over Vertically Partitioned Data",2003 ACM 1581137370/03/0008

[25] Marieke E. Timmerman & Eva Ceulemans& Kim De Roover& Karla Van Leeuwen, "Subspace K-means clustering",Published online: 23 March 2013 # Psychonomic Society, Inc. 2013

[26] NehaAggarwal, KirtiAggarwal " A Mid – Point based k-mean Clustering Algorithm for Data mining", IJCSE,2012

[27] Guan yong, Liu Xumin , Guan yong , "Research on k-means Clustering AlgorithmIEEE 2010