Enhancement of Web Proxy Caching Using Simple k-Means Clustering

Julian Benadit.P,Sagayaraj Francis.F, Nadhiya.M

Abstract— The Simple k-Means is an ensemble learning method for Web data clustering. In this study, we attempt to improve the performance of the traditional Web proxy cache replacement policies such as LRU and GDSF by integrating machine learning technique for enhance the performance of the Web proxy cache. Web proxy caches are used to improve performance of the web. Web proxy cache reduces both network traffic and response time. In the first part of this paper , clustering method as the simple k-Means cluster to learn from proxy log data and predict then group the classes of objects to be revisited or not. In second part, a simple k-Means is incorporated with traditional Web proxy caching policies to form novel caching approaches known as k-Means-LRU and k-Means-GDSF. These proposed k-Means-LRU and k-Means-GDSF significantly improve the performances of LRU and GDSF respectively.

Index Terms— Cache replacement, Clustering, k-Means cluster, Integration, Prediction, Proxy server, Web caching.

1 Introduction

For the past few years many researches are going on in Web proxy caching and integration of supervised techniques in Web cache replacement. This paper also comes under this category. Web proxy caching plays a significant part in improving Web performance by conversing web objects that are likely to be visited again in the proxy server close to the user. This internet proxy caching aids in decreasing user perceived latency, i.e. delay from the time missive of request is issued till response is received, reducing network information measure[4],[15].

Cache space is restricted; the space should be uses competently. A cache replacement principle is required handle the cache content [4,][11]. If the cache is full when an object desires to be stored, the replacement strategy will work out which objects to be evicted to permit space for the new object.

TABLE 1
CACHE REPLACEMENT POLICIES

Policy	Brief description
LRU	The least recently used objects are taken first.
LFU	The least frequently utilized objects are taken first.
SIZE	Big objects are removed first.
GDS	It assigns a key value to each object in the cache. the object with the low key value is evicted .
GDSF	It expands GDS algorithm by integrating the frequency component into the key worth.

- *Julian Benadit .P is currently pursuing Ph.D program in computer science Engineering , Pondicherry Engineering college, Pondicherry University,Pondicherry,INDIA. E-mail: benaditjulian@gmail.com*
- *Sagayraj Francis.F, Professor ,Department of computer science Engineering,Pondicherry Engineering college , Pondicherry ,INDIA. E-mail: fsfrancis @pec.edu*
- Nadhiya.M , PG scholar, Dr.SJS Paul Memorial college of Engg&Technology, Pondicherry University, INDIA E-mail:nadhiya.1241@gmail.com

The most common internet caching ways (Table 1) aren't effective enough and flout alternative factors that aren't often visited. This decreases the effective cache size and affects the performance of the online proxy caching negatively. Therefore, a supervised mechanism is needed to manage internet cache content with efficiency. In preceding papers exploiting supervised learning methods to cope with the matter [1],[6],[7],[9],[10],[12],[15]. Most of these surveys use an adaptive neuro-fuzzy inference system (ANFIS) in World Wide Web caching. Though ANFIS training might consume wide amounts of time and need further process overheads.

In this paper, we attempted to increase the performance of the web cache replacement strategies by integrating Clustering method of simple k-Means Clustering. In conclusion, we achieved a large-scale evaluation with other supervised learning algorithm on different log files and the proposed methodology has enhanced the performance of the web proxy cache.

2 Simple K-Means Clustering

Simple k-Means clustering is popular machine learning method. The k-Means algorithm clusters N data point into k disjoint groups, where k is a predefined fator. The steps in the simple k-Means clustering-based log data prediction method [9] are as follows:

1.Select k random instances from the training data subclass as the centroids of the clusters $C_1, C_2, \ldots . C_k$.

2. For every training instance X.

$D(C_i, X), i = 1 \ldots k$

a. Calculate the Euclidean distance

Discover cluster that is closest to X.

b. Assign X to Cq. Update the centroid of Cq. (The centroid of a group is the arithmetic mean of the instances in the cluster).

3. Repeat Step 2 until the centroids of clusters $C_1, C_2, \ldots . C_k$ stabilize in terms of mean-squared-error criterion.

4. For each test instance Z:

a. Compute the Euclidean distance $D(C_i,Z), i = 1 \dots k$. Find cluster C_r that is closest to Z.
b. Classify Z as an revited or not visited instance using either the Threshold rule or the Bayes Decision rule. The threshold rule for classifying a test instance Z that belongs to cluster C_r is:
Assign Z →1 if P (W_{0r}| Z ∈ C_r) > T
Otherwise Z →0,
Where "0" and "1" represent revisited,not visited classes, W_{0r} represents the not visited class in cluster C_r, P (W_{0r}| Z ∈ C_r) represents the probability of not requested instance in C_r and T is a predefined Threshold. In our Experiments, the threshold is set to 0.5 so that a test instance is classified as not visited only if it belongs to a cluster that has not visited instancesin majority. The Bayes Decision rule is:
Assign Z →1 if P (W_{0r}| Z ∈ C_r) > P (W_{1r}| Z ∈ C_r);
Otherwise Z →0,
Where W_{1r} represents the class in cluster C_r and P (W_{1r}| Z ∈ C_r) is the probability of revisited instances in cluster C_r.

3 Proposed Novel Web Proxy Caching Approaches

The proposed system will present a framework (Fig. 1) for novel Web proxy caching approaches based on machine learning techniques [2],[5],[19].

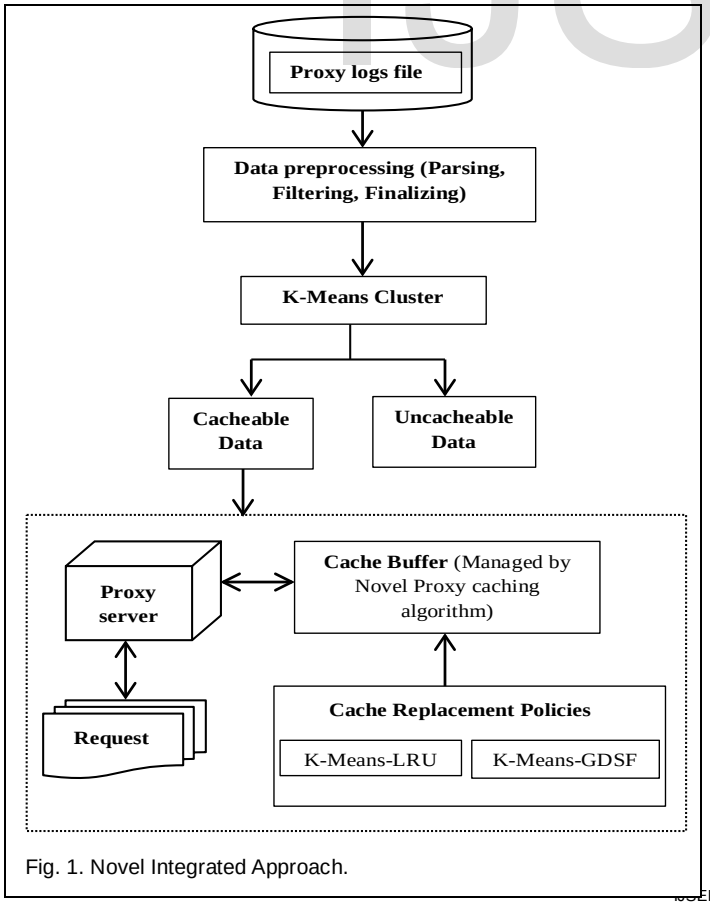

Fig. 1. Novel Integrated Approach.

In first part, once the dataset is prepared, the machine learning techniques are trained depending on the concluded dataset to order the web objects into objects that may be revisited or not. In second part, we present novel Web proxy caching approaches which depend on integrating supervised techniques with traditional Web caching algorithms

3.1 K-Means-GDSF

The main advantage of the GDSF [16] principle is that it executes well in terms of the hit ratio. However, the byte hit ratio of GDSF principle is too reduced. Thus, the k-Means clustering is integrated with GDSF for advancing the performance in terms of the byte hit ratio of GDSF. The suggested novel proxy caching approach is called k-Means-GDSF

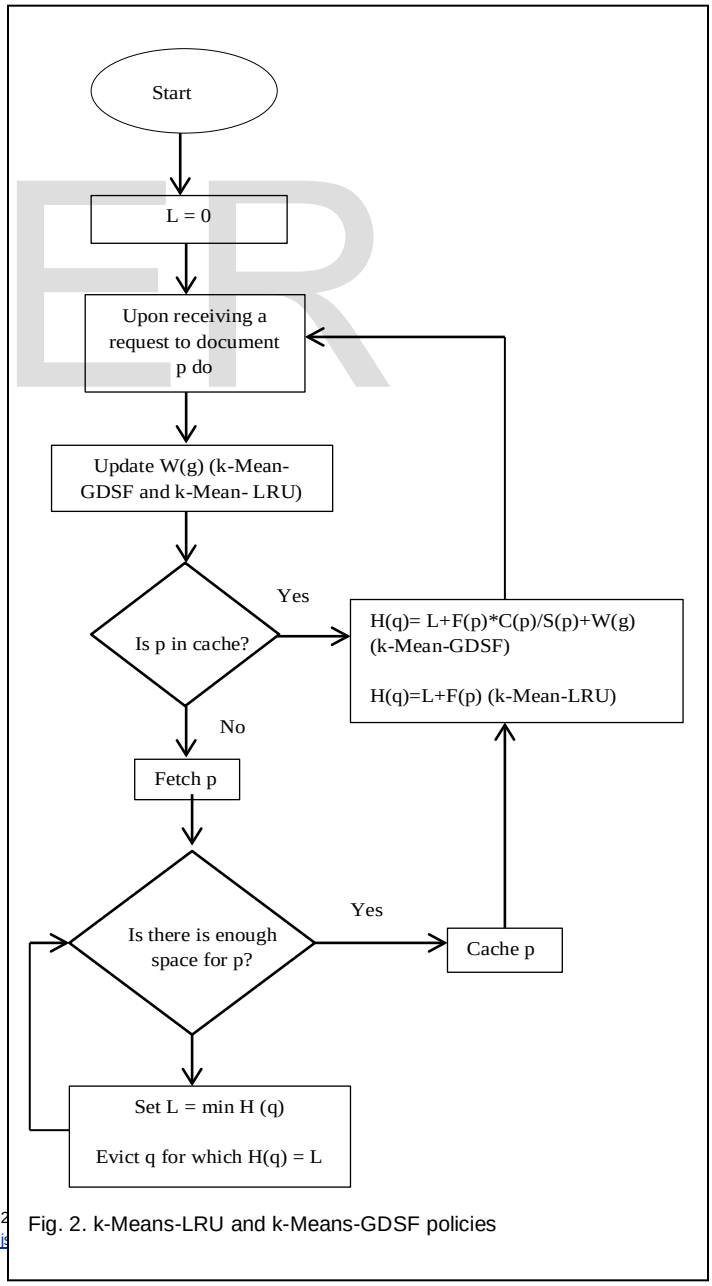

Fig. 2. k-Means-LRU and k-Means-GDSF policies

In k-Means-GDSF, a trained k-Means clustering is used to predict the classes of web objects either objects may be re-visited later or not. After this, the clustering assessment is integrated into cache replacement policy (GDSF) to give a key value for each object in the cache buffer; the lowest values are removed first. The proposed k-Means-GDSF illustrated Fig. 2.

3.2 K-Means-LRU

LRU policy [18] is the most common web proxy caching scheme among all the Web proxy caching algorithms [1],[9]. But, LRU policy suffers from cache pollution, which means that unpopular data's will remain in the cache for a long period. For reducing cache pollution in LRU, a k-Means cluster is joint with LRU to form a novel approach (Fig. 2) called k-Means-LRU.

TABLE 2
DIFFERENT PROXY LOG FILES

Proxy Data set	Proxy server name	Location	Duration of Collection
UC	uc.us.ircache.net	Urbana-Champaign,	1/8-4/9/2011
BO2	bo2.us.ircache.nt	Boulder-Colorado,	1/8-4/9/2011
SV	sv.us.ircache.net	Silicon, Valley,	1/8-4/9/2011
SD	sd.us.ircache.net	San Diego,	1/8-4/8/2011
NY	ny.us.ircache.net	New York	1/8-4/9/2011

An access proxy log entry generally consists of the consequent fields: timestamp, lapsed time, log tag, message protocol code, size, user identification, request approach, URL, hierarchy documents and hostname, and content type

4 EXPERIMENTAL RESULT

4.1 Data Pre-processing

In the data pre-processing [14], irrelevant and not valid request are removed from the logs proxy files. The pre-processing, including parsing, filtering and finalizing, has a strong influence on the performance; therefore, a correct preparation is required in order to achieve results reflecting the behavior of the algorithms. After the pre-processing, the final format of our data consist of URL ID, timestamp, lapsed time, size and set of Mesh data (type) as shown in Table 3.

TABLE 3
SAMPLE OF PRE-PROCESSED DATA

URL id	Timestamp	Lapsed time	size	Type
1	1082348905.73	53	43097	Application
2	1082348907.41	703	14179	Application
3	1082348908.47	284	1276	image/jpeg
4	1082349578.75	263	25812	image/jpeg
1	1082349661.61	71	43097	application
5	1082349675.35	203	8592	text/html
6	1082349688.90	231	24196	text/html
4	1082349753.72	875	25812	text/html
4	1082350464.01	173	25812	text/html
1	1082351887.76	115	43097	application
4	1082352609.09	35	25812	text/html
1	1082352861.56	311	43097	application

4.2 Training Phase

The training datasets are prepared; the desired characteristics of Web objects are extracted from pre-processed proxy logs files. These features comprise of URL id, timestamp, lapsed time, size and category of Web object (type).

Consequently, these features are transformed to the input and output dataset or training forms in the format < a_1, a_2, a_3, a_4, a_5, a_6, b>. a_1 is recency of mesh data based on sliding window, a_2 is frequency of mesh data, a_3 is frequency of mesh data based on sliding window, a_4 is retrieval time of mesh data a_5 is size of mesh data, a_6 is category of mesh data. a_1… a_6 represent the inputs and b represents the output of the requested mesh data. a_1 and a_3 are extracted based on sliding window. The sliding window of a request is that the period afore and later once the demand was created

TABLE 4
SAMPLE OF TRAINING DATASETS

Inputs						
Recency	Frequency	SWL frequency	Retrieve time	size	Type	output
900	1	1	53	43097	5	1
900	1	1	703	14179	5	0
900	1	1	284	1276	2	0
900	1	1	263	25812	2	1
900	2	2	71	43097	5	0
900	1	1	203	8592	1	0
900	1	1	231	24196	1	0
900	2	2	875	25812	1	1
900	3	3	173	25812	1	0
1226.15	3	1	115	43097	5	1
1145.08	4	1	35	25812	5	0
900	4	2	311	43097	5	0

In additional, the sliding window ought to be around the signify time that the data usually stays during a cache (SWL is 15 min).

In a similar way , the category of the data is classified into five types: HTML with worth 1, image with worth 2, audio

with worth 3, video with worth 4, application with worth 5 and others with worth 0. The worth of b will be assigned to 1 if the object might be re-visited again within the progressive sliding window. Otherwise the output should be assigned to 0. One time the dataset is prepared (see Table 4), the machine learning techniques are taught depending on the concluded dataset to categorize the World Wide Web objects into objects that will be re-visited or not.

Each proxy dataset is then separated randomly into training data (75%) and testing data (25%). Consequently, the dataset is normalized according into the series [0, 1]. When the dataset is arranged and normalized, the machine learning methods are applied using WEKA3.7.10 [20] see Fig. 3 and 4.

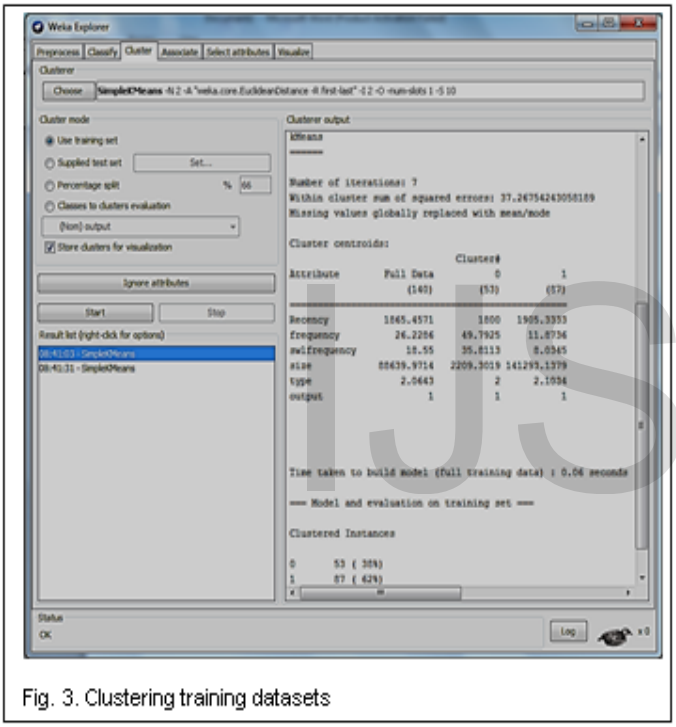

Fig. 3. Clustering training datasets

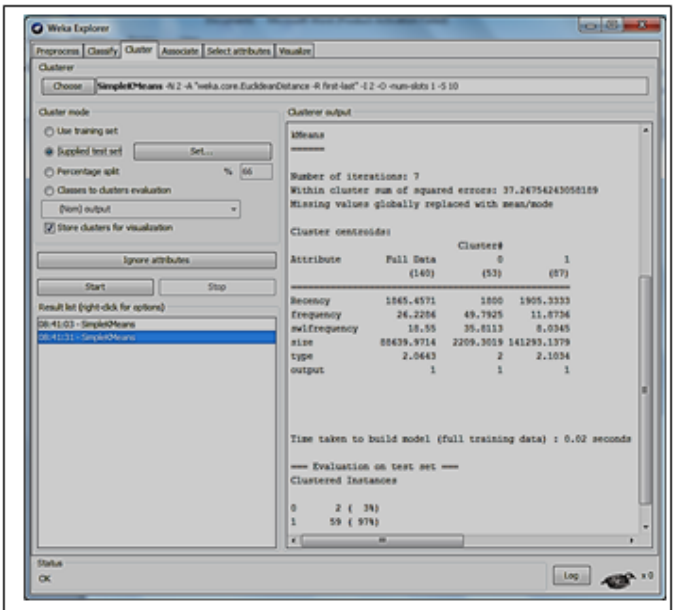

Fig. 4. Clustering testing datasets

4.4 Web Proxy Cache Simulation

The simulator WebTraff [13] can be modified to rendezvous our suggested proxy caching approaches. WebTraff simulator is an open source simulator used to evaluating distinct replacement Policies such as LRU, LFU, GDS, GDSF, FIFO and RAND policies Fig. 4. 1The trained clusters are integrated with WebTraff to simulate the suggested novel World Wide Web proxy caching approaches. The WebTraff simulator receives the arranged log proxy document as input and develops file encompassing performance measures as outputs.

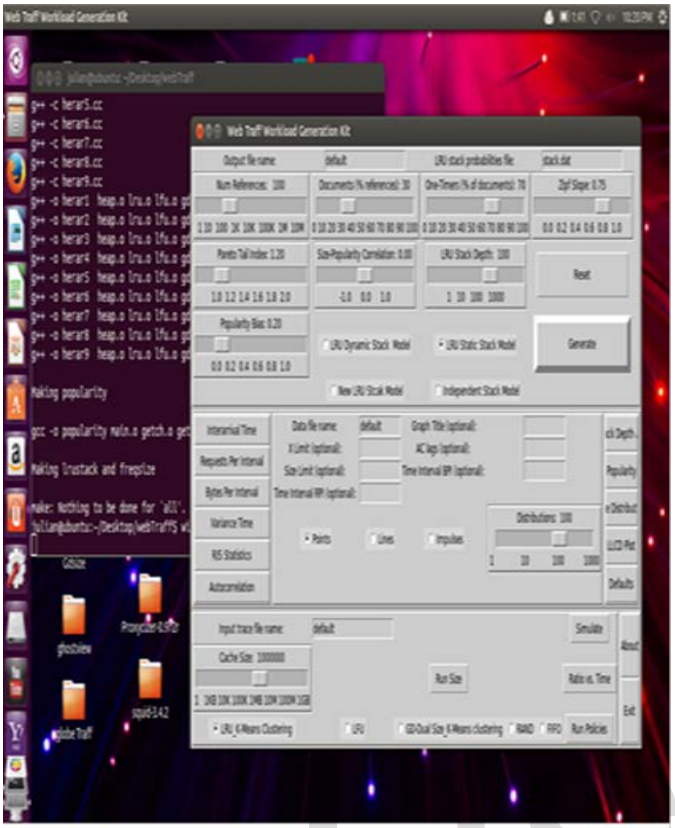

Fig 4.1. WebTraff simulator

TABLE 5
THE MOST COMMON MEASURES

Measure name	Formula
Correct clustering ratio	$CCR = \frac{\#correctely\ classified\ examples}{\#total\ examples}(\%)$
True positive ratio	$TPR = \frac{TP}{TP+TN}(\%)$
True negative ratio	$\mathrm{TNR} = \frac{TN}{TN+FP}(\%)$
G mean	$GM = \sqrt{TPR * \mathrm{TNR}}(\%)$

TABLE 6
THE CONFUSION MATRIX

	Assessed positive	Assessed negative
Actual positive	True Positive (TP)	False Negative (FN)
Actual negative	False Positive (FP)	True Negative (TN)

TABLE 7
CCR FOR DIFFERENT DATASETS

Datasets	CCR of training dataset		CCR of testing dataset	
	k-Mean	ANFIS	k-Mean	ANFIS
BO2	0.901	0.845	0.885	0.781
NY	0.862	0.689	0.813	0.724
UC	0.889	0.872	0.856	0.770
5V	0.892	0.707	0.892	0.725
5D	0.898	0.642	0.823	0.751
Average	0.888	0.751	0.854	0.750

5. PERFORMANCE EVALUATION

5.1 Cluster Evaluation

A correct clustering ratio (CCR) is a measure for estimate ng cluster. However, CCR alone is deficient for evaluat ing the performance of a cluster, particularly if the data is unbalanced. In an unbalanced data item, where the da taset covers significantly more popular class than smaller class instances, one can always select the popular class and obtain good CCR [4] see Table 7

We address that the object will belong to the positives class if the object is re-visited again either the forward-looking SWL.

TABLE8
TPR and TNR for training datasets

Datasets	TPR for training set		TNR for training set	
	k-Mean	ANFIS	k-Mean	ANFIS
BO2	0.792	0.708	0.868	0.982
NY	0.982	0.591	0.828	0.786
UC	0.758	0.681	0.868	0.883
5V	0.798	0.552	0.796	0.861
5D	0.898	0.484	0.870	0.799
Average	0.846	0.603	0.840	0.862

TABLE 9
TPR and TNR for testing datasets

Datasets	TPR for testing set		TNR for testing set	
	k-Mean	ANFIS	k-Mean	ANFIS
BO2	0.725	0.681	0.763	0.881
NY	0.712	0.591	0.895	0.857
UC	0.869	0.693	0.813	0.847
SV	0.786	0.552	0.736	0.898
SD	0.745	0.508	0.756	0.994
Average	0.767	0.605	0.793	0.856

TABLE 10
Gmean For Different Datasets

Datasets	G mean for training set		G mean for testing set	
	k-Mean	ANFIS	k-Mean	ANFIS
BO2	0.875	0.571	0.888	0.778
NY	0.853	0.791	0.745	0.889
UC	0.812	0.875	0.817	0.787
SV	0.875	0.852	0.862	0.863
SD	0.962	0.808	0.982	0.858
Average	0.875	0.779	0.861	0.835

Otherwise, the Web object will belong to the contradictory class. From proxy files, we can observe that most World Wide Web objects are remained just one time using the users. Hence, the contradictory class describes the most class, while the positive class contains the smaller class, which is the utmost important class in Web caching. Therefore, the true positive ratio (TPR) and the true negative ratio (TNR) can furthermore be utilized to assess the performance of the machine learning methods using some common measures as shown in Table 5 and 6.Gmean (GM) is used to estimate the overall performance of the machine learning methods, as shown in table 5.

Table 8 and 9 display a relationship among the performance measures of k-Means and ANFIS for five dissimilar proxy datasets in the training and testing stage. As can be discerned from Table 8 and 9, all of k-Means and ANFIS yield good performance. Table 8 and 10 apparently displays that the k-Means accomplish the best TPR and Gmean for all datasets. On the Contrary, ANFIS achieve the worst TPR and Gmean for all datasets. This contributes to getting the highest TNR of ANFIS. A higher weight is set to a positive class, while fewer weights are fixed to a negative class. Thus, k-Means has better TPR when related to further approaches; this specifies that k-Means can forecast the positive or lesser class which comprises the objects that might be re-visited within the close to future.

TABLE11
TRAINING TIME IN (SEC) FOR DIFFERENT DATASETS

Datasets	Training time(in seconds)	
	k-Mean	ANFIS
BO2	0.03	20.39
NY	0.15	22.66
UC	0.09	18.54
SV	0.21	16.18
SD	1.56	16.92

In addition, the computational time for training k-Means, ANFIS can be measured on the same computer for dissimilar datasets, as seen in Table 11. As expected, k-Means is faster than ANFIS for all datasets. Thus, we can conclude that the applications of k-Means in web proxy caching are more valuable and effective when related to other algorithm.

5.2 Evaluation of Integrated Web Proxy Caching

5.2.1 Performance Measures

In web caching, hit ratio (HR) and byte hit ratio (BHR) are two commonly utilized metrics for assessing the performance of web proxy caching strategies [1],[9],[15]. HR is well-defined as the ratio of the number of demands served from the proxy cache and the complete number of demands. BHR denotes to the number of bytes assisted from the cache, riven up by the complete number of byte assisted. It is important to memo that HR and BHR work in slightly opposite ways.

It is very difficult to accomplish the best performance for both metrics [1]. This is due to the fact that the strategies that increase HR typically give preference to little objects, but these strategies are inclined to decline BHR by giving less concern to bigger objects. On the contrary, the strategies that do not give preference to small objects tend to increase BHR at the expense of HR [1].

In terms of HR, the outcomes of Fig.5 clearly show that k-Means-LRU and k-Means-GDSF advance the performance in terms of HR for GDSF and LRU respectively for all proxy datasets. On the opposing, the HR of LRU-k-Means is similar or slightly not as good as than the HR of GDSF.

In terms of BHR, Fig. 6 illustrates that BHR of LRU-k-Means is better than BHR of GDSF-k-Means for the five proxy datasets. This is anticipated, since LRU policy eliminates the old objects despite of their sizes.

It is very difficult to accomplish the best performance for both metrics [1]. This is due to the fact that the strategies that increase HR typically give preference to little objects, but these strategies are inclined to decline BHR by giving less concern tobigger objects. On the contrary, the strategies that do not give preference to small objects tend to increase BHR at the expense of HR [1].

The norms of HR and BHR for five proxy datasets in all specific cache size are computed as Eq. (1). Wherever, ER is the percent of enhancement attained by the proposed tech-

nique (PT) over the conventional technique (CT).

$$ER = \frac{(PT-CT)}{CT} \times 100\ (\%) \quad (1)$$

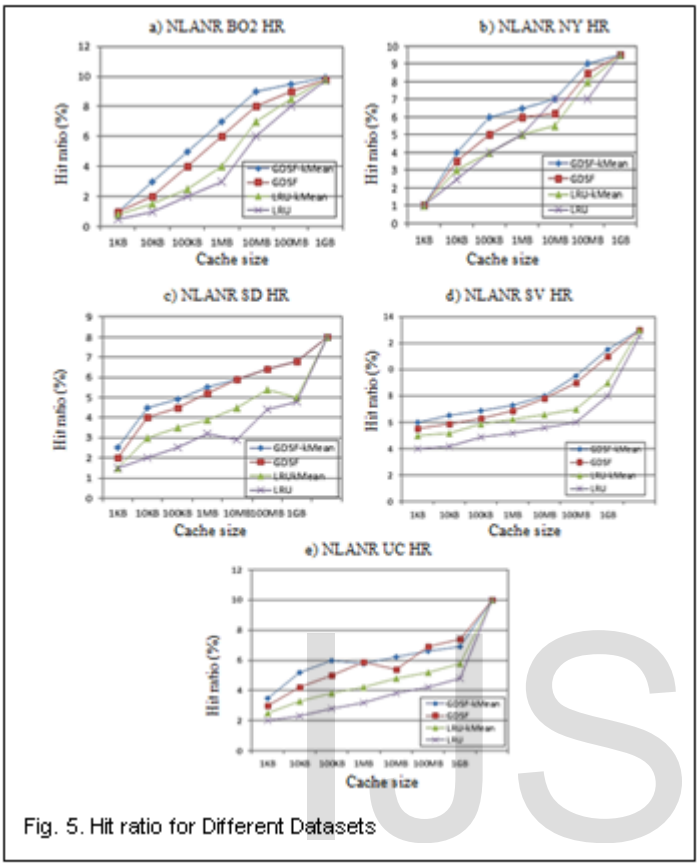

Fig. 5. Hit ratio for Different Datasets

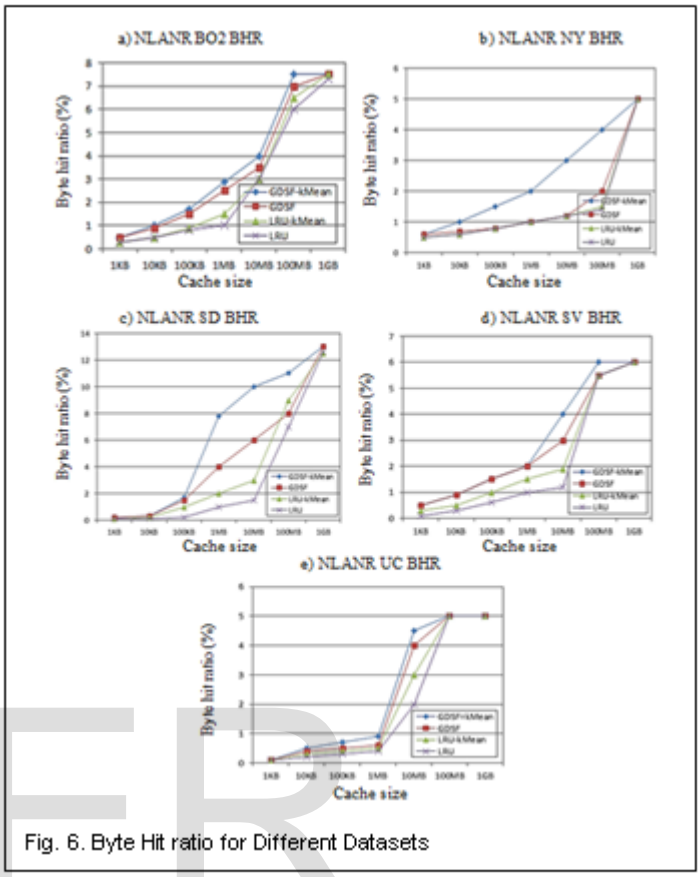

Fig. 6. Byte Hit ratio for Different Datasets

The enhancement ratios (ER) of the performances in terms of HR and BHR which are attained using the suggested approaches are determined and concise in Table 12. The outcomes in Table 12 specify that k-Means-GDSF increases GDSF performance in terms of HR up to 20.90% and in terms of BHR by up to 95.46% and k-Means-LRU over LRU is up to 31.87% in terms of HR and up to 32.34% in terms of BHR.

TABLE 12
ENHANCEMENT RATIO

Cache size	kMean-GDSF Over GDSF		kMean-LRU Over LRU	
	HR	BHR	HR	BHR
1KB	20.90	33.01	26.64	24.17
10KB	18.19	97.46	31.87	27.68
100KB	14.27	34.69	15.04	32.34
1MB	12.33	47.69	26.70	27.95
10MB	10.94	95.46	30.77	18.53
100MB	9.61	86.66	8.86	15.06
1GB	9.27	58.77	61.08	16.38

6 CONCLUSION

This study has suggested two novel web proxy caching ap-

proaches, namely k-Means-LRU, and k-Means-GDSF for improving the performance of the conventional World Wide Web proxy caching algorithms. Primarily, k-Means discovers from World Wide Web proxy log file to forecast the categories of objects to be revisited or not. Experimental results have revealed that k-Means achieve much better true positive rates and performance much faster than ANFIS in all proxy datasets. More importantly, the trained clusters are combined effectually with conventional Web proxy caching to provide more productive proxy caching policies.

REFERENCES

[1] S.Romano, H.ElAarag, "A neural network proxy cache replacement strategy and its implementation in the squid proxy server," Neural Computing and Applications, vol. 20,pp. 59-78, 2011.

[2] G. Sajeev, M. Sebastian, "A novel content classification Scheme for web caches," Evolving Systems vol. 2 pp. 101–118, , 2011.

[3] C,-J. Huang, Y.W. Wang, T.-H. Haung, C.-F. Lin, C.-Y. Li, H.-M. Chen, P.C. Chen, J.-J. Liao, "Applications of machine learning techniques to a sensor- network-based prosthesis training system," Applied Soft Computing, vol.11,pp. 3229-3237, 2011.

[4] H.T. Chen,"Pre-fetching and Re-fetching in web caching system", Algorithms and Simulation, Trent University, Peterborough, Ontario, Canada, 2008.

[5] C.Kumar, J.B. Norris, A new approach for a proxy-level web caching mechanism," Decision support System , vol. 46, , 2008, pp.52-60.

[6] Cobb, J., & ElAarag, H. Web proxy cache replacement scheme based on back-propagation neural network. Journal of Systems and Software,vol. 81, pp.1539-1558, 2008.

[7] Farhan, "Intelligent web caching architecture," Faculty of Computer Science and Information System, UTM University, Johor, Malaysia, 2007.

[8] B. Liu, "Web data mining: exploiting hyperlinks,contents, and usage data," Springer, 2007.

[9] Shekhar R. Gaddam, Vir V. Phoha, Senior Member, IEEE, and Kiran S. Balagani,"K-Means+ID3: A Novel Method for Supervised Anomaly Detection by Cascading K-Means Clustering and ID3 Decision Tree Learning Methods",ieee transactions on knowledge and data engineering, vol. 19, no. 3, march 2007.

[10] W.Kin-Yeung, "Web cache replacement policies a pragmatic approach", IEEE Network, vol.20, pp.28-34, 2006.

[11] S. Podlipnig, L. Boszormenyi, "A survey of web cache replacement strategies," ACM Computing Surveys, vol. 35, pp. 374-398, 2003.

[12] Koskela, T., Heikkonen, J., & Kaski, K.. Web cache optimization with nonlinear model using object features. Computer Networks, 43, 805-817, 2003.

[13] N. Markatchev, C. Williamson, "WebTraff: a GUI for Web proxy cache workload modeling and analysis," Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, IEEE Computer Society, pp.356, 2002.

[14] J. Han, M. Kamber, "Data Mining: Concepts and Techniques," Morgan Kaufmann, 2001.

[15] I.Bose, H.K.Cheng," Performance models of a firms proxy cache server," Decision Support System and Eletronic Commerce, Vol.29, pp.45-57, 2000.

[16] L. Cherkasova, "ImprovingWWWProxies Performance with Greedy-Dual-Size-Frequency Caching Policy", Technical Report HPL-98-69R1, Hewlett-Packard Laboratories, November 1998.

[17] P. Lorenzetti and L. Rizzo, "Replacement Policies for a Proxy Cache", Technical Report, University Pisa, December1996.

[18] E. O'Neil, P. O'Neil and G. Weikum, "The LRU-K Page Replacement Algorithm for Database Disk Buffering", *Proceedings of SIGMOD '93*, Washington, DC, May 1993.

[19] Lan H. Witten,Eibe Frank,Mark A.Hall,Data Mining:Practical Machine Learning Tools and Techniques, Morgan Kauffmann, 2011.

[20] WEKA tool: Available at http://www.cs.waikato.ac.nz/ml/weka/.

[21] NLANR, National Lab of Applied Network Research (NLANR), Sanitized Access Logs: Available at http://www.ircache.net/2010.