Research on WEB Cache Prediction Recommend Mechanism Based on Usage Pattern

LIN Jianhui[1,2], HUANG Tianshu[1], and YANG Chao[2]
1 School of Electronic Information, Wuhan University
2 Department of Information Technology, Hubei University of Police
linjh_eis@hotmail.com

Abstract

Cache prefetching technique can improve the hit ratio and expedite users visiting speed. After analyzed the recommend system in E-Business, this paper studied the characteristics how user visit web page and proposed a web prefetching recommender system based on usage pattern. This system cluster user behavior through an improved ant colony algorithm, then usage pattern can be abstracted from these classes through sequence mining. These sequence patterns are applied to forecast the coming behavior of users thus improve the hit ratio of system. Experiment result proves the validity of the system.

1. Introduction

Cache technique is a common technology which can store the nearest collected information in order to use it in future, these information are thought to be used more frequently than others. But as is known from [9], there is exponential relation between the increasing of cache size and the hit ratio of cache. Even though with an infinite cache, the hit of ratio can reach only at the range from 40% to about 50%.

Consequently, we can learn that the hit ratio is the important index to estimate the performance of cache, it is affected by cache size, updating tragedy, user visit habit and many other factors [1]. In order to improve the hit ratio of cache, cache prefetching technique is proposed. If only prediction is correct, both the hit ratio of cache and the visiting speed can be improved.

Present prefetching system, however, cast much emphasis on statistical information of the rules. It depends on the probability of the appearance of a rule, and then decides whether this rule is adopted. It does not take different users' preference into account and neglects the habit of different individual; as a result, the precision of the prediction may be affected. In this paper, we proposed a recommender system which abstracted the web usage pattern as a prediction rule from WEB log. These rules were used to forecast the user's coming usage page so system can in advance prefetch the object this user may visit into cache.

2. System Model

2.1 Recommendation Process

Recommender systems were first used in E-Business to estimate the interest of customer and recommend the goods customers may interest in to them. Referring to recommender system, if considering prediction rules as goods, we can take the same measure to forecast the page user may visit and prefetch them into cache. Fig.1 shows the rough process of prefetching system.

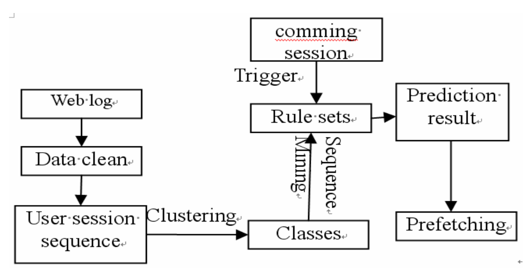

Fig. 1 An ordinary model of the prefetching system

As is shown in fig. 1, the whole prediction process is as follows:

Clean the web usage log. Retain only those recorders which may be useful in abstracting rules and find the user information (IP address or Cookies), divide web usage recorders to obtain the user session.

a. Clustering. Cluster the user session sequence and obtain certain classes according the similarity of sequence.

0-7695-3090-7/08 $25.00 © 2008 IEEE
DOI 10.1109/WKDD.2008.9

b. Mining usage rules. Mining web usage patterns as rules to forecast from different classes.

c. Classification. Decide user's membership information in each class according to membership information of sessions.

d. Prediction and Updating. While a new session comes, determine which user it belongs to, select prediction rules according to certain usage pattern. Lastly, update the membership information of the user.

2.2 Analyze the Log

The information user visited agent cache was saved in usage log, its format comply with W3C standard, shown in table 1.

Table 1. Log segment

204.186.186.83[08/Apr/2000:05:18:44-0400]"GET aukce/rings/ prstynky3.jpg HTTP/1.1"200 12551"http://cgi.ca.ebay.com/aw-cgi/eBayISAPI.dll?ViewItem&item =298708327" "Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt)"
202.168.131.33[08/Apr/2000:08:39:44 -0400] "GET /warez/upload/AutoCad/ HTTP/1.0" 404 18897 "-" "Mozilla/4.0 (compatible; MSIE 5.0; Windows 98; DigExt)"

Each recorder includes many fields, including visit data, time, user IP address, method, URL resources, server responding status, user proxy, browsing time, etc. to simple the processing procedure, we deleted automatic link downloaded item such as .jpeg, .gif file, RSS file, etc., request visit failure recorders and those recorders whose method field is not GET are also deleted.

2.3 Identify User Visit Transaction

a. Identify users. Identify each user according to IP address and cookies fields, then divide log file into some independent usage recorder sets.

b. Identify sessions. If the interval between request time of two pages exceed a certain threshold, we claim this user began a new session. In practice, we set this time threshold at 30 minutes.

Up to now, we can express the web usage event of users within a certain time as a $n \in Z$ dimension sequence:

$S_i = (url_1, url_2, ..., url_n)$

Since a URL is a string, which is not convenient to handle, we transferred URLs to some long integral with MD5 hash function. Some transferred user sessions were shown in table 2.

Table 2. Some transferred user sessions

2431125→550196→4883866→4883973→4884143→4884218→4884374→4884374→4884374→19765→536326→4202061→4857252
4398→4398→90914→91559→1814→4807071
4807086→4821878
4853087→4853155→1008618→996929
4523010→61026→1007925→4868643→4869908→4870250→4871088→4523735→4871355→487503
4785143→2923171→915564→541851→446428→4887393
234407→4889429→2036964→2037491→3239292
752107→2563357→2563060→3820374→857060→960454
802099→158235→2855301→2855315

3. Clustering Session Sequence

3.1 Similarity of Directed Graph

In this paper, we not only compared the similarity of URL content, but also lay emphasis on the different position information between URLs. In order to highlight the affect that sequence order imposed on its similarity, we suggested a definition of similarity through comparing directed graph.

Suppose there were 2 user visiting sequence Seq1, Seq2, represented with directed graph as G1, G2.Select public element from two sequences then express directed graphs as matrix M1, M2. Take corresponding element of two matrix and make "AND" operation, then a new matrix G is obtained.

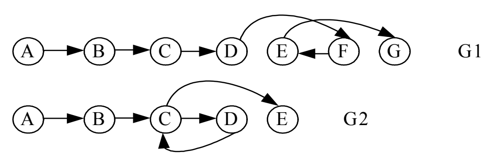

Fig.2 Usage sequence represented by directed graph

Definition 1:Define the similarity between seq1 and seq2 as:

$$r_{AB} = \sum_{i=1,j=1}^{|seq_1 \cap seq_2|} (a_{ij} AND b_{ij}) / \max(|seq_1|,|seq_2|) \qquad (1)$$

$|seq_1|$ and $|seq_2|$ is the length of sequence, $|seq_1 \cap seq_2|$ is the number of public element. a_{ij} and b_{ij} respectively denote the element of M1 and M2. $0 < r_{AB} < 1$,the bigger r_{AB} is, the more similar two sequences are.

3.2 Ant colonies cluster algorithm

3.2.1 Algorithm description

a. Individual initialization

Assigned user session sequence to artificial ants, it was considered as ant's gene sequence. In standard ant colony algorithm, it randomly selects two ants and computed the similarity. But in practical computing process, because of the pseudo-random number, the opportunity each two ants meet is unequal, some ants rarely are selected. We improved on the process of the ant selection.

Suppose the total number of the ant colony in the set is M, and it has been iterated for $IT_{current}$ times. If the ant i has been selected $n_{icurrent}$ times, then at the next moment the probability it is selected can be express as formula:

$$P = (1 - n_{icurrent} / IT_{current}) / M \qquad (2)$$

b. Update template

When ants met, if their odor is compatible, they would exchange the odor information with each other to update the template. The longer two ants had met, the smaller the affect between two templates was. Consequently, we introduce the Attenuation effect into the process of the template updating.

Definition 2: Supposed that ant i in turn met ant j ($j \in \{1,2,3,4,\dots k, k+1, \dots n\}$), their similarity is $Sim(i,j)$, then after ant i has met k+1 ants,

$$\overline{Sim(i,\bullet)}_{k+1} = \frac{Sim(i,\bullet)_k + Sim(i,k+1)}{2} \qquad (3)$$

The effect imposed by the information of met ants attenuate at the ratio of 2^{-n}.

c. Algorithm description

Algorithm: Generate different classes through ant colony cluster

Input: Clustering sample set W; the sum of sample N, iteration number IT, evaluator M_i, M_i^+ *,class label of ant* $lable_i$ *,the information template of ant i* $Template_i$

Output: Clustered classes

Method:

Generate N artificial ants, and initialize them.

$M_i \leftarrow 0, M_i^+ \leftarrow 0, A_i \leftarrow 0$, $lable_i = 0$;

for($IT_{current}$*<IT)AND(not match iteration condition)*

{

Select randomly two ants anti and ant_j *from ant colony with probability P, let them meet and calculate* $Sim(i,j)$;

if($Sim(i,j) > Template_i$*)AND(*$Sim(i,j) > Template_j$*)*

{ $Accep\tan ce(i,j) = TRUE$

Update($Template_i$*);Update(*$Template_j$*);}   //update each template;*

else

{ $Accep\tan ce(i,j) = FALSE$ }

end

$if\,(Label_i = Label_j = 0)$ *AND* $Accep\tan ce(i,j) = TRUE$

{ *generate new class label* $Label_{NEW}$;

$Label_i \leftarrow Label_{NEW}$; $Label_j \leftarrow Label_{NEW}$;

}

$if\,(Label_i = 0 \wedge Label_j \neq 0)$ *AND* $Accep\tan ce(i,j) = TRUE$

{ $Label_i \leftarrow Label_j$ }

$if\,(Label_i \neq 0 \wedge Label_j = 0)$ *AND* $Accep\tan ce(i,j) = TRUE$

{ $Label_j \leftarrow Label_i$ }

$if\,(Label_i = Label_j) \wedge (Label_i \neq 0) \wedge (Label_j \neq 0)$ *AND* $Accep\tan ce(i,j) = TRUE$

{*Increase(*M_i, M_j, M_i^+, M_j^+*)*}

$if\,(Label_i = Label_j) \wedge (Label_i \neq 0) \wedge (Label_j \neq 0)$ *AND* $Accep\tan ce(i,j) = FALSE$

{*Increase(*M_i, M_j*); Decrease(*M_i^+, M_j^+*);*

if(the smaller ant in M_i, M_j *satisfies* $(x \mid M_x^+ = Min_{k \in [i,j]} M_k)$*)*

{ $Label_x \leftarrow 0, M_x \leftarrow 0, M_x^+ \leftarrow 0)$ }

}

$if\,(Label_i \neq Label_j)$ *AND* $Accep\tan ce(i,j) = TRUE$

{*Decrease(*M_i, M_j*);*

Merge the smaller ant in M_i, M_j *into the bigger class*}

}

3.2.2 Clustering result

Select DEC96-9-6 log file as experiment data source. The log file recorded the usage recorders from 7 A.M., 6th, Sep., 1996 to 7 A.M., 6th, Sep., 1997, 1271582 usage request recorders, 1083 users in total. After cleaning the data, 9382 usage request remained. Clustered them according our algorithm, the experiment result was shown in fig. 3, 72 classes were obtained.

Fig. 3 shows that the number of member belongs to different class are unequal. We set the threshold at 1%, then many class contain less members are regarded as noise and deleted; only 19 classes remain. Please use a 9-point Times Roman font, or other Roman font with serifs, as close as possible in appearance to Times Roman in which these guidelines have been set. The goal is to have a 9-point text, as you see here. Please use sans-serif or non-proportional fonts only for special purposes, such as distinguishing source code text. If Times Roman is not available, try the font named Computer Modern Roman. On a Macintosh, use the font named Times. Right margins should be justified, not ragged.

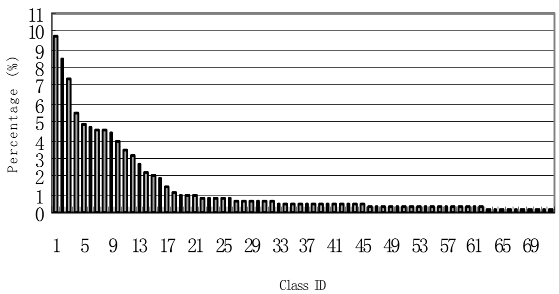

Fig.3 Clustering result

4. Ming Usage Sequence Pattern

Sequence pattern mining was first suggested by Agrawal and Srikant[4], it tried to find frequent sub-sequence from the data set consisted of sequences. Algorithm Apriori [5] is designed to mine frequent sequence from WEB usage sequence. Unlike [8], the sequence rule in this paper is used to predict the next step of a user, then page object in frequent sequence should be continuous, thus the primary algorithm has to be modified.

In prediction process, the criteria we selected rules are as follows:

a. Prior select the rules with higher probability to forecast;

b. If there are not matched rules in rules set with a higher Similaryi , then try to find in those rules set with little $Similary_i$, until find it successfully or no rules matched.

c. If there are different prediction rules generated from one rules set, then select rules according their length and similarity.

d. Because usage pattern varies with time span, once a new session comes, it should used to update the sequence set.

Experiment and Conclusion

Take DEC agent cache log as data resource we made our simulation. Divided the log into two parts; abstracted prediction rules from DEC96-9-6 and then these rules is tested with DEC96-9-7 log. In experiment, time span T=24h was divided into 6 segment, each time is 4 hours.

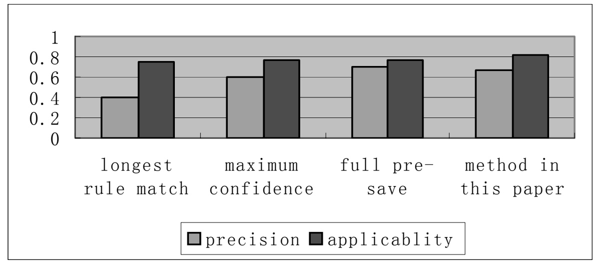

Fig.4 Accuracy and adoptability of different method

Comprehensively considered prediction accuracy and adoptability, we select 0.15 as the minimal support. Comparing our method with conventional method, the result is shown in Fig. 4. We can learn that the adaptability of prediction rules is improved because of filtering noise and enhancing data. On anther hand, since the full pre-save method expense higher network bandwidth to cache all possible prediction result , the accuracy of our method is little lower but higher then its.

Acknowledgments

This research work is partially supported by China Hubei Province NSF under Grand No. 2007ABA151.

References

[1] M.Abrams, C.R.Standridge, G.Abdulla,S.Williams, and E.A.Fox, Caching proxies: Limitations and potentials, Proceedings of the 4th International WWW Conference, Boston,MA,Dec.1995.

[2] Colomi A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies. In: Proc of 1st European Conf on Artifical Life . Pans France:Elsevier ,1991.

[3] Nicolas Labroche,Nicolas Monmarché,Gilles Venturini,A new clustering algorithm based on the chemical recognition system of ants, ECAI 2002: 345-349

[4] AGRAWAL R, SRIKANT R. Mining sequential patterns A Proc International Conference on Data Engineering [C]. Taipei: Taipei Press, 1995.

[5] Jiawei Han, Micheline Kamber. The conception and technology of data mining [M].Machine industry press.2001.8.