International Journal of Grid Distribution Computing
Vol.7, No.5 (2014), pp.149-158
http://dx.doi.org/10.14257/ijgdc.2014.7.5.14

On the Locality Modeling of Web Access Stream

Huixia Wang[1] and Yao Yao[2]

[1]*Department of Information Engineering, HeNan Radio & Television University Zhengzhou, China 450046*
[2]*School of Information Engineering, Zhongzhou University, Zhengzhou 450044, China, Zhengzhou, China 450046*
[1]*441239488@qq.com*, [2]*55539495@qq.com*

Abstract

Using mathematical method to model Web workload locality characteristics is established based on the study of entropy and coefficient of variation. Experiments show that these metrics can depict locality characteristics more properly and are much easier to use.

Keywords: *reference locality, modeling, spatial locality, temporal locality*

1. Introduction

The research of web workload characteristics is directly relevant to the improvement of network performance. Caching, prefetching and strategy of loading balance are all is inspired from the research of these characteristics. In the attribute of network workload, reference locality characteristic is the most important factor to influence the caching. The temporal locality as the basis of caching, the spatial locality as the prerequisite of prefetching, these principles have already had a good application to memory system.

Although the access characteristics of web request stream have already had many research achievements, for example Zipf's law modeling for popularity of access stream, heavy-tailed distribution modeling for size of access stream etc, the model of local characteristics of access files lacks unity and the definition is also relatively ambiguous. This paper put forward a model to describe and study the locality characteristics of reference stream, which overcomes shortcomings of modeling locality characteristics, such as the information of catching too ambiguous to satisfy the research demand and easily confused origins of locality, *etc.*, This paper takes advantage of popularity to model temporal locality based on the study of entropy, and utilize access correlation to model spatial locality between the same files based on the study of the coefficient of variation. By modeling locality characteristics, locality characteristics can be analyzed between different reference streams, thus making use of these characteristics can design algorithm and strategy to further improve the caching performance.

The purpose of this paper is to study the mathematical modeling and measurement method of reference locality. Quantization of reference locality helps us to judge it strong or weak in the log and how does it influence the caching performance of web.

In the remainder of this section we describe the data on which our measurements and trace simulations are based. In Section 1, we characterize the popularity of Web documents and show a cache reference model. This section also presents the evidence of this locality of reference properties. In Section 2, we characterize these properties and present statistical models that capture these characteristics. In Section 3, we

ISSN: 2005-4262 IJGDC

examine related work. We conclude in Section 4 with a summary and directions for future work.

2. Related Work

Access characteristics is modeled based on the files' access distance model, and analyzed and studied based on the ideal IRM model. By analyzing the characteristics and weakness of stack distance model and the Zip's law, it's confirmed that this modeling method is more suitable to catch information of access locality.

2.1. Inter Arrival Distance (IAD)

Inter arrival distance (IAD) is the amount of these files divided by other web files between two reference to the same file. In this paper IAD is used to stand for the reference distance of web documents, which represents the distributed characteristics of access locality.

The n caching document, N={1 , … , n}, *i.e.*, i=1 , … , n, a series of request to cache, {Rt , t=0, 1, …}, if Rt=i, then the Tth reference document is i, the popularity of {Rt, t=0, 1, …} is : P=(P(i),…P(N)) ,

$$P(i) = \lim_{t->\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} 1(R_\tau = i), i = 1, \cdots N \quad (1)$$

The temporal correlation of request sequence is: r(s,t) = Cov[Rs, Rt], s , t=0 , 1 , …

2.2. Independent Reference Model

All accesses to objects in a stream were completely uncorrelated, which could modeled by the Independent Reference Model (IRM). In the IRM, each object has associated a probability of being referenced, and each reference is independent of any other reference. This model is in fact an ideal model, and no access to objects in a stream is match this kind of distributes, because each reference in logs has some relativity to other reference.

In this model the inter arrival distance (IAD) follows a geometric distribution, which is a memory less distribution. For each object *i* with reference probability *pi*, the probability $d_i(k)$ of its associated IAT distribution is:

$$d_i(k) = P_i(1 - P_i)^{k-1} \quad (2)$$

Total probability function of IAT is:

$$d(k) = \sum_{i=1}^{n} P_i d_i(k) \quad (3)$$

2.3. Stack Distance Model

A stack distance model was defined which captures the temporal locality relationships present in the web pages visited. Let us define the LRU stack St, which is an ordering of all objects visited recently. Thus, at virtual time t, the LRU stack is given by St = { O_1, O_2 ,… ,O_N }, where O_1, O_2 ,… ,O_N are objects referenced and O_1 is the

most recently accessed object, O_2 the next most recently referenced, *etc*., If O_1 is the most recently accessed document, then Rt = O_1 . Whenever a reference is made to an object, the stack must be updated. Considering that $R_{t+1} = O_i$, then the stack becomes $S_{t+1} = \{O_i , O_1, O_2, \ldots, O_{i-1}, O_{i+1}, \ldots, O_N\}$.

Let d_t denote the stack distance at time t, then if $R_t = O_i$ then d_t = i. Thus, for any request string $R_t = R_1 , R_2 , \ldots R_t$, there is a corresponding distance string $\delta = d_1, d_2, \ldots, d_t$. Small stack distances result from frequent references to a document. In other words, the documents referenced most frequently tend to be close to the top of the stack. The average stack distance is the direct index of the temporal locality. If an ordering of objects visited is scrambled randomly, the average stack distance will become apparently greater.

2.4. Zip's Law

As we know, high frequency objects correspond to the main body of the Web requests, and low ones correspond to the Web objects such as one-time referencing. In this paper, Zipf 1st law is employed to model the high frequency objects; Zipf 2nd law is borrowed to simulate the low frequency objects. The reason is due to the fact that Zipf 1st law is valid for the high frequency objects, and the low frequency objects can be described by Zipf 2nd law, which is based on the study of the original research work of Zipf and many related work.

Zipf's first law: if an object's frequency of reference is P and its rank is i, they follow the relationship:

$$P^i = K/i^{\alpha} \tag{4}$$

where K is a constant, $\alpha \in [0.5 , 1]$.

Zipf's second law: if the number of objects whose frequencies are 1 is x, then the number of objects whose frequencies are 2, 3, 4…n are $x/2^2$, $x/3^2$, $x/4^2$…, x/n^2 respectively. Booth modified this formula to the following form:

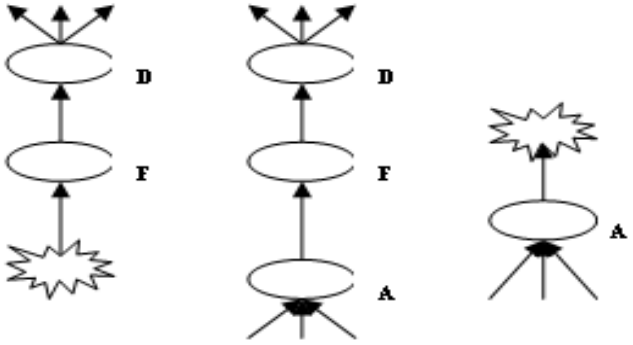
$$In/I_1 = 2/n(n+1) \tag{5}$$

In is the number of objects whose frequencies are n.

2.5. ADF Model

Fonseca, *et al.*, proposed a general ADF (Aggregation-Disaggregation-Filtering) framework, which depicts three basic transformations namely aggregation, disaggregation, and filtering.

Aggregation refers to multiple streams being merged into a single stream based on their arrival times. A typical example is the aggregation of requests from multiple sources (clients and proxies) at a server. Disaggregation is the reverse of aggregation where a single stream is split into multiple streams based on destination address. A typical example is the forwarding of requests by a proxy server to different origin servers. Filtering is a by-product of caching wherein some requests in a stream are absorbed by the proxy as cache hits, while others (those that result in cache misses) are forwarded to a higher-level proxy, or the origin server. Figure 1 depicts these transformation experienced by the request streams at the client, the proxy server and the origin server, which can be used to analyze the transformation at the multi-cache.



(a) Client (b) Proxy Cache (c) End Server

Figure 1. Node A Denotes Aggregator, D Denotes Disaggregator, F Denotes Filter, the Curve Denotes end Nodes, which is the Clients and the Original Servers

3. Measuring Locality of Web Object Referenced

Many documents study characteristics of reference locality and modeling method, but there are still many limitations. The stack distance model described in 1.3 section is easily influenced by variation of a single file and easily confuse the origin of reference locality. So a new modeling method of reference locality is proposed in this paper to overcome these disadvantages.

It may be observed that reference Locality is closely related to the two factors: files' popularity and files' reference relativity. For example, in the reference sequence XABXXCXCXXXEFX…, locality arises from the popularity of 'X', which directly determines strong or weak of the temporal locality of the reference sequence. In the reference sequence GGHHIIJJKKLL…, locality arises from the correlation of file referenced, which directly determines strong or weak of the spatial locality of the same files. An important difference between the two is that, in a random permutation of the sequence, the first property is preserved whereas the second is not, which can be used to distinguish the source of access locality and strength.

In order to describe the main characteristics of reference locality, the popularity is used to measure the temporal locality and the correlation is used to measure the spatial locality between the same files in reference stream.

3.1. Modeling of Popularity: Zip's Law

Zipf's law is used to define universal characteristics of web reference stream in general, which includes imbalance highly of popularity's distribution and parameter in terms of α value in order to capture the popularity's characteristics.

To model file's popularity, the constant K can be calculated in terms of Zipf's second law, and the popularity P in high frequency terms can be calculated in terms of Zipf's first law.

Algorithm 1: modeling the file's popularity

Input: N (the total number of requests to be generated), the percentage of distinct files, the number of files in the low frequency terms, the Zipf's exponent β

Output: popularity of high frequency files

Method:

Step1: Calculate the number of files of low frequency in terms of equation (5);

Step2: Estimate K in terms of $K=P_m (a+I_m/2)^{\beta}$, where a is the number of the distinct files in the high frequency terms;

Step3: Generate the popularity P of each file in the high frequency terms using equation (4);

3.2. Measuring of Popularity: Entropy

Entropy as a different measure method is used based on the weakness of Zipf's law introduced above. Entropy is used to measure the deviation of P_i from 1/n for some values of i and is not required to meet a certain distribution, so entropy is more flexible and easy to use than the Zipf parameter α.

The entropy of a random variable X taking on n possible values with probability $P_{i\ is}$ simply:

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad (6)$$

The properties of H(X) measure the deviation of X's distribution from the uniform distribution in which every files have the same reference probability. It takes on its maximum value ($\log_2 n$) in the case where all values of X are equally likely, *i.e.*, $p_i=1/n$ （i=1，2，…n）. It takes on its minimum value (zero) in the case where only one observation can occur, that is, when the outcome is deterministic.

In order to estimate H(X) from a given log, the requested objects in the log are numbered according to the IP address. Approximately the probability p_i of a given object i being referenced is calculated as the number of times it appears in the log, divided by the total references in the log. We thus obtain an empirical probability distribution of objects in the log. From the entropy H(X) defined in equation(4), H(X) only depends on the probabilities of occurrence of the different objects and the number of distinct objects that are referenced in the log , and not on the relative order in which they are occur. Because the number of distinct objects referenced increases with the size of a log, there is different number of distinct objects in a log. In order to compare entropy of logs entropy is normalized, therefore the metric for popularity we will use is the normalized entropy:

$$H_1 = H(X) / H_0(N) \quad (7)$$

Where N is the number of distinct references in the log, $H_0(N)=\log_2^N$. Because H_1 is often close to 1, the following transformation of H_1 is used:

$$H_2 = -\log_{10}^{(1-H_1)} \quad (8)$$

According to analysis above, entropy is calculated as follows:

Algorithm 3: measurement of the web files' popularity

Input：*Numall*（the total number of log），*Disnum*（the number of distinct files）

Output：entropy

Method：

Step1：Calculate the access probability P_i of distinct files in terms of log;

Step2：Calculate H(x) in terms of equation (6);

Step3：Calculate $H_1(x)$ in terms of equation (7);

Step4：Calculate the final value of entropy $H_2(x)$ in terms of equation (8).

3.3. Measuring of Correlation: the Coefficient of Variation

To measure correlation of reference locality, metric of the Coefficient of Variation can describe it well. The Coefficient of Variation is relative number and dimensionless, and can be used to compare the degree of variation between logs. The coefficient of variation of a distribution is its standard deviation divided by its mean. Coefficient of variation (CV) is a simple measure of relative dispersion of a distribution.

As shown in Section 2.2, if all accesses to objects in a stream were completely uncorrelated, the Independent Reference Model (IRM) could be used to model the request generating stream. In the IRM, each object has associated a probability of being referenced, and each reference is independent of any other reference.

Correlation for the accesses to the same object should cause a deviation from the geometric distribution; we introduce a metric, *i.e.*, CV that is very sensitive to this deviation. Given a geometric distribution with mean given by $\mu=1/p$, and variance given by $\sigma^2=(1-p)/p^2$, the CV is given by:

$$CV = \sqrt{1-p} \tag{9}$$

In Web reference streams, in which even the most heavily accessed objects, have a very low probability of reference, the expected CV, in the case of no temporal correlation, is very close to 1. Thus, CV values close to unity are associated with distributions close to the IRM, while values larger than one represent a distribution with large relative variance.

Each unique object in a trace has its own IAD-CV. In order to summarize an entire trace we must combine these individual values. The method we use is based on the per-reference IAD-CV. since we are concerned with temporal locality which can impact caching systems; the measure method is influenced by each accessed object. The median is robust metric and is independent of the length of the trace, so the distributional median is used as the total measurement. The test proves that this method is effective extremely.

According to analysis above, entropy is calculated as follows:

Algorithm 4: measurement of the web object's correlation

Input：*Numall*（the total number of log），*Disnum*（the number of distinct files）

Output：CV

Method：

Step1：Calculate the IAD distribution of distinct files in terms of log;

Step2：Calculate CV of the IAD distribution in terms of distinct files;

Step3：Calculate the median of CV after reorder CV.

Through analysis, entropy can be used to model the temporal locality and the temporal locality is strong or weak depend on the value of entropy. The CV can be used

to model the spatial locality between the same files and can reflect the spatial locality whether strong or weak.

4. Experiments

4.1. Data Collection

The Web is a large-scale distributed information system based on client-server architecture. Thus, the workload viewed from a server standpoint consists of a number of requests originated at many different clients.

In order to analyze the locality in reference stream, we examined the access logs for different Web servers, namely: NASA, Clark, World cup 98, Berkeley, USask. Five empirical logs are used in the experiment and Table 1 summarizes the statistics about the logs of the five Web servers.

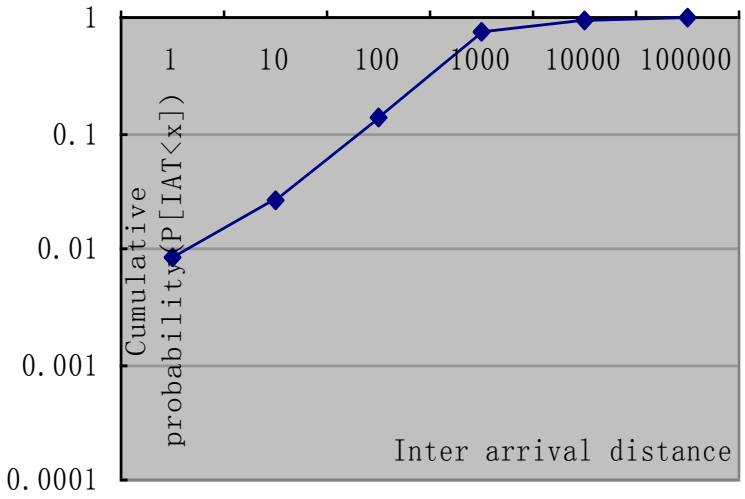
To test hit ratio in the cache, cache replacement policy is the LRU and the cache size accounts for 4% of the total size.

Table 1. The Logs used in Experiments

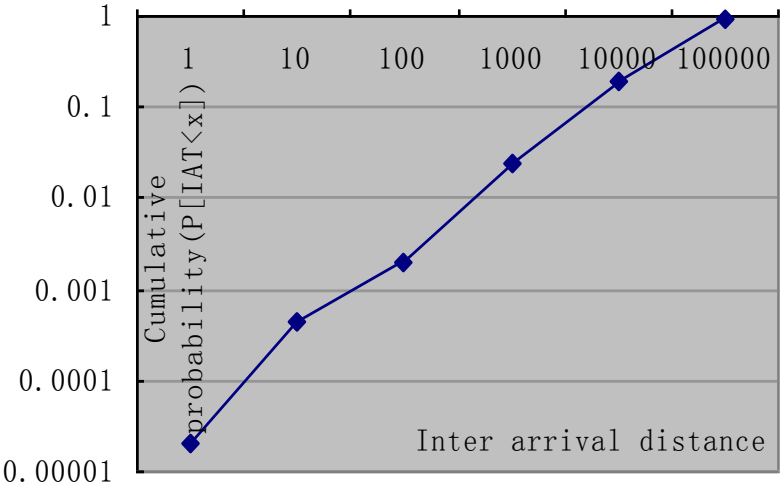
Name	Number of files	Number of distinct files
NASA	123210	9741
Clark	468141	32279
World cup 98	497061	8838
Berkeley	501708	10372
USask	696252	60057

4.2. Validaton of Files' Reference Distance

The size of reference distance determines the strength of reference locality in the log. In order to cache this difference, the empirical log (Clark) is compared to the scramble log and their reference distance too. As shown in Figure 2(b), the reference distance comes from the scramble log becomes very large and the cumulative probability at 1000 files intervals or below is very small, but the empirical log goes conversely. It shows that the empirical log has the characteristics of reference correlation, but the scramble log destroys this correlation and more closes to IRM. Then we concluded that the reference locality depends only on popularity of files and the popularity is very important and not easily disappear.



(a). The Empirical Log



(b). The Scramble Log

Figure 2. Cumulative Probability Distribution of IAD

4.3. Validation of Two Measure Method

4.3.1. Validation of Entropy: As shown in Figure 3, hit ratio in the scramble log increases gradually (from 2.82 to 18.33) with the entropy reducing (from 0.56 to 0.39). In order to remove correlation of reference locality and prompt hit ratio to depend only on popularity of files, the scramble log is used in this paper. Because entropy reflects the aggregation of popularity of files in the log, smaller entropy (such as Berkeley 0.39) means higher aggregation of popularity and stronger temporal locality and higher hit ratio (achieving 18.33). By contrast, larger entropy (such as Clark 0.56) means more disperse of files referenced and weaker temporal locality and lower hit ratio (only achieving 2.82).

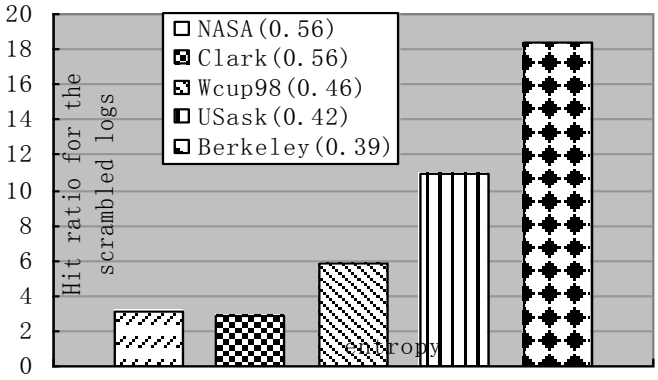


Figure 3. Entropy

4.3.2. Validation of CV: As shown in Figure 4, the difference of hit ratio between the empirical log and the scramble log becomes larger (from 22.51 to 53.94) with CV increasing (from 1.23 to 4.58). When CV becomes larger (such as Berkeley 4.58), the log deviates more from IRM, and reference correlation in the log and spatial locality between the same files becomes more strong and hit ration in the cache becomes higher and so hit ration difference becomes larger (achieving 53.94).

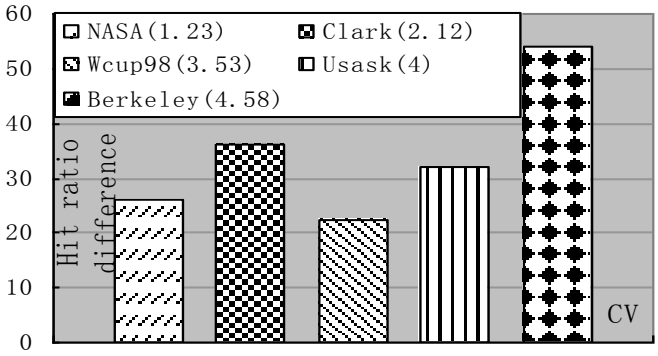


Figure 4. CV

5. Conclusion

New metrics is proposed to capture the two causes of reference locality: popularity and correlation. Entropy is defined as a natural metric for measuring the skew in the relative popularity of different objects in a request stream, and the Coefficient of Variation of the IAT distribution was used as a metric for spatial locality between the same files, motivated by the fact that the presence of such a correlation between accesses to the same object. It is validated of these metrics using wide logs from different area and these metrics are intuitive and effective.

The modeling of the temporal locality and spatial locality is the important foundation to caching and prefetching. This paper studies the local characteristics of the access logs, and proposes the new model for Measurement of the local characteristics. These models can be used to compare the reference stream origin from different source to test

and predict the cache performance, thus algorithm can be proposed to improve the performance of cache.

Acknowledgments

This work is supported by Key project of science & technology research of The Education Department Henan Province (project number: 14B520052), Humanities and Social Sciences planning fund of The Education Department Henan Province (project number: 2014-gh-215)

References

[1] A. Mahanti and C. Williamson, "Locality characteristics of Web streams revisited", Proceedings of Symp. Philadelphia: Performance Evaluation of Computer and Telecommunication Systems, **(2005)**, pp. 795-803.
[2] S. Vanichpun and A. M. Makowski, "The output of a cache under the independent reference model - where did the locality of reference go", Proceedings of the joint international conference on Measurement and modeling of computer systems. New York: ACM Press, **(2004),** pp. 295-306.
[3] S. Jin and A. Bestavros, "Source and characteristics of web temporal locality", In Proceedings of the 8th MASCOTS, San Francisco: IEEE Computer Society Press, **(2000),** pp. 28-35.
[4] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications", In Proceedings of IEEE Infocom, New York: IEEE Computer and Communications Societies, **(1999),** pp. 126-134.
[5] A. Mahanti, "Web proxy workload characterization and modeling", Department of computer science, University of Saskatchewan, **(1999)**.
[6] R. Fonseca, V. Almeida and M. Crovella, "Locality in a Web of streams", Communications of the ACM, vol. 48, no. 1, **(2003),** pp. 82-88.
[7] L. Cherkasova and G. Ciardo, "Characterizing temporal locality and its impact on web server performance", In Proceedings of IEEE ICCCN, Las Vegas: IEEE International Conference on Computer Communication and Networks, **(2000),** pp. 434-441.
[8] S. Jin and A. Bestavros, "Temporal locality in web request streams", Technical Report, Boston University Computer Science Department, **(2002)**.
[9] A. Mahanti, C. Williamson and D. Eager, "Traffic Analysis of a Web Proxy Caching Hierarchy", IEEE Network, vol. 14, **(2000),** pp. 16-23.

Authors

Huixia Wang, she is currently lecturer in the department of information Engineering, HeNan Radio & Television University. She received her Master of engineering degree of Application of computer from department of information engineering, University of Zhengzhou, China in 2007. The author has been teaching in HeNan Radio & Television University for 6 years. Her primary research interest includes Artificial intelligence, Web prefetching and Web mining.

Yao Yao, she is currently a lecturer in the school of information Engineering, Zhongzhou University. She received her Master of engineering degree of Application of computer from department of information engineering, University of Zhengzhou, China in 2008. Her research interests include web prefetching and web mining.

