ECAI 2008
M. Ghallab et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-891-5-251

Web Page Prediction Based on Conditional Random Fields

Yong Zhen Guo and **Kotagiri Ramamohanarao** and **Laurence A. F. Park** [1]

Abstract. Web page prefetching is used to reduce the access latency of the Internet. However, if most prefetched Web pages are not visited by the users in their subsequent accesses, the limited network bandwidth and server resources will not be used efficiently and may worsen the access delay problem. Therefore, it is critical that we have an accurate prediction method during prefetching. Conditional Random Fields (CRFs), which are popular sequential learning models, have already been successfully used for many Natural Language Processing (NLP) tasks such as POS tagging, name entity recognition (NER) and segmentation. In this paper, we propose the use of CRFs in the field of Web page prediction. We treat the accessing sessions of previous Web users as observation sequences and label each element of these observation sequences to get the corresponding label sequences, then based on these observation and label sequences we use CRFs to train a prediction model and predict the probable subsequent Web pages for the current users. Our experimental results show that CRFs can produce higher Web page prediction accuracy effectively when compared with other popular techniques like plain Markov Chains and Hidden Markov Models (HMMs).

1 Introduction

While the Internet is developing rapidly, the number of users surfing the Internet is dramatically increasing. Even though the construction of the Internet infrastructure is developing very quickly, many users still connect to the Internet through slow connections. According to [28], in 2007 about 20% of the 162 millions Internet users in China access the Internet using slow dial-up connections. Meanwhile, because of the popularization and convenience of wireless connection, many users have begun to use mobile phones or PDAs to surf the Internet. For example, in 2007, at least 34% of the Internet users had wireless devices in both China [28] and America [29], and this ratio keeps increasing steadily. On account of the limited bandwidth and low-speed connection, usually many dial-up and wireless Internet users need to spend long periods of time waiting for the Web pages they are visiting to be transferred to them through Internet, which may lead to intolerable delays. Moreover, the access latency problem of broadband users is noticeable as well and can be improved.

In order to decrease the access latency of the Internet, a variety of different approaches have been proposed, among which *caching* and *prefetching* are two primary methods.

The caching technique has been widely used on the Internet. It greatly improves access speed by saving local copies of the Web pages that users are currently visiting, so that their browsers will not need to connect to the Internet to download these pages during future visits. However, the caching technique has some shortcomings. Firstly, a Web page can not be cached if it has not been previously accessed. Secondly, its function will be nullified if the Web resources on the Internet have been modified or updated (for example, the Web pages' contents are changed). In addition, to maintain the consistency of copies at the client side and the corresponding Web pages at the server side is quite expensive. Thirdly, if the caches are saved at the client side, when a user uses another computer to surf the Internet, the caches that have already been saved in his original computer will be useless for his current access to the Internet. These problems reduce the attractiveness of caching.

Web page prefetching techniques are introduced as another effective way to address the access latency problem and thus improve the usability and user retention of a Web site. By analyzing the Web log and a user's current access path in combination with the link structure using different methods (such as association rules mining, Markov models or neural network), the Web pages that the user may access in the immediate future can be predicted by the Web site server and transferred before the user requests them. When the user accesses the page, there is no latency since the page has already been downloaded. It has been proven by many practical applications that the Web page prefetching technique is able to decrease a user's access delay dramatically and thus enhance the service quality of the World Wide Web [2]. The results from the simulations in [3] show that a 36% reduction in the latency perceived by an Internet user can be achieved at the cost of a 40% increase in the network traffic. Moreover, the studies in [2] indicate that by using the "rate-controlled" prefetching to smooth the transfer rates of prefetched pages can significantly reduce the network traffic congestion caused by "aggressive" prefetching and, therefore, improve the performance of the Internet.

However, if most prefetched Web pages are not visited by the users in their subsequent accesses (implying that the prefetching method has predicted these users' actions poorly), the limited network bandwidth and server resources will not be used efficiently, and hence may worsen the access latency problem. Therefore, the success of a prefetching method relies mainly on the prediction accuracy.

In this paper, we propose a novel Web page prediction approach based on Conditional Random Fields (CRFs) [1] to improve the prediction accuracy. CRFs are powerful probabilistic framework for labeling and segmenting sequential data. Owing to their conditional nature, CRFs have the ability to model the dependencies among observation elements; they can also incorporate various features from observation sequences to increase the prediction accuracy. CRFs have already been used with success to many labeling-related tasks, such as text chunking [4], part-of-speech (POS) tagging [1], intrusion detection [5] and even predicting the secondary structures of protein

[1] Department of Computer Science and Software Engineering, University of Melbourne, Australia, email: yzguo@csse.unimelb.edu.au

sequences [6]. If we consider the access sessions of previous Internet users as observation sequences, and in each observation sequence we use each pageview's subsequent pageview as its label to get the corresponding label sequence (each pageview is an observation element), then we can employ CRFs to model the access behaviors of all previous users and predict the possible Web pages that a current user will request in his subsequent access. We show in this paper that the CRF-based Web page prediction approaches have distinct advantages over other well known techniques such as plain Markov Chains and Hidden Markov Models (HMMs).

The rest of this paper is organized as follow: In Section 2 we briefly review the related works concerning Web page prediction and personalization. In Section 3, we first discuss the main differences between *generative* models and *discriminative* models, and then briefly introduce the basic principle of CRFs. The novel CRF-based Web page prediction approaches are presented in Section 4 along with the experimental results and evaluations. Finally, we conclude in Section 5 with our future work.

2 Related Works

Ming Syan Chen *et al.* [7] introduced the notion of *"maximal forward reference (MFR)"* to identify users' transactions and employed data mining techniques (such as association rules discovery) to mine frequently-accessed paths and make predictions. They first converted the original log data sequence into a set of maximal forward references and eliminated the effect of some backward references, then they presented algorithms to recognize the frequent traversal patterns from the maximal forward references obtained, which can be used to predict the user's future requests.

T. I. Ibrahim *et al.* [8] introduced a neural networks model to implement the semantics-based Web page prediction. This model extracts the semantics of a Web page according to the keywords of its URL anchor text. It employs these keywords as the input of the neural network to construct the semantic network of URLs, and predicts user's future requests based on the output of the neural network. In order to reduce the influence of the ambiguity of key words, this model builds a predictor for every different category of Web pages, which enhances the prediction accuracy but also decreases the applicability of this model.

M. Eirinaki *et al.* [9] proposed a novel Web personalization approach: *Usage-based PageRank (UPR)*, which combines both Web usage information and Web link structure information to conduct Web page ranking and prediction. This approach employs UPR to rank the Web pages in a relevant personalized navigational graph and predicts the probable pages in terms of their ranking values. By using the number of times a page was visited and the number of times the page was visited right after another page by previous users as the biasing factors, UPR favors the pages and paths that have been accessed more frequently by previous users. Yong Zhen Guo *et al.* [10] extended the UPR approach by introducing the access time duration of each Web page as another biasing factor, which will yield more accurate prediction.

Schechter [11] constructed an access path tree for the current user and used the longest-match method to find a history path which matched the user's current navigational path. In this way the user's following access requests can be predicted, but the construction of path trees and the match of history paths are expensive in terms of both computing and storage.

Sarukkai [12] employed a 1^{st}-order Markov model to analyze access paths and make predictions. In this model, every Web page is considered as a different state, and one state can transfer to another state with a certain probability according to previous users' access paths. After all transition probabilities are computed from training Web logs, the model can predict the most probable next page for the current user in terms of the transition probability matrix. However, when making predictions, this approach only takes users' current access requests into consideration but not the whole access paths, which will influence the prediction accuracy. In order to deal with this problem, higher-order Markov models [13] are proposed, which take into account more states when computing the transition probability, and thus improve the prediction accuracy. However, the increase of the order will increase the state space complexity. M. Deshpande *et al.* [14] discussed the shortcomings of higher-order Markov models in predicting Web users' browsing behaviors, and presented three schemes to eliminate the state space complexity of higher-order Markov models without influencing the performance.

A Hidden Markov Model (HMM) [15] is a dual-stochastic process which is very popular for labeling sequences, one stochastic process is an invisible Markov chain that describes the transition between states (labels) while the other reflects the statistical relationship between states and observations. Xin Jin *et al.* [16] proposed a HMM-based prefetching model in which they employed HMM to capture and mine the latent concepts of information requirement implied by Web users' access paths, and then used the obtained information to make semantic-based prefetching decisions.

In this paper we propose a CRF-based Web page prediction model and compare its prediction accuracy with that of plain Markov Chain models and Hidden Markov Models.

3 Conditional Random Fields

There are two predominant kinds of models for the tasks of sequence labeling and segmentation: *generative* models and *discriminative* models. Hidden Markov Models (HMMs) are *generative* models with a directed graphical structure. Similar to other *generative* models, HMMs define a joint probability distribution $p(x, y)$ where x and y are random variables over observation sequences to be labeled and their corresponding label sequences respectively. On account of the nature of modeling the joint probability distribution, *generative* models have some major drawbacks. First of all, the purpose of sequence labeling tasks is to label the given observations, which corresponds to the conditional distribution $p(y|x)$. Therefore, the joint probability distribution $p(x, y)$ defined by *generative* models is not the probability distribution of interest since the observation sequence x is already known and visible in both training and testing datasets. Secondly, in order to calculate the conditional distribution $p(y|x)$ from the joint distribution $p(x, y)$, the marginal distribution $p(x)$ is required according to the Bayes rule. However, because usually the amount of the training data is limited, it is difficult to enumerate all possible observation sequences, thus the calculation of $p(x)$ is only an approximation to the real distribution, which will decrease the accuracy of the model [17]. Furthermore, the calculation of $p(x)$ also requires strict independence assumptions over observation elements, which is not always possible in reality since most observation sequences in reality contain long-range dependencies and highly interacting features between observation elements [1].

On the contrary, *discriminative* models directly model the conditional distribution $p(y|x)$, they do not need to model the visible observation sequences x, which results in the relaxation of unwarranted independence assumptions over observation sequences. Moreover, owing to the conditional nature, *discriminative* models are able to model arbitrary features of observation sequences, regardless of the relationships between them. Therefore, *discriminative* models can overcome the inherent shortcomings of *generative* models and obtain higher labeling and prediction accuracy. The Maximum Entropy Markov Models (MEMMs) [18] are *discriminative* models. Because MEMMs conduct per-state normalization for the conditional probability of every next state given the current state and the observation sequence, they achieve a local optimum which can cause the "label bias" problem [1]. Conditional Random Fields (CRFs) are extensions of MEMMs, they have all the advantages of MEMMs and avoid the label bias problem. CRFs have a single exponential model for the conditional probability of the entire label sequence given the observation sequence [1], therefore, each state needs not to preserve the probability "mass" over its outgoing transitions and the whole model can achieve a global optimum.

A Conditional Random Field is an undirected graphical model, it defines a conditional probability distribution of a label sequence Y, given an observation sequence X. All components Y_i of Y are assumed to range over a finite label set. Let $G = (V, E)$ be a graph where V denotes the set of vertices, E represents the set of edges and $Y = (Y_v)_{v \in V}$, then (X, Y) is a conditional random field if, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph: $p(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$, where $w \sim v$ means that w and v are neighbors in G [1]. Although theoretically the structure of a Conditional Random Field can be an arbitrary undirected graph that obeys the Markov property, for the tasks of labeling the most common graphical structure is an undirected linear chain of first-order among label sequence Y, which can be seen in Figure 1 [19]. In our experiments, we made use of this linear chain model for the implementation of CRFs, where $X = (x_1, x_2, \cdots, x_n)$ denotes an observation of a user's accessing session of length n and $Y = (y_1, y_2, \cdots, y_n)$ denotes the corresponding label sequence of X. A linear chain CRF has the form as below:

$$P_\theta(Y|X) = \frac{1}{Z(X)} \exp\left[\sum_{t=1}^{T}\left(\sum_i \lambda_i f_i(y_{t-1}, y_t, X, t) + \sum_j \mu_j s_j(y_t, X, t)\right)\right]. \tag{1}$$

Where $f_i\,(y_{t-1}, y_t, X, t)$ is a transition feature function between the states (labels) at position $t-1$ and t, while $s_j\,(y_t, X, t)$ is a state feature function of the state at position t. $Z(X)$ is a global normalization factor over all possible label sequences with the following format:

$$Z(X) = \sum_Y \exp\left[\sum_{t=1}^{T}\left(\sum_i \lambda_i f_i(y_{t-1}, y_t, X, t) + \sum_j \mu_j s_j(y_t, X, t)\right)\right]. \tag{2}$$

The parameters $\theta = (\lambda_i, \mu_j)$ can be estimated from training data using many different approaches such as GIS[20], IIS[24] and L-BFGS[22, 23]. After the parameters are trained, the Viterbi [21] algorithm can be used to label the testing data and perform the prediction.

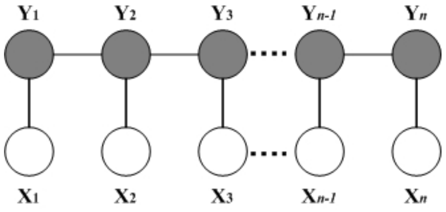

Figure 1. First-order linear chain graphical structure of a Conditional Random Field. The unshaded nodes indicate that the corresponding variables are observed and not generated by the model.

4 Experiments

In this section we present a set of experiments that we performed to evaluate the performance of using CRFs in Web page prediction. We compared the prediction results of CRF-based approaches to that of plain Markov Chain-based and Hidden Markov Model-based methods. The experimental results show an overall enhancement in the prediction accuracy by using CRF-based measures.

4.1 Experimental Dataset and Preprocessings

We used the publicly accessible *msnbc.com* anonymous Web data [26] as the dataset in our experiments. The *msnbc* dataset is obtained from the Web logs of *www.msnbc.com* and contains page visits of users who visited this website on September 28, 1999. All the user visits are recorded in session format at the level of page categories defined by the website administrator, such as *weather*, *health*, *sports* and so on. There are 17 different page categories in this dataset which can also be treated as 17 distinct pageviews. In addition, each page category was assigned one integer ranging from *1* to *17*, for example, the category *weather* was assigned the integer *8* while the category *sports* was assigned the integer *12*. An example of a user session in this dataset is: *6 9 4 4 4 10 3 3 10 5 10 4 4*. There are 989,818 user sessions with more than one pageview of this kind in this dataset.

We also performed a preprocessing to remove the consecutive repetitions of a same page from all of the user sessions. For instance, after this preprocessing, the sample user session above will be reduced to: *6 9 4 10 3 10 5 10 4*. Then we randomly selected 50,000 distinct sessions with length more than 5 and less than 100 from the preprocessed dataset and divided them equally into ten subsets, which will be used to perform 10-fold cross validation in our experiments.

Furthermore, we labeled the sessions in both training data and testing data. We treated each user session as an observation sequence. In the Web page prefetching scenario, we can use every observation element's subsequent element as its label. Therefore, since each observation element's subsequent element can be any of the 17 page categories, there are 17 different labels in total.

4.2 Experimental Setups

In our experiments we created seven different prediction methods to compare the Web page prediction accuracy of plain Markov Chains and Hidden Markov Models with that of Conditional Random Fields. The first two methods are the first-order plain Markov Chain (referred to as 1$^{\text{st}}$-PMC) and the second-order plain Markov Chain (referred to as 2$^{\text{nd}}$-PMC). We trained the 1$^{\text{st}}$-PMC and 2$^{\text{nd}}$-PMC to obtain their state transition probability matrices, and then labeled the testing dataset according to the entries of the corresponding transition matrix.

We implemented the $1^{\rm st}$- and $2^{\rm nd}$-order Hidden Markov Models as well, which will be referred to as $1^{\rm st}$-HMM and $2^{\rm nd}$-HMM respectively. All parameters of a HMM model can be estimated by the Forward-Backward algorithm [15]. However, because in our case the training is fully supervised, we can use a statistical method, which is quicker and more accurate, to acquire the parameters $\lambda = (\pi, A, B)$, where π is the initial probability distribution of states, A is the state transition probability matrix and B is the observation probability distribution matrix. After training, the Viterbi algorithm is used to label the testing dataset in terms of the trained model.

For the implementation of the Conditional Random Fields, we use the CRF++ toolkit [27]. CRF++ is a simple, customizable implementation of $1^{\rm st}$-order Conditional Random Fields which ensures fast training by using L-BFGS. We used three different CRF++ feature templates in our experiments. In the first template (referred to as CRF0), we define the current observation as the only unigram feature; in the second template (CRF1), we use the current and previous one observation and their combination as the unigram features; for the third template (CRF2), we use the current and previous two observations and their combinations as the unigram features. All these three templates share a same bigram feature, which will automatically generate a combination of the current label and previous label as the feature function. The more abundant and detailed features are used, the more powerful CRF models can be attained. Therefore, we expect that CRF2 has the best performance out of these three CRF models. Then we use CRF++ to label the testing dataset by employing the trained models.

4.3 Experimental Results

In our experiments we performed the 10-fold cross validation to evaluate the experimental results. We used the aforementioned seven methods to train the training dataset and obtained seven corresponding models, which are applied to label the testing dataset respectively later. For each observation in the testing data we predicted a series of labels that are ranged in a descending order according to their probabilities. Then we evaluated the prediction accuracy of these seven methods by using three different accuracy measures. The first measure is the "top-1 accuracy", in which we used the first label, which is also the most probable label, of the predicted label series as the current observation's label. The accuracy is simply the ratio of the number of correctly predicted labels to the total number of predicted labels. The second measure is the "top-3 accuracy", in this measure we use the top-3 most probable labels of the predicted label series to form a candidate label set for the current observation, if the real label of the current observation is in this candidate label set, we then consider this labeling as a "correct labeling". The third measure "top-5 accuracy" has the similar definition to the "top-3 accuracy". The reason we chose to measure the "top-3 accuracy" and the "top-5 accuracy" is because they resemble what happens in reality better, for instance, usually the prefetching systems will predict 3 or 5 possible "next" pages for the current user. The statistically significant prediction accuracies of these three accuracy measures for the seven methods are depicted in Figure 2, Figure 3 and Figure 4 respectively.

From the results we can see that the $1^{\rm st}$-PMC model has the worst performance in all of the seven models for all three accuracy metrics. Among the four non-CRF models, the second-order models achieve higher prediction accuracies than their corresponding first-order models, while the $2^{\rm nd}$-HMM outperforms the other three non-CRF models in all of the three accuracy measures. When compared CRF-based models with non-CRF models, we observed that although the "top-1 accuracy" of CRF0 is slightly lower than that of the second-order non-CRF models (that is, $2^{\rm nd}$-PMC and $2^{\rm nd}$-HMM), it behaves much better in both top-3 and top-5 accuracy measures. In the "top-1 accuracy" measure, CRF1 can achieve slightly better performance than $2^{\rm nd}$-HMM, while CRF2 outperforms $2^{\rm nd}$-HMM undoubtedly. In addition, all three CRF-based models provide higher accuracy than the $2^{\rm nd}$-HMM in both "top-3 accuracy" and "top-5 accuracy". Finally, when compared the models of CRF0, CRF1 and CRF2, we found out that the more detailed features a CRF model used, the higher prediction accuracy it can obtain. Moreover, CRF2 has the best performance out of the three CRF-based models in all of the three accuracy measures, which is in accordance with our expectation.

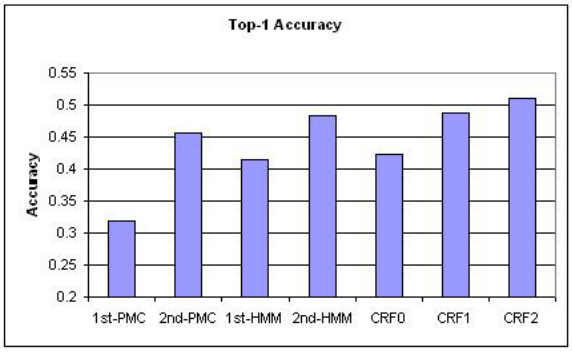

Figure 2. Top-1 accuracies for the 7 methods.

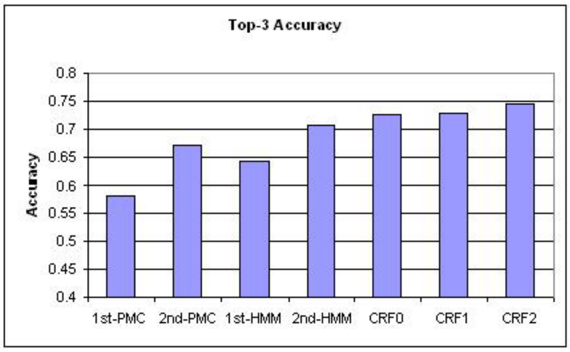

Figure 3. Top-3 accuracies for the 7 methods.

It is obvious from our experimental results that the selection of features is crucial to the performance of CRFs models. A good prediction model has poor performance without good features, while a less powerful prediction model may also perform well with a set of deliberately chosen features [25]. In our experiments, we only used the current and previous observations as the unigram features, more useful features can be incorporated to enhance the prediction accuracy, *i.e.*, *"the length of the observation sequence"*. Moreover, notice that although all the CRF-based models in our experiments are of first order, their performance (except that of CRF0 in the "top-1

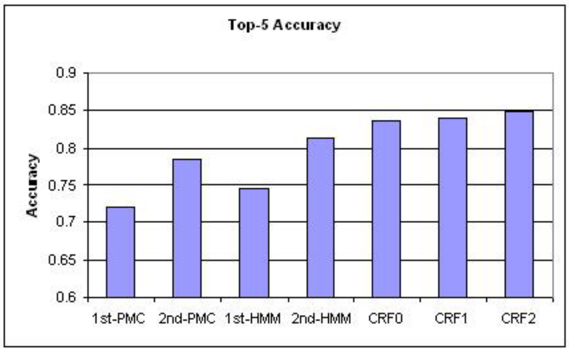

Figure 4. Top-5 accuracies for the 7 methods.

accuracy") have already exceeded that of the second-order Hidden Markov Models, we can consider that Conditional Random Fields are superior models for Web page prediction than Hidden Markov Models.

However, we noticed that the training of a CRF model is expensive and thus slower than that of PMC and HMM models, but once it is trained, its performance is robust and the speed of labeling the testing data is very fast, which is comparable to that of the other two models. Therefore, the CRF-based Web page prefetching can be efficiently applied online.

5 Conclusion and Future Work

In this paper, we discussed the main differences between *generative* models and *discriminative* models and showed through experimentation that the Conditional Random Fields can be effectively applied in the task of Web page prediction. The ability to model the long range dependencies among observation elements and the combination of arbitrary and overlapping features from observation sequences allow Conditional Random Fields to overcome the inherent disadvantages of the most popular Web page prediction models such as plain Markov Chains and Hidden Markov Models, and thus produce much more accurate predictions. The positive experimental results also revealed that by using richer features, the CRF-based prediction models can achieve better performance.

We should point out that the training of Conditional Random Fields converges considerably slowly when compared to HMMs and plain Markov Chains. The training complexity of a CRF is quadratic with respect to the number of labels. When the number of labels is very large, the training of a CRF may become very expensive and even intractable. Therefore, in our future work, we will focus on dealing with the problem of using CRFs to perform Web page prediction in a larger dataset with many thousands of distinct pages.

REFERENCES

[1] J. Lafferty, A. McCallum, F. Pereira. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*, In: Proceedings of the 18th International Conference on Machine Learning (ICML), 282-289, 2001.

[2] Crovella M, Barford P. *The Network Effects of Prefetching*, In: Proceedings of the IEEE Conference on Computer and Communications (INFOCOM' 98), 1232-1240, 1998.

[3] Venkata N., Padmanabhan. *Improving World Wide Web Latency*, Technical Report CSD-95-875, Computer Science Department, University of California at Berkeley, 1995.

[4] Charles Sutton, Khashayar Rohanimanesh, Andrew McCallum. *Dynamic Conditional Random Fields: Factorized Probabilistic Models for Labeling and Segmenting Sequence Data*, In Proceedings of the 21st International Conference on Machine Learning, 99-106, 2004.

[5] Kapil Kumar Gupta, Baikunth Nath, Ramamohanarao Kotagiri. *Layered Approach using Conditional Random Fields for Intrusion Detection*, IEEE Transactions on Dependable and Secure Computing, 2008.

[6] John Lafferty, Xiaojin Zhu, Yan Liu. *Kernel Conditional Random Fields: Representation and Clique Selection*, In Proceedings of the 21st International Conference on Machine Learning, 2004.

[7] Ming Syan Chen, Jong Soo Park. *Data Mining for Path Traversal Patterns in a Web Environment*, In: Proceedings of the 16th International Conference on Distributed Computing Systems, 385-392, 1996.

[8] T. I. Ibrahim, Cheng Zhong Xu. *Neural Nets Based Predictive Prefetching to Tolerate WWW Latency*, In: Proceedings of the 20th IEEE Conference on Distributed Computing Systems, 636-643, 2000.

[9] M. Eirinaki, M. Vazirgiannis, *Usage-based PageRank for Web Personalization*, In: Proceedings of the 5th IEEE International Conference on Data Mining (ICDM'05), 2005.

[10] Yong Zhen Guo, Kotagiri Ramamohanarao, Laurence A. F. Park. *Personalized PageRank for Web Page Prediction Based on Access Time-Length and Frequency*, In: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Web Intelligence (WI-07), 2007.

[11] Schechter S, Krishnan M, Michael DS. *Using Path Profiles to Predict Http Requests*, In: Proceedings of the 7th International World Wide Web Conference, 1998.

[12] Ramesh R. Sarukkai. *Link Prediction and Path Analysis Using Markov Chains*, Computer Networks, 33(1-6), 377-386, 2000.

[13] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, Pang-Ning Tan. *Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data*, SIGKDD Explorations, Vol. 1, Issue 2, 12-23, 2000.

[14] Mukund Deshpande, George Karypis. *Selective Markov Models for Predicting Web-Page Accesses*, In: Proceedings SIAM International Conference on Data Mining (SDM2001), 2001.

[15] Lawrence R. Rabiner. *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*, In: Proceedings of the IEEE, 257-286, 1989.

[16] Xin Jin, Huanqing Xu. *An Approach to Intelligent Web Pre-fetching Based on Hidden Markov Model*, In: Proceedings of the 42nd IEEE Conference on Decision and Control, 2003.

[17] Charles Sutton, Andrew McCallum. *Introduction to Statistical Relational Learning: An Introduction to Conditional Random Fields for Relational Learning*, MIT Press, 2006.

[18] A. McCallum, D. Freitag, F. Pereira. *Maximum Entropy Markov Models for Information Extraction and Segmentation*, In: Proceedings of the 17th International Conference on Machine Learning (ICML), 591-598, 2000.

[19] Hanna M. Wallach. *Conditional Random Fields: An Introduction*, CIS Technical Report MS-CIS-04-21, University of Pennsylvania, 2004.

[20] J. N. Darroch, D. Ratcliff. *Generalized Iterative Scaling for Log-Linear Models*, In: The Annals of Mathematical Statistics, vol. 43, 1972.

[21] Andrew J. Viterbi. *Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm*, IEEE Transactions on Information Theory, 260-269, 1967.

[22] Jorge Nocedal. *Updating Quasi-Newton Matrices with Limited Storage*, Mathematics of Computation, 773-782, 1980.

[23] D. C. Liu, J. Nocedal. *On the limited memory BFGS method for large scale optimization*, IMathematical Programming: Series A and B, 503-528, 1989.

[24] Adam Berger. *The Improved Iterative Scaling Algorithm: A Gentle Introduction*, Technical Report, School of Computer Science, Carnegie Mellon University, 1997.

[25] N. Smith, D. Vail, J. Lafferty. *Computationally Efficient M-Estimation of Log-Linear Structure Models*, In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2007.

[26] UCI KDD Archive: msnbc.com anonymous Web data. *http://kdd.ics.uci.edu/databases/msnbc/msnbc.html*, last accessible on February 10, 2008.

[27] CRF++: Yet another CRF toolkit. *http://crfpp.sourceforge.net/*, last accessible on February 10, 2008.

[28] The Website of the China Internet Network Information Center, *http://www.cnnic.com.cn/*, last accessible on February 10, 2008.

[29] PEW Internet & American Life Project, *http://www.pewinternet.org/*, last accessible on February 10, 2008.