International Journal of Computational Intelligence and Applications
Vol. 2, No. 3 (2002) 303–312
© Imperial College Press

PREDICTION OF NAVIGATION PROFILES IN A DISTRIBUTED INTERNET ENVIRONMENT THROUGH LEARNING OF GRAPH DISTRIBUTIONS

DIRK KUKULENZ
Institut f. Informationssysteme, Universitaet zu Luebeck
Osterweide 8, 23562 Luebeck, Germany
kukulenz@ifis.uni-luebeck.de

JOSEF PAULI
Fraunhofer Institut IITB, Abteilung ERS, Fraunhoferstr. 1
76131 Karlsruhe, Germany
Pauli@iitb.fraunhofer.de

Received 31 November 2001
Revised 25 March 2002

Collaborative filtering techniques in the Internet are a means to make predictions about the behavior of a certain user based on the observation of former users. Frequently in literature the exploited information is contained in the access-log files of web servers storing requested data objects. However with additional effort on the server side it is possible to register, from which to which data object a client actually navigates. In this article the profile of a user in a distributed web environment will be modeled by the set of his navigation decisions between data objects. Such a set can be regarded as a graph with the nodes being the requested data objects and the edges being the decisions. A method is presented to learn the distribution of such graphs based on distance functions between graphs and the application of clustering techniques. The estimated distribution is used to predict future navigation decisions of new users. Results with randomly generated graphs show properties of the new algorithm. A measure to estimate the prediction quality for observed profiles is presented.

Keywords: Collaborative filtering; prediction of user profiles.

1. Introduction

In many applications in the field of Internet research it is important to estimate the relevance of data objects available in the web for a specific user or a group of users. In the field of *content-based learning* the estimation is based on the (former) behavior of a specific user. *Collaborative filtering* techniques make it possible to learn from former usages of other users in order to make predictions for a new user. These estimations can e.g. be used to make a navigation on a web site easier, to improve the quality of web sites or to find groups of consumers or interest groups. In Ref. 10, a procedure was presented to apply a collaborative filtering technique for the creation of index lists, i.e. new web pages containing lists of hyperlinks relevant for a certain topic. The technique is based on an observation of sets of requested data objects. In Ref. 2, a navigation support system is presented that learns from search words and browsing decisions of users, applying a reinforcement learning technique. In Refs. 11 and 13, techniques for pre-sending documents on the WWW are described, that apply different kinds of Markov-learning techniques.

The information that is known about a specific user in the case of Refs. 10 and 13, is the log data of web servers. Each request is stored in a so called access-log file containing information about the time of a request, the IP-address of a client and the URL-address of the requested data object. However, different caching strategies are used in the web with the purpose to reduce net traffic and to increase the speed of requests. As a consequence, not all requests of clients actually reach the original server. Thus only a subset of requested data objects of a specific client is known on the server side.

In Ref. 13, the actual navigation path is estimated from the access-log information. We will however use an idea presented in Ref. 2 to register the actual set of navigation decisions of a client on the server side. An extended proxy server in the connection between server and client modifies each requested web page in a way that all hyperlinks point to that proxy server. The new links contain additional information like the originally requested page, the page where the link is located and an id-number assigned to the client. By this means navigation decisions of web users on the considered web-site can be registered on the server side. This method makes it also possible to register navigation decisions in a distributed web environment consisting of a number of web servers.

Our collaborative filtering procedure is based on these sets of navigation decisions of users. In the field of data mining algorithms were presented to find sets with high frequencies.[1] Related to that, in Ref. 6, an algorithm is presented to find frequent navigation sequences in the web. The approach described here is based on the distances between patterns. A set of navigation decisions can be regarded as a set of directed edges between data objects. These edges represent a graph structure with vertices being the requested data objects and the edges being the decisions. In the field of pattern recognition different distance functions between graph structures were presented e.g. in Refs. 4, 9 and 12. We will use one of these functions together with an application of nearest neighborhood clustering[7] to estimate the shape of the distribution of the graph profiles. Knowing this distribution, simple classification procedures can be applied to classify a new profile and thereby to predict future decisions. The advantage of this technique compared to Markov models as presented e.g. in Ref. 13 is that we don't have to consider the order of a Markov process. Such a predefinition may cause classification errors or otherwise cause an unnecessary increase of complexity.

In the next section the technique for the estimation of graph distributions and the prediction technique of future navigation decisions will be described. In Sec. 3, we present some estimation examples with randomly generated graphs showing properties of the distribution estimation and the prediction technique. Section 4 gives a summary and mentions further research issues.

2. Estimation of Graph Distributions

2.1. *Definitions and model*

As described in the introduction, the information we know about a specific web user on the server side is the set (or at least a subset) of his navigation decisions. These navigation decisions take place between certain data objects being available on the considered web site, like web pages, images, etc. Let "D" denote the set of data objects on the web site with an (own) URL address.

A user profile, measured by the system, is then a graph structure:

Definition 2.1. A (profile-) graph or navigation profile is a 4-Tupel $G = (V, E, \mu, \nu)$. V is a set of nodes and $E \subseteq V \times V$ is a set of edges. Function $\mu : V \to L_V \subset D$ assigns labels to the nodes. Function $\nu : E \to L_E$ assigns labels to the edges.

Let Γ be the set of all graphs following the preceding definition. This set will be denoted as "graph space" based on D.

L_V is a subset of D or a set of pointers to D. The edges considered here are in the most common case hyperlinks, that are present on certain web pages, Java applets or scripts. However, with the help of a search engine, the user can get from one data object to possibly any other object.

In the following sections definitions of a subgraph and the maximal-common subgraph are used, which are common in the field of graph theory or artificial intelligence and which are e.g. presented in Refs. 3 and 9.

2.2. *Characterizations of graph distributions*

It is our aim to classify a new profile graph according to a set of former profiles supplied by users. For this purpose it is helpful to know the distribution of graph profiles or at least to get an idea of the shape of this distribution. It is possible to regard Γ as a discrete set and to assign a probability value to each element depending on the relative frequency. However, graphs may be similar according to certain aspects which may not be taken into account by the discrete formulation. It is very likely that people having the same question in mind produce similar navigation profiles that are however slightly distorted because of Internet caching, different starting points or different searching strategies. Vice versa, similar profiles are likely to result from similar questions or intentions of users which is the main assumption we make.[a] We therefore assume that the profiles are distributed in a

[a] Similar assumptions are examined in Refs. 5 and 8.

way that one or a number of profiles in some "places" in the graph space have a high likelihood and the other profiles, being more and more distant from one of these "center" profiles, have a decreasing likelihood with respect to a distance function that will be defined in Sec. 2.3. Give a set of graphs $G_1, \ldots, G_m \in \Gamma$, $m \in \mathbb{N}$, this distribution can then be characterized by a function Charac1:

Definition 2.2. Charac1: $\{G_1, \ldots, G_m\} \longrightarrow \{1, \ldots, n\}$.

Every profile is associated with one of the ($n \in \mathbb{N}$) clusters. Another method is to consider the centers of the clusters and to take into account some characteristics of the inner cluster structure. Such a characterization may be the set of these cluster attributes (Charac2):

Definition 2.3. Charac2:$= \bigcup_{i=1,\ldots,n}\{(\mu_i, \sigma_i, A_i)\}$.

Here μ_i is the center graph of cluster i, σ_i is a measure for the distribution within the cluster, e.g. the mean value of the distances of the elements in the cluster i from the center element μ_i and A_i is the number of elements in the cluster. The center values μ_i can easily be found from Charac1 by determining the element in the cluster with the smallest sum of the distances to all the other elements in the same cluster.

In the following we will use a simplification of the graph distribution characterization in Definition 2.3 by taking only the center points into account. We define Charac3:$= \bigcup_{i=1,\ldots,n}\{\mu_i\}$.

2.3. *Graph metrices*

The "shape" of the graph distribution as being characterized by Definitions 2.2 or 2.3 depends strongly not only on the data elements but also on the distance measure between graphs. Several definitions of graph distances are known from the field of pattern recognition.

A simple idea to define such a distance function is to count the number of identical nodes. To achieve a better segmentation of the set of graph profiles, the structure of the connections, i.e. edges in the graphs, should be taken into account, too. A measure for such a structural similarity is the size of the largest common subgraph.

In Ref. 4, it was shown that for two non-empty graphs G_1 and G_2 and the largest common subgraph $lcS(G_1, G_2)$, the function $d(\cdot, \cdot)$ has the mathematical properties of a metrics:

Definition 2.4. $d(G_1, G_2) := 1 - \frac{|lcS(G_1,G_2)|}{\max(|G_1|,|G_2|)}$.

($|\cdot|$ denotes the number of nodes in a graph.) A similar graph distance was defined in Ref. 12.

2.4. *Estimation of a graph distribution and prediction*

The previously defined metrices can be applied to estimate the shape of a graph distribution considering the distribution characterization Charac3 in Sec. 2.2. The navigation graphs can be clustered using a common clustering techniques like nearest neighborhood clustering as described in Ref. 7 and by using one of the distance functions given in Sec. 2.3. Further investigations concerning the shape of the inner cluster distributions according to Charac2 in Sec. 2.2 can then be made.

In order to measure the quality of such a distribution estimation it may be helpful to determine the distance between a real distribution that is known in advance and an estimation of this distribution.

Let $G_1, \ldots, G_m$ be a number of elements in Γ, $\mu_1, \ldots, \mu_n$ $(n \in \mathbb{N})$ be the real cluster centers characterizing the graph distribution and $d(\cdot, \cdot)$ be the distance between two graphs according to one of the definitions in Sec. 2.3. Let $\delta(G) := \min\{d(G, \mu_j) | j = 1, \ldots, n\}$ with $G \in \{G_1, \ldots, G_m\}$. Given an estimation of the cluster centers $\hat{\mu}_1, \ldots, \hat{\mu}_n$, define the estimation error (*err*):

Definition 2.5. $\mathrm{err} := \sum_{i=1,\ldots,n} \delta(\hat{\mu}_i)$.

Obviously, *err* decreases, if the estimation result gets better, i.e. the estimated cluster centers move towards the real ones.

Knowing the estimated distribution of navigation graphs we can describe a prediction technique to find future navigation steps of a specific user, if we assume that the new profile follows the same distribution as the former ones. One way is to compare the new navigation profile to the estimated cluster centers and to find the closest center. Given the estimated cluster centers $\hat{\mu}_1, \ldots, \hat{\mu}_n$ and the new profile G, in this method $d1_j := d(G, \hat{\mu}_j)$ has to be minimized in j, where $d(G, \hat{\mu}_j)$ is a distance of G to the cluster center $\hat{\mu}_j$ as defined in Sec. 2.3. This center element $\hat{\mu}_j$ is the prediction of the profile G.

A further possibility is to take into account the absolute probability that a user profile belongs to a cluster. This probability can be estimated by the relative number of elements in the cluster. The minimization of $d2_j := d(G, \hat{\mu}_j)\frac{1}{1+A_j/A}$ in j takes this absolute probability into account, where A is the number of observed profiles, A_j is the number of patterns in cluster j. These functions will be tested in the following. Figure 1 shows the basic steps of the estimation and the prediction algorithm. The distribution estimation as described above can be done offline. For most of the applications, like navigation support however, the prediction step has to be done in real-time.

In the following experiments the subsequent method was applied to measure the prediction quality for real or generated profiles: a set of profiles is decomposed into a testing and a training set. The training set is used for the distribution estimation as described above. The testing set is used to estimate the prediction quality. Each graph G in the testing set is decomposed into two graphs (H_G and T_G) with regard to a previously fixed size of H_G. H_G is classified with respect to the estimated

Prediction procedure

(offline) Data acquisition
Computation of the distance matrix
Clustering procedure
Distribution estimation

(online) Registration of a new (partial) user profile
Computation of the distances to the estimated cluster centers
Classification of the new profile according to the estimated distribution and a classification function
Prediction of future navigation decisions according to the classification result

Fig. 1. Procedures for distribution estimation and prediction of new navigation decisions.

distribution. H_G is used to find the prediction P_H (the nearest center graph). Then the prediction P_H and the "original" subgraph T_G are compared. The prediction quality values $1 - d(P_H, T_G)$ are added up (the sum will be refered to as *recall* value) and represent the measure for the prediction quality.

3. Experiments

It is the aim to show some of the properties of the described distribution estimation and classification with randomly generated navigation profiles, where the distribution (i.e. Charac3 in Sec. 2.2) of the original data is known in advance and can be compared to the estimation results. The simulation process starts by defining a graph space Γ as defined in Sec. 2.1. A number of graphs is then computed randomly with equal distribution, the number of nodes being identical and a fix number of edges. These graphs represent the real center graphs. Then a sequence of graphs, the simulated graph data, is computed. Each graph is obtained by randomly choosing one of the real center graphs and a number for the label errors. The error value is chosen according to (the positive part of) a discrete Gaussian $N(0, \sigma)$ distribution (σ will be refered to as *deviation* in the following). The simulated graph is computed by changing a number of node labels of the center graph, equal to the number of label errors.

In a first experiment we examined the prediction quality, supposing that the distribution characterization is already known. Two different classification functions are tested. The number of elements in D is 30, the number of nodes in each graph is 25, with 30 edges. The graphs were computed from 2 original graphs ($n = 2$), being the real distribution characterization. The number of identical simulations was 10. The estimation error defined in Sec. 2.4 was used here. Figure 2 shows the estimation error based on the minimization of $d1(\bullet)$ and $d2(+)$ in Sec. 2. In the experiment the deviation of label errors is changed. As can be seen, the prediction based on minimization of $d2$ shows better results for higher values of the label error.

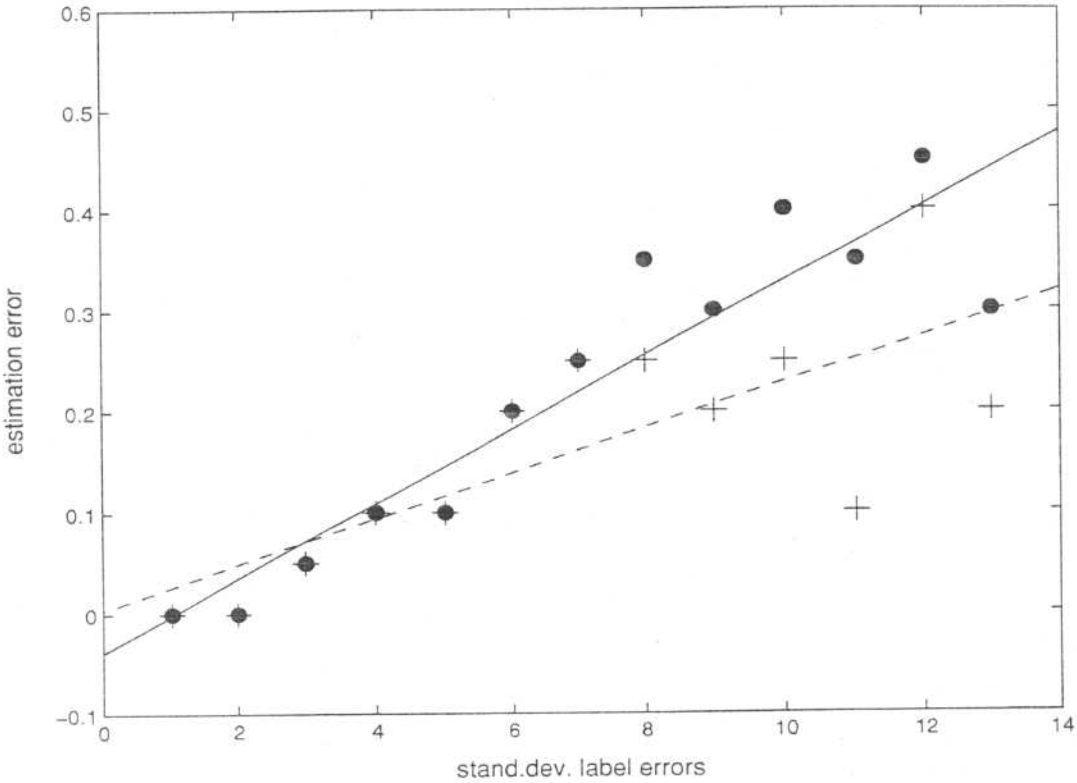

Fig. 2. Classification experiment of user profiles by minimizing $d1(\bullet)$ and $d2(+)$ in Sec. 2.

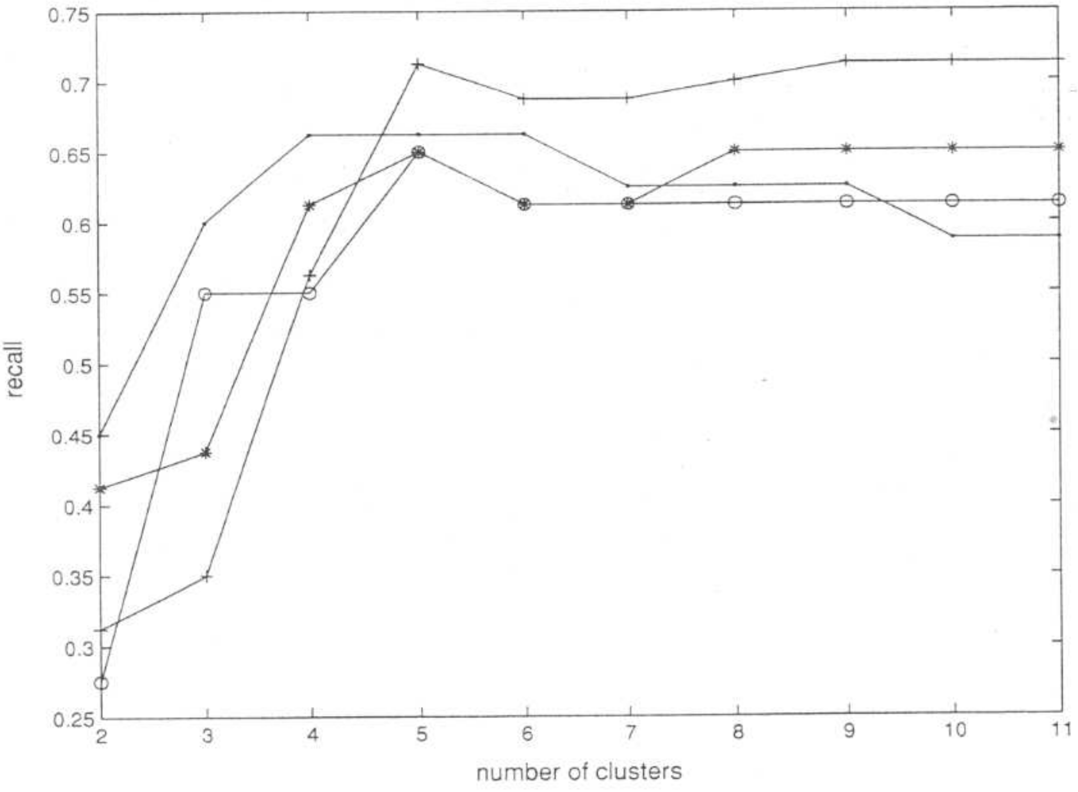

Fig. 3. Prediction quality (recall) for 5 original clusters and a changing number of assumed clusters (identical simulation parameters are used 4 times).

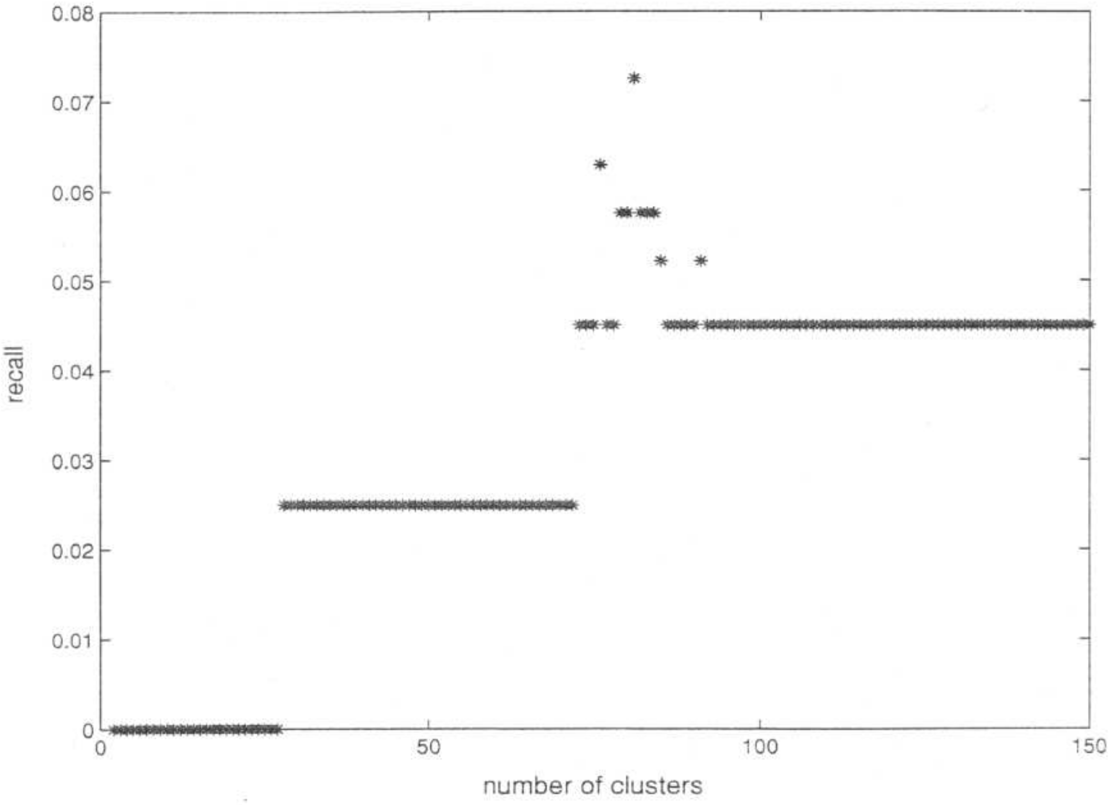

Fig. 4. Prediction quality (recall) for observed profiles, 20 testing and 150 training data.

This result was expected since more information about the shape of the distribution is used in the case of $d2$.

In the previous experiment the original number of clusters was known. Instead, in a second experiment we analyzed the effect of an assumed number of clusters on the prediction quality measure presented in Sec. 2.4. Figure 3 shows the result of this experiment, where 4 identical simulations were performed, the original number of clusters is 5, the deviation is 2. In the experiment 80 training data and 20 testing data were generated. The x-axis in Fig. 3 shows the assumed number of clusters in the prediction process, the y-axis shows the prediction quality (recall).

It can be seen, that the prediction quality reaches a (first) maximum, when the assumed cluster number is near the original cluster number. This method makes it thus possible to estimate the real cluster number by maximizing the prediction quality with regard to the assumed cluster number.

Figure 4 shows the same experiment for observed navigation profiles, which were extracted from access-log files of the web server of our institute. 150 training profiles were used for the distribution estimation and 20 profiles were used for the testing process. The graph shows a maximum near a value of 80 clusters. In this experiment, the "original" cluster number is not known and the experiment only shows, that a cluster value of about 80 leads to a maximal prediction quality with respect to the considered data set.

4. Conclusion and Further Aspects

In the article an estimation technique for graph distributions was presented, that applies clustering of a set of graphs based on a definition of a distance between graphs. The process is based on a characterization of the distribution of graphs, which is difficult to describe directly. The characterization can then be applied for a prediction of a new user profile, presuming that the new navigation graph follows the same distribution.

Some properties of the algorithm like the convergence for two different classification functions were shown with the use of randomly generated graphs. The advantage of the use of simulated data is the knowledge about the distribution that isn't known for real data. A method was presented and applied to real observations to measure the prediction quality.

The presented prediction method has the advantage compared to Markov modelling that a multi-step-prediction can easily be done and that not only sequences of navigation steps but also navigation graphs i.e. sets of navigation steps can be taken into account. A graph modelling of user decisions can be of advantage if e.g. caching strategies in the web cause distorted navigation profiles or if the actual navigation decisions have to be considered.

There are more refined methods to describe a distribution of graphs conceivable. A first improved method is given in Definition 2.3, however further improvements should be developed. More refined graph distances can be defined, e.g. string-edit distances.[9] Additionally the prediction quality has to be examined closely for real data. The time requirements of the prediction algorithm are very important because this step has to be done in real-time if the prediction result is used e.g. for a navigation support tool. Further improvements of the system with respect to learning from additional information about a user or the web site are of interest.

References

1. R. Agrawal, T. Imielinski and A. N. Swami, Mining association rules between sets of items in large databases, eds. P. Buneman and S. Jajodia, *Proc. 1993 ACM SIGMOD Int. Conf. Manag. Data* (1993) 207–216.
2. R. Armstrong, D. Freitag, T. Joachims and T. Mitchell, WebWatcher: A learning apprentice for the World Wide Web, *AAAI Spring Symp. Inf. Gathering Heterogeneous Distr. Env.* (1995) 6–12.
3. Bollobas, *Graph Theory*, 3rd edn. (Springer, 1999).
4. H. Bunke and K. Shearer, A graph distance metric based on the maximal common subgraph, *Pattern Recognition Lett.* **19** (1998) 255–259.
5. E. Carmel, S. Crawford and H. Chen, Browsing in hypertext: A cognitive study, *Trans. Syst. Man Cybernet.* **22** (1992) 865–883.
6. M. S. Chen, J. S. Park and P. S. Yu, Data mining for path traversal patterns in a web environment, *16th Int. Conf. Distr. Comput. Syst. (ICDCS)* (1996) 385–392.
7. B. S. Everitt, *Cluster Analysis*, 3rd edn. (Edward Arnold, 1993).
8. C. Hoelscher and G. Strube, Web search behavior of Internet experts and newbies, *Proc. 9th Int. World Wide Web Conf.* (2000) 337–346.

9. B. Messmer and H. Bunke, Efficient graph matching algorithms for preprocessed model graphs, PhD Thesis, Bern University, 1996.
10. M. Perkowitz and O. Etzioni, Towards adaptive web sites: Conceptual framework and case study, *Artif. Intell.* **118**, 1–2 (2000) 245–275.
11. R. Sarukkai, Link prediction and path analysis using markov chains, *Comput. Netw.* **33**, 1–6 (2000) 377–386.
12. W. D. Wallis, P. Shoubridge, M. Kraetz and D. Ray, Graph distances using graph union, *Pattern Recognition Lett.* **22**, 6–7 (2001) 701–704.
13. I. Zukerman, D. Albrecht and A. Nicholson, Predicting users' requests on the WWW, *Proc. 7th Int. Conf. User Modeling, UM'99* (1999) 275–284.