Prediction of Navigation Profiles in a Distributed Internet Environment through Learning of Graph Distributions

Dirk Kukulenz

Kiel University, Institute of Computer Science,
Preusserstr. 1-9, 24105 Kiel, Germany
dku@ks.informatik.uni-kiel.de

Abstract. Collaborative filtering techniques in the Internet are a means to make predictions about the behaviour of a certain user based on the observation of former users. Frequently in literature the information that is made use of is contained in the access-log files of Internet 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 Internet 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 beeing 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 will make it possible to predict future navigation decisions of new users. Results with randomly generated graphs show properties of the new algorithm.

1 Introduction

In many applications in the field of Internet research it is important to estimate the relevance of data objects available in the Internet for a specific user or a group of users. In the field of *content-based learning* the estimation is based on the behaviour 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 navigation on an Internet site easier, to improve the quality of web sites or to find groups of consumers or interest groups.

In [10] a procedure is 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, that are based on sets of requested data objects. In [2] a navigation support system is presented that learns from search words and browsing decisions of users, applying a reinforcement learning technique. In [14] and [12] techniques for presending documents on the WWW are described that apply different kinds of Markov-learning techniques.

P. De Bra, P. Brusilovsky, and R. Conejo (Eds.): AH 2002, LNCS 2347, pp. 233–241.

The information that is known about a specific user in the case of [10] and [14] is the log data of Internet servers. Each request is stored in the so-called access-log file containing information about the time of a request, the IP-address of a client and the (IP-address of the) requested data object. However, different caching strategies are used in the Internet, with the intention 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's side.

In [14] the actual navigation path is estimated using the access-log information. Here we will however use an idea presented in [2] to register the actual set of navigation decisions of a client on the server's side. A specially developed 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 and an id-number assigned to the client. By this means navigation decisions of Internet users on the considered website can be registered on the server's side. This method makes it also possible to register the number of navigation decisions in a distributed Internet environment, i.e. the navigation decisions between data objects on a number of Internet servers.

Our collaborative filtering procedure is based on these sets of navigation decisions of users. In the field of data mining algorithms are presented to find sets with high frequencies [1]. Related to that, in [6] an algorithm is presented to find frequent navigation sequences in the Internet. 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 constitute 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 are presented e.g. in [9], [4], [13]. We will use one of these functions together with an application of nearest neighbourhood 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 [14], 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 distibutions and the prediction technique of future navigation decisions will be described. In Sect. 3 some estimation examples with randomly generated graphs are presented 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 that can be acquired about a specific Internet user on the server's side with the described method 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 Internet site, like web pages, images, scripts, etc . Let 'D' denote the set of data objects in the Internet site, having an (own) URL address.

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

Definition 1. *A (profile-) graph or navigation profile is a 4-Tupel* G=(V,E,μ,ν). *V is a set of nodes and $E \subseteq V \times V$ is a set of edges. Function $\mu : V \rightarrow L_V \subset D$ assigns labels to the nodes. Function $\nu : E \rightarrow L_E$ assigns labels to the edges. Let $< G >$ be the set of all graphs following the preceding definition. This set will be denoted as 'graph space' based on D. Let $\{G\} \subseteq < G >$ denote a set of graphs.*

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 the definitions of a subgraph, a graph and subgraph isomorphism, graph-edit operations and an error correcting subgraph isomorphism are used that are common in the field of graph theory or artificial intelligence and that are presented e.g. in [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 $< G >$ 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 [5], [8]. We therefore assume that the profiles are distributed in a 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 'central' profiles have a decreasing likelihood with respect to a distance function that will be defined in Sect. 2.3. This distribution can then be characterized by a function:

$$Characl : \{G\} \longrightarrow \{1, ..n\}$$

Here, every profile is associated with one of the clusters. Another method is to consider the centers of the clusters and to take into account some characteristics of the inner cluster structure. A characterization of the graph distribution is then the set of these cluster properties:

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

where 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 $Charac.1$ 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 $Charac2$ by taking only the center elements into account.
We define: $Charac2' : \bigcup_{i=1,..n}\{\mu_i\}$.

2.3 Graph Metrices

The 'shape' of the graph distribution as being characterized by $Charac1$ or $Charac2$ 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 however, 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 [4] it was shown that for two non-empty graphs G_1 und G_2 and the largest common subgraph $lcS(G_1, G_2)$ the function

Definition 2. $d(G_1, G_2) := 1 - \frac{|lcS(G_1,G_2)|}{max(|G_1|,|G_2|)}$

has the mathematical properties of a metrics ($|.|$ denotes the number of nodes in a graph). A similar graph distance was defined in [13]. The disadvantage of this metrics is that possible similarities between different nodes can't be taken into account. Such similarities between the type of nodes that are considered here, i.e. data objects, have been examined for textual data in the field of information retrieval [11]. They are important for the automatic indexing of web pages for the realization of search engines. One well-known distance measure is the *tfidf*-Norm, in which text pages are converted into vectors of weights of words that can be compared with the help of the cosine between the vectors.

A distance measure for two graphs G_1 and G_2 making it possible to take such similarities into account is the following function, where Δ is a set of graph-edit operations and C is a cost function for the edit oprations as described in [9]:

Definition 3. $d(G_1, G_2) := min_{\Delta}\{\ C(\Delta)\ |$ *there exists an error-correcting-subgraph-isomorphism* f_{Δ} *from* G_1 *to* $G_2\}$

The distance function in definition 3 is not symmetric. In order to create a symmetric distance function it is possible to take the minimum of $d(G_1, G_2)$ and $d(G_2, G_1)$.

In the following we will however work with the distance in definition 2 which is easier to implement and faster to compute.

2.4 Estimation of a Graph Distribution

The previously defined metrices or distance functions can now be applied to estimate the shape of a graph distribution considering the distribution characterization $Charac2'$ in Sect. 2.2. The navigation graphs can be clustered using a common clustering technique like nearest neigbourhood clustering as described in [7] and by using one of the distance functions given in Sect. 2.3. Further investigations concerning the shape of the inner cluster distributions according to $Charac2$ in Sect. 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 $G_1, ..G_n$ be the elements in $\{G\}$, $H_1, ..H_m$ $(m \leq n)$ be the real cluster centers characterizing the graph distribution and $d(G_1, G_2)$ be the distance between two graphs according to one of the definitions in Sect. 2.3. Let $\delta(G) := min\{d(G, H_j)|j = 1, ..m\}$ with $G \in \{G\}$.

Definition 4. *Given an estimation of the cluster centers* $\hat{H}_1, ..\hat{H}_m$*, let* $err := \sum_{i=1,..m} \delta(\hat{H}_i)$.

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 $\hat{H}_1, ..\hat{H}_m$ and the new profile G, in this method $d1_j := d(G, \hat{H}_j)$ has to be minimized in j where $d(G, \hat{H}_j)$ is a distance of G to the cluster center $\hat{H}_j$ as defined in Sect. 2.3. This center element $\hat{H}_j$ can then be expected to have a high relevance for the user.

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 $d2_j := d(G, \hat{H}_j)\frac{1}{1+A_j/A}$ 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 section. The basic steps of the estimation and prediction algorithm are:

- Data acquisition

Distribution estimation (offline)

- Computation of the distance matrix
- Clustering procedure
- Distribution estimation

Prediction (online)

- Registration of a new (partial) user profile
- Computation of 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 results

The distribution estimation as described above can be done offline. For most of the applications like navigation support, the prediction step has to be done in real-time.

3 Simulation Results with Randomly Generated Graphs

It is our aim to show some of the properties of the described distribution estimation and classification with randomly generated navigation profiles where the distribution (i.e. $Charac2'$ in Sect. 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 $< G >$ as defined in Sect. 2.1. A number of graphs will then be 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 will be computed presenting the simulated graph data. 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 a discrete Gaussian $N(0, \sigma)$ distribution. The simulated graph is computed by changing a number of node labels of the center graph, equal to the number of label errors.

In Fig. 1 the dependence of the estimation quality according to definition 4 on the number of graphs in the sequence of navigation profiles is shown. 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 (m=2), constituing the real distribution characterization. The number of identical simulations was 10. In Fig. 1 each value is the mean value of the estimation errors in the identical simulations. The graph metrics applied here for the clustering and the estimation quality measurement is the subgraph metrics in definition 2. As can be expected, the estimation error decreases, when the number of graphs increases since more information about the distribution is available for the estimation process.

In a second experiment we examined the prediction quality supposing that the distribution characterization is already known. A number of profiles were

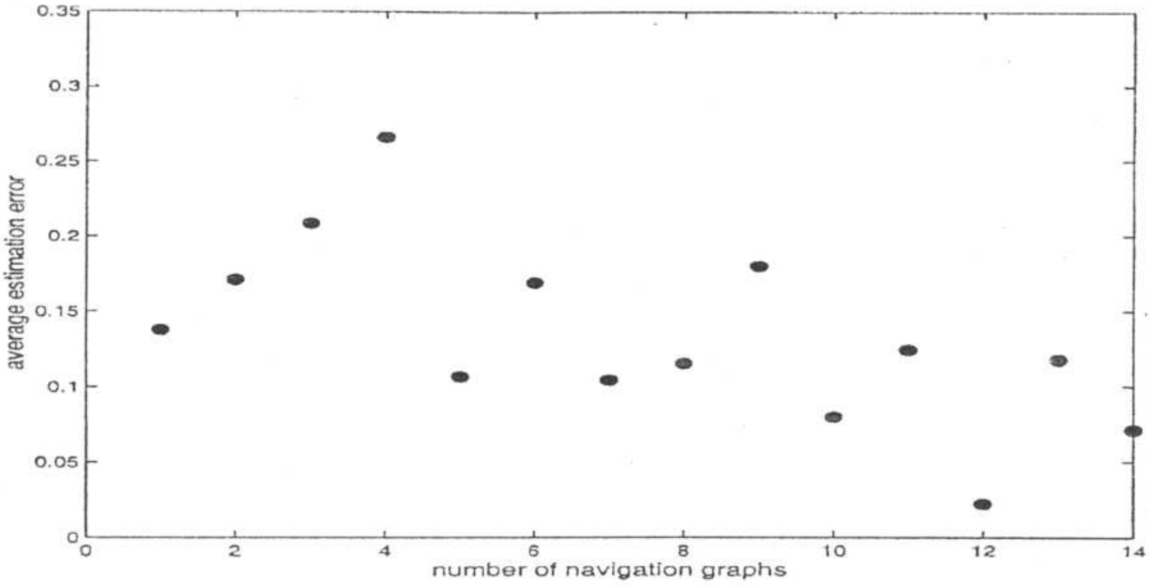

Fig. 1. Dependence of the estimation error on number of graphs

generated, following this distribution as described above. The percentage ($\times\frac{1}{100}$) of missclassifications was determined, denoted as 'classification error'.

Figure 2 shows the classification error based upon the minimization of $d1$ (•) and $d2$ (+) in Sect. 2.4. In the experiment the deviation of label errors is changed. As can be seen, the prediction based upon minimization of $d2$ shows better results for higher values of the label error. This result was expected since more information about the shape of the distribution is used in the case of $d2$.

4 Conclusion and Further Aspects

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

Some characteristics of the algorithm like the convergence for an increasing number of patterns were shown by means of randomly generated graphs. The advantage of the use of simulated data is the knowlege about the distribution that can't be known for real data.

Compared to Markov modelling this estimation method has the advantage 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 Internet cause distorted navigation profiles or if the actual set of navigation decisions has to be considered.

One problem to discuss when recommending navigation decisions is the so-called 'snowball effect'. If the system learns a wrong path and presents it to other

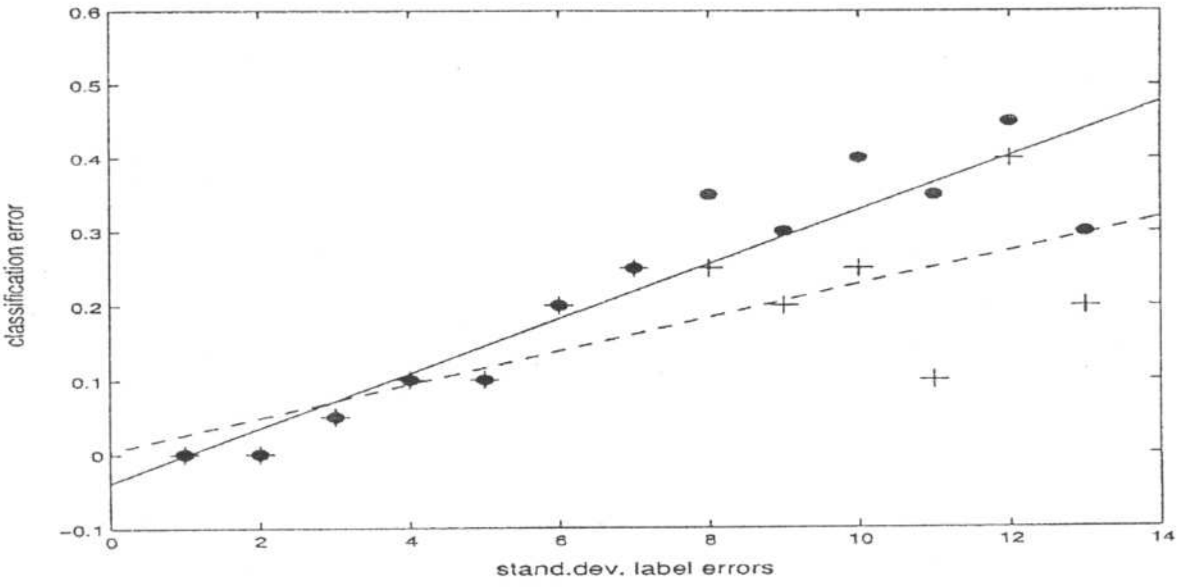

Fig. 2. Classification experiment of user profiles by minimizing d1 (•) and d2 (+) in Sect. 2.4.

users, they may also follow this wrong path and the system will learn again the wrong path. This problem however becomes only important if a high percentage of users actually use the support system. The registration of navigation decisions described in Sect. 1 is also possible for users who don't apply the support system.

There are more refined methods conceivable to describe a distribution of graphs. A first improved method is given in definition 2.2, however further improvements should be developed. Different and more refined graph distances can be defined e.g. taking into account node distances as described in definition 3. 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 Internet site are of interest.

References

1. R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc.of the ACM SIGMOD Conference on Management of Data*, 1993.
2. R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell. Web watcher: A learning apprentice for the www. In *AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, pages 6–12, 1995.
3. Bollobas. *Graph theory*. Springer, 3 edition, 1999.
4. H. Bunke and K. Shearer. A graph distance metric based on the maximal common subgraph. In *Pattern Recognition Letters*, volume 19, pages 255–259, 1998.
5. E. Carmel, S. Crawford, and H. Chen. Browsing in hypertext: A cognitive study. In *Transactions on System, Man and Cybernetics*, volume 22, pages 865–883, 1992.

6. M. Chen, J.S. Park, and P.S. Yu. Data mining for path traversal patterns in a web environment. In *Proc. of the 16th ICDCS*, volume 16, pages 385–392, 1996.
7. B.S. Everitt. *Cluster Analysis.* Edward Arnold, 3 edition, 1993.
8. C. Hoelscher and G. Strube. Web search behavior of internet experts and newbies. In *World Wide Web Conf*, volume 9, 2000.
9. B. Messmer and H. Bunke. *Efficient graph matching algorithms for preprocessed model graphs.* PhD thesis, Bern University, 1996.
10. Mike Perkowitz and Oren Etzioni. Towards adaptive web sites: Conceptual framework and case study. *Artificial Intelligence*, 118(1–2):245–275, 2000.
11. G. Salton. Developments in automatic text retrieval. *Science*, 253:974–979, 1991.
12. R. Sarukkai. Link prediction and path analysis using markov chains. In *Intern. World Wide Web Conf.*, 2000.
13. W.D. Wallis, P. Shoubridge, M. Kraetz, and D. Ray. Graph distances using graph union. In *Pattern Recognition*, volume 22, pages 701–704, 2001.
14. I. Zukerman, D.Albrecht, and A.Nicholson. Predicting users' requests on the www. In *UM99 – Proceedings of the Seventh International Conference on User Modeling*, 1999.