A Recommendation Model Based on Latent Principal Factors in Web Navigation Data

Yanzan Zhou, Xin Jin, Bamshad Mobasher
{yzhou,xjin,mobasher}@cs.depaul.edu
Center for Web Intelligence
School of Computer Science, Telecommunication, and Information Systems
DePaul University, Chicago, Illinois, USA

Abstract

Discovery of factors that lead to common navigational patterns can help in improving online information presentation as well as in providing personalized content to users. It is, therefore, necessary to develop techniques that can automatically characterize the users' underlying navigational objectives and to discover the hidden semantic relationships among users as well as between users and Web objects. Typical approaches to Web usage mining, such as clustering of user sessions, can discover usage patterns directly, but cannot identify the latent factors, intrinsic in users' navigational behavior, that lead to such patterns. In this paper, we propose an approach based on a latent variable model, called *Iterative Principal Factor Analysis*, to discover such hidden factors in Web usage data. The hidden factors are then used to create aggregate models of common user profiles which are, in turn, used to provide dynamic recommendations to users. Our experimental results, performed on real Web usage data, verify that the proposed principal factor approach results in better predictive user models, when compared to more traditional approaches such as clustering and principal component analysis.

1 Introduction

Users of a Web site, generally exhibit task-oriented navigational behavior which may involve interactions with one or more functional units within the site. The tasks performed by users are reflected in sets of pages or Web objects that are commonly accessed together. Pages may be related in this way because they have similar content, because they serve the same user task (such as a product purchase, or interacting with an online application), or because they related to products that are frequently purchased together.

Observations of task-oriented navigational patterns can shed light on the "behavior types" associated with typical site users. For example, in an e-commerce site, there may be many user groups with different (but overlapping) behavior types. These may include visitors who are goal-oriented showing interests in a specific product category, or visitors who tend to first browse many similar products in the same category before placing items in their shopping cart (e.g., bargain shoppers), or visitors who engage in "window shopping" by browsing through a variety of product pages in different categories without placing any items into their shopping carts. Identifying these user tasks and behavior types may, for example, allow a site to distinguish between those who show a high propensity to buy versus whose who don't. This, in turn, can lead to automatic tools that can tailor the content of pages for those users accordingly.

Web usage mining techniques [21], which capture navigational patterns of Web users, have achieved great success in various application areas such as Web personalization [13, 15, 17], link prediction and analysis [18], Web site evaluation or reorganization [19, 20], and e-commerce data analysis [10]. A variety of Web usage mining algorithms have been developed to discover posterior Web site usage patterns, including using association rules to derive frequently co-occurring pageview sets [14, 11], distance or model-based clustering on either pageviews or user sessions [12, 16], and using stochastic models to make predictions based on sequential patterns [18, 6]. Generally, these techniques capture standalone usage patterns at the

pageview level. They, however, do not capture the intrinsic characteristics of Web users' activities, nor can they quantify the underlying and unobservable factors that lead to specific navigational patterns.

Latent variable models, such as Principal Component Analysis (PCA), have been widely used in a variety of areas, most prominently for latent semantic indexing in information retrieval [5, 1]. More recently, probabilistic Latent Semantic Analysis [8, 3] has gained attention because of its flexibility and its ability to leverage probabilistic inference. Factor analysis models, which are established statistical models [7], are less discussed in data mining research than PCA, possibly due to the model complexity and solution indeterminacy. Although the latent variable models, such those mentioned above, have been studied and applied extensively, few of these approaches have been employed in the context of Web mining, in general, or for the discovery of Web usage patterns, in particular.

In this paper we propose a Web usage mining approach to personalization based on Principal Factor Analysis (PFA). Specifically, we use an algorithm called *Iterative Principal Factor* (IPF) Analysis to discover the latent factors which characterize relationships among pages based on their usage. We, then, use the discovered factors to create aggregate representations of common navigational patterns among user. We call such patterns *aggregate usage profiles* (or simply usage profiles). Finally, the usage profiles are used, together with a current user's active session, to generate dynamic recommendations for the active user.

The primary assumption behind all latent variable models is that there is a set of common hidden factors which "explain" a set of observations in co-occurrence data. In our case, we are interested in automatically identifying the common factors that lead groups of users to visit certain pageviews together. Thus, in the Web usage mining context, user sessions represent the set of observations, while pageviews contained in the sessions represent the variables.

It should be noted that the more common Principal Component Analysis (PCA) can also be used to identify latent variables (principal components). However, while PCA finds the principal components that maximize the *total variance* of variables, the principal factor model discovers the structural factors underlying the *common variance* of variables and excludes unique variances not related to other variables. Because of this difference, PCA is commonly used for tasks such as dimensionality reduction in which preserving the overall relationships between the variables is important. On the other hand, the Principal Factor Analysis is better suited for identifying the factors that characterize unique relationships among sets of variables. Indeed, our experimental results verify that the IPF approach results in better predictive models for generating dynamic recommendations.

The paper is organized as follows. In Section 2 we discuss the modeling aspects of the principal factor model for Web usage data, and we present the iterative principal factor extraction algorithm. In Section 3 we introduce our approach for constructing aggregate usage profiles on the basis of extracted latent factors. Our recommendation algorithm based on the discovered usage profiles is introduced in Section 4. Section 5 presents our experimental results based on real usage data.

2 Web Usage Patterns as Latent Factors

We begin with the raw server logs of the site under analysis and perform the appropriate preprocessing steps such as data cleaning, pageview identification, sessionization, and Web robot detection. A detailed discussion of the various steps in usage data preprocessing is beyond the scope of the current paper. Interested reader can find these details in [4]. The preprocessing tasks result in a set of n pageviews $P = \{p_1, p_2, \cdots, p_n\}$ and a set of m user sessions $U = \{u_1, u_2, \cdots, u_m\}$. A *pageview* is an abstraction representing a set of objects associated with a single user request. So, a pageview may consist of an individual Web page, multiple pages (such as in the case of HTML frames), or a combination of pages and objects (e.g., database records associated with dynamic applications). Following multivariate data analysis conventions, if we treat pageviews as variables and user sessions as observations, we can represent the usage data as an $m \times n$ user-pageview matrix $\mathbf{S} = [w(u_i, p_j)]$, where each element $w(u_i, p_j)$ is the significance weight (a function of time duration) of pageview j in user session i.

Our goal is to discover a set of latent factors $C = \{c_1, c_2, \cdots, c_k\}$ from this data that "explain" the underlying relationships among pageviews. These latent factors will be closely related to Web site's functional structure and Web users' actual navigational tasks. The identification of such factors allows us to transform

the usual representation of users sessions as a set or sequence of pageviews (pageview-level representation) to a higher-level representation over the space of latent factors. Specifically, given the matrix $\mathbf{S}$, of user-pageview observations, each user session, $\vec{u_i} = [w(u_i, p_j)]_{1\times n}$, can be transformed into a higher-level factor space representation $\vec{u_i^c} = [\omega(u_i, c_j)]_{1\times k}$, where ω_{ij} is a significance weight of user u_i with respect to factor c_j.

During a given session, a user may perform a single task or multiple tasks simultaneously. These tasks a captured by the discovered latent factors. Thus, the higher-order factor-level representation of user sessions can be used to identify the primary tasks performed by users according to the associated weights in each factor dimension. Users with relatively high weights on a given factor dimension could be regarded as prototypical users performing the associated task. By aggregating the profiles of such prototypical users we can create user models that can be used for generating recommendations to other users exhibiting similar navigational behavior. We divide this learning task into two major steps. First, we extract latent factors representing pageview association patterns from usage data. Secondly, aggregate usage profiles representing prototypical user navigational patterns are derived to be used in dynamic Web personalization.

2.1 The Latent Factor Model for Web Navigational Patterns

Let $E(p_i)$ denote the expectation of random variable p_i and $var(p_i)$ denote the variance of p_i (pageviews). To simplify our discussion we assume $E(p_i) = 0$ and $var(p_i) = 1$. This assumption can be made without loss of generality given that we can always obtain a z-score normalized observed value by subtracting the sample mean $\bar{p}_i$ and dividing by the standard deviation). Let $\mathbf{P}^T = [p_1, p_2, \cdots, p_n]$. Then the covariance matrix of $\mathbf{P}$ is defined as $cov(\mathbf{P}) = [E(p_i p_j)]_{n\times n}$, and we have correlation matrix $corr(\mathbf{P}) = cov(\mathbf{P})$ given our zero mean and unit variance assumptions.

We assume that Web users' accesses to a pageview p_i are influenced by a set of common latent factors $\mathbf{C}^T = [c_1, c_2, \cdots, c_k]$ as described in the linear function: $p_i = l_{i1}c_1 + l_{i2}c_2 + ... + l_{ik}c_k + \Delta_i$, where coefficient l_{ij} is the loading weight of pageview p_i on unobservable common factor c_j. Unique factor Δ_i represents unobservable unique portion of p_i that are not accounted for by k common factors (such as individual variable-specific variation and measurement error). This can be written in matrix format as

$$\mathbf{P} = \mathbf{LC} + \Delta, \tag{1}$$

where $\mathbf{L}$ is the loading coefficient matrix. Since both $\mathbf{C}$ and Δ are unobservable, there are too many unknown components in equation 1, we cannot solve the equation directly. However, we can approach the solution by making the following assumptions on $\mathbf{C}$ and Δ:

1. Both common and unique factors have zero mean: $E(\mathbf{C}) = \mathbf{0}$, and $E(\Delta) = \mathbf{0}$;
2. Common factors have unit variances and are uncorrelated with each other (orthonormal), i.e., $cov(\mathbf{C}) = E(\mathbf{CC}^T) = \mathbf{I}_{k\times k}$ (identity matrix)
3. Unique factors have their own variances while they are uncorrelated with each other. Let the diagonal matrix $\Psi = [\psi_i]_{n\times n}$, where $\psi_i = var(\Delta_i)$, then $cov(\Delta) = E(\Delta\Delta^T) = \Psi$,
4. Common factors are uncorrelated with unique factors, i.e., $\text{cov}(\mathbf{C}, \Delta) = E(\mathbf{C}\Delta^T) = \mathbf{0}$.

Note that these assumptions are necessary to obtain an orthogonal factor model and the second assumption could be relaxed to be $cov(\mathbf{C}) \neq \mathbf{I}$ resulting in a more complicated *oblique factor model* [9]. That is, some of these latent factors will be correlated. As a rule of thumb, we can always solve the orthogonal factor model first, followed by some oblique rotation criteria on factor axes as needed. Based on the above assumptions, we can now derive important covariance relationships between pageviews, factors and loading coefficients. For example, let $cov(\mathbf{P}, \mathbf{C}) = [E(p_i, c_j)]_{n\times k}$. Then the covariance between $\mathbf{p}$ and $\mathbf{C}$ can be described as:

$$\begin{aligned} cov(\mathbf{P}, \mathbf{C}) &= E\left[\mathbf{pC}^T\right] = E\left[(\mathbf{LC} + \Delta)\mathbf{C}^T\right] \\ &= \mathbf{L}E(\mathbf{CC}^T) + E(\Delta\mathbf{C}^T) = \mathbf{L}. \end{aligned} \tag{2}$$

The covariance matrix of $\mathbf{p}$ can be described as:

$$\begin{aligned} cov(\mathbf{p}) &= E\left[\mathbf{p}\mathbf{p}^T\right] = E\left[(\mathbf{LC}+\Delta)(\mathbf{LC}+\Delta)^T\right] \\ &= \mathbf{L}E(\mathbf{C}\mathbf{C}^T)\mathbf{L}^T + \mathbf{L}E(\mathbf{C}\Delta^T) + E(\Delta\mathbf{C}^T)\mathbf{L}^T + E(\Delta\Delta^T) = \mathbf{L}\mathbf{L}^T + \Psi. \end{aligned} \tag{3}$$

Equation (3) tells us that the variance of p_i is just the sum of its squared loading coefficients plus a unique variance not related to other variables:

$$var(p_i) = \sum_{j=1}^{k} l_{ij}^2 + \psi_i. \tag{4}$$

Thus, the total variance of pageview p_i is comprised of two parts: the *communality* part, denoted as $h(p_i) = \sum_{j=1}^{k} l_{ij}^2$, which is the common variance that is accounted for by k common factors; and the unique variance part ψ_i that is not explained by common factors. The communality is essentially equivalent to the squared multiple correlation coefficient $\mathbf{R^2}$ in the multiple regression sense, i.e., $h(p_i)$ is the proportion of total variance in p_i that is predictable by the k predictors (in this case, factors).

Equation (2) tells us that loading coefficients have clear statistical meaning: l_{ij} is just the covariance between pageview p_i and factor c_j. That is $\text{cov}(p_i, c_j) = l_{ij}$. l_{ij} is also the correlation coefficient of pageview p_i and factor c_j given our variable standardization assumption.

2.2 The IPF Algorithm for Factor Extraction in Web Navigation Data

There are several algorithms available to estimate the loadings and factors, including maximum likelihood (ML) method, the centroid method, and image factor analysis. For a discussion of these and other approaches see [7]. The ML method usually assumes that the co-occurrence data has a Gaussian distribution. This, however, does not generally hold in the context of Web usage data. Thus, we adapt a more robust and computationally efficient method called iterative principal factor analysis (IPF) to perform the factor extraction.

In IPF, the estimation of factor loading matrix can be solely based on the sample correlation matrix. The sample correlation matrix of the usage data is simply computed as $\mathbf{R}(\mathbf{S}) = \mathbf{S}^T\mathbf{S}$ given that the p_i are z-score standardized. There are three major steps in the IPF algorithm:

1. Initial communality estimates $h(p_i)$. The most popular method is squared multiple correlation coefficient (SMC) [9, 7], which is defined as $h(p_i) = 1 - \frac{1}{\gamma_{ii}}$, where γ_{ii} is the i-th diagonal element of inverse correlation matrix $[\mathbf{R}(\mathbf{S})]^{-1}$. Thus, estimated unique variance $\psi_i = 1 - h(p_i)$ and the reduced correlation matrix becomes $\mathbf{R}^*(\mathbf{S}) = \mathbf{R}(\mathbf{S}) - \Psi$. Note that the diagonal elements of $\mathbf{R}^*(\mathbf{S})$ represent the common variances of pageviews (i.e., communalities).

2. Decomposition of the reduced correlation matrix $\mathbf{R}^*(\mathbf{S})$. Consider an eigen decomposition of a symmetric matrix, we can approximate $\mathbf{R}^*(\mathbf{S})$ by keeping k leading eigen vectors and corresponding k largest eigen values:

$$\mathbf{R}^*(\mathbf{S}) = \mathbf{H}_{(k)}\Lambda_{(k)}^{\frac{1}{2}} \times \Lambda_{(k)}^{\frac{1}{2}}\mathbf{H}_{(k)}^T = \mathbf{L}\mathbf{L}^T$$

   where $\mathbf{H}_{(k)}$ is a matrix consisting of k columns of eigen-vectors corresponding to the descendingly ranked k largest eigen-values of $\mathbf{R}^*(\mathbf{S})$, $\Lambda_{(k)}^{\frac{1}{2}}$ denotes a diagonal matrix whose diagonal elements are square roots of corresponding k largest eigen values of $\mathbf{R}^*(\mathbf{S})$. Thus loading matrix could be estimated as $\mathbf{L} = \mathbf{H}_{(k)}\Lambda_{(k)}^{\frac{1}{2}}$.

3. Updates of communality estimates. From $\mathbf{L}$, we get re-estimated communality as $h(p_i) = \sum_{j=1}^{k} l_{ij}^2$

After the above steps, the diagonal elements of $\mathbf{R}^*(\mathbf{S})$ are replaced with the updated communalities. Then, we iterates steps 2 and 3 until the difference between updated and previous communalities are minimized within a certain threshold.

Given that initial solutions of loading matrices are usually not easily interpretable, appropriate transformations such as orthogonal or oblique factor rotation procedures are usually performed on the loading matrix to arrive at a simpler structure for better interpretability [9, 7]. Based on our experience, the rotated factor patterns generally lead to better quality usage profiles while showing more interpretable patterns as well.

3 Discovery of Aggregate Usage Profiles Based on Latent Factors

In this section we present the algorithm for deriving aggregate usage profiles based on the factor loading matrix obtained through IPF.

3.1 The Factor Loading Matrix

The loading matrix $\mathbf{L}$ tells us how closely each pageview is related to each factor. We can further interpret the meaning of each factor by selecting the most related pageviews. Each user's activity pattern could be represented as a mixture of these factors. These factors summarize the leading k prominent types of user navigational tasks. The weight of a pageview in a certain factor reflects the relative importance of describing the factor.

Furthermore, we can represent each factor as a pageview space vector for later comparison with user sessions (also in pageview space). We can derive an aggregate usage profile for other similar users who have the same dominant factor present in their sessions. This could be done by first comparing the similarity between a user activity session with these factors. Specifically, for each column of the loading matrix $\mathbf{L}$ in the factor model, we can generate a corresponding vector $\vec{c_j} = [l_{1j}, l_{2j}, \cdots, l_{nj}]$, where l_{ij} is the loading of pageview p_i on factor c_j.

3.2 Deriving Aggregate Usage Profiles from Factor Loadings

To obtain aggregate profiles from the discovered factors, we begin by projecting the user sessions (represented as vectors of pagviews) onto the reduced-dimension space of latent factors. The component (factor) scores can be regarded as an inner-product similarity computation of a user with the reduced dimensions. In order to normalize the effect of diverse user browsing styles such as slow-surfers versus fast surfers we use cosine coefficient to compute the user interest weight with respect to different factors. Specifically, we can conceptualize user sessions by transforming a user-pageview matrix $\mathbf{S}$ to a user-factor matrix $\mathbf{S}^c$ as follows:

$$\mathbf{S}^c = \mathbf{SL} = [\omega(u_i, c_j)]_{m \times k},$$

where $\omega(u_i, c_j) = \vec{u_i} \cdot \vec{c_j}$, $\vec{u_i}$ and $\vec{c_j}$ are both normalized to have unit length.

Thus, we have a conceptualized view of user activities beyond the pageview level, and we can further discover the dominant and the secondary factors by examining the associated weights with the corresponding factors. For example, some users may have only one significantly weighted factor, which might indicate that the user is engaged in a single specific task; while other users may exhibit multiple interests during a session, as reflected by high significant weights on more than one factor.

Given this representation, for each of the k latent factors (columns in the user-factor matrix $\mathbf{S}^C$), we choose those users whose factor loading weight is greater than a certain threshold as a representative user associated with that factor. Such a set of users $U^c = \{u_1^c, u_2^c, \cdots, u_m^c\}$ forms a user segment whose members have similar dominant interests while having probably diverse minor interests. In other words, users' cross-interests would be captured in such segments and each member will have different contributing weights. Differentiating these contributing weights is important since they are directly used for generating the aggregate usage profiles.

To create an aggregate representation of the user segment U^c, we compute the centroid of all the vectors in U^c, resulting in a representation of the segment as a set of factor-weight pairs. This is a similar approach as the profile aggregation method, PACT, discussed in [15]. In PACT, user segments were generated by performing clustering on the set of user sessions, and the cluster centroid was regarded as the aggregate usage profile. In contrast, the user segments is the present approach are derived based on the factor loadings obtained from the IPF algorithm. The algorithm of generating aggregate usage profiles is as follows:

1. For each factor c, choose all the user sessions with $\omega(u_i, c) \geq \mu$ to get a candidate session set U^c, where μ is predefined threshold.

2. Represent each user session $u_i \in U^c$ as a pageview vector and compute their centroid pageview vector $\vec{v} = (1/|U^c|) \sum \vec{u_i} \cdot \omega(u_i, c)$, where $|U^c|$ denotes the total number of sessions in set U^c.

3. For each factor c, output page vector $\vec{v}$. This pageview vector consists of a set of weights for pageviews in P, which represent the relative significance of that pageview for the user segment associate with c.

The pageview-weight pairs obtained using the above aggregation process can be further ordered according to the associated weights. In contrast to individual factors, these aggregate usage profiles contain information about changing contexts during Web user's online real navigation. In particular, different (possibly related) tasks performed by a user during a session is reflected by pageviews in the aggregate profile that are contributed by different latent factors. We regard a dominantly weighted pageviews in a usage profile as reflecting the main"theme" (interest) of the associated user segment. Furthermore, non-dominant pageviews, reflect the diversity of minor "themes" in other factors and could be regarded as additional contextual information. We provide some real examples of discovered aggregate usage profiles in Section 5.

4 Using the Latent Factors for Personalization

Web personalization usually refers to dynamically tailoring the presentation of a Web site according to the interests or preferences of individual or groups of users. This can be accomplished by recommending Web resources, such as Web pages or products, to a user by considering the current user's active behavior with his own historic patterns or best matched learned models of other users. The usage profiles generated based on the algorithm of Section 3.2 provide an aggregate representation of all individual users' navigational activities in a particular group. They also provide the basis for automatically generating relevant pageview recommendations.

Here, we use the discovered usage profiles from the IPF algorithm to generate top-N recommendations based on a current user's active session:

1. Each aggregate usage profile described in Section 3 can be conveniently represented as an n-dimensional pageview vector $\vec{v} = [w_1^c, w_2^c, \ldots, w_n^c]$, where w_i^c is the weight associated with pageview p_i in this profile. Similarly, the active user session $\vec{a}$ is represented as $\vec{a} = [a_1, a_2, \cdots, a_n]$, where $a_i = 1$, if pageview p_i is visited, and otherwise $a_i = 0$.

2. Choose the usage profile that best matches the active user session. The similarity match score is defined using the cosine coefficient:

$$sim(\vec{a}, \vec{v}) = \frac{\sum_i (a_i \times w_i^c)}{\sqrt{\sum_i (a_i)^2 \times \sum_i (w_i^c)^2}}$$

3. For the top matched usage profile $\vec{v}^c_{max} = argmax_{\vec{v}} sim(\vec{a}, \vec{v})$ together with the active session $\vec{a}$, we compute a recommendation score: $rec(\vec{a}, p_i)$, for each pageview $p_i \in \vec{v}^c_{max}$:

$$rec(\vec{a}, p_i) = \sqrt{w_i^c \times sim(\vec{a}, \vec{v}^c_{max})}.$$

   Thus, each pageview will receive a normalized recommendation score between 0 and 1. If the pageview p_i is already in the current active session $\vec{a}$, its recommendation score is set to zero.

4. Select the N pageviews with the highest recommendation scores as the top-N recommendation set.

We will present our evaluation results on real usage data for the top-N recommendations in Section 5.

5 Experiments and Evaluation

In this section we evaluate the effectiveness of the PFA model for recommendations on two different data sets. We also provide some examples of the generated aggregate usage profiles.

5.1 Data Sets and Experimental Methodology

The data sets used in our experiments are:

1. **CTI data.** This data set is based on the server logs of the host Computer Science department spanning a one-month period. The initial preprocessed data set contained more 100,000 sessions and over 4,000 pageviews. Further aggregation was performed to "roll up" low-support (infrequently accessed) pageviews to their common root node in the site hierarchy. Furthermore, small sessions (containing less than 6 pageviews) were filtered out. This resulted in a final data set containing 21,299 user sessions and 692 Web pageviews. The site is highly dynamic, involving numerous online applications, including online admissions application, online advising, online registration, and faculty-specific Intranet applications. Thus, we expect the discovered usage patterns to reflect various functional tasks performed by diverse groups of users.

2. **NC data.** This data set is from *Network Chicago* which combines the programs and activities of the Chicago Public Television and Radio (*www.networkchicago.com*). The data was collected over a period of one month. Similar preprocessing and aggregation steps as in the case of the CTI data were also applied here, resulting in a total of 4,987 user sessions and 295 pageviews were included for analysis. In contrast to the CTI data, this site is comprised primarily of static pages grouped together based on their association with specific content areas. In this case, we expect the generated usage profiles to reflect common interests of users in one or more programs represented by the content areas.

Each data set was randomly divided into multiple training and test sets to use with 10-fold cross-validation. The training sets were used to build the models while the test sets were used to evaluate the user segments and the recommendations generated by the models. In all experiments, the results represent averages over the 10 folds.

We extracted 24 factors for both CTI and NC data sets based on the initial scree plots [2]. The scree plot is a plot of eigen-values (communalities of the reduced correlation matrix) against their descending order. In our case, the initial 24 factors, corresponding to the largest eigen values, explain $\geq 50\%$ of the common variance based on the initial communality estimations.

5.2 Examples of Usage Patterns Based on the Latent Factors

Figure 1 shows examples of three aggregate usage profiles from the CTI data set. For each profile, the top-weighted pageviews are listed using the associated pageview titles. For each pageview in a given profile the associated latent factor (a number between 1 and 24), contributing that pageview to the profile, is also given. Finally, we have provided an interpretation of each profile by identifying the dominant and secondary tasks captured by the profile (with their associated factors).

For instance, Profile 1 clearly captures two functional tasks. The first (associated with factor 1) represents the activity of faculty engaged in various online advising tasks, including login to the faculty Intranet, searching for student records, viewing students' course histories, etc. The second (less dominant) task (associated with factor 6) represents other related faculty-specific tasks performed in the same section, such looking at class rosters and events calendar.

	Factor #	Aggregate usage profiles		Interpretation
Profile #1	1	1.00	Intranet main page	
	1	0.71	Online advising – student history	**Dominant Factor:**
	1	0.59	Intranet main page – login	- Online advising system
	6	0.50	Intranet – news	(factor #1)
	1	0.38	Online advising – student search	
	6	0.27	Intranet – calendar	**Secondary Factors:**
	6	0.25	Intranet – faculty page	-Faculty Intranet system
	1	0.24	Intranet – online advising page	(factor #6)
	6	0.18	Intranet – faculty rosters	
Profile #2	2	0.70	Online application-step1	**Dominant Factor:**
	2	0.70	Online application-step2	- Online applications
	2	0.58	Online application-finish	(factor #2)
	2	0.42	Online application-restart	
	2	0.38	Online application-login	**Secondary Factors:**
	9	0.34	Admission- main page	- Admission info search
	9	0.29	Admissions-requirements	(factor #9)
Profile #3	3	0.89	People - faculty evaluation	**Dominant Factor:**
	3	0.89	People - faculty search	- faculty info search
	3	0.75	People - faculty info main page	(factor #3)
	3	0.74	People - full-time faculty list	
	3	0.56	People - faculty news	**Secondary Factors:**
	15	0.46	Programs – course information	- Course info search,
	9	0.30	Student Intranet login	- Making appointments,
	15	0.29	Course news main page	- Course news
	3	0.25	People - part-time faculty list	(factor #9 and factor #15)

Figure 1: Aggregate usage profile examples from CTI data

As another example, Profile 2 also captures two different but related tasks, namely that of a prospective student going through the online admissions application process (factor 2), and that of searching for general admissions information and requirements (factor 9). Note that there may be groups of students who perform each of these tasks independently and separately (reflected in other discovered profiles). However, this particular profile captures the behavior of those students who, after checking on various admissions requirements, have decided to go through the application process during the same session.

As a final example, Profile 3 shows, in part, the activity of current students who are in the process of registering for courses. As reflected by the dominant factor, the profile indicates that one important determining criteria for students' course selection is the faculty teaching the courses, and particularly the student evaluations of the faculty in past courses.

5.3 Evaluation of Recommendations

For evaluating the recommendation effectiveness we use a measure called *hit ratio* in the context of top-N recommendation. For each user session in the evaluation set T, we took the first j pages as a representative of an active user to generate a top-N recommendation set as described in section 4. We then compare the recommendation set in the top matched profile $\vec{v}^{c}_{max}$ with the pageview $(j+1)$ in the evaluation session. If the pageview $(j+1)$ appears in this recommendation set, we consider it as a *hit*. We define the hit ratio as the total number of hits divided by the total number of sessions in the evaluation set.

Note that the hit ratio increases as the value of N (number of recommendations) increases. Thus, we pay special attention to smaller number of recommendations that result in good hit ratios in our experiments (generally between 5 and 10 recommendations).

For each of the data sets, we compare the hit ratio based on the aggregate profiles generated using the IPF algorithm, against those derived based on PCA and PACT approaches. In the case of PCA, we used the standard SVD (Singular Valued Decomposition) approach to identify the principal components. We then used an approach similar to our proposed method, based on IPF, to derive the aggregate usage profiles. In the case of PACT (described earlier), first clusters of user sessions were obtained using k-means clustering, and then the aggregate usage profiles were derived as the cluster centroids.

Figure 2 and 3 depicts the hit ratios vs. recommendation set sizes for both CTI and NC data sets, across the three approaches. The results show that the profiles based on the latent factor model have consistently higher hit ratio values than both PACT and PCA. Such advantage continues in all ranges of recommendation

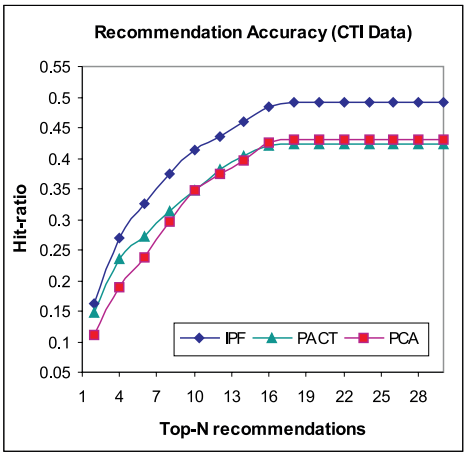

Figure 2: Hit ratio evaluation on CTI data

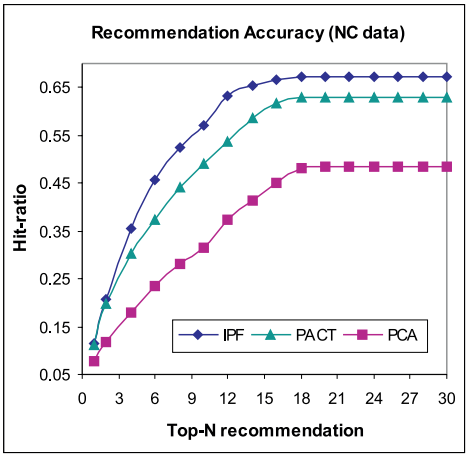

Figure 3: Hit ratio evaluation on NC data

set sizes. Interestingly, the results show that PACT performs in par with, or better than, PCA for the purpose of generating dynamic recommendations. This may suggest that the standard approach of clustering user sessions is better suited for distinguishing among Web user segments. This is likely due to the fact that principal components are designed to maximize the total variance among variables, instead of isolating their communalities (as is the case for the IPF model). However, clustering approahces, in contrast to PCA, cannot directly identify latent factors which describe inter-relationships among pageviews.

6 Conclusions

Every Web site contains functional units that represent a variety of user tasks performed within those units. The navigational patterns of users are, therefore, semantically related to the functional structure of the site. Web site owners may have pre-defined functions and page categories. However, we are particularly interested in how exactly users are utilizing these functions during real visits and the latent factors that underly the task-oriented behavior of users.

We have introduced an approach based on iterative principal factor analysis that can automatically learn hidden factors and to generate aggregate usage profiles, based on these factors, that represent common navigational patterns. The discovered usage profiles can be used to dynamically predict a new user's navigational interests and recommend relevant pages or products. Our results show that this approach can successfully uncover the patterns that characterize a Web site's functional structure, and distinguish between different types of user interests and tasks. Our experimental results show that the proposed factor model is better able to capture the hidden relationships among users and Web objects, than both the standard PCA-based approach and the traditional clustering approach. Furthermore, the standard clustering approaches, such as PACT, that are often used for user segmentation, are generally unable to automatically identify the hidden factors that "explain" the underlying reasons behind the discovered usage patterns.

References

[1] M.W. Berry, S.T. Dumais, and G.W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37:573–595, 1995.

[2] R.B. Cattell. The scree test for the number of factors. *Multivariate Behavioral Research*, 1:245–276, 1966.

[3] D. Cohn and H. Chang. Learning to probabilistically identify authoritative documents. In *Proceedings of the 17th International Conference on Machine Learning*, Stanford, June 2000.

[4] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems*, 1(1), 1999.

[5] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Hashman. Indexing by latent semantic indexing. *Journal of the American Society for Information Science*, 41(6), 1990.

[6] M. Deshpande and G. Karypis. Selective markov models for predicting web-page accesses. In *Proceedings of the First International SIAM Conference on Data Mining*, Chicago, April 2001.

[7] H. H. Harman. *Modern Factor Analysis*. University of Chicago Press, Chicago, 3rd edition, 1976.

[8] T. Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the 15tg Conference on Uncertainty in Artificial Intelligence*, Stockholm, 1999.

[9] R. Johnson and D. Wichern. *Applied Multivariate Statistical Analysis*. Prentice Hall, New Jersey, 5th edition, 2002.

[10] R. Kohavi, L. Mason, R. Parekh, and Z. Zheng. Lessons and challenges from mining retail e-commerce data. *To appear in Machine Learning*, 2004.

[11] W. Lin, S. A. Alvarez, and C. Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6:83–105, 2002.

[12] B. Mobasher, R. Cooley, and J. Srivastava. Creating adaptive web sites through usage-based clustering of urls. In *Proceedings of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX'99)*, Chicago, Illinois, November 1999.

[13] B. Mobasher, R. Cooley, and J. Srivastava. Automatic personalization based on web usage mining. *Communications of the ACM*, 43(8):142–151, 2000.

[14] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Effective personalization based on association rule discovery from web usage data. In *Proceedings of the 3rd ACM Workshop on Web Information and Data Management (WIDM01)*, Atlanta, Georgia, November 2001.

[15] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Discovery and evaluation of aggregate usage profiles for web personalization. *Data Mining and Knowledge Discovery*, 6:61–82, 2002.

[16] O. Nasraoui, R. Krishnapuram, and A. Joshi. Mining web access logs using a fuzzy relational clustering algorithm based on a robust estimator. In *Proceedings of the 8th International World Wide Web Conference*, Toronto, Canada, May 1999.

[17] O. Nasraoui, R. Krishnapuram, A. Joshi, and T. Kamdar. Automatic web user profiling and personalization using robust fuzzy relational clustering. In J. Segovia, P. Szczepaniak, and M. Niedzwiedzinski, editors, *Studies in Fuzziness and Soft Computing*. Springer-Verlag, 2002.

[18] R.R. Sarukkai. Link prediction and path analysis using markov chains. In *Proceedings of the 9th International World Wide Web Conference*, Amsterdam, May 2000.

[19] M. Spiliopoulou. Web usage mining for web site evaluation. *Communications of the ACM*, 43(8):127–134, 2000.

[20] R. Srikant and Y. Yang. Mining web logs to improve website organization. In *Proceedings of the 10th International World Wide Web Conference*, Hong Kong, May 2001.

[21] J. Srivastava, R. Cooley, M. Deshpande, and P. Tan. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23, 2000.