Journal of Southeast University (English Edition) Vol.19 No.3 Sept. 2003 ISSN 1003—7985

Matrix dimensionality reduction for mining typical user profiles

Lu Jianjiang[1,2] Xu Baowen[1,3] Huang Gangshi[2] Zhang Yafei[2]

([1] Department of Computer Science and Engineering, Southeast University, Nanjing 210096, China)

([2] School of Science, PLA University of Science and Technology, Nanjing 210007, China)

([3] School of Computer Science, National University of Defence Technology, Changsha 410073, China)

Abstract: Recently clustering techniques have been used to automatically discover typical user profiles. In general, it is a challenging problem to design effective similarity measure between the session vectors which are usually high-dimensional and sparse. Two approaches for mining typical user profiles, based on matrix dimensionality reduction, are presented. In these approaches, non-negative matrix factorization is applied to reduce dimensionality of the session-URL matrix, and the projecting vectors of the user-session vectors are clustered into typical user-session profiles using the spherical k-means algorithm. The results show that two algorithms are successful in mining many typical user profiles in the user sessions.

Key words: Web usage mining; non-negative matrix factorization; spherical k-means algorithm

Traditional approaches to Web personalization fall into three major categories: manual-decision rule systems, collaborative filtering systems, and content-based filtering agents. There are several well-known drawbacks to these traditional content-based or rule-based filtering techniques for personalization. The type of input is often a subjective description of the users by the users themselves, and thus is prone to biases. The profiles are often static, obtained through user registration, and thus the system performance degrades over time as the profiles age. Collaborative filtering is used to address some of these issues[1]. However, collaborative filtering techniques have their own potentially serious limitations. For instance, it becomes hard to scale collaborative filtering techniques to a large number of items, while maintaining reasonable prediction performance and accuracy.

Several proposals have explored Web-usage mining as an enabling mechanism to overcome some of the problems associated with more traditional techniques. Data-mining techniques have been recently proposed to mine typical user profiles from the vast amount of historical data stored in server or access logs. Associations and sequential patterns between Web transactions are discovered based on the association rule algorithms[2,3]. Clustering has been used to automatically discover Web user profiles from the historic Web log files stored on a Web server[4-6]. In the context of discovering Web user profiles based on clustering, a vector space model is used to represent the user sessions by assigning each vector attribute to a given URL on the Web site, and the similarity measure between the session vectors is defined to mine typical user profiles. However, for the case of Web sessions, it is well known that the user sessions form extremely high dimensional and sparse data matrices. In general, it is a challenging problem to design similarity functions for high dimensional applications because of the fact that the aggregate behavior of high dimensional feature vectors contains a lot of information which cannot be inferred from individual attributes[7]. In order to design effective similarity measure, this paper applies non-negative matrix factorization (NMF)[8-10] to dimensionality reduction of the session-URL matrix; two approaches to mine typical user profiles are presented.

The rest of this paper is organized as follows. In section 1, NMF is applied to dimensionality reduction of the session-URL matrix. In section 2, the projecting vectors of the user session vectors are clustered into typical user session profiles by the spherical k-means algorithm. In section 3, we give an experiment. The conclusions are briefly given in section 4.

1 Non-Negative Matrix Factorization

Each access log entry consists of: ① user's IP

Received 2003-02-21.

Foundation items: The National Natural Science Foundation of China (60073012), National Grand Fundamental Research 973 Program of China (2002CB312000), National Research Foundation for the Doctoral Program of Higher Education of China and Opening Foundation of Jiangsu Key Laboratory of Computer Information Processing Technology in Soochow University.

Biographies: Lu Jianjiang (1968—), male, associate professor; Xu Baowen (corresponding author), male, doctor, professor, bwxu@seu.edu.cn.

address; ② access time; ③ request method, ④ URL of the page accessed; ⑤ data transmission protocol; ⑥ return code; ⑦ number of bytes transmitted. First, we filter out log entries that are not germane to our task. These include entries that: ① Result in any error; ② Use a request method other than "GET"; or ③ Record accesses to image files. Next, analogous to[11], the individual log entries are grouped into user sessions. A user session is defined as a sequence of temporally compact accesses by a user. Since Web servers do not typically log usernames, we define a user session as accesses from the same IP address such that the duration of elapsed time between two consecutive accesses in the session is within a prespecified threshold.

Each URL in the site is assigned a unique number $i \in \{1,2,\cdots,m\}$, where m is the total number of valid URLs. Thus, the j-th user session is encoded as an m dimensional binary attribute vector $\boldsymbol{x}_j = \{x_{1j}, x_{2j},\cdots,x_{mj}\}^{\mathrm{T}}$, where $x_{ij} = 1$ if user accessed i-th URL during j- th session, otherwise $x_{ij} = 0$. Let the number of user sessions be n, then the user session vectors consist of a non-negative matrix $\boldsymbol{X} = (x_{ij})_{m\times n}$, which is called a session-URL matrix. The components of the vector $\boldsymbol{x}_j$ are usually normalized to a unit vector, that is,

$$x_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \qquad i = 1,2,\cdots,m$$

Intuitively, the effect of normalization is to retain only the direction of the vector. This ensures that sessions dealing with the same subject matter, but differing in length lead to similar session vectors.

However, for the case of Web sessions, it is well known that the session-URL matrix is extremely high dimensional and sparse. In general, it is a challenging problem to design similarity functions for high dimensional applications because of the fact that the aggregate behavior of high-dimensional feature vectors contain a lot of information which cannot be inferred from individual attributes. In order to design effective similarity measures, NMF is applied to reduce dimensionality of the session-URL matrix.

Given a non-negative matrix $\boldsymbol{X} = (x_{ij})_{m\times n}$, NMF finds the non-negative $m \times r$ matrix $\boldsymbol{U} = (u_{ij})_{m\times r}$ and the non-negative $r \times n$ matrix $\boldsymbol{V} = (v_{ij})_{r\times n}$ such that

$$\boldsymbol{X} \approx \boldsymbol{UV} \tag{1}$$

The r is generally chosen to satisfy $(n + m)r < nm$, so that the product $\boldsymbol{UV}$ can be regarded as a compressed form of the data in $\boldsymbol{X}$.

Eq.(1) can be rewritten column by columns as

$$\boldsymbol{x} \approx \boldsymbol{Uv} \tag{2}$$

where $\boldsymbol{x}$ and $\boldsymbol{v}$ are the corresponding columns of $\boldsymbol{X}$ and $\boldsymbol{V}$. Each vector $\boldsymbol{x}$ is approximated by a linear combination of the columns of $\boldsymbol{U}$, weighted by the components of $\boldsymbol{v}$. Therefore, $\boldsymbol{U}$ can be regarded as containing a basis vector that is optimized for the linear approximation of the vector in $\boldsymbol{X}$. Since relatively few basis vectors are used to represent many vectors, good approximation can only be achieved if the basis vectors discover structure that is latent in the vectors.

Here, we introduce an algorithm based on iterative estimation of $\boldsymbol{U}$ and $\boldsymbol{V}$. At each iteration of the algorithm, the new values of $\boldsymbol{U}$ and $\boldsymbol{V}$ are found by multiplying the current values by some factors that depends on the quality of the approximation in Eq. (1). Repeated iterations of the updated rules are guaranteed to converge to a locally optimal matrix factorization.

We introduce the updated rules given in the next equations[10],

$$v_{ij} \leftarrow \sqrt{v_{ij}\sum_k u_{ki}\frac{x_{ki}}{y_{kj}}} \tag{3}$$

$$u_{ij} \leftarrow u_{ij}\sum_k \frac{x_{ik}}{y_{ik}}v_{jk} \tag{4}$$

$$u_{ij} \leftarrow \frac{u_{ij}}{\sum_k u_{kj}} \tag{5}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are initial stochastic matrices.

The update rules maximize the following objective function:

$$F(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i,j}\left(x_{ij}\log\frac{x_{ij}}{y_{ij}} - x_{ij} + y_{ij}\right) + \alpha\sum_{i,j}a_{ij} - \beta\sum_i b_{ii} \tag{6}$$

where a_{ij} are the components of $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U}$; b_{ii} are the diagonal components of $\boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}$; $\alpha,\beta > 0$ are some constants; $\boldsymbol{Y} = \boldsymbol{UV} = (y_{ij})_{m\times n}$.

2 Mining Typical User Profiles

By the NMF, the m dimensional user session vector $\boldsymbol{x}_j$ is projected into the r dimensional vector $\boldsymbol{v}_j$, where $\boldsymbol{x}_j = \{x_{1j}, x_{2j},\cdots,x_{mj}\}^{\mathrm{T}}$, $\boldsymbol{v}_j = \{v_{1j}, v_{2j},\cdots, v_{rj}\}^{\mathrm{T}}$, $j = 1,2,\cdots,n$. Since the new space has lower dimensions, an effective similarity measure can be designed. Next, we normalize $\boldsymbol{v}_j$ to a unit vector, which is still denoted as $\boldsymbol{v}_j$, and cluster $\boldsymbol{v}_j$ with the spherical k-means algorithm[12].

Definition 1 Given two r dimensional vectors $\boldsymbol{v}_i$

and $\boldsymbol{v}_j$, the similarity between $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ is defined as the inner product

$$\boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{v}_j = \sum_{k=1}^{r} v_{ki} \times v_{kj} \tag{7}$$

Given vectors $\boldsymbol{v}_j, j = 1,2,\cdots,n$. Let $\pi_1, \pi_2, \cdots, \pi_k$ denote a partitioning of the vectors into k disjoint clusters. For each fixed $1 \leqslant j \leqslant k$, the mean vector of the vectors contained in the cluster π_j is

$$\boldsymbol{m}_j = \frac{1}{n_j} \sum_{\boldsymbol{v}_i \in \pi_j} \boldsymbol{v}_i \tag{8}$$

where n_j is the number of vectors in π_j. Note that the mean vector $\boldsymbol{m}_j$ is not a unit vector; we can capture its direction by writing the corresponding concept vector as

$$\boldsymbol{c}_j = \frac{\boldsymbol{m}_j}{\| \boldsymbol{m}_j \|} \tag{9}$$

We measure the quality of any given partitioning $\{\pi_j\}_{j=1}^{k}$ using the following objective function:

$$Q(\{\pi_j\}_{j=1}^{k}) = \sum_{j=1}^{k} \sum_{\boldsymbol{v}_i \in \pi_j} \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{c}_j \tag{10}$$

Spherical k-means algorithm

① Start with an arbitrary partitioning of the vectors, namely, $\{\boldsymbol{\pi}_j^{(0)}\}_{j=1}^{k}$. Let $\{\boldsymbol{c}_j^{(0)}\}_{j=1}^{k}$ denote the concept vectors associated with the given partitioning. Set the index of iteration $t = 0$.

② Compute the new partitioning $\{\boldsymbol{\pi}_j^{(t+1)}\}_{j=1}^{k}$ induced by the old concept vectors $\{\boldsymbol{c}_j^{(t)}\}_{j=1}^{k}$:

$$\boldsymbol{\pi}_j^{(t+1)} = \{\boldsymbol{v}_i \in \{\boldsymbol{v}_i\}_{i=1}^{n} : \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{c}_j^{(t)} > \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{c}_l^{(t)}, \quad 1 \leqslant l \leqslant k, l \neq j\} \qquad 1 \leqslant j \leqslant k$$

where $\boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{c}_j^{(t)}$ is the inner product proposed in definition 1.

③ Compute the new concept vectors corresponding to the partitioning:

$$\boldsymbol{c}_j^{(t+1)} = \frac{\boldsymbol{m}_j^{(t+1)}}{\| \boldsymbol{m}_j^{(t+1)} \|} \qquad 1 \leqslant j \leqslant k$$

where $\boldsymbol{m}_j^{(t+1)}$ denotes the mean of the vectors in cluster $\pi_j^{(t+1)}$.

④ Stop if $\left| Q(\{\boldsymbol{\pi}_j^{(t)}\}_{j=1}^{k}) - Q(\{\boldsymbol{\pi}_j^{(t+1)}\}_{j=1}^{k}) \right| \leqslant \varepsilon$, for some suitably chosen $\varepsilon > 0$.

Suppose vectors $\boldsymbol{v}_j$ $(j = 1,2,\cdots,n)$ are partitioned into k disjoint clusters $\pi_1, \pi_2, \cdots, \pi_k$ by the spherical k-means algorithm. Next, we give two methods to summarize k typical session profiles from k disjoint clusters. As an example, we only consider the cluster π_1. Other $k-1$ clusters can be considered with the same way.

1) The first method

Let $\pi_1 = \{\boldsymbol{v}_1^1, \boldsymbol{v}_2^1, \cdots, \boldsymbol{v}_l^1\}$, where $\boldsymbol{v}_j^1 = \{v_{1j}^1, v_{2j}^1, \cdots, v_{rj}^1\}^{\mathrm{T}}$, $j = 1,2,\cdots,l$. We use the vectors in the cluster π_1 to construct an m dimensional vector $\boldsymbol{p}_1$ as

$$\boldsymbol{p}_1 = \frac{1}{l}(U\boldsymbol{v}_1^1 + U\boldsymbol{v}_2^1 + \cdots + U\boldsymbol{v}_l^1) \tag{11}$$

Let the mean vector of π_1 be $\boldsymbol{m}_1 = \{o_1^1, o_2^1, \cdots o_r^1\}$, then

$$\boldsymbol{m}_1 = \frac{1}{l} \sum_{j=1}^{l} \boldsymbol{v}_j^1 \tag{12}$$

$$\boldsymbol{p}_1 = \boldsymbol{U}\boldsymbol{m}_1 = \boldsymbol{u}_1 o_1^1 + \boldsymbol{u}_2 o_2^1 + \cdots + \boldsymbol{u}_r o_r^1 \tag{13}$$

We select vector $\boldsymbol{p}_1$ as the typical user profile for the cluster π_1.

2) The second method

Let $\pi_1 = \{\boldsymbol{v}_1^1, \boldsymbol{v}_2^1, \cdots, \boldsymbol{v}_l^1\}$, other residual vectors are denoted as $\{\boldsymbol{v}_{l+1}^1, \boldsymbol{v}_{l+2}^1, \cdots, \boldsymbol{v}_n^1\}$, where $\boldsymbol{v}_j^1 = \{v_{1j}^1, v_{2j}^1, \cdots, v_{rj}^1\}^{\mathrm{T}}$, $j = 1,2,\cdots,n$. We define the class discriminative degree of the basis vectors in order to obtain an effective typical user profile.

Definition 2 The class discriminative degrees of the basis vectors $\boldsymbol{u}_s$, $s = 1,2,\cdots,r$ to the cluster π_1 are defined as

$$d_s = \frac{1}{l} \sum_{j=1}^{l} v_{sj}^1 - \frac{1}{n-l} \sum_{j=n-l}^{n} v_{sj}^1 \qquad s = 1,2,\cdots,r \tag{14}$$

If the average weight $\frac{1}{l} \sum_{j=1}^{l} v_{sj}^1$ of the basis vector $\boldsymbol{u}_s$ in the cluster π_1 is big, and the average weight $\frac{1}{n-l} \sum_{j=n-l}^{n} v_{sj}^1$ of the basis vector $\boldsymbol{u}_s$ in other clusters is small, then the class discriminative degree d_s is large. That is to say, the basis vector $\boldsymbol{u}_s$ has strong discriminative ability between the sessions in the cluster π_1 and the sessions in the other clusters. Choose s basis vectors with high class discriminative degrees, let be $\boldsymbol{u}_1^1, \boldsymbol{u}_2^1, \cdots, \boldsymbol{u}_s^1$. These basis vectors can effectively characterize the session profile represented by the cluster π_1. Similarly to Eq.(13), we use these s basis vectors to construct an m dimensional vector of the URLs as

$$\boldsymbol{u}_1 o_1^1 + \boldsymbol{u}_2 o_2^1 + \cdots + \boldsymbol{u}_s o_s^1 \tag{15}$$

Let this vector be $\boldsymbol{P}_1 = \{P_{11}, P_{21}, \cdots, P_{m1}\}^{\mathrm{T}}$, then a typical session profile with respect to the cluster π_1 can be summarized a vector $\boldsymbol{P}_1$.

In general, the components of $\boldsymbol{P}_1$ represent the probability of access of each URL during the profile. The URL weights P_{i1} measure the significance of a given URL to the profile. Besides summarizing profiles, the components of the profile vector can be used to recognize an invalid profile which has no strong or frequent access pattern. For such a profile, all the

URL weights will be low. Using two methods above, we can obtain k typical session profiles which are represented with $\boldsymbol{P}_j = \{P_{1j}, P_{2j}, \cdots, P_{mj}\}^{\mathrm{T}}$, $j = 1, 2, \cdots, k$.

3 Experimental Results

Two methods are used to mine typical user session profiles on the log data from http://www.cs.washington.edu/research/adaptive. Data during a period of 1/1/98 to 6/1/98 is used. URLs are properly first dealt with. For example,/machines/ecards/templates/,/machines/ecards/templates and machines/ecards/templates are regarded as the same URLs. The number of distinct URLs accessed in valid entries is 4 274. We delete the URLs whose total accessed number is less than 11; the last number of distinct URLs is 778. After filtering out irrelevant entries, the data is segmented into 8 004 sessions. The maximum elapsed time between two consecutive accesses in the same session is set to 45 min. Let $r = 30$, $k = 16$, Tab.1 illustrates four profiles by the first method, Tab.2 illustrates four profiles mined by the second method. Where only the significant URLs ($P_{is} > 0.06$) are displayed, and the individual components are displayed in the format {P_{is} – i-th URL}. A listing of all 16 profiles is not presented here due to paucity of space. The results show that two algorithms are successful in delineating many different profiles in the user sessions.

Tab.1 Profile examples mined by the first method

s	P_s
1	{0.1753 – /music/machines/links} {0.1081 – /music/machines} {0.0809 – /music/machines/links/sites.html}
2	{0.1454 – /music/machines/manufacturers} {0.1002 – /music/machines/manufacturers/Yamaha} {0.0730 – /music/machines}
3	{0.6838 – /music/machines/Analogue-Heaven} {0.0937 – /music/machines}
4	{0.342 – /music/machines/categories/midi-cv-sync/midi} {0.075 – /music/machines/categories/midi-cv-sync/midi/midi-history} {0.075 – /music/machines/categories/midi-cv-sync/midi/midi-specs} {0.074 – /music/machines/categories/midi-cv-sync/midi/midi-controller}

Tab.2 Profile examples mined by the second method

s	P_s
1	{0.3841 – /music/machines/links} {0.1260 – /music/machines/links/sites.html} {0.0989 – /music/machines/links/machines.html} {0.0722 – /music/machines/links/links.html}
2	{0.4851 – /music/machines/manufacturers} {0.1249 – /music/machines/manufacturers/Yamaha} {0.0652 – /music/machines/gearlists}
3	{0.8781 – /music/machines/Analogue-Heaven}
4	{0.3691 – /music/machines/categories/midi-cv-sync/midi} {0.0748 – /music/machines/categories/midi-cv-sync/midi/midi-history} {0.0748 – /music/machines/categories/midi-cv-sync/midi/midi-specs} {0.0738 – /music/machines/categories/midi-cv-sync/midi/midicontroller}

4 Conclusion

In this paper, we have presented two new approaches for automatic discovery of user session profiles in Web log data. In order to design effective similarity measure, we apply non-negative matrix factorization to dimensionality reduction of the session-URL matrix; the projecting vectors of the user session vectors are clustered into typical user session profiles by the spherical k-means algorithm. The results of experiments show that our algorithm can mine typical user profiles effectively.

References

[1] Konstan J, Miller B, Maltz D. et al. GroupLens: applying collaborative filtering to usenet news [J]. *Communications of the ACM*, 1997, **40**(3): 77 – 87.

[2] Cooley R, Mobasher B, Srivastava J. Web mining: Information and pattern discovery on the World Wide Web [A]. In: *Proc IEEE Intl Conf Tools with AI* [C]. Newport Beach, CA, 1997. 558 – 567.

[3] Chen M S, Park J S, Yu P S. Efficient data mining for path traversal patterns [J]. *IEEE Trans on Knowledge and Data Engineering*, 1998, **10**(2): 209 – 221.

[4] Nasraoui O, Frigui H, Krishnapuram R. et al. Extracting Web user profiles using relational competitive fuzzy clustering [J]. *International Journal on Artifical Intelligence Tools*, 2000, **9**(4): 509 – 526.

[5] Nasraoui O, Krishnapuram R, Joshi A. Mining Web access logs using relational clustering algorithm based on a robust estimator [A]. In: *NAFIPS Conference* [C]. New York, NY, 1999.705 – 709.

[6] Nasraoui O, Krishnapuram R. One step evolutionary mining of context sensitive associations and Web navigation patterns [A]. In: *Second SIAM International Conference on Data Mining* [C]. Arlington, VA, 2002. 531 – 547.

[7] Hinneburg A, Aggarwal C C, Keim D A. What is the nearest neighbor in high dimensional spaces? [A]. In: *Proceedings of the VLDB Conference* [C]. Cario, Egypt, 2000. 506 – 515.

[8] Lee D, Seung H. Learning the parts of objects by non-negative matrix factorization [J]. *Nature*, 1999, **401**:788 – 791.

[9] Lee D, Seung H. Algorithms for non-negative matrix factorization [A]. In: *Adv Neural Info Proc Syst*[C]. 2001, **13**:556 – 562.

[10] Li S Z, Hou X W, Zhang H J. Learning spatially localized parts-based representation [A]. In: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition* [C]. Hawaii, 2001. 207 – 212.

[11] Mobasher B, Jain N, Han E-H. et al. Web mining [R]. U. of Minnesota, 1996.

[12] Inderjit S D, Dharmendra S M. Concept decompositions for large sparse text using clustering [J]. *Machine Learning*, 2001, **42**(1): 143 - 175.

基于矩阵降维的典型用户文件发现方法

陆建江[1,2] 徐宝文[1,3] 黄刚石[2] 张亚非[2]

([1] 东南大学计算机科学与工程系,南京 210096)
([2] 解放军理工大学理学院,南京 210007)
([3] 国防科学技术大学计算机学院,长沙 410073)

摘 要 应用聚类技术能够自动地发现典型用户文件,但是由于会话向量通常是高维的稀疏向量,因此很难在会话向量之间设计有效的相似度度量.本文提出 2 种基于矩阵降维的典型用户文件发现方法.这些方法应用非负矩阵分解技术降低会话-URL 矩阵的维数,并通过球形的 k-均值算法对用户会话向量的投影向量聚类,由此得到典型用户文件.实验结果表明,这些算法能够有效地从用户会话中发现典型的用户文件.

关键词 Web 挖掘; 非负矩阵分解; 球形的 k-均值算法

中图分类号 TP18