International Journal of Hybrid Information Technology
Vol.8, No.9 (2015), pp.185-202
http://dx.doi.org/10.14257/ijhit.2015.8.9.18

An Online Learning Model of Mobile User Preference Based on Context Quantification

Yancui Shi, Congcong Xiong, Jucheng Yang*, Yarui Chen and Jianhua Cao

Institute of Computer Science and Information Engineering
Tianjin University of Science & Technology, Tianjin China
jcyang@tust.edu.cn

Abstract

In mobile network, the mobile user has the strict requirement for the performance of accessing the information. In order to provide the personalized service for mobile user timely and accurately, an online learning model of mobile user preference based on context quantification is proposed. In the model, a context quantification method is proposed, which can enhance the accuracy of learned mobile user preference; and the sliding window and the online extreme leaning machine (O-ELM) are introduced to realize the online learning. Firstly, it needs to judge whether the mobile user preference is affected by the context through analyzing the mobile user behaviors. Secondly, the context is quantified according to the context relevancy and the context similarity. And then, the sliding window is employed to select the samples that need to be learned when updating the mobile user preference. Finally, O-ELM is employed to learn the mobile user preference. The experimental results show that the proposed method surpasses the existing methods in the performance.

Keywords: *Context Quantification; Mobile User Behavior; Sliding Window; Online Extreme Learning Machine; Online Learning*

1. Introduction

With the popularity of the commercial 3G network and the improvement of the performance of the intelligent mobile terminal, the acquisition and push of information can happen at any time, at anywhere, at anyhow, and it is possible to provide the mobile network service for mobile user in real time [1]. And the fixed users who use desktop shift gradually to mobile users who use mobile terminal. However, with the continuously emergence of new mobile network service and wide application, the kinds of mobile network services and the growth of information content have exceeded the acceptable range of the mobile user. In addition, the mobile terminal have some defects, such as small display, limited terminal handling capacity, short duration power, and so on. So, it leads the problem of serious information overload [2,3]. In e-commerce, how to provide the personalized mobile network service for mobile user is very important, which can improve the user satisfaction and prevent customer churn. Therefore, the research about mobile user preference learning has become a hot issue.

ISSN: 1738-9968 IJHIT

In mobile network, the mobile user preference is affected by the context (e.g. time, location) more evidently. On the other hand, in the mobile network, it is easy to get the context information when mobile user uses the mobile network service. For example, the time information can be get by the mobile terminal directly, and the location information can be get by the GPS. While some high-level context can be get by the reasoning towards the lower-level context, e.g. mood, activity. Therefore, in order to locate the mobile user preference accurately, the researcher added the context into the learning model of mobile user preference [4]. However, there is no unified context quantification when introducing the context into the mobile user preference learning model. Hence, in the paper, a context quantification is proposed by analyzing the mobile user behavior. In addition, the mobile user preference will change as the time goes by, so it needs to update the mobile user preference timely. In order to solve the problem, in the paper, we employ the sliding window to select the samples that need to learn and employ O-ELM to learn the mobile user preference.

The structure of the paper is as follows. The review is followed by an introduction; section 3 describes the proposed context quantification; the online learning model is described in section 4; section 5 shows the experimental studies for verifying the proposed model; section 6 includes some conclusions plus some ideas for the future work.

2. Overview of Mobile User Preference Learning

In the existing learning models of mobile user preference, the classification is often employed to learn the mobile user preference, such as Bayesian classifier, neural network, association rules, decision tree, support vector machine (SVM) and so on [5]. In order to locate the mobile user preference accurately, these methods all introduce the context information. It is a urgent problem to solve about how to introduce the context into the classification reasonably. It was shown in [6], the context was fallen into three representation methods: the scalar context, e.g., temperature (15°C); the ordinal context, e.g., the number of hotel room; and categorical context, e.g., mood (sad, happy). For the scalar context, the closer the values of the contexts are, the more similar the two contexts are. The scalar context can be directly applied into the classification, so it doesn't need to be quantified. But the ordinal context and the categorical context are difficult for the contextual mobile user preference learning. There is no unified approach about how to introduce the ordinal context and the categorical context into the classification reasonably. Some researchers quantified the context according to their own habits. As shown in [7], Oku et al. gave the following rules: 0=Monday to 6=Sunday. Dao et al. gave the similar rules to quantify the context: 1=workday (Monday-Friday), 2=weekend (Saturday-Sunday) [8]. According to the quantification rules in [7], the obtained similarity between Saturday and Sunday is less than 1, while the similarity between Saturday and Sunday is equal to 1 obtained by the quantification rules in [8]. Obviously, the above methods are unreasonable and not universal.

In addition, the mobile user preference will change as time goes by. In order to solve the problem, Sarah et al. [9] proposed a dynamic learning model of mobile user preference, Daidalos. In Daidalos, the user behaviors are stored into two memories, the long-term memory (LTM) stored all user actions until the last updating time and the short-term memory (STM) stored user behaviors which have happen since the last updating time. When updating the mobile user preference, only the user behaviors in STM needs to be learned. Shi et al. [10] proposed an adaptive learning method of mobile user preference, in which only the changed mobile user preferences need to be learned. But, above methods are all offline learning and can't need the demand of mobile user timely. Rana et al. [11] proposed a evolutionary clustering algorithm to learn user preference, and the incoming relevant new data at the current was maximized and the clustering drifted from the historical data was minimized in the method. This method can update the user preference timely.

3. The Context Quantification

In this section, we will elaborate the proposed method of context quantification which includes the determination of contexts that impact mobile user behaviors, the calculation of the context relevancy and the context similarity, and the quantification of context.

3.1. The determination of context

In the paper, the volatility of mobile user behaviors is employed to judge whether the mobile user preference is affected by the given context [12]. The volatility is represented as follows:

$$vol_{C_l,u_i,s_j} = \frac{\sum_{c_{lq}\in C_l} |Vol_{u_i,s_j,c_{lq}} - \overline{Vol_{u_i,s_j,C_l}}|}{Vol_{u_i,s_j,C_l}} \tag{1}$$

Where, $u_i \in U$ represents the mobile user i, U represents the set of mobile users; $s_j \in S$ represents the mobile network service j(e.g., game, music, video), S represents the set of mobile network services; C_l represents some context, such as time, location; $c_{lq} \in C_l$ represents the specific context instance, e.g., morning, noon, afternoon, evening; $Vol_{u_i,s_j,c_{lq}}$ represents the usage volume that u_i used s_j under c_{lq}.

The units of the usage volume of different mobile network services are different, and the common units include duration, times, flow etc. In the paper, the formula of usage volume is as follows:

$$Vol_{u_i,s_j,c_{lq}} = \upsilon * \frac{L_{u_i,s_j,c_{lq}}}{L_{u_i,c_{lq}}} + (1-\upsilon) * \frac{N_{u_i,s_j,c_{lq}}}{N_{u_i,c_{lq}}} \tag{2}$$

Where, $L_{u_i,s_j,c_{lq}}$ represents the duration or flow that u_i used s_j under c_{lq}; $L_{u_i,c_{lq}}$ represents the total duration or flow that u_i used mobile network services under c_{lq}; $N_{u_i,s_j,c_{lq}}$ represents the times that u_i used s_j under c_{lq}; $N_{u_i,c_{lq}}$ represents the total times that u_i used mobile network services under c_{lq}; υ is the weight parameter.

Vol_{u_i,s_j,C_l} represents the total usage volume that u_i used s_j under C_l. $\overline{Vol_{u_i,s_j,C_l}}$ is the mean of the usage volume that u_i used s_j under C_l and it is represented as follows:

$$\overline{Vol_{u_i,s_j,C_l}} = \frac{1}{n_l} \sum_{c_{lq} \in C_l} Vol_{u_i,s_j,c_{lq}} \tag{3}$$

Where, n_l represents the number of context instances that C_l contains. n_l will change when the different context partition methods are used. Introduced by the case of the time context, if it is divided into morning, afternoon, evening, night, n_l=4; if it is divided into hour, n_l=24.

The larger the volatility is, the greater the impact that context towards mobile user preference is. Introduced for the case of time context and location context, the voice call duration of some mobile user under the given context is as follows: {morning, afternoon, evening, night}={20,20,60,0}, the volatility obtained by Equation (2) is 0.7; {at home, at work}={52,48}, the volatility obtained by Equation (2) is 0.04. We can know that the voice call affected by the time context is greater than that by the location context in the above example. In the paper, we need to set the threshold of the volatility. If the obtained volatility is greater than the setting threshold, we judge the mobile user preference is affected by the given context; else, we judge the mobile user preference isn't affected by the given context.

The principle that sets the threshold of the volatility is as follows. When the threshold is relatively small, it's loose for the volatility, so there are many contexts that are judged to impact the mobile user preference. There is a Cartesian Product proportional relationship between the number of contexts and the number of the mobile user preferences [13]. Therefore, there are many mobile user preferences and the running time is longer, but the accuracy of learned mobile user preferences is higher. When the threshold is relatively great, the contexts that are judged to impact the mobile user preference are few. There are few mobile user preferences and the running time is shorter, but the accuracy of learned mobile user preferences is incomplete. In the above example, when the threshold of the volatility is set 0.1, the voice call is affected only by time context and the number of the context instances is 4; when the threshold of the volatility is set 0.01, the voice call is affected by the time context and the location context, and the number of the context instances is 8(4*2). Therefore, it needs to compromise between the accuracy and the running time when setting the threshold of the volatility.

We can calculate the impact that context towards the mobile user preference according to Equation (4):

$$w_{C_l,u_i,s_j} = \frac{vol_{C_l,u_i,s_j}}{\sum_{m=1}^{n_{s_j}} vol_{C_m,u_i,s_j}} \tag{4}$$

Where, n_{s_j} represents the number of contexts that impacted u_i when he used s_j. Integrating the obtained weight of context and the usage volume that mobile user used mobile network service under given context instance, the weight of context instance can be calculated by Equation (5):

$$\omega_{c_{lq},u_i,s_j} = w_{C_l,u_i,s_j} * \frac{Vol_{u_i,s_j,c_{lq}}}{Vol_{u_i,s_j,C_l}} \tag{5}$$

3.2. The Context Relevancy

There is some relevancy between the context instances that impact mobile user preference. In order to mine the relevancy between the context instances, the weighted association rules [14] is employed to judge whether there is some relevancy between contexts. If so, we use the weighted association rules to compute the context relevancy.

Firstly, it needs to calculate the support which can be obtained by the Equation (6):

$$S_{c_{lq,mp},u_i,s_j} = \frac{Vol_{u_i,s_j,c_{lq}\cap c_{mp}}}{Vol_{u_i,s_j}} \tag{6}$$

Where, $Vol_{u_i,s_j,c_{lq}\cap c_{mp}}$ represents the usage volume that u_i used s_j under the common constraint of c_{lq} and c_{mp}; Vol_{u_i,s_j} represents the total usage volume that u_i used s_j. When the support is greater than the setting minimum support, it needs to calculate the confidence. The formula of confidence is as follows:

$$T_{c_{lq,mp},u_i,s_j} = \frac{Vol_{u_i,s_j,c_{lq}\cap c_{mp}}}{Vol_{u_i,s_j,c_{lq}}} \tag{7}$$

If the obtained confidence is greater than the setting minimum confidence, it judges that the two context instances are related. And the context relevancy can be calculated by Equation (8):

$$Co_{c_{lq,mp},u_i,s_j} = \frac{\omega_{c_{mp},u_i,s_j}}{\omega_{c_{lq},u_i,s_j}} * S_{c_{lq,mp},u_i,s_j} * T_{c_{lq,mp},u_i,s_j} \tag{8}$$

3.3. The Context Similarity

There is also some similarity between the context instances that impact mobile user preference, such as morning and afternoon, Saturday and Sunday. The context similarity can be obtained by analyzing the similarity of the mobile user preferences under the given contexts. To the best of our knowledge, until now little research has considered the weight of context instance when calculating the context similarity [15]. However, the impacts of different context instances towards mobile network service are different. For example, if the mobile user likes to play games in the evening, the impact of the evening is greater than the morning or afternoon. Therefore, it's necessary to introduce the weight of context instance into the calculation of context similarity.

We employ the improved Pearson Correlation Coefficient to calculate the context similarity. The formula is as follows:

$$Sim_{c_{lq},c_{lp},u_i} = \frac{\theta * \sum_{s_j\in S_c} \beta_1 * \beta_2 (Vol_{u_i,s_j,c_{lq}} - \overline{Vol_{u_i,c_{lq}}})(Vol_{u_i,s_j,c_{lp}} - \overline{Vol_{u_i,c_{lp}}})}{\sqrt{\sum_{s_j\in S_c} \beta_1 (Vol_{u_i,s_j,c_{lq}} - \overline{Vol_{u_i,c_{lq}}})^2 \sum_{s_j\in S_c} \beta_2 (Vol_{u_i,s_j,c_{lp}} - \overline{Vol_{u_i,c_{lp}}})^2}} \tag{9}$$

Where, S_c represents the set of the common mobile network services that u_i used under c_{lq}

and c_{lp} respectively; $\beta_1 = \omega_{c_{lq},u_i,s_j}$, $\beta_2 = \omega_{c_{lp},u_i,s_j}$. The traditional Pearson Correlation Coefficient

Doesn't consider the impact of the size of the set S_c. While the size of the set S_c plays an important role at the accuracy of the calculated context similarity. For example, mobile user A uses 100 kinds of mobile network services, and mobile user B uses 2 kinds of mobile network services, while the number of the common mobile network services between A and B only is one and the value of preference is the same. If the similarity between A and B is calculated according to the traditional Pearson Correlation Coefficient, the obtained similarity is 1. Obviously, the obtained result isn't accurate. Therefore, we also consider the impact of the size of S_c when computing the context similarity, that is the number of the common preferences. θ is employed to measure the impact of the number of the common preferences, and it is calculated by the Equation (10):

$$\theta = \frac{|S_{c_{lq},u_i} \cap S_{c_{lp},u_i}|}{|S_{c_{lq},u_i}|} \tag{10}$$

Where, S_{c_{lq},u_i} represents the set of the mobile network services that u_i used under c_{lq}, so $S_c = S_{c_{lq},u_i} \cap S_{c_{lp},u_i}$. According to Equation (10), we can know that the similarity between context instances is asymmetry, that is $Sim_{c_{lp},c_{lq},u_i} \neq Sim_{c_{lq},c_{lp},u_i}$.

3.4. The Step of the Context Quantification

Firstly, it needs to select some context as the basis to quantify. And then, the other contexts

will be quantified according to the following rules. $C_{u_i,s_j} = \{C_1, C_2, \cdots, C_{n_{s_j}}\}$ represents the set of

contexts that impacted u_i when he used s_j.

Assuming C_1 is the basis context, the rules of context quantification are as follows:

(1) According to the weight of context instance and the matrix of context similarity, we can quantify the context instances that belong to C_1, referring to Equation (11).

$$Q_{C_1,u_i,s_j} = W_{C_1,u_i,s_j} * H_{Sim_{C_1}} \tag{11}$$

Where, $W_{C_1,u_i,s_j} = \{\omega_{c_{11},u_i,s_j}, \omega_{c_{12},u_i,s_j}, \cdots, \omega_{c_{1n_1},u_i,s_j}\}$ is the vector of the weight of the context instances; n_1 represents the number of context instances that belong to C_1; $Q_{C_1,u_i,s_j} = \{q_{c_{11},u_i,s_j}, q_{c_{12},u_i,s_j}, \cdots, q_{c_{1n_1},u_i,s_j}\}$ represents the vector of the quantified value of the context instances; $H_{Sim_{C_1}}$ is the matrix of context similarity, and its size is $n_1 \times n_1$; $h_{q,p} = Sim_{c_{1q},c_{1p},u_i} \in H_{Sim_{C_1}}$ is the element in the matrix of the context similarity and $h_{q,q} = 1, q = 1,2,\cdots,n_1$.

(2) The other contexts are quantified according to the quantified values of the basis context instances, the matrix of context relevancy and the matrix of context similarity, referring to Equation (12).

$$Q_{C_l,u_i,s_j} = Q_{C_1,u_i,s_j} * R_{C_1,C_l} * H_{Sim_{C_l}} \tag{12}$$

Where, $C_l \in C_{u_i,s_j}$, $l \neq 1$; R_{C_1,C_l} represents the matrix of the relevancy of C_1 and C_l, and its size is $n_1 \times n_l$, n_l represents the number of context instances that belong to C_l; $r_{q,p} = Co_{c_{1q},c_{lp},u_i,s_j} \in R_{C_1,C_l}$.

3.5. The Rational Analysis of the Proposed Quantification

(1) The introduction of the weight of context instance.

The weight of context instance is calculated according to the usage volume that mobile user use mobile network services under the given context, and which can reflect the degree of influence that context instance impacts the mobile user behavior. Shi et al. [10] employed the following rules to quantify location context: {at home, work, elsewhere}={0,1,2}. The similarity between at home and work is greater than that between at home and elsewhere according to the quantification rules in [10] and it is obvious that the quantification rule didn't consider the real mobile user behavior, so it is irrational. Supposed that the usage volume that mobile user used some service under the location context is as follows: {at home, work, elsewhere}={40,80,30}. Suppose that the weight of the location is 0.5 obtained by Equation (5), the quantified values of location context instance obtained by the proposed quantification in the paper is as follows: {at home, work, elsewhere}={0.133,0.267,0.1}. The similarity between at home and elsewhere is greater than that between at home and work according to the above quantified value and it is more accurate for measure the context similarity. Therefore, we can get the conclusion: when the context is quantified, it is necessary to introduce the weight of the context instance.

(2) the introduction of the context similarity

The weight of context instance is calculated according to the usage volume that mobile user used some mobile network service under the given context instance, while the context similarity between context instances is calculated according to the the usage volume that mobile user used all mobile network services under the given context instances. Therefore, the context similarity reflects the comprehensive measure. Supposed the similarity matrix of the location context instances obtained by Equation (9) is as follows:

$$H = \begin{bmatrix} 1 & 0.2 & 0.7 \\ 0.19 & 1 & 0.22 \\ 0.67 & 0.21 & 1 \end{bmatrix} \tag{13}$$

The quantified results by the Equation (11) or Equation (12) are as follows: {at home, work, elsewhere}={0.2507,0.3146,0.2518}. Compared with the quantified results only introducing the context instance, the quantified values can more accurately reflect the mobile user behavior under different context instances.

(3) The introduction of the context relevancy

The contexts that impact the mobile user behavior are not dependent, and there may be some relevancy. For example, the mobile user behavior may be different at the different time but in the same location. Therefore, it is reasonable to consider the context relevancy in the context quantification.

4. The Proposed Online Learning Model

In the section, the online learning model of mobile user preference will be described in detail. Before introducing the construction of model, we briefly describe the sliding window and O-ELM algorithm.

4.1. The Sliding Window

The sliding window is mainly applied to the research about data stream or time series. Since in the data stream or time series, the new data will be added constantly, and the time complexity is very great when doing research on the data stream. So, the researcher use the sliding window to select the recent data to do some research [16]. In the mobile network, the mobile user behavior will happen constantly, so we can regard the mobile user preference as data stream or time series. In addition, the mobile user preference will be affected more greatly by the recent data. Therefore, we can employ the sliding window to select the recent data to model for achieving the online learning.

In the sliding window, the new data will be added into and the outdated data will be removed constantly. If each time the new data arrives, the sliding window moves ahead and the online learning method is triggered, and then the time complexity is very great. Hence, we introduce the basic window. Only the new data fill in the basic window, the sliding window moves ahead and the updating process is triggered.

The length of the sliding window (T) can be a fixed number or a fixed duration time. While the length of the basic window (T_1) can be defined T/Nb, here, Nb is a natural number and can be set according to the actual needs, shown as Figure 1. At the time t_0, when the basic window W_{Nb+1} is filled with the new arrival data, the sliding window moves ahead, the basic window W_1 is removed from the sliding window and the basic window W_{Nb+1} is added into sliding window. The new updating process is triggered.

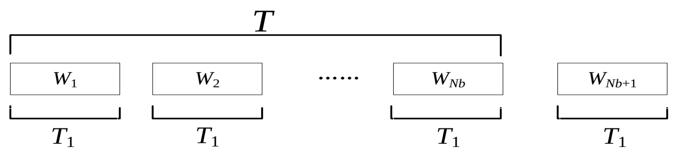


Figure 1. The Sliding Window and the Basic Window

4.2. The O-ELM

Extreme learning machine (ELM) is a single-hidden layer feed-forward neural network (SLFN) proposed by Huang [17]. Compared with the traditional feed-forward network learning like back-propagation (BP) algorithm, SVM, the learning speed of ELM is not only faster, but also it can obtain better generalization performance. In the ELM, the hidden layer needn't be tuned and the training parameters are fewer.

4.2.1. ELM: given a data set including N training samples, $S=\{(x_i,t_i)|x_i\in R^n,t_i\in R^m\}$, $i=1,2,\cdots,N$, the number of hidden nodes is L, thus, the ELM can be expressed as follows:

$$\sum_{l=1}^{L}\beta_l g(x_i)=\sum_{l=1}^{L}\beta_l g(w_l\bullet x_i+b_l)=t_i, i=1,2,\cdots,N \tag{14}$$

Where, β_l represents the weight vector that connects the lth hidden node and the output node; $g()$ represents the non-linear activation function of the hidden node, and which can be the identify sigmoid or Gaussian function; w_l represents the weight vector that connects the lth hidden node and the input node; b_l represents the threshold of the lth hidden node. The Equation (14) can be simplified as follows:

$$H\beta = T \tag{15}$$

Where,

$$H = \begin{bmatrix} g(w_1, b_1, x_1) & \cdots & g(w_L, b_L, x_1) \\ \vdots & \cdots & \vdots \\ g(w_1, b_1, x_N) & \cdots & g(w_L, b_L, x_N) \end{bmatrix}_{N \times L} \tag{16}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}_{L \times m} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \tag{17}$$

In the Equation (14), w_l and b_l are randomly set and g() is selected as sigmod function, so the matrix H is determined. According to the definition of the Moore-Perrose generalized inverse, we can obtain the β according to the Equation (18).

$$\beta = H'T \tag{18}$$

Where, H' represents the Moore-Perrose generalized inverse of H, and it can be get according to the Equation (19)

$$H' = (H^T H)^{-1} H^T \tag{19}$$

4.2.2. Online ELM: In the practical application, the training data set may be dynamic, which can be regarded as data stream. So, Yuan et al. proposed the online sequential ELM (OS-ELM) which achieves the incremental learning based on the initial training samples [18]. However, in the sliding window, not only the new incoming samples are added into, but also the outdated samples will be removed from the sliding window. In the sliding window, only the new incoming samples are not be learned, while the other samples have been learned at the last training process. Hence, there are many redundant data. Based on the OS-ELM, we propose the online ELM(O-ELM).

We set $P=H^TH$, $Q=H^TT$, at the time t_0, the H_0 and T_0 are represented as follows:

$$H_0 = \begin{bmatrix} H_{00} \\ H_{01} \end{bmatrix} \text{ and } T_0 = \begin{bmatrix} T_{00} \\ T_{01} \end{bmatrix} \tag{20}$$

Where, H_{00} represents the hidden layer output matrix obtained according to the samples in the first basic window; H_{01} represents the hidden layer output matrix obtained according to the samples besides the first basic window; T_{00} represents the target matrix obtained according to the samples in the first basic window; T_{01} represents the target matrix obtained according to the samples besides the first basic window. According to the Equation (20), we can obtain the following results:

$$P_0 = \begin{bmatrix} H_{00} \\ H_{01} \end{bmatrix}^T \begin{bmatrix} H_{00} \\ H_{01} \end{bmatrix} = H_{00}^T H_{00} + H_{01}^T H_{01} \tag{21}$$

$$Q_0 = \begin{bmatrix} H_{00} \\ H_{01} \end{bmatrix}^T \begin{bmatrix} T_{00} \\ T_{01} \end{bmatrix} = H_{00}^T T_{00} + H_{01}^T T_{01} \tag{22}$$

$$\beta_0 = P_0^{-1} Q_0 \tag{23}$$

At the time t_1, when the new incoming samples fill into the (Nb+1)th basic window, it will trigger the updating process, and Equation (21)~Equation (23) are transformed as follows:

$$P_1 = \begin{bmatrix} H_{01} \\ H_{10} \end{bmatrix}^T \begin{bmatrix} H_{01} \\ H_{10} \end{bmatrix} = H_{01}^T H_{01} + H_{10}^T H_{10} = P_0 - H_{00}^T H_{00} + H_{10}^T H_{10} \tag{24}$$

$$Q_1 = \begin{bmatrix} H_{01} \\ H_{10} \end{bmatrix}^T \begin{bmatrix} T_{01} \\ T_{10} \end{bmatrix} = H_{01}^T T_{01} + H_{10}^T T_{10} \tag{25}$$

$$\beta_1 = P_1^{-1} Q_1 \tag{26}$$

Where, H_{10} represents the hidden layer output matrix obtained according to the samples in the (Nb+1)th basic window; T_{10} represents the target matrix obtained according to the samples in the (Nb+1)th basic window; According to the Equation (21)~Equation (26), we can do following derivation:

$$\because H_{01}^T T_{01} = Q_0 - H_{00}^T T_{00} \quad , Q_0 = P_0 \beta_0$$

$$\therefore H_{01}^T T_{01} = P_0 \beta_0 - H_{00}^T T_{00}$$

$$\because P_0 = P_1 + H_{00}^T H_{00} - H_{10}^T H_{10}$$

$$\therefore H_{01}^T T_{01} = (P_1 + H_{00}^T H_{00} - H_{10}^T H_{10})\beta_0 - H_{00}^T T_{00} = P_1 \beta_0 + H_{00}^T H_{00} \beta_0 - H_{10}^T H_{10} \beta_0 - H_{00}^T T_{00}$$

$$\because \beta_1 = P_1^{-1} Q_1 = P_1^{-1}(H_{01}^T T_{01} + H_{10}^T T_{10}) = P_1^{-1} H_{01}^T T_{01} + P_1^{-1} H_{10}^T T_{10}$$

$$\therefore$$

$$\begin{aligned}
\beta_1 &= P_1^{-1} H_{01}^{\ T} T_{01} + P_1^{-1} H_{10}^{\ T} T_{10} \\
&= P_1^{-1} (P^1 \beta_0 + H_{00}^{\ T} H_{00} \beta_0 - H_{10}^{\ T} H_{10} \beta_0 - H_{00}^{\ T} T_{00}) + P_1^{-1} H_{10}^{\ T} T_{10} \\
&= \beta_0 + P_1^{-1} H_{00}^{\ T} H_{00} \beta_0 - P_1^{-1} H_{10}^{\ T} H_{10} \beta_0 - P_1^{-1} H_{00}^{\ T} T_{00} + P_1^{-1} H_{10}^{\ T} T_{10} \\
&= \beta_0 + P_1^{-1} H_{00}^{\ T} (H_{00} \beta_0 - T_{00}) - P_1^{-1} H_{10}^{\ T} (H_{10} \beta_0 - T_{10})
\end{aligned}$$

According to the derivation result, we can get the weight matrix that connects the hidden nodes and the output node at the time k:

$$\beta_k = \beta_{k-1} + P_k^{-1} H_{(k-1)0}^{\ T} (H_{(k-1)0} \beta_{k-1} - T_{(k-1)0}) - P_k^{-1} H_{k0}^{\ T} (H_{k0} \beta_{k-1} - T_{k0}) \tag{27}$$

Therefore, we can update the weight matrix based on the last obtained weight matrix.

4.3. The Construction of the Model

The input vector of the model of mobile user preference that introduces the context can be expressed as follows: $x_i=\{C_i,V_i\}$. Where, C_i is a vector of context instances. If mobile user preference toward the given service isn't impacted by some context, the value is set 0; else, the value is set according to the proposed quantification. Suppose that in the mobile network, there are three types of context: time, location and companion, if some user playing game is only impacted by time and location context, the vector of context instances is as follow: $\{0,q_t,q_l\}$, q_t and q_l are respectively the quantified values of some time and some location context instances. V_i represents the usage volume that mobile user uses the given service under common constraints of the context instances that belong to C_i. The output of the model $t_i=p_i$, p_i is the value of corresponding mobile user preference and $p_i \in [1,5]$ is a integer data.

The construction steps of the proposed model are as follows:

(1) Determine the contexts that impact mobile user preference according to the volatility obtained by the Equation (1).

(2) Quantify the context instances by the proposed context quantification method in the paper.

(3) Determine the sizes of the sliding window and the basic window by many experiments.

(4) Employ the O-ELM to learn the mobile user preference when the (Nb+1)th basic window is filled with new incoming data.

5. The Experimental Results And Analysis

In the section, we verify the performance of the proposed model using the real data set and analyze the experimental results.

5.1. The Data Set

We use the real data set to verify the performance of the proposed model in the paper [19]. The data set includes the behavior information of 94 mobile users from September

2004 to June 2005. There are more than 200 kinds of mobile network services in the data set. Before the experiment, we need to do some pretreatment. We add the week information according to the specific date into the data set. And then, we divide the time context according to the following rules, shown as Table 1 and Table 2.

Table 1. The Partition of the Time (Week)

week	Monday to Friday	Saturday	Sunday
context	workday	weekend	weekend

Table 2. The partition of the time (hour)

Time (hour)	6:00~ 12:00	12:00~ 18:00	18:00~ 24:00	0:00~ 6:00
context	morning	afternoon	evening	night

According to the time information and the base station information of the location of the mobile user, we divide the location context into three types: at home, at work and elsewhere.

5.2. Evaluation

In order to evaluate the proposed model, we choose MAE, precision and the running time as the evaluation.

MAE refers to Equation (28):

$$MAE = \frac{\sum_{i=1}^{N} \left| p_i - q_i \right|}{N_o} \tag{28}$$

Where, p_i represents the learned mobile user preference; q_i represents the actual mobile user preference; N_o represents the number of learned mobile user preferences. The smaller *MAE* is, the closer the learned mobile user preferences and the actual mobile user preferences are.

The *precision* refers to Equation (29):

$$precision = \frac{N_0}{N_o} \tag{29}$$

Where, N_0 represents the number of the learned accurate mobile user preferences. The larger the *precision* is, the higher the accuracy of the learned mobile user preferences is.

In addition, the running time is employed to evaluate the efficiency of the proposed model.

5.3. The Experimental Step

Step 1. Determining the value of υ . We set υ =0.2,0.3,0.4,0.5,0.6,0.7, and select the best results to do the following experiments.

Step 2. Determining the threshold of the volatility. We set the threshold as follows: 0,0.4,0.5,0.6,0.7,0.8. According to the experimental results, we select the appropriate value to do the further experiment.

Step 3. Determining the minimum support and minimum confidence. We set the minimum support as follows: 0.02,0.03,0.04,0.05, and set the minimum confidence as follows: 0.05,0.1,0.15,0.2. We select the best results to do further experiment.

Step 4. Comparing the proposed quantification method with the existing methods. We select the methods in reference [7] (method 1), reference [8] (method 2) and the proposed method to quantify the context respectively.

Step 5. Determining the size of the sliding window and basic window. We set the size of the sliding window and the basic window is a fixed period time. The size of the sliding window is set 4,8,12,16, and the size of the basic window is set 1,2,3,4, the unit is week.

Step 6. Comparing the different online learning methods: O-ELM, OS-ELM, O-LSSVM [10], O-BP [20].

5.4. The Experimental Results and Analysis

(1) the impact of υ . Where, the threshold of the volatility is set 0, the minimum support and minimum confidence are set 0.02 and 0.05 respectively. The results are shown as Figure 2.

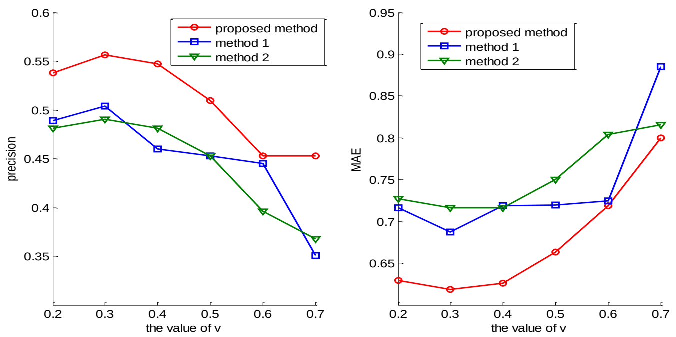


Figure 2. The Results Learned By the Different Methods When υ Is Set Different Values

The Figure 2 shows that when υ =0.3, the proposed method and the comparison methods all obtain the best results. It indicates the usage times plays a more important role than the duration. In the following experiment, we set υ =0.3.

(2) The impact of the threshold of the volatility. Where, the minimum support and minimum confidence are set 0.02 and 0.05 respectively. The results are shown as Figure 3.

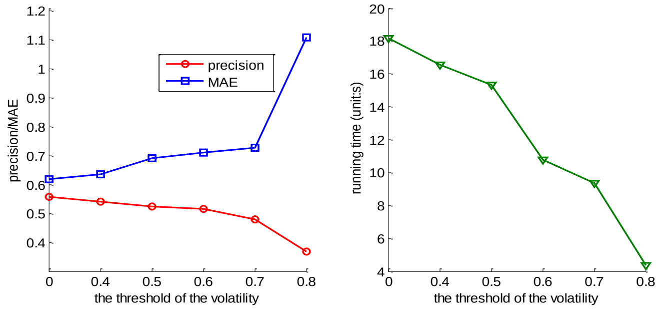


Figure 3. The Comparison Of Results And The Running Time When The Threshold Of The Volatility Is Set Different Values

From Figure 3, we can know that the accuracy changes worse with the increasing of the threshold of the volatility, but the running time changes shorter. This is because the mobile user preferences will decrease with the increasing of the threshold of the volatility. Considering the accuracy and the running time, the threshold of the volatility is set 0.5 in the further experiment.

(3) The impact of the minimum support and the minimum confidence. The results are shown as Figure 4. Firstly, we set the minimum support, and then set the minimum confidence based on the appropriate minimum support.

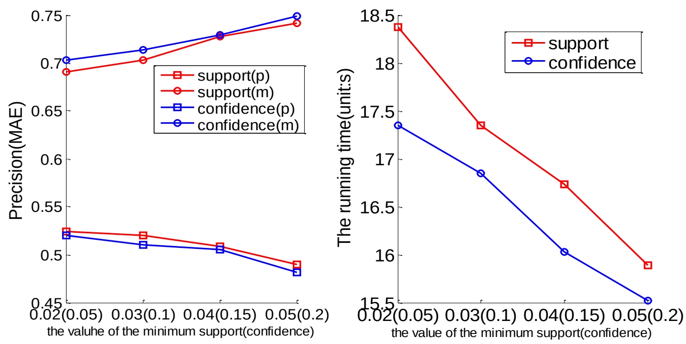


Figure 4. The Comparison of Results and the Running Time When the Minimum Support and Minimum Confidence Are Set Different Values Respectively

The Figure 4 shows that the learned results change worse with the increasing of the minimum support and minimum confidence, but the running time changes shorter. The result is similar with the impact of the threshold of the volatility. This is because that when the minimum support or the minimum confidence is smaller, it is very loose for the context relevancy and there are many contexts that are judged related. Therefore, the obtained results are better, but the complexity of the time is greater. In the further experiment, the minimum support and minimum confidence are set 0.03 and 0.1 respectively.

(4) The comparison of different quantification methods. The results are shown as Figure 1 and Table 3.

Table 3. The Precision of Different Methods

	N_0	N_1	N_2	N_3
method 1	50.36%	35.63%	12.68%	1.33%
method 2	49.06%	37.08%	11.12%	2.74%
proposed method	51.04%	37.09%	11.87%	0

note: N_1 represents the number where p_i-q_i=1; N_2 represents the number where p_i-q_i=2; N_3 represents the number where p_i-q_i=3.

From the Figure 1, we can know that the proposed method is higher 5.3% than method 1 and 6.6% than the method 2 in *precision*, smaller 0.0686 than the method 1 and 0.0977 than the method 2 in *MAE* in the best condition. The table 3 shows the more specific difference of the different methods, and it shows the proposed method is feasible in improving the results of learned mobile user preference. The reason is that we consider the context similarity and context relevancy in the process of context quantification.

(5) The impact of the size of the sliding window. The results are shown in the Figure 5.

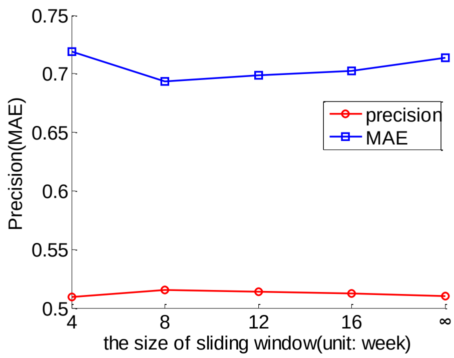


Figure 5. The Comparison of Results When the Size of the Sliding Window Is Set Different Values

From the Figure 5, we can know that when the size of the sliding window is set 8 week, the obtained results are the best. As the size increases, the performance decreases gradually; when the size of sliding window is set 4 week, the obtained results are unsatisfactory. The reason is as follows: when the size of the sliding window is set 4 week, the training samples are fewer, and the obtained mobile user preferences are not comprehensive, so the performance is worse; when the size is set more that 8 week, the training samples will introduce some noise data which can lead the negative impact, so the obtained performance becomes worse gradually. When the size of the sliding window is set infinity, the online learning transfers into the offline leaning, that is O-ELM transfers into ELM, so the obtained results are the same with that obtained by the step 4.

(6)The comparison of different online learning methods. The results are shown as Figure 6.

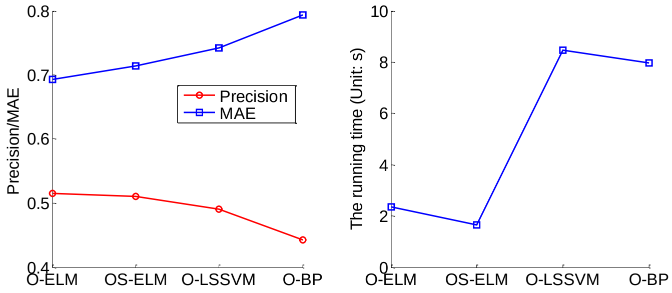


Figure 6. The Comparison of the Results Obtained By Different Online Learning Methods

In the figure 6, it shows that the accuracy of the mobile user preference obtained by the O-ELM is the best. The reason is that the OS-ELM considers the whole data set, so there are some noise data. But the running time of O-ELM is longer than OS-ELM. This is because that the OS-ELM only consider the new incoming data, while O-ELM considers both the new incoming data and the dated data. Due to the reason of the algorithm itself, the O-LSSVM and O-BP are worse in both accuracy and running time.

6. Conclusion

In order to improve the performance of learning model of the mobile user preference, the paper focuses on how to quantify the context reasonably and online learning. When quantifying the context, do some research about the following problems: how to judge whether the mobile user preference is affected by the context, how to calculate the context relevancy and the context similarity. Based on the above research, a context quantification method is proposed. In the proposed online learning model, the sliding window is employed to select the samples that need to be trained. In order to shorten the running time, we improve the OS-ELM and use the O-ELM to learn the mobile user preference. The experimental results verify that the proposed model can get better performance than the existing methods.

In the further work, we plan to introduce the social relationship of mobile user into the model. But, how to use the relationship needs more in-depth study.

Acknowledgement

This research was supported by the National Natural Science Foundation of China (NSFC) (Grant No. 61272509, 61402331 and 61402332).

References

[1] S. Ruchita, V. M. Thakare and S. S. Swati, “Mobile multimedia system approach for development of mobile applications & services with data visualization”, International Journal Of Computer Science And Applications, vol.6, no.2, pp. 107-112, (**2013**).
[2] L. C. Wang, X. W. Meng and Y. J. Zhang, “Context-aware recommender systems: a survey of the state-of-the-art and possible extensions”. Journal of Software, vol.23, no.1, pp. 1-20, (**2012**).
[3] M. Asif and J. Krogstie, “Mobile services personalization evaluation model”, International Journal of u- and e- Service, Science and Technology, vol.6, no.2, pp. 1-12, (**2013**).

[4] K. Stefanidis, E. Pitoura and P. Vassiliadis, "Adding context to preferences", Proceeding of the IEEE 23rd International Conference on Data Engineering, (**2007**) April 15-20; Istanbul, Terky.

[5] X. W. Meng, Y. C. Shi, L. C. Wang and Y. J. Zhang, "Review on learning mobile user preferences for mobile network services", Journal on Communications, vol.34, no.2, pp. 147-155, (**2013**).

[6] L. W. Liu, F. Lecue, N. Mehandjiev and L. Xu, "Using context similarity for service recommendation", Proceeding of the 2010 IEEE Fourth International Conference on Semantic Computing (ICSC), (**2010**) September 20-22; Manchester, UK.

[7] K. Oku, S. Nakajima, J. Miyazaki and S. Uemura, "Context-aware SVM for context-dependent information recommendation", Proceeding of the 7th International Conference on Mobile Data Management (MDM'06), (**2006**), May 10-12, Nara, Japan.

[8] T. H. Dao, S. R. Jeong and H. Ahn, "A novel recommendation model of location-based advertising: context-aware collaborative filtering using GA approach", Expert Systems with Applications, vol.39, pp. 3731-3739, (**2012**).

[9] Sarah M, Eliza P and Nick T. Adapting pervasive environments through machine learning and dynamic personalization. Proceeding of the International Symposium on Parallel and Distributed Processing with Applications. (**2008**), Decebor 10-12; Sydney, NSW.

[10] Y.C. Shi, X. W. Meng and Y. J. Zhang. "Adaptive learning approach of contextual mobile user preferences", Journal of Software, vol.23, no.10, pp. 2533-2549, (**2012**).

[11] C. Rana and S. K. Jain, "An evolutionary clustering algorithm based on temporal features for dynamic recommender systems", Swarm and Evolutionary Computation, vol.14, pp. 21-30, (**2014**)

[12] L. C. Wang, X. W. Meng and Y. J. Zhang, "A cognitive psychology-based approach to user preferences elicitation for mobile network service", Acta Electronica Sinica, vol.39, no.11, pp. 2547-2553, (**2011**).

[13] Y. Shi, A. Karatzoglou, L. Baltrunas and M. Larson, "TFMAP: optimizing MAP for top-n context-aware recommendation", Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, (**2012**), August 12-16, Portland, USA.

[14] H. Yang, "Improved collaborative filtering recommendation algorithm based on weighted association rules". Applied Mechanics and Materials, vol. 411, pp. 94-97, (**2013**).

[15] F. Carmen; A. Isaac and L. Javier, "Building trust from context similarity measures". Computer Standards & Interfaces, vol.36, no.4, pp. 792-800, (**2014**).

[16] H. T. Lam, W. J. Pei, A. Prado, B. Jeudy and É. Fromont, "Mining top-K largest tiles in a data stream", pp. 82-97, (**2014**).

[17] G. B. Huang, H. M. Zhou, X. J. Ding and R. Zhang, "Extreme learning machine for regression and multiclass classification", IEEE Transactions on Systems, Man, and Cybernetics, vol.42, no.2, pp. 513-529, (**2012**).

[18] L. Yuan, C. S. Yeng and G. B. Huang, "Ensemble of online sequential extreme learning machine" Neurocomputing, vol.72, pp. 3391-3395, (**2009**).

[19] N Eagle, A Pentland and D Lazer, "Inferring social network structure using mobile phone data", the National Academy of Sciences (PNAS), vol.106, no.36, pp. 15274-15278, (**2009**).

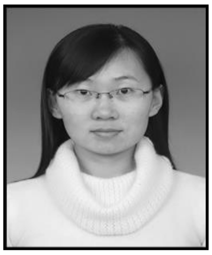
[20] R. Zhang, Z. B. Xu, G. B. Huang and D. Wang, "Global convergence of online BP training with dynamic learning rate", IEEE Transactions on Neural Network and Learning Systems, vol.23, no.2, pp. 330-341, (**2012**).

Authors

Yancui Shi received the bachelor's degree in Telecommunication Engineering from Hebei University of Science and Technology, MS and PhD degrees from North China Electric Power University and Beijing University of Posts and Telecommunications respectively. Now, she is a lecturer in the school of computer science and information technology, Tianjin University of Science and Technology. And her research focuses on the learning of mobile user preference, recommender system.

Congcong Xiong received the bachelor's degree in Computer Science from Shenyang University of Aeronautics and Astronautic and master's degree in Machinery Design and Theory from Tianjin University of Science & Technology. She is working in school of computer science and and information technology, Tianjin University of Science & Technology. Her research focuses on cloud computing, Distributed Network and Wireless Resource Management.

Jucheng Yang is a full professor in College of Computer Science and Information Engineering, Tianjin University of Science and Technology. He is a Specially-appointed Professor of Tianjin City and Haihe Scholar. He received his B.S. degree from South-Central University for Nationalities, in 2002, MS and PhD degrees from Chonbuk National University, Republic of Korea in 2004 and 2008. His research interests include image processing, biometrics, pattern recognition, and neural networks.



Yarui Chen is an associate professor in College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, P.R.China. She received her B.S. Degree from Hebei University of Technology, China, MS and PhD degrees from Tianjin University, China. Her research interests include machine learning, neural networks, probabilistic inference, and approximate inference.

Jianhua Cao, he is now an associate professor in School of Computer Science and Information Engineering, Tianjin University of Science and Technology. He got his doctor degree from China University of Geosciences(Beijing). He has more than 10 years of working experience in geophysical data processing and interpretation. He is now interested in data mining, pattern recognition, image processing, and their application in geophysical interpretation.

