



# A recommendation mechanism for contextualized mobile advertising

Soe-Tsyur Yuan\*, Y.W. Tsao

Department of Information Management, Fu-Jen University, 510 Chung Cheng Road, Hsinchuang, Taipei Hsien 24205, Taiwan, ROC

## Abstract

Mobile advertising complements the Internet and interactive television advertising and makes it possible for advertisers to create tailor-made campaigns targeting users according to where they are, their needs of the moment and the devices they are using (i.e. contextualized mobile advertising). Therefore, it is necessary that a fully personalized mobile advertising infrastructure be made. In this paper, we present such a personalized contextualized mobile advertising infrastructure for the advertisement of commercial/non-commercial activities. We name this infrastructure MALCR, in which the primary ingredient is a recommendation mechanism that is supported by the following concepts: (1) minimize users' inputs (a typical interaction metaphor for mobile devices) for implicit browsing behaviors to be best utilized; (2) implicit browsing behaviors are then analyzed with a view to understanding the users' interests in the values of features of advertisements; (3) having understood the users' interests, Mobile Ads relevant to a designated location are subsequently scored and ranked; (4) Top-*N* scored advertisements are recommended. The recommendation mechanism is novel in its combination of two-level Neural Network learning, Neural Network sensitivity analysis, and attribute-based filtering. This recommendation mechanism is also justified (by thorough evaluations) to show its ability in furnishing effective personalized contextualized mobile advertising.

© 2003 Elsevier Science Ltd. All rights reserved.

**Keywords:** Mobile commerce; Mobile advertising; Neural Network; Sensitivity; Analysis; Information filtering; Recommender systems

## 1. Introduction

With the popularity of mobile devices (such as wireless phones, PDAs, vehicle-mounted devices, etc.), technologies and applications have increasingly been focusing on new sets of tasks, problems, and domains that appreciate the advent of wireless computing and wireless Internet. For instance, mobile commerce refers to any activities related to a (potential) commercial transaction conducted through communication networks that interface with mobile devices. Varshey identified a few fields of applications in mobile commerce (Varshey & Vetter, 2001), such as mobile financial applications, mobile advertising, mobile inventory management, product locating and shipping, mobile entertainment, etc.

According to Ovum's report (Interactive Advertising: New Revenue Streams for Fixed & Mobile Operators) (Nelson, 2000), mobile advertising will begin to reach critical mass between 2002 and 2003 and ultimately generate USD16.4 billion by 2005. However, Ovum warned that the quickest way to alienate users is to inundate them with messages. Mobile advertising must be carried out with

a basic intention of offering something of value to the consumers. *Mobile advertising can complement Internet and interactive television advertising and make it possible for advertisers to create tailor-made campaigns targeting users according to where they are, their needs of the moment and the device they are using (i.e. contextualized mobile advertising).* Therefore, it is essential that a fully personalized mobile advertising infrastructure be made.

In this paper, we present a personalized contextualized mobile advertising infrastructure for advertising the commercial/non-commercial activities. We name this infrastructure MALCR—an abbreviation for Mobile Advertising by Location-based Customized Recommendation. The concepts behind MALCR are depicted in Fig. 1. The advertisements (obtained from Mobile AD service providers) represent the kind of information available in the environment. The users' mobile devices serve as points of access to the environment. MALCR then supplies the application that provides a device-independent gateway to information in the environment (Mark, 1999).

MALCR's contributions are three-fold:

1. Furnish a new mobile advertising infrastructure that can unfold both modes of interactive advertising (pull and push) characterized by location-based (Tewari et al.,

\* Corresponding author. Tel./fax: +886-2-2369-3220.

E-mail address: [yuans@tpts1.seed.net.tw](mailto:yuans@tpts1.seed.net.tw) (S.T. Yuan).

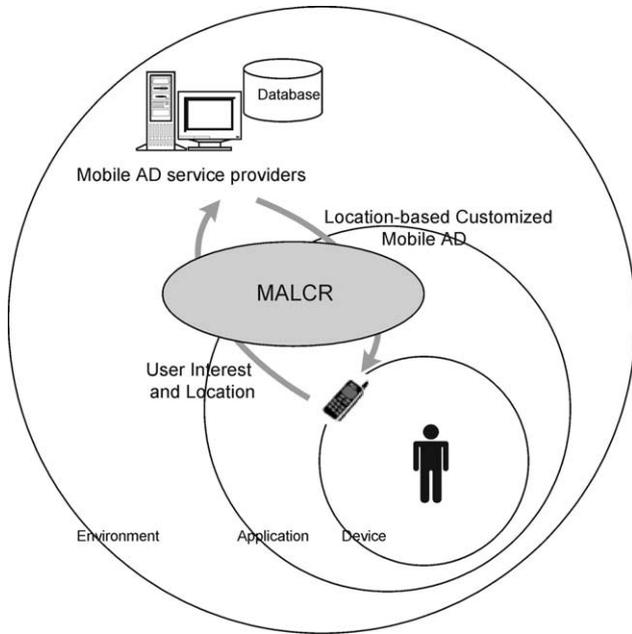


Fig. 1. MALCR's concepts.

2000) and customized recommendations (i.e. contextualized recommendations). We believe that such a method of advertising affects customers' participation of advertised activities better than non-location-based/non-customized advertisements. For instance, an advertisement of sale promotion activity at a particular location would better drive the participation of a customer who is nearby and who is interested in the items for sale than advertisements that do not apply on both scores.

2. Provide a representation space (called vector-based representation space) that is suitable for both the representation of the features in advertised activities and the analysis of users' interests.
3. Devise a recommendation mechanism that efficiently learns from users' handset-screen-browsing implicit behaviors and captures users' preferences in order to

provide good recommendations (the evaluations are to be shown in Section 5). This mechanism is mainly a combination of two-level Neural Network learning, Neural Network sensitivity analysis, and attribute-based filtering.

This paper is organized as follows: Section 2 provides the architecture of MALCR. Section 3 defines the vector-based representation space that is used in the recommendation mechanism that is presented in Section 4. Section 5 provides the evaluation results. Finally, a discussion and a conclusion are given in Sections 6 and 7, respectively.

## 2. The architecture of MALCR

This section provides MALCR's architecture (as shown in Fig. 2) and describes how the push and the pull modes of mobile advertising unfold (as shown in Fig. 3).

Fig. 2 shows that the primary tasks involved in MALCR are representing Mobile Ads, learning users' profiles, and providing recommendations by furnishing Mobile Ads that are most similar to a given user's profile and relevant to users' locations. In other words, there should be such a common representation space (to be described in Section 3) as to represent both Mobile Ads and users' profiles before they can be compared. This concept was initially deployed in text filtering (Oard & Marchionini, 1996), but it was first exerted in the area of mobile advertising.

By virtue of the limitations of mobile devices, it is better to learn users' profiles (understanding of user's needs) mostly from implicit browsing behaviors than to request users' interests from direct keypad inputs. As a result, an approach for learning users' profiles (represented in the common representation space) has to be devised and Section 4 will detail this mechanism.

Having Mobile Ads and users' profiles represented in the same space, what follows in the recommendation is

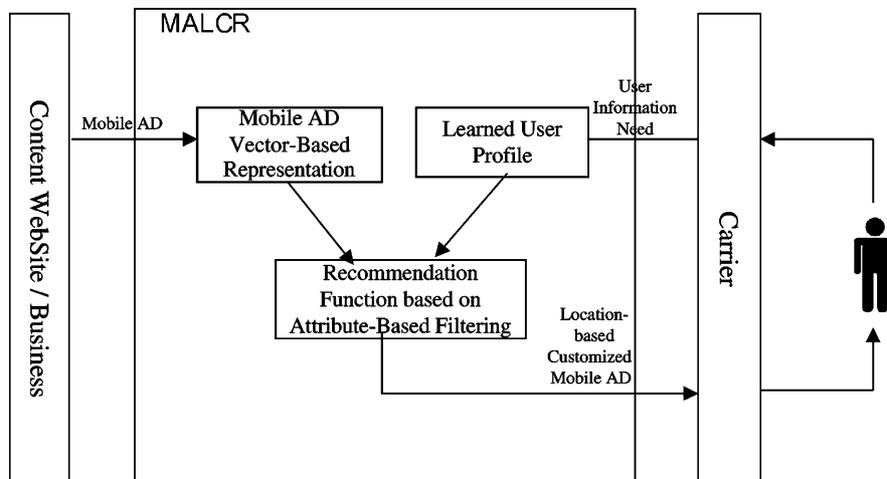


Fig. 2. The architecture of MALCR.

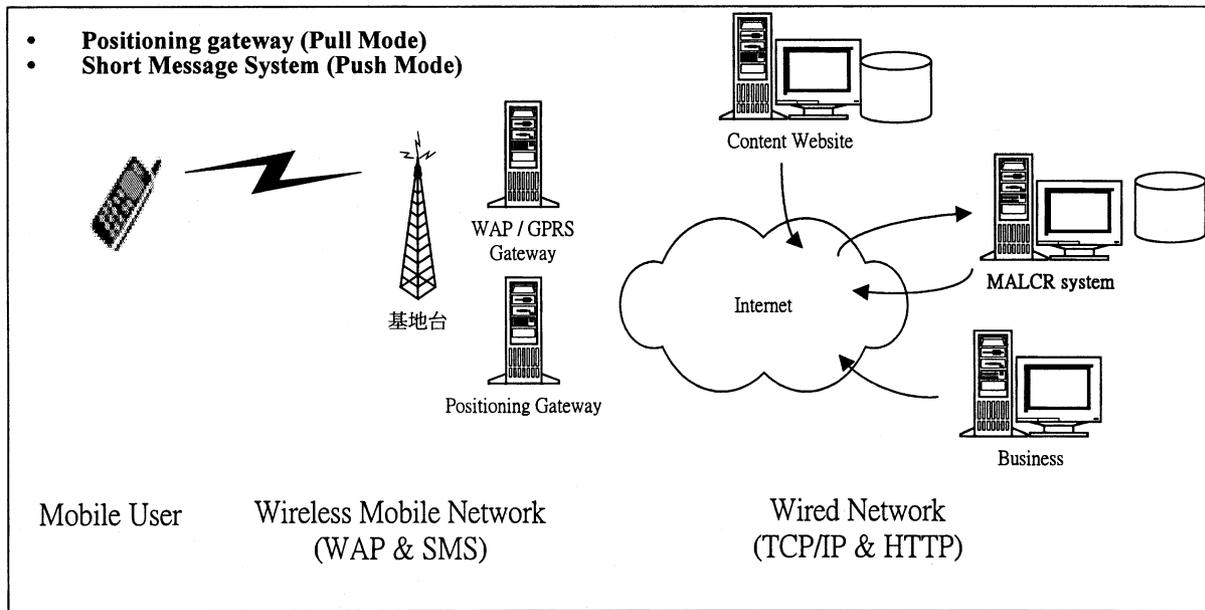


Fig. 3. Pull/push modes of mobile advertising.

the provision of a scoring mechanism (that will be detailed in Section 4).

Advertising through MALCR proceeds in two ways—the pull mode (the dominating mode) and the push mode (being feasible provided that permission from users is granted). From Fig. 3, working with the function of the positioning gateway residing in wireless carriers (to acquire the location of a given mobile user), the pull mode unfolds simply by the wireless shipping of the requested recommendations obtained from MALCR (that compares with the given user’s profile the Mobile Ads relevant to the location where the user invokes this pull). On the other hand, when a user grants permission, SMS is used to ship the recommendations obtained from MALCR (that compares with

the given user’s profile the Mobile Ads relevant to a location where the user last makes use of his/her mobile device so as to avoid the expensive location tracking of the user).

The in depth version of MALCR’s architecture is shown in Fig. 4 that details the components required to operate the tasks of representing Mobile Ads (the components of Mobile AS Extractor and Mobile AD Database), learning users’ profiles (the components of Personalization Agent, User Stereotype KB, and User Profile database), providing recommendations (the component of Recommendation Function).

Mobile AD Extractor transforms Mobile Ads with the vector-based representation and stores the transformed Ads

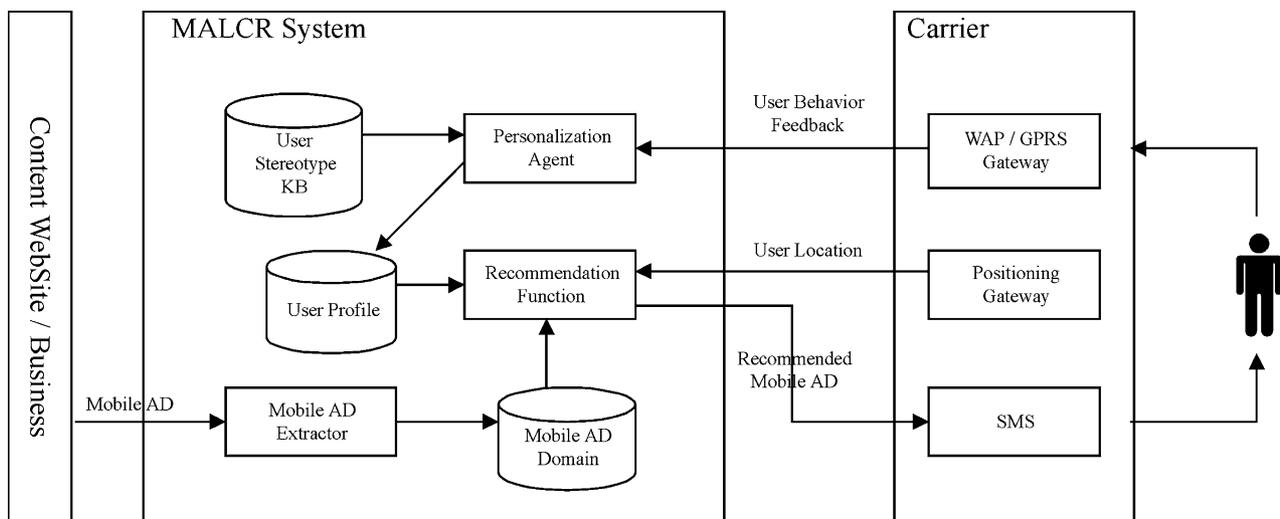


Fig. 4. The detailed version of MALCR’s architecture.

in the Mobile Ad Database. Personalization Agent (a combination of two-level Neural Network learning and Neural Network sensitivity analysis) learns users' profiles with the help of User Stereotype Knowledge Base (that is exerted to expedite the process of Neural Network learning) and stores the learned profiles into the User Profile database. The details of key components will be described in Section 4.

### 3. Vector-based representation space

In this section, we identify the features of advertised activities and present a suitable representation space (called vector-based representation space) that can be used in representing advertised activity Mobile Ads and users' profiles.

One major feature of advertised activities is their categories, exemplified in the first row of Table 1, based on their characteristics. The categories are wholesale and retail, arts and entertainment, etc. For instance, the activity of a sale promotion in a mall falls in the category of wholesale and retail, while the activity of a live performance of a singer belongs in the category of arts and entertainment. The usefulness of advertised activities to users often also depends on the other features of the activities such as day, time, place, fee, performers, etc. Each feature has its prevailing values, such as weekdays and weekends in the day attribute, free and fee-based in the fee attribute, etc. It is often the case that an advertised activity spans multiple values of a feature, such as the activity of a sale promotion in a mall takes place on both weekdays and weekends.

From the features of advertised activities described above, we believe a suitable representation for Mobile Ads should have the following traits: (1) a simple representation as Mobile Ads are short in lifespan and are updated frequently; (2) a representation allowing the encoding of the span of multiple values of multiple features; (3) a representation enabling simple comparison. As a result, the vector-based representation is chosen in this paper and is defined in the following two definitions (Definitions 1 and 2).

**Definition 1.** A Mobile Ad is represented as follows:

$$(I_1a_1, I_1a_2, \dots, I_1a_{m_1}, I_2a_1, I_2a_2, \dots, I_2a_{m_2}, \dots, I_na_1, I_na_2, \dots, I_na_{m_n})$$

$$I_ia_j \in \{0, 1\}, 1 \leq i \leq n, 1 \leq j \leq m_i$$

where  $n$  is the total number of features characterizing advertised activities;  $m_i$ , the number of possible values for the  $i$ th feature.

$$I_ia_j = \begin{cases} 1, & \text{if a given advertisement embodies the } j\text{th value} \\ & \text{of the designated } i\text{th feature} \\ 0 & \text{otherwise.} \end{cases}$$

**Definition 2.** A user profile is represented as follows:

$$(W_{11a1}, W_{11a2}, \dots, W_{11am_1}, W_{12a1}, W_{12a2}, \dots, W_{12am_2}, \dots, W_{1na1}, W_{1na2}, \dots, W_{1nam_n})$$

$$0 \leq W_{iiaj} \leq 1, 1 \leq i \leq n, 1 \leq j \leq m_i$$

where  $n$  is the total number of features characterizing advertised activities;  $m_i$ , the number of possible values for the  $i$ th feature;  $W_{iiaj}$ , a numerical value (ranging from 0 to 1) indicating a user's interest in the  $j$ th value of the designated  $i$ th feature.

For example, a given advertisement is about a SOGO sale promotion that features as follows (using the feature set of Table 1): Wholesale and Retail, Weekdays and Weekends, A time slot and B time slot, Indoors and Informal, Fee-Based, no performers. Accordingly, this advertisement is represented as the vector (1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0) as this advertisement is about a Wholesale and Retail activity and thus the values of  $I_1a_1, I_1a_2, \dots, I_1a_{m_1}$  ( $m_1 = 3$ ) are 1, 0, 0, respectively (similar reasoning for other features).

For a user profile example, take as an exemplar the user who is only interested in the activities of indoor promotion sales (such as sales in malls). Chances are the profile of this user looks like (0.52, 0, 0, 0, 0.05, 0, 0.1, 0, 0, 0.33, 0, 0, 0, 0) that shows non-zero preference values at the feature values corresponding to the user's interest. The computing of magnitude of preference at the feature values will be detailed in Section 4.

### 4. The recommendation mechanism

*The concepts underlying the recommendation mechanism are four-fold: (1) minimize users' inputs (a typical interaction metaphor for mobile devices) and thus implicit browsing behaviors are best utilized; (2) implicit browsing behaviors are then analyzed to the understanding of users' interests to values of features of advertisements; (3) with the understanding of users' interests, Mobile Ads relevant to*

Table 1  
Features in commercial/non-commercial advertisements

Attributes	Attribute values
Category	Wholesale and retail, arts and entertainments, others
Day	Weekdays, weekend
Time	A time slot (17:00 pm before), B time slot (17:00 pm after)
Place	Outdoors, indoors and formal, indoors and informal
Fee	Free, fee-based
Performer	Top celebrities, others

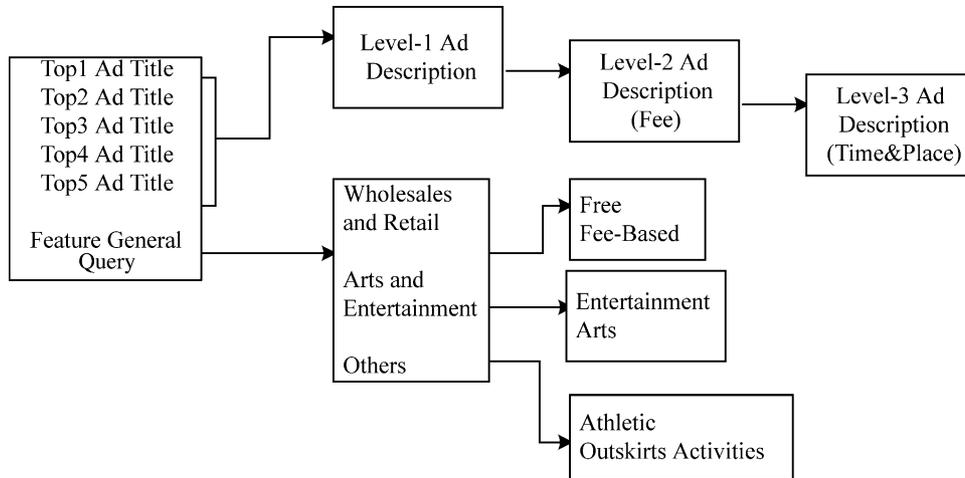


Fig. 5. Presumed browsing interface at mobile devices.

a designated location are subsequently scored and ranked; (4) Top-N (Karypis, 2001) scored advertisements are recommended.

After describing the vector space representation, what remains for the recommendation mechanism is the description of the task of learning users' profiles from implicit browsing behaviors and providing recommendations based on the understanding of users' interests to values of features of advertisements. We detail the task of learning users' profiles as follows: describe the objectives of User Stereotype KB and the contents in User Profile database (in Section 4.1) and present the functions of Personalization Agent (in Section 4.2). Subsequently, in Section 4.3 we delineate Recommendation Function that computes the recommendations from the learned user profiles and Mobile Ads (both of which are encoded in the vector space representation).

Before going into the details of subsequent sections, we first depict the browsing interface (as shown in Fig. 5) that we presume at the mobile devices. This browsing interface reveals the types of implicit browsing behaviors that can be analyzed in the task learning users' profiles.

Fig. 5 shows how the main screen displays the top five recommendations of activities each of which can be further clicked for more details (three levels of details are under consideration for the analysis of users' interests<sup>1</sup>). At the bottom of the main screen, a general query based on the advertisement features is also furnished in case the top five recommendations are not favored by the user.

Given the display design in Fig. 5, the implicit browsing behaviors, accordingly, can be of the variety of clicking order, clicking depth, and clicking count that are to be taken into account for understanding users' interests. Clicking

order means the order of clicking rendered on an item of recommendation. Clicking depth represents the number of levels of details clicked for an item of recommendation. Clicking count indicates the count of the requests for the complete details of an item of recommendation (i.e., the count of requests of the clicking depth of 3).

#### 4.1. User stereotype KB and user profile database

The objective of User Stereotype KB is to expedite the learning of the users' interests in Personalized Agent (that exerts two-level Neural Network learning).

The performance of a well-trained Neural Network has been recognized as being marvelous, but Neural Network learning does suffer from lengthy training and learning. However, a pre-training phase is capable of mitigating the problem of lengthy training (Yuan & Liu, 2000).

User Stereotype KB stores a set of pre-trained user stereotype vectors (and the corresponding learned Neural Network weights) representing a variety of typical users' interests, such as those who love activities of weekend arts and entertainment. Each pre-trained user stereotype vector is obtained by the following steps: (1) pre-train a Neural Network by feeding it with training examples each of which is composed of either a stereotype advertisement (an advertisement attracting the designated user stereotype) vector (please see Appendix A for details) and a score of 1 or an advertisement (that does not attract the designated stereotype user) vector and a score of 0; (2) perform Neural Network sensitivity analysis (to be described in Section 4.2) to the pre-trained Neural Network model and obtain a stereotype vector.

In User Profile Database, a user may be associated with multiple user stereotype vectors, each of which is initially brought in from User Stereotype KB when the user is new to MALCR and invokes the general query (shown in Fig. 5) (or the user is not new but invokes a new general query) and thus identifies the applicable user stereotype vectors.

<sup>1</sup> We believe it is rare for a mobile user browse beyond three levels of details due to the size limitation of the mobile devices' screens. However, if it is the case, for simplicity we take beyond three levels of details as detailed as three levels of details when analyzing users' interests.

The *pre-trained user stereotype vectors* associated with a user in the User Profile Database will evolve and thus become *customized user stereotype vectors* corresponding to the knowledge learned from the user’s implicit browsing behaviors. This evolution will be described in Section 4.2.

Accordingly, the user profile of a given user is defined as in Definition 3.

**Definition 3.** If a user is associated with the following user stereotype vectors:

User Stereotype 1 ( $W_{11}, W_{21}, W_{31}, \dots, W_{n1}$ )

User Stereotype 2 ( $W_{12}, W_{22}, W_{32}, \dots, W_{n2}$ )

⋮

User Stereotype  $m$  ( $W_{1j}, W_{2j}, W_{3j}, \dots, W_{nj}$ )

Then User Profile of this user is defined as follows:

User Profile ( $W_1, W_2, W_3, \dots, W_n$ )

where

$$W_i = \frac{\sum_{j=1}^m W_{ij}R_j}{m}$$

$$0 \leq W_i \leq 1, \quad 1 \leq i \leq n, \quad 1 \leq j \leq m, \quad 0 \leq R_j \leq 1$$

where  $n$  is the total number of feature values in the vector space representation;  $m$ , the total number of stereotype vectors associated with the user;  $R_j$  is the ratio of the reference of the  $j$ th stereotype vector to the total number of reference to all stereotype vectors.

For instance, a user that is associated with two stereotype vectors, (0, 0.36, 0, 0, 0.31, 0.03, 0.06, 0, 0.11, 0, 0, 0.05, 0.08, 0) and (0.42, 0, 0, 0.35, 0, 0, 0.13, 0, 0, 0.04, 0.01, 0.05, 0, 0), each of which is referenced (due to the user’s

browsing behaviors) 7 times and 3 times, respectively ( $R_1$  and  $R_2$  are 7/10 and 3/10), then this user’s User Profile is (0.126, 0.252, 0, 0.105, 0.217, 0.021, 0.081, 0, 0.077, 0.012, 0.003, 0.05, 0.056, 0).

4.2. Personalization Agent

Personalization Agent aims to learn by Neural Networks to understand users’ interests to values of features of advertisements from users’ implicit browsing behaviors. Personalization Agent employs two-level Neural Network learning together with Neural Network sensitivity analysis to achieve this end. In this section, we will explain the rationale behind two-level Neural Network learning (Section 4.2.1), describe the main steps involved in Personalization Agent, and how Neural Network sensitivity analysis is accomplished (Section 4.2.3).

4.2.1. Rationale behind two-level Neural Network learning

Neural Networks are often used with event triggers as inputs and event predictions as outputs and a set of event triggers and corresponding pre-known event predictions as training examples. We call this deployment of Neural Networks, one-level Neural Network learning.

In our case of learning to understand a user’s interests, pre-known event predictions have to be explicitly furnished by the user (such as event triggers’ scores,  $Score_U$ ) for corresponding event triggers (Mobile AD representation) when one-level Neural Network learning is employed (as shown in Fig. 6(a)), resulting in the learned and recommended event predictions ( $Score_R$ ).

However, because of the physical limitations of mobile devices (tiny keypads and screens, etc.), frequent requests of explicit inputs from users are not eligible. Therefore, an alternative way of deploying Neural Network learning has to be devised. Two-level Neural Network learning, accordingly (as shown in Fig. 6(b)), is such an appropriate deployment.

Instead of requesting explicit inputs of  $Score_U$  from users, *two-level Neural Network learning exerts the first*

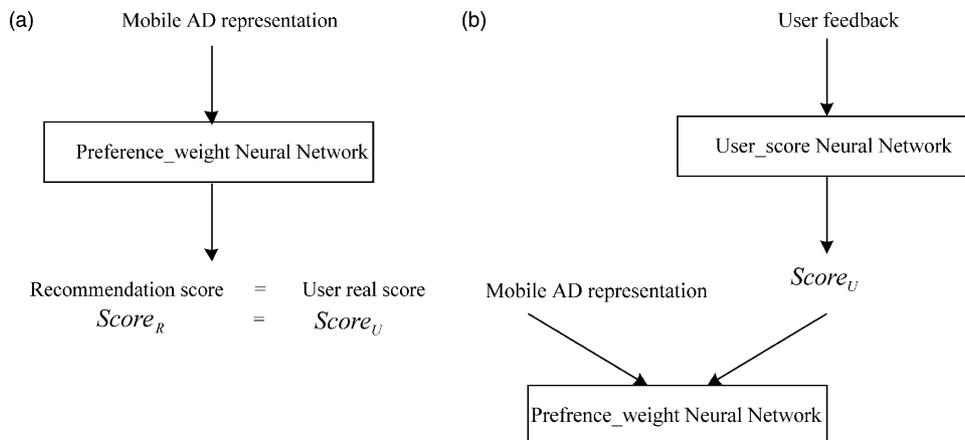


Fig. 6. one-level and two-level Neural Network learning shown in (a) and (b) respectively.

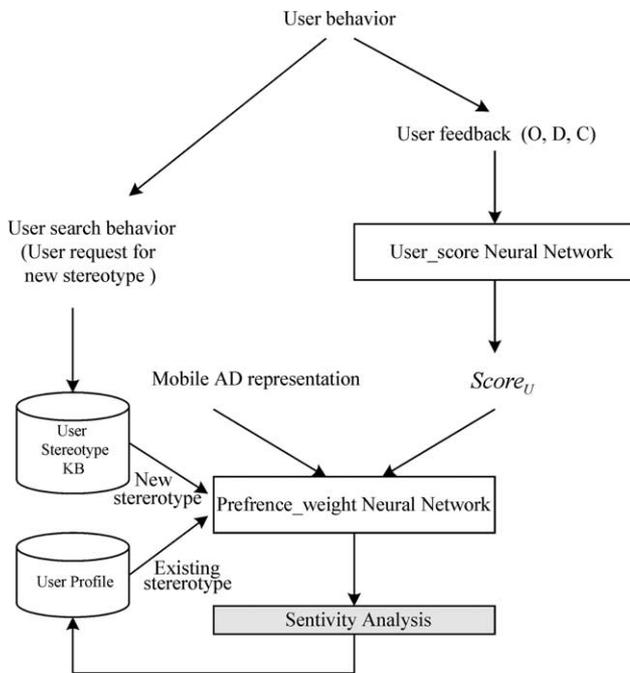


Fig. 7. Flow of steps in Personalization Agent.

*Neural Network (User\_Score Neural Network, abbreviated as USNN) to generate  $Score_U$  from users' implicit behaviors, such as clicking order ( $O$ ), clicking depth ( $D$ ), and clicking count ( $C$ ), and applies the second Neural Network (Preference\_Weight Neural Network, abbreviated as PWNN) to learn users' interests.*

USNN is such a pre-trained Neural Network that generates reasonable  $Score_U$  from the values of ( $O, D, C$ ). For instance, an item of recommendation with ( $O, D, C$ ) being (1,3,2) should have a higher  $Score_U$  than another item of recommendation with ( $O, D, C$ ) being (2, 1, 0).

#### 4.2.2. The algorithm of Personalization Agent

After giving the justifications of the use of two-level Neural Network learning, what remains is to present the complete flow of the steps as shown in Fig. 7 (i.e. the algorithm as shown below) of Personalization Agent.

From Fig. 7, the procedures of Personalization Agent for understanding a user's interests on values of features of advertisements are two-fold:

1. *On the request of a new stereotype.* Personalization Agent brings from User Stereotype KB the pre-trained user stereotype vector and the corresponding Neural Network weights into the User Profile. Compute the customized user stereotype vector by training PWNN (initialized by the Neural Network weights acquired from User Stereotype KB) with the training example that is composed of a Mobile Ad and the user's  $Score_U$  (predicted by using USNN with the user's implicit browsing behaviors ( $O, D, C$ ) to the Mobile Ad), and performing a sensitivity analysis to PWNN.

2. *On the use of existing stereotype<sup>2</sup>.* Evolve the customized user stereotype vector by training PWNN with the training example that is composed of a Mobile Ad and the user's  $Score_U$  (predicted by using USNN with the user's implicit browsing behaviors ( $O, D, C$ ) to the Mobile Ad), and performing Sensitivity Analysis to PWNN.

*Personalization\_Agent (Stype, M\_AD, O, D, C,)*

Stype is the pre-trained User Stereotype a user requests to add into his/her User Profile when using the query function looking for M\_AD Mobile AD. ( $O, D, C$ ) are the parameters of user feedback when a user browses M\_ADs,  $O$  represents order,  $D$  represents depth,  $C$  represents count.

1. Insert Stype into the User Profile if it is a new User Stereotype requested by the user.
2. Input ( $O, D, C$ ) to USNN (User\_Score Neural Network) and get  $Score_U$  as output.
3. Compose ( $M\_AD, Score_U$ ) as a training example of a User Stereotype which the M\_AD belongs to.
4. Use the User Stereotype's Neural Network weights as the initial weights of PWNN (Preference\_Weight Neural Network) if it is the case of a new User Stereotype requested by the user.
5. Train PWNN (that corresponds to the User Stereotype indicated in Step 3) with the training examples obtained from Step 3.
6. Use Sensitivity Analysis (SA\_Function) to generate the attribute preference weights from the PWNN, normalize the sum to 1, and store them back to the User Profile.

*SA\_Function ( )*

1. For  $i = 1$  to  $n$  do  
 $Score_i \leftarrow PWNN_{pretrained}(X_1, X_2, X_3, \dots, X_n)$

$$1 \leq i \leq n$$

$$X_j = 1, j = i$$

$$X_j = 0, j \neq i$$

for  $i$  indicates each input attribute of PWNN,  $X_i$  is each input value, and  $Score_i$  is the output value of pre-trained PWNN.

2. Compute  $Score_{sum}$

$$Score_{sum} = \sum_{i=1}^n Score_i$$

<sup>2</sup> Without loss of generality, each Mobile Ad corresponds to a User Stereotype, and thus any Mobile Ad browsed by a user contributes to the evolution of a User Stereotype vector (regardless of it is a new stereotype or an existing stereotype).

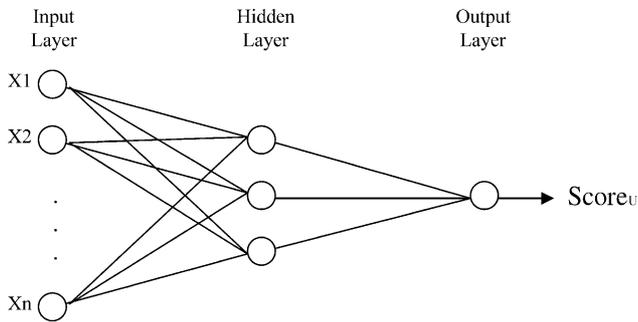


Fig. 8. PWNN structure.

### 3. Compute $W_i$

$$W_i = \frac{\text{Score}_i}{\text{Score}_{\text{sum}}}$$

where  $W_i$  is the preference weight of  $X_i$  in the User Stereotype.

The use of multiple User Stereotypes (and thus PWNNs) captured in User Profile prevents the quality of Neural Network learning from downfall due to abrupt drastic change in user's interests. In other words, the differentiation between Mobile Ads are to be well taken care of by their corresponding PWNNs and what needs to be done is to devise a way to integrate the customized user stereotype vectors obtained from PWNNs (this will be described in Section 4.3).

#### 4.2.3. Sensitivity analysis

In order to transform trained PWNNs (the understanding of a user's interests) into the vector-based representation, it is necessary to obtain the understanding of a user's interests in values of features in advertisements. Neural Network sensitivity analysis is often employed to this end (Frost & Karri, 1999; Han & Kamber, 2001)

The structure of PWNNs we employ (as shown in Fig. 8) includes 14 input nodes ( $X_1, X_2, X_3, \dots, X_n$ ) (corresponding to the representation of Mobile Ads shown in Table 1), three hidden-layer nodes, one output node (corresponding to  $\text{Score}_U$ ).

For a trained PWNN, the learned data are embedded in the weights of the PWNN. Sensitivity analysis aims to transform the black box of learned weights into a vector showing the user's interests among values of features in advertisement. The concepts are as simple as follows: (1) among 14 input nodes (i.e. 14 values of features), assign 1 to one feature value and 0 to the remaining feature values and compute the corresponding output ( $\text{Score}_i$ ); (2) repeat Step 1 to all of the input nodes; (3) sum those 14 output scores and obtain  $\text{Score}_{\text{sum}}$ ; (4) compute the percentage of  $\text{Score}_i$  to  $\text{Score}_{\text{sum}}$  and obtain the understanding of the user's relative preference to the designated feature value.

### 4.3. Recommendation Function

With the learned understanding of users' interests, Recommendation Function aims to provide a scoring mechanism that scores and ranks the Mobile Ads relevant to a designated location and then recommends the Top- $N$  scored advertisements. The algorithms of Recommendation Function are then shown below.

#### Recommendation\_Function ( $P, M\_ADs$ )

$P$  is the User Profile of a specific user.  $M\_ADs$  are the candidate Mobile Advertisements relevant to the specific user location.

#### 1. For each $M\_AD$ do

$$\text{Score}_R \leftarrow \sum_{i=1}^n \sum_{j=1}^m W_{I_{iaj}} I_{iaj}$$

$$1 \leq i \leq n, 1 \leq j \leq m$$

$I_{iaj}$  indicates the  $j$ th value of the  $i$ th feature in the  $M\_AD$

$W_{I_{iaj}}$  is the preference weight of the  $j$ th value of the  $i$ th feature in  $P$

#### 2. Rank the scores of $M\_ADs$

#### 3. Recommend Top- $N$ $M\_ADs$ if in the Pull mode

#### 4. Push Top-1 $M\_AD$ to the user if in the Push mode

The main idea behind Recommendation Function is *attribute-based filtering* that is described as follows: with the User Profile (described in Definition 3), the score of a given Mobile Ad can be simply the inner product of the vectors of the Mobile Ad and the User Profile. In other words, a Mobile Ad (represented by a vector of 0/1 over the feature values) that is of interest to a user (that is, the value of 1 mostly occurs to the feature values the user prefers) would induce a high score as the User Profile (represented by a vector of preference weight over values of features in advertisements) also embodies high values in those preference weights corresponding to 1's feature values in the Mobile Ad. Contrarily, a Mobile Ad that does not attract the user often has most of the 0 value go to the feature values that the user prefers (and thus corresponds to high preference weights in User Profile) and thus results in small value in score due to cancelling out of zero production.

## 5. Evaluation

For the nature of recommendation mechanisms in general, evaluations rest on the quality of the recommendations (Ben Schafer et al., 1999; Sarwar et al., 2000). Therefore, this section aims to provide the evaluation results on recommendation quality.

However, we need to first define the measurements we employ to perform the evaluation (Averaged Score<sub>U</sub> Growth, Instance Precision, Instance Recall, and Instance Fallout as described in Section 5.1), identify experimental user types (Extremely Focused, Extremely Scattered, Middle as described in Section 5.2), and then present the evaluation results for different cases. For example, the case where users will not change their interests and their experimental user types (Section 5.3); the case where users will change their interests but not their experimental user types (Section 5.4) and the case where users will change their interests but not their experimental user types (Section 5.5). In other words, we endeavor to furnish a thorough investigation of how MALCR's recommendation mechanism responds to a variety of situations as completely as we can, in order to justify the contributions of MALCR. (The simulated recommendation system is then shown in Appendix B.)

### 5.1. Recommendation quality measurements

This section defines and explains the four quality measurements (used in evaluating MALCR's recommendation mechanism), Averaged Score<sub>U</sub> Growth, Instance Precision, Instance Recall, and Instance Fallout:

- *Averaged Score<sub>U</sub> Growth.* MALCR's objective is to recommend Mobile Ads that are best suited to the user. Score<sub>U</sub> is a score computed from a user's implicit browsing behaviors (*O, D, C*) rendered on a recommendation that in turn manifests how close this recommendation matches the user's real interests. Therefore, average the Score<sub>U</sub> among the recommendations and attain an averaged Score<sub>U</sub> showing how close the Top-*N* recommendations match the user's interests. The nearer the averaged score is to the value of 1, the closer the Top-*N* recommendations match the user's interest. Accordingly, the growth of the averaged Score<sub>U</sub> between the user's feedback exhibits the velocity in attaining the accuracy of the recommendation mechanism.
- *Instance Precision, Instance Recall and Instance Fallout.* The measurements of Precision, Recall and Fallout have been widely used in the area of information filtering (Lanquillon, 1999) and recommendation systems. However, the measurements we employ are slightly different from previous definitions of these measurements, and thus they are called Instance Precision, Instance Recall and Instance Fallout. The word 'Instance', we take to mean that we see Precision, Recall, and Fallout from the aspect of the microcosm. That is, Precision, Recall, and Fallout are measured by comparing the elements between a learned vector representation (i.e. a Mobile Ad recommendation) and a target vector representation (i.e. a given user's preferred Mobile Ad). Table 2 then shows the microcosm view of element comparison

Table 2  
Instance-based measurements

		Target R1 representation	
		Feature value is 1	Feature value is 0
Learned representation	Feature value is 1	Found	False alarm
	Feature value is 0	Missed	Correctly rejected

between a learned vector representation and a target representation.

In Table 2, a matched value of 1 in a pair of corresponding elements between learned representation and target representation is interpreted as Found as it represents the feature value (corresponding to the element pair) preferred by a user and also appears at the learned recommendation. Similarly, a matched value of 0 then is interpreted as Correctly Rejected as the feature value disliked by the user also disappears at the learned recommendation. On the other hand, an unmatched pair of corresponding elements with the values 0 in the learning representation and 1 in the target representation is interpreted as Missed as the feature value preferred by the user disappears at the learned representation. Similarly, an unmatched pair of corresponding elements with the values 1 in the learning representation and 0 in the target representation is interpreted as False Alarm as the feature value disliked by the user nevertheless appears at the learned representation.

Based on the interpretations shown in Table 1 for Found, Correctly Rejected, Missed, and False Alarm, it is logical to define the instance-based measurements as follows:

#### Definition 4. Instance Precision

$$\text{Instance Precision} = \text{Found} / (\text{Found} + \text{False Alarm})$$

#### Definition 5. Instance Recall

$$\text{Instance Recall} = \text{Found} / (\text{Found} + \text{Missed})$$

#### Definition 6. Instance Fallout

$$\text{Instance Fallout} = \text{False Alarm} / (\text{False Alarm} + \text{Correctly Reject})$$

For a given recommendation, Instance Precision, Instance Recall, and Instance Fallout measure the percentage of the accurate hit among recommended feature values, the percentage of the accurate hit among the user's truly preferred feature values and the percentage of the inaccurate hit among the user's disliked feature values. Therefore, the higher the values in Instance Precision and Instance Recall,

the better the quality of MALCR's recommendations, however, the lower value in Instance Fallout, the better the recommendation quality. What follows are examples of the computation of these three measurements when the target representation is (1001111010100) and the learned representation is (10001111001100):

- Instance Precision =  $6/6 + 1 = 0.86$  (Found = 6, False Alarm = 1)
- Instance Recall =  $6/6 + 2 = 0.75$  (Found = 6, Missed = 2)
- Instance Fallout =  $1/1 + 6 = 0.14$  (False Alarm = 1, Correctly Reject = 6)

These three measurements complement each other in the following ways: (1) when high Instance Precision (e.g. False Alarm = 0) occurs but feature values still exist that are preferred by a user but not learned by the recommendation mechanism (i.e. Missed > 0), Instance Recall then can complement this void. In other words, *high quality in recommendations is justified only by high values in both Instance Precision and Instance Recall*. (2) Instance Fallout is able to differentiate the two cases where it is low in Instance Precision but one case is due to small Found (and thus small Fallout) and the other case is owing to large False Alarm (and thus large Fallout).

### 5.2. Experimental user types

Without loss of generality, certain experimental user types (Extremely Focused, Extremely Scattered, Middle) are assumed in order to investigate how robust MALCR's recommendation mechanism is in the following three sets of experiments discussed in this section.

The three sets of experiments are deployed as follows: (1) for each set of experiments, there are three types of users, each of which is comprised of 50 users (each of which exercises MALCR 10 times (represented as Login1, Login2,...,Login10) after his/her MALCR's first use (represented as Login0)); (2) record the measurements of  $Score_U$ , Instance Precision, Instance Recall, and Instance Fallout that are produced from each use of MALCR.

The following are then the descriptions of the three experimental user types that differentiate with each other mainly in the frequencies of the use of general query:

- *Extremely Focused (U1)*. This user type exemplifies the group of users whose interests are highly concentrated. Without loss of generality, this type of user is simulated as follows: (1) a general query (and thus a pre-trained User Stereotype) is randomly generated to emulate the first use of MALCR (Login0) by such a user; (2) subsequent recommendations from MALCR (i.e. recommendations obtained in Login0, Login2,..., Login10) are assumed to conform to this user's interests.

- *Extremely Scattered (U2)*. This user type represents the group of users whose interests are spread over a wide variety of advertisements. Therefore, we simulate this type of user as follows: three general queries (and thus three pre-trained User Stereotypes) are generated via MALCR (the number 3 is heuristically chosen for manifesting the semantics of 'scattered' that is feasible with the use of mobile devices by the user).
- *Middle (U3)*. This user type indicates the group of users acting between the previous two extreme types of users. Therefore, this user type is simulated as follows: (1) two general queries are generated in each use of MALCR from Login1 to Login5; (2) subsequent recommendations from MALCR (i.e. recommendations obtained from Login6 to Login10) are assumed to conform to this user's interests.

### 5.3. Stable User's interests and experimental user type

This set of experiments aims to investigate how the quality of MALCR varies with numerous uses of MALCR on the condition that a target representation (representing a user's interests) is randomly assigned prior to Login0 and stays the same throughout Login1–Login10.

The evaluation results of  $Score_U$  and those instance-based measurements are shown in Figs. 9 and 10 and they disclose the following observations:

- The exhibition of 1 in  $Score_U$  (shown in Fig. 9) for all of the three user types (U1, U2, U3) manifests that users' interests are well captured after the general query invoked at Login0 regardless of the different number of user stereotypes subsequently employed by the three different user types.
- Fig. 10 exhibits an averaged Instance Precision of 0.95, an averaged Instance Recall of 0.88, and an averaged Instance Fallout of 0.06 at Login10 (similarly for Login1–Login9). That is, a combination of the high values in Instance Precision and Instance Recall and the low values in Instance Fallout exactly indicate a high recommendation quality of MALCR. In other words, MALCR exhibits the competence of learning user's interests effectively in this set of experiments.

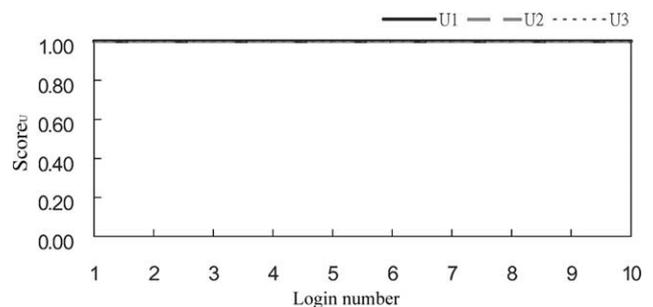


Fig. 9. Averaged  $Score_U$  Growth for the case of stable user interests and stable user type.

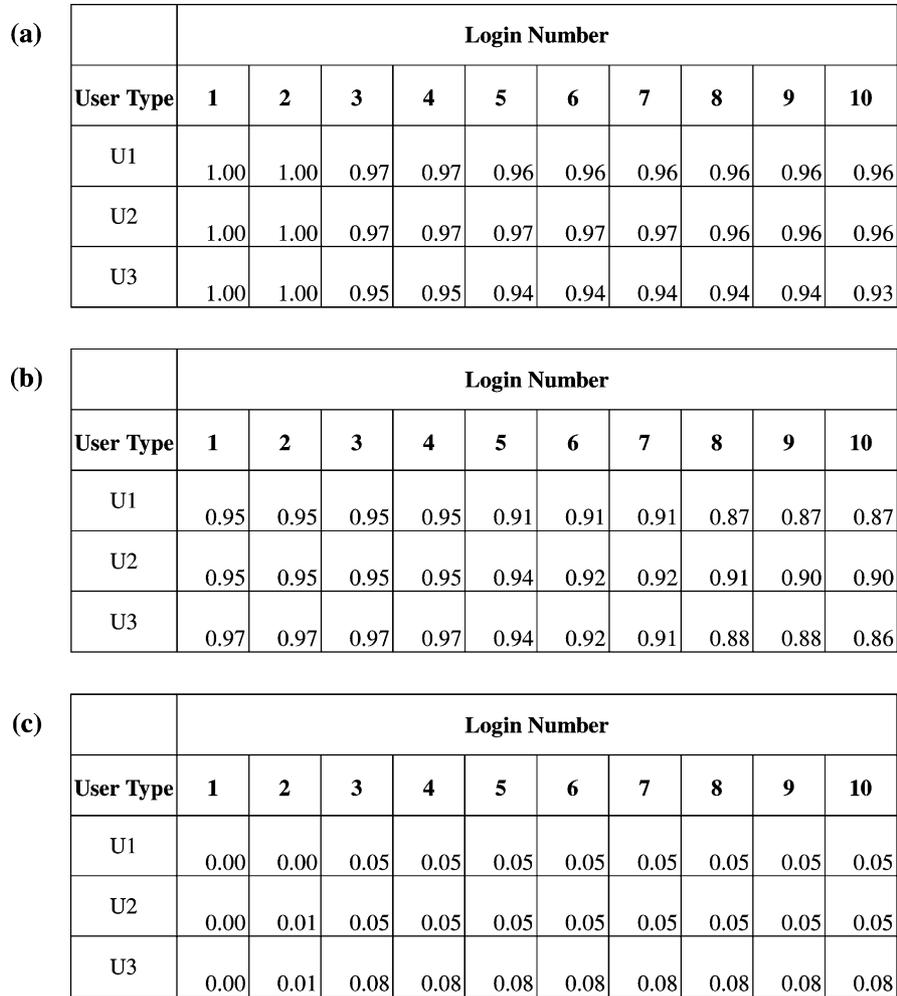


Fig. 10. The measurements of Instance Precision (a), Instance Recall (b), and Instance Fallout (c) for the case of stable user interests and stable user type.

5.4. Unstable user's interests but stable user type

This set of experiments investigates how the quality of MALCR varies with numerous uses of MALCR when there is a change in the interests of a user who, nonetheless, will not change his/her user type.

The experiment setting is then deployed as follows: (1) There are two randomly assigned target representations; (2) the first one is generated prior to Login0 to represent the user's initial interests; (3) the second one is generated at Login3 to signal the change of the user's interests.

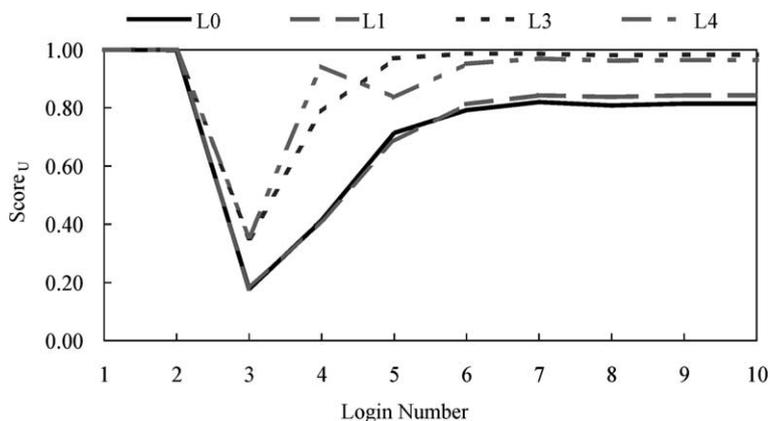


Fig. 11. Score<sub>u</sub> results for the case of unstable user's interests and stable user's user type.

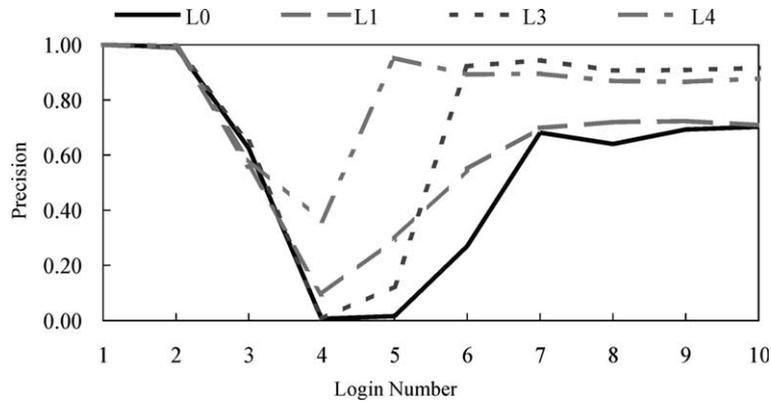


Fig. 12. Instance Precision results for the case of unstable user’s interests and stable user’s user type.

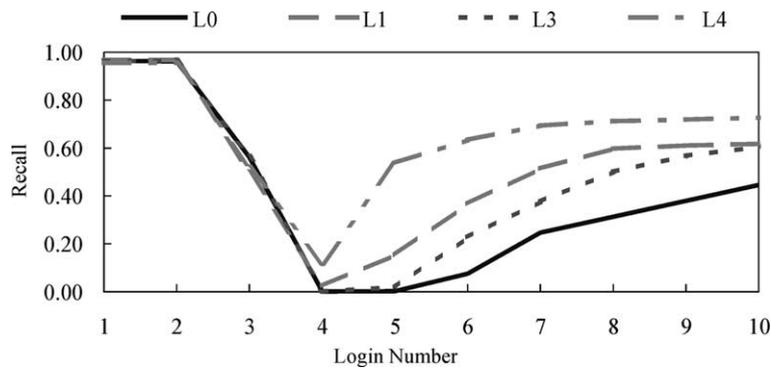


Fig. 13. Instance Recall results for the case of unstable user’s interests and stable user’s user type.

However, the change of a user’s interests is operated by the user in two ways: (1) explicit change in which the user drives the change by invoking a new general query (and thus bringing in a new pre-trained User Stereotype); (2) implicit change in which the user changes his/her interests with the browsing behaviors and no new pre-trained User Stereotype is brought in. Implicit change in the user’s interests is anticipated to give rise to more complexity in learning which consequently results in a case of explicit change. We would like to explore how robust MALCR can be in the case of implicit change.

Another experiment variable employed in the experiment setting is if we weigh the most recent User Stereotype (stored in database of User Profile) more than the others due to the consideration of the time at which User Stereotypes are brought in (i.e. an intuition that the newly identified User Stereotype is supposed to reflect more the user’s contemporary interests). For simplicity, we investigate this issue with the principle of 80/20 (i.e. the most recently employed User Stereotype is weighed 80% of importance in comparison with 20% of importance rendering on the other User Stereotypes).

Combining the factors of explicit change/implicit change and yes/no weighing on the most current User Stereotype, there are four situations shown as below:

- Implicit change and no weighing on the most current User Stereotype (L0)
- Implicit change and yes weighing on the most current User Stereotype (L1)
- Explicit change and no weighing on the most current User Stereotype (L3)
- Explicit change and yes weighing on the most current User Stereotype (L4)

In this set of experiments, for each user type (U1–U3), we conduct the experiments for each of the situations (L0–L4). As a result, there are plenty of figures (for details please

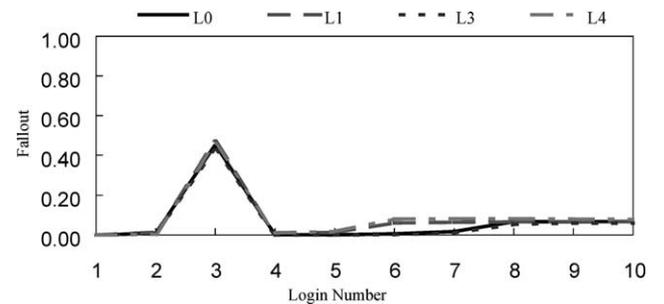


Fig. 14. Instance Fallout results for the case of unstable user’s interests and stable user’s user type.

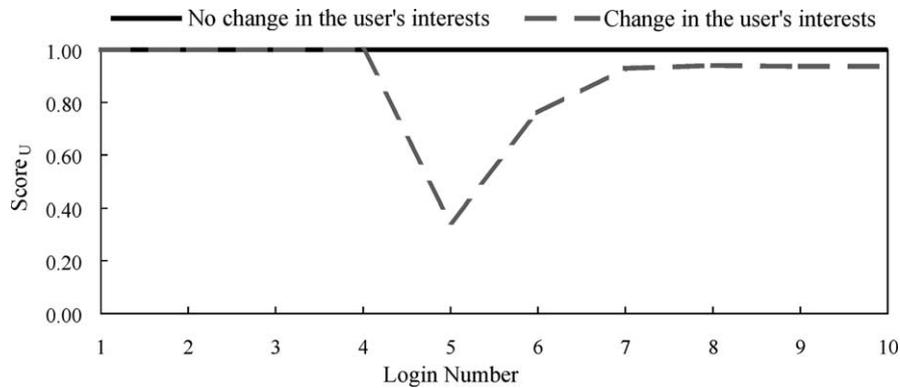


Fig. 15. Score<sub>U</sub> results for the case of unstable user type.

see Taso and Yuan (2002)). In order to condense the results, we average over the user types and obtain only one set of the averaged results for L0–L4. The evaluation results of Score<sub>U</sub> and those instance-based measurements for the four situations are then shown in Figs. 11–14 that disclose the following observations:

- From Fig. 11, Score<sub>U</sub> in the situations of L0 and L1 (i.e. the cases of implicit change), in general, is inferior to that in the situations of L2 and L3 (i.e. the cases of explicit change). However, after three times of learning (Login3–Login5), Score<sub>U</sub> in the cases of implicit change still performs as well as an accuracy of 0.8. On the other hand, this figure also shows that the factor of yes/no weighing on the most current User Stereotype does not significantly affect the accuracy rate.
- From Fig. 11, Score<sub>U</sub> in the situation of L0 reaches as high as an accuracy of 0.97 after two times of learning (Login4 and Login5), but that of L1 oscillates at Login3–Login5 even though the accuracy rate is still high. The rationale behind this oscillation can be explained as follows: the pre-trained User Stereotype (that is brought in by a user who explicitly invokes the general query) is quite different in nature from Login3's randomly generated target representation (as described in the above experiment setting).
- From Figs. 12–14, we know there is no significant difference between yes/no weighing on the most current

User Stereotype. In the cases of explicit change, the situation of no weighing is even slightly better than the situation of yes weighing.

- Instance Precision and Instance Recall in the cases of explicit change, in general, are superior to those in the cases of implicit change. However, they all exhibit quite a satisfactory level of quality in learning the target representations (about Instance Precision of 75–95% and Instance Recall of 50–75%).
- Instance Fallout is very low for all situations (L0–L4).
- In summary, the quality of MALCR's recommendations in this set of experiments, in general, is good because of satisfactory Precision Instance and Precision Recall and low Precision Fallout (regardless of the user types U1–U3, the situations of implicit/explicit change and yes/no weighing on the most current User Stereotype).

### 5.5. Unstable experimental user type

This set of experiments investigates how the quality of MALCR varies with numerous uses of MALCR when there is a change in the user type (from Extremely Focused to Extremely Scattered).

The experiment setting is then deployed as follows: (1) the change in the user type occurs at Login4. That is, Login1–Login3 take on the user type of Extremely Focused and Login4–Login10 take on the user type of Extremely Scattered; (2) there are two subsets of experiments—one

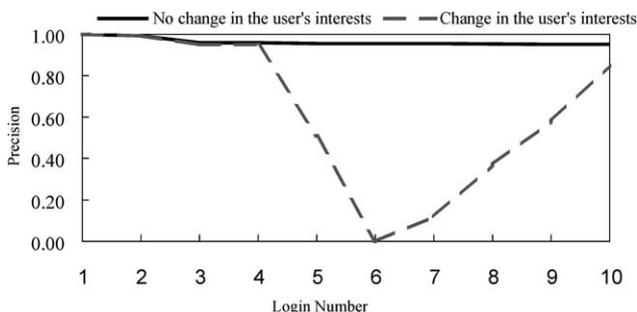


Fig. 16. Instance Precision for the case of unstable user type.

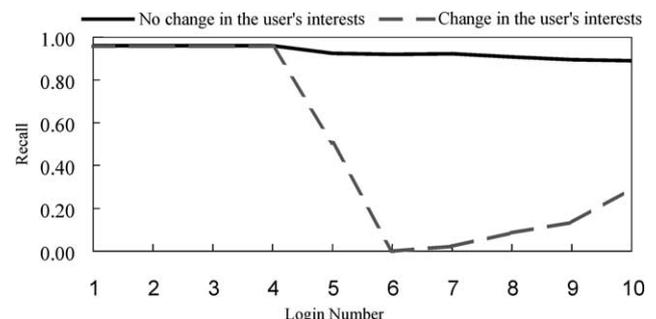


Fig. 17. Instance Recall for the case of unstable user type.

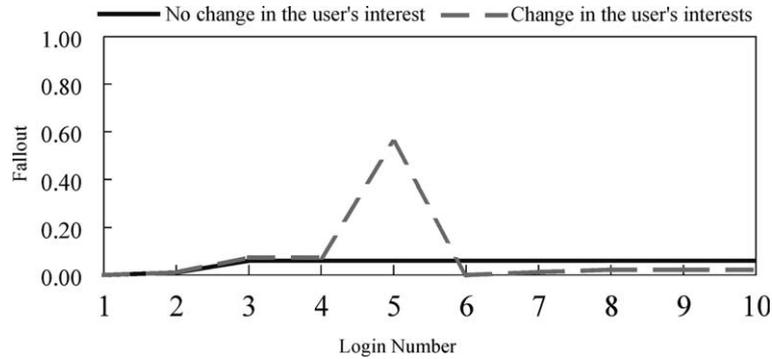


Fig. 18. Instance Fallout for the case of unstable user type.

for the case at which there is no change in the user's interests and the other for the case at which there is a change in the user's interests (that takes place at Login5). The change in the user's interests is exercised with the approach of explicit change and no weighing on the most current User Stereotype (that performs the best as discussed in Section 5.4)

The evaluation results of  $Score_U$  and the instance-based measurements for the four situations are then shown in Figs. 15–18 that disclose the following observations:

- The case in which there is no change in the user's interests always performs well (in terms of  $Score_U$  and instance-based measurements) regardless of the user types employed.
- At the case in which there is a change in the user's interests at Login5,  $Score_U$  is back to a level as good as 0.88 after two times of learning since Login5. Instance Precision and Instance Fallout exhibit MALCR's high accuracy in learning the user's preferred feature values, but Instance Recall shows the slow progress of learning the whole set of the user's preferred feature values. However,  $Score_U$  of 0.88 and high Instance Precision already adequately show MALCR's good performance.

## 6. Discussion

Although this paper presents a Mobile Ads' recommendation mechanism that is justified with good evaluation results in terms of the quality of recommendations, there are still other ways of evaluating the advertising effects such as communicative effects and sales effects (Hsu, 2000).

The effects of Internet-based advertising are often measured by such methods as the number of page visits, the number of visitors, etc. (Lee, 2001). For Mobile Ads, appropriate methods of evaluating the advertising effects are worthy of further investigation.

For MALCR, we devise such a method represented as a measurement (defined in Definition 7) that will be evaluated in our future work. This method measures the advertising effect at the use of MALCR's push mode to a user (i.e. a pushed Top-1 recommendation). The concepts behind this

method are two-fold: (1) the Top-1 recommendation affects the user if this user responds by exerting MALCR; (2) this response is correlated with the Top-1 recommendation only when this response is made within a certain amount of time; (3)  $Score_U$  rendered on the Top-1 recommendation is used to represent the magnitude of this advertising effect.

### Definition 7. Advertising effect measurement

$$\text{effect} = L \times \frac{1}{\log T} \times \text{Score}_U$$

where  $L$  (login) is 1 if a user exerts MALCR after receiving the Top-1 recommendation pushed by MALCR and 0, otherwise;  $T$  (time), the lapse of time between the push of the Top-1 recommendation by MALCR and the exertion of MALCR by a user;  $S$  ( $Score_U$ ) is the  $Score_U$  rendered on the Top-1 recommendation when  $L = 1$ .

## 7. Conclusion

In this paper, we present a personalized, contextualized mobile advertising infrastructure (MALCR) for the advertisement of commercial/non-commercial activities. The contributions of MALCR are three-fold: (1) furnish a new mobile advertising infrastructure that can unfold both modes of interactive advertising (pull and push) characterized with location-based and customized recommendations (i.e. contextualized recommendations); (2) provide a representation space (called vector-based representation space) that is suitable for both the representation of the features in advertised activities and the analysis of users' interests; (3) devise a recommendation mechanism that efficiently learns implicit behaviors from users' handset-screen-browsing and captures users' preferences in order to provide good location sensitive recommendations. This mechanism is largely a combination of attribute-based filtering, two-level Neural Network learning, and Neural Network sensitivity analysis. The evaluation results exhibit good recommendation quality embodied in MALCR. This paper also devises a method for measuring MALCR's advertising effect (that is to be investigated in our future work).

Table A1

Features	Category			Day		Time		Place			Fee		Performers		Example
	Whole sales and retail	Arts and entertainment	Others	Week day	Week end	A Slot	B Slot	Out doors	Indoors and formal	Indoors and informal	Free	Fee -based	Top-100 performers	Others	
Stereo type	I <sub>1a1</sub>	I <sub>1a2</sub>	I <sub>1a3</sub>	I <sub>2a1</sub>	I <sub>2a2</sub>	I <sub>3a1</sub>	I <sub>3a2</sub>	I <sub>4a1</sub>	I <sub>4a2</sub>	I <sub>4a3</sub>	I <sub>5a1</sub>	I <sub>5a2</sub>	I <sub>6a1</sub>	I <sub>6a2</sub>	
S01	1	0	0	1	1	1	1	1	0	1	0	1	0	0	Discounted sales
S02	1	0	0	1	1	1	1	0	0	1	1	0	0	0	Member-based sales
S03	1	0	0	0	1	1	0	1	0	0	1	1	1	0	Stars promoting sales
S04	0	1	0	1	0	0	1	0	1	0	0	1	1	0	Weekday arts/entertainment
S05	0	1	0	0	1	1	1	0	1	0	0	1	1	0	Weekend arts/entertainment
S06	0	1	0	1	1	1	0	1	0	0	1	1	0	0	Local culture arts exhibition
S07	0	1	0	1	1	1	0	0	1	0	0	1	0	0	Country culture arts exhibition
S08	0	1	0	0	1	0	1	0	1	0	0	1	1	0	Concerts
S09	0	1	0	0	1	1	1	1	0	0	1	0	1	1	Fee-to-charity performance
S10	0	1	0	0	1	0	1	1	1	1	1	0	1	1	Singer contact /movie preview
S11	0	0	1	0	1	1	0	0	1	0	0	1	1	0	Formal athletic competition
S12	0	0	1	1	1	1	1	1	0	1	1	0	0	0	Athletic activities
S13	0	0	1	1	1	1	0	1	0	0	1	1	0	0	Garden arts exhibition
S14	0	0	1	0	1	1	0	1	0	0	1	1	0	0	Outskirts activities

Appendix A. The set of stereotype activities

Table A1

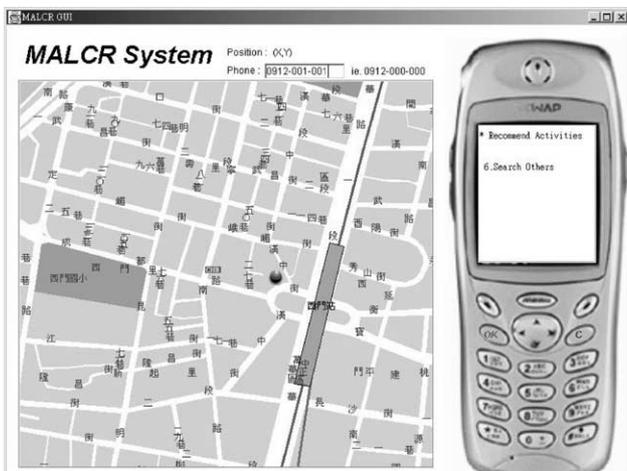


Fig. A1.

Appendix B. MALCR’s simulated user interface

Fig. A1.

References

Ben Schafer, J., Konstan, J., & Riedi, J. (1999). Recommender systems in e-commerce. *Proceedings of the First ACM Conference on Electronic Commerce*, 158–166.

Frost, F., & Karri, V. (1999). Determining the influence of input parameters on BP neural network output error using sensitivity analysis. *Third International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '99)*, 45–49.

Han, J., & Kamber, M. (2001). *Data mining concepts and techniques*. Morgan Kaufmann Publishers, pp. 303–311.

Hsu, S. F (2000). *A study on the types and properties of internet advertising effects*. Master Thesis, Graduate Institute of Business Administration, National Taiwan University, Taiwan.

Karypis, G (2000). *Evaluation of item-based Top-N recommendation algorithms*. Technical report CS-TR-00-46, Computer Science Department, University of Minnesota.

Lanquillon, C. (1999). Information filtering in changing domains. *Workshop on Machine Learning for Information Filtering, International Joint Conference on Artificial Intelligence (IJCAI'99)*.

- Lee, R. C (2001). *Evaluation of internet-based advertisements*, [http://magazines.sina.com.tw/brain/contents/300/300-004\\_1.html](http://magazines.sina.com.tw/brain/contents/300/300-004_1.html).
- Mark, W. (1999). Turning pervasive computing into mediated spaces. *IBM Systems Journal, Pervasive Computing*, 38(4).
- Nelson, R (2000). *Interactive advertising: New revenue streams for fixed and mobile operators*. An ovum report.
- Oard, D.W., & Marchionini, G (1996). *A conceptual framework for text filtering*. Technical report CS-TR3643, University of Maryland, College Park, MD.
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2000). Analysis of recommendation algorithms for e-commerce. *Proceedings of the Second ACM Conference on Electronic Commerce*, 158–167.
- Taso, E., & Yuan, S.-T (2002). *Location-based and customized mechanism for mobile advertising*. Technical report, Information Management Department, Fu-Jen University, Taiwan.
- Tewari, G., Youll, J., & Maes, P. (2000). Personalized location-based brokering using an agent-based intermediary architecture. *Proceedings of the 2000 International Conference on Electronic Commerce, Seoul, Korea*.
- Varshey, U., & Vetter, R. (2001). A framework for the emerging mobile commerce applications. *Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34)*.
- Yuan, S.-T., & Liu, A. (2000). Next-generation agent-enabled comparison shopping. *International Journal of Expert Systems with Applications*, 18(4), 283–297.