A PRACTICAL APPROACH FOR MOBILE RECOMMENDATION AGENTS OVER THE INTERNET

*Timothy K. Shih, Chuan-Feng Chiu and Hui-huang Hsu**

Department of Computer Science and Information Engineering
Tamkang University, Tamsui, Taipei, 251, Taiwan
Email: {tshih, cfchiu}@cs.tku.edu.tw
*Takming College, Neihu, Taipei, 114, Taiwan
Email: hhsu@mail.takming.edu.tw

ABSTRACT

The Internet is a popular medium for information delivery and exchange. Recommendation systems and mobile technology have become important academic research topics in electronic commerce (EC). In this paper we propose a mobile architecture for recommendation systems and two algorithms for making recommendation in an EC environment. The mechanism is based on the correlations among customers, product items, and product features. Analysis of such an approach is given.

1. INTRODUCTION

The Internet has gained great popularity for information exchange. No spatial and temporal constraints are the advantages of the Internet. However, information overload is a critical problem, especially for applications in the electronic commerce arena. Moreover, product brokerage is essential in electronic commerce. Recommendation systems have been introduced by Resnick and Varian in [2]. The idea is that users cannot make a choice without sufficient personal experience or other people’s suggestion. Thus, automated recommendation by the computer system would be very helpful in filtering product information. We focus on the recommendation part of the electronic commerce application in this research.

Balabanovic and Shoham proposed a recommendation system — Fab in 1997 [1]. They combined content-based filtering and collaborative filtering to design their system. Content-based recommendation recommends product items to users who have purchase records or showed interest from browsing on the web. On the other hand, collaborative recommendation classifies users into several groups. Users in the same group are of similar behavior or interests on product items. And a recommendation system [7] which recommend netnews to users has been proposed in 1994.

Agent technology is a popular technology in distributed and mobile computing. Petrie declared a demand for web-based intelligent agents on the Internet [3]. Thomas and Fischer proposed a customized agent system [4]. Their system can enhance the Internet link based on users' bookmarks. And they also designed a mechanism to update content while any information or bookmark changes. Dharap and Freeman proposed a web-based information system using KQML as the agent communication language [5]. The system can reduce network loading and navigation complexity. We believe that mobile agent technology is good for personalization of the recommendation system.

In this paper we propose a system that makes product recommendation available to the user. Our proposed system can find the potential customers and users of interest on a particular product. The goals for the proposed system can be described as follows:

- Make maximum chance for merchants to sell products on the World Wide Web.
- Make suitable recommendations to potential customers or users who have little product information.
- Find the relations between products and users.

The system focuses on users' navigation behavior on the electronic commerce portal. Also, the system is based on the mobile technology. Users can browse the portal anywhere via any computers. The user preferences can follow the user on the client side to different web sites.

The remainder of this paper is organized as the following. An overview of the system is given in Section 2. The agent-based recommendation is presented in Section 3. The design of the practical algorithm is delineated in Section 4. The implementation is explained in Section 5. Finally, a brief conclusion is drawn in Section 6.

2. OVERVIEW OF THE SYSTEM ARCHITECTURE

Our proposed system includes two basic components. The first one is the mobile recommendation agent in the client’s machine. And, the second one is the mobile recommendation and the agent platform on the server side.

In the mobile recommendation agent, the basic information and agent tasks are as follows:

User real-time action: The action information includes links the user clicks, types of the links, and products the user orders.
User profile: The agent holds the user preference when the user logins to the system.
Gateway: This component is the bridge between the system platform and the client machine.
Monitor: This component is the administrator that captures the user action.

Figure 1 shows the framework of the mobile recommendation and the agent platform. The platform includes several components in the framework.

Interface agent: The agent is the communication channel between the platform and the mobile recommendation agent in the client machine.
Extraction agent: The task of this agent is to transform the user real-time action and navigation paths to suitable meta-data for the mobile recommendation agent.
Predict and analysis agent: The agent analyzes the user behavior and predicts the best recommendation products for the user.
MA: It is the merchant agent that holds the merchant strategy to assist generating the recommendation.

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE

Recommendation generation agent: After processing all the information, the result is sent to the agent to generate the recommendation.

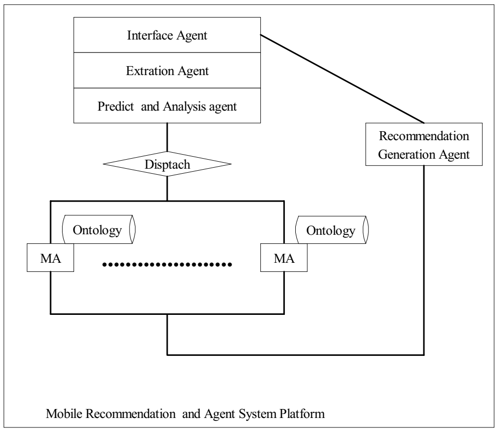

Figure 1. The mobile recommendation and agent platform

3. THE AGENT-BASED RECOMMENDATION SYSTEM ARCHITECTURE

In this section, we propose an agent computing architecture for the recommendation. Our proposed architecture is a multi-agent architecture within which the agents work collaboratively. In Figure 2 we deploy a mobile agent — the CARAgent, in the client's machine, which records the user behavior. The agent is also the mediator for the agents in the server. The agent accepts recommendation from the recommendation agent on the server side. For this reason the CARAgent is a mobile agent that can assist the user and give personalized recommendation. On the other hand, the MARAgent is deployed in the merchant machine. The merchant can place their products on the electronic commerce site with the assistance of the MARAgent.

Upon the entry to the server, an agent group including the user identification agent, the behavior extraction agent and the control agent, is formed to be the manager and the communication interface of the mobile agents. The user identification agent handles the authentication task and plays as the security manager in the system. The behavior extraction agent decodes the navigation information into suitable categories and then turns to the next processing agents. The control agent controls the agent life cycle and monitors the agent status (alive or dead).

When the agent group finishes the extraction task, the information is sent to the user model agent. The user model agent builds the user profile according to the information from both the agent group and the individual user. And then the user profile is given to the information processing agent and is stored into information center by the agent. The information agent is the coordinator of database access. The agent handles all the processes in the information center which includes an identification database, a site topology database, a user profile database and a recommendation pool. The behavior processing agent analyzes the user behavior using the CPF model proposed in the next section, and sends the recommendation target to the recommendation agent. The final work is then finished by the recommendation agent. The recommendation agent accepts the recommendation target and finds the corresponding product contents for the users.

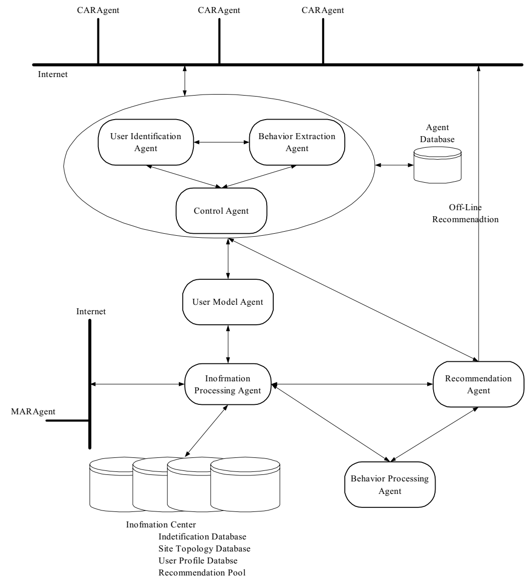

Figure 2. A Collaborative Agent Architecture

4. THE CPF MODEL AND ALGORITHMS FOR RECOMMENDATION

We propose a graphic model to analyze the user behavior. The model is referred to as the CPF analysis model. The model has three dimensions including the product dimension, the customer dimension and the feature dimension. Figure 3 shows the model with respective information.

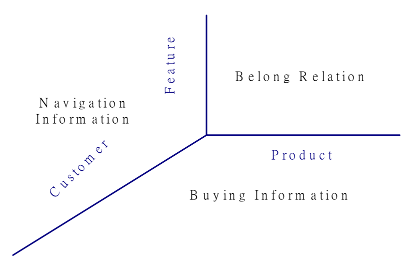

Figure 3. The CPF analysis model

The model has three views that are explained in the following.

Product-Customer view: This view stores the relation between the product and the customer. The value in the view shows whether customer *i* buys product *j* or not. The value is of the set {0,1}. If customer *i* buys product *j* on the E-commerce site, the corresponding value is "1". Otherwise, the value is "0".

Product Identification

Customer Identification

	P_1	P_2					P_m
C_1	C_1P_1	C_2P_2					
C_2	C_2P_1						
				C_iP_i			
C_n	C_nP_1						C_nP_m

Table 1. Customer-Product View

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE
0-7695-1198-8/01/$10.00 (C) 2001 IEEE

Table 1 shows the buying relation arrangement between customers and products. In this view we can discover potential customers, which bought most product in the past. And we can find the potential products, which are bought by most users.

Product-Feature view: This view shows the relation between the product and the feature. The value in the view represents whether product *j* has feature *k* or not. The value is defined in the set {Y, N}. The symbol "Y" represents a "have" relation. On the contrary, the symbol "N" denotes a "not-have" relation. Table 2 shows the arrangement of the belong relation. The statistical information, which records the product all customers navigated, is summarized.

Feature Indetification

Product Identification

	F_1	F_2					F_m
P_1	P_1F_1	P_2F_2					
P_2	P_2F_1						
				P_iF_j			
P_n	P_nF_1						P_nF_m

Table 2. Product-Feature view

Customer-Feature view: This view shows the relation between the customer and the feature. When the customer browses the site, his/her interest on certain features of the product would reveal. The value is in the real number domain. And in this view each row is a navigation vector corresponding to a customer. Customers are clustered into different groups according to the navigation vectors. The element *k* in the navigation vector *i* is calculated by the following formula.

$$\text{Navigation Element}_k = \frac{\sum f_k}{K} \times Avg(f_k)$$

$$Avg(f_k) = \frac{time\_of(f_k)}{total\_time(p_j)}$$

where K is the number of features of product j on which customer i navigates, $time\_of(f_k)$ is the total time that the customer navigates feature k of the product, and $total\_time(p_j)$ is the total time that the customer navigates on all the features of product j.

Table 3 shows the arrangement of the customer-feature view. Each value in the matrix is calculated by the above formula. Each customer's row is corresponding to the features as a navigation vector that used for computing similarity with others.

Feature Identification

Customer Identification

	F_1	F_2					F_m
C_1	C_1F_1	C_2F_2					
C_2	C_2F_1						
				C_iF_j			
C_n	C_nF_1						C_nF_m

Table 3. Customer-Feature view

After gathering the behavior information, we propose two algorithms for classifying customers and for finding the buying patterns.

Algorithm 1: Generate User Groups for Products
Input: *CF Matrix, k as the cluster number*
Output: *User Groups*

```
{
    for i = 1 to k
        Gi <- Ci
    do
    {
        for i = 1 to n
        {
            most <- ∞
            for j =1 to k
            {
```

$$temp = \sqrt{\sum_{z=1}^{m} \left| C_i F_z - G_j F_z \right|^2}$$

```
                if temp < most
                    index = j
            }
            Gindex <- Ci
        }
        for i = 1 to k
        {
            temp <- 0
            for j = 1 to Gi.length
                temp <- temp + Gi, j
            Gi <- temp / Gi.length
        }
    } while set G changes
}
```

Algorithm 2: Discover Buying Pattern
Input: *CP Matrix, minimum support count*
Output: *Buying Patterns*

```
{
    for each Pj
    {
        count <- 0
        for each Ci
            if CiPj = 1
                count <- count + 1
        S <- j
        Is <- count
    }
    do
    {
        for i =1 to I.length
            for j = i to I.length
            {
                count <- 0
                for each Ck
                {
                    S <- i ∪ j
                    if CkPi = 1
                        count = count + 1
                    if CkPj = 1
                        count = count + 1
                }
                if count >= MSC
                    Is <- count
            }
    } while I.length = 0
    output I
}
```


2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE

0-7695-1198-8/01/$10.00 (C) 2001 IEEE

With Algorithm 1, which is based on the k-means classification method customers can be classified into different classification groups with respect to a certain product. Combining with merchant strategies, the resulting information then can be used to make personal recommendation. Also from the classification result, some problems can be identified to recommend merchants to improve their strategies. For example, suppose that the buying rate is greater than a threshold in a group, but someone in that group does not buy the product. Then we can recommend merchants to push product information or offer discount to that particular user. *A priori* algorithm is a method for finding frequent itemsets [6]. We enhance the algorithm for the discovery of buying patterns. The general sequences of buying behavior can be found in Algorithm 2. With the information about buying patterns, we can decide what to recommend to the users who have not bought.

In Figure 4 we conclude our recommendation process in several stages. The first stage is the CBNR user profile, which recorded user navigation behavior and buying information will be generate from the agent in the client's machine. The result is combined with CPF analysis model through information extraction procedure. And then the user classification algorithm (Algorithm 1) and the buying pattern discovery algorithm (Algorithm 2) are used to find products to be recommended. At the last stage, the agent generates personalized recommendation that is suitable for each customer. Each customer can get product information of his/her interest.

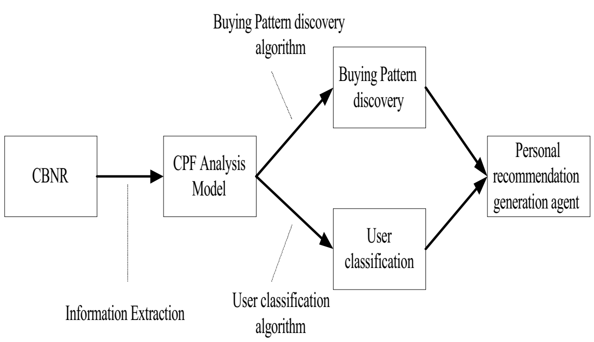

Figure 4. Recommendation process for existing users

5. IMPLEMENTATION

Our system is based on the Microsoft Windows platform. The implementation of the system follows the popular software development trend – distributed computing. Hence we develop a mobile agent server platform that can host agents from the client's machine. We use the agent control proposed by Microsoft as the agent program interface. When the users navigate the E-commerce site after login to the system, the agent is launched from the client. Wherever the user goes the agent would follow. In the mobile agent server, we program a platform based on the MS-Windows environment. The agent communication mechanism with the server is built based on the Windows SDK. The platform also provides agent naming service, so that each user can be identified.

Three servers are included in the basic runtime environment. That includes the web server, the agent server and the database server. The IIS is used as the web server that hosts all web documents and materials of the E-commerce site. And the agent server is programmed based on the Windows DLL library. All data are stored in the database server that is the MS-SQL server in our implementation. Figure 5 shows our implementation architecture.

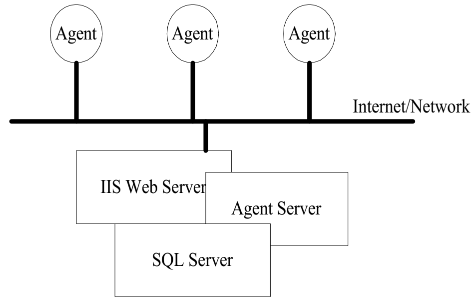

Figure 5. Implementation Architecture

6. CONCLUSION

In this paper we propose a recommendation system based on mobile agent technology. We not only propose a framework for recommendation system, but also propose practical algorithms for identifying user classification and buying patterns. The personal agent with user preferences is added to the system. It can follow the user. That is, the agent has the mobile capability. We believe that personalization is an important issue for the recommendation system in electronic commerce. In the future we can improve the platform in two manners. First, it should be able to host a chain of retail stores within which users can shop around and communicate with each other. Secondly, the E-commerce portal site can be integrated to form a virtual community. We hope this system would become a useful application for electronic commerce and gained the advantages of the mobile agent technology in the E-business environment.

7. REFERENCES

[1] Marko Balabanovic and Yoav Shoham. "Fab: Content-based collaborative recommendation." *Communication of the ACM*, 40 (3), pp. 64-72, 1997.

[2] Paul Resnick, and Hal R. Varian. "Recommendation systems." *Communication of the ACM*, 40 (3), pp. 56-58, 1997.

[3] Charles J. Petrie, "Agent-Based Engineering the Web and Intelligence." *IEEE Expert*, Vol. 11, No. 6, pp. 24-29, 1996.

[4] Christoph G. Thomas, and Gerhard Fischer, "Using Agents to Personalize the Web." *In Proceedings of the ACM IUI Conference*, Orlando, U. S. A., pp. 53-60, 1997.

[5] Chanda Dharap and Martin Freeman, "Information agents for Automated Browsing.", *In Proceedings of the 1996 ACM CIKM Conference*, Rockville, MD, U.S.A., pp. 296-305, 1996.

[6] Agrawal, R. Imielinski, T. and Swami A."Mining association rules between sets of items in large database." *In Proceedings of the ACM SIGMOD Conference on Management of Data*, pp. 207-216, Washington D.C., 1993.

[7] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom and John Riedl. "GroupLens : An open architecture for collaborative filtering of netnews." *In Proceedings of the 1994 Computer Supported Cooperative Work Conference*, New York, 1994.

2001 IEEE International Conference on Multimedia and Expo
ISBN 0-7695-1198-8/01 $17.00 © 2001 IEEE
0-7695-1198-8/01/$10.00 (C) 2001 IEEE