

PERSONALIZATION FOR NETWORK MARKETING

Tsai-Yen Li
National Chengchi Univ.
Computer Science Dept.
Tel: 886-2-29387170
Fax: 886-2-2939-0414
li@nccu.edu.tw

Sheng-Luen Chung
National Taiwan Univ. of Science and Technology
Electrical Engineering Dept.
slchung@mail.ntust.edu.tw

Kuo-Kan Liao
National Taiwan Univ. of Science and Technology
Electrical Engineering Dept.
M8907235@mail.ntust.edu.tw

Key Words: personalized network marketing; user profile collection and analysis; data mining; personalized human-machine interface

Subject Index: CO1, C09, and CO00

ABSTRACT

Personalized network marketing refers to the technique that allows network-marketing decisions to be made based on the results of the collection and analysis of user profiles. In this paper, we propose an interactive questionnaire mechanism whose design objective is to produce timely and reliable user profile attributes in an incremental fashion through an intelligent interface accessible from web environment. Subsequently, a profile analyzer that utilizes data mining techniques is employed to select fitted users from the user profile database thus collected, and then feed the result back to the intelligent interface mechanism to improve the quality of the collected database profiles. In particular, questions used to reveal attributes can be prioritized based on an adjustable mechanism geared to the particular marketing objectives. To implement the proposed interactive questionnaire mechanism, we have made use of graphical questionnaire design tools, such as question editor, profile editor, and rule editor. For a vivid illustration of the personalized network marketing technique, we have applied our implementation to a web site to demonstrate how user profiles are collected, and how, in turn, marketing decisions are made accordingly.

摘要

個人化網路行銷技術係透過使用者個性屬性的收集與分析，提高網路行銷的決策品質。本論文提議以智慧型的 Web 瀏覽介面及漸進的互動問答機制，誘導並歸納出即時且可靠的使用者個性屬性；再以資料挖掘的技術設計個性屬性分析程式，從所收集的個性資料庫中，選出適合的使用者，並將此分析結果回饋至智慧型屬性收集介面，以提高所收集之屬性資料庫的品質。在此問答機制中，我們可以依據不同的行銷目標，調整個別問卷問題的優先度，以反應個別行銷專案的急迫性。我們已實做出圖形化的問卷設計工具，如問題編輯器、個性編輯器、及規則編輯器等，以輔助互動問答機制的資料建立。另外，為能展示此個人化網路行銷技術，我們已將此技術在一個網站上實做出來，以說明此技術如何透過個人屬性資料收集，提升網路行銷的品質。

I. INTRODUCTION

1. Personalized Network Marketing

With the rapid development of the Internet, network marketing has become an indispensable marketing means. The theoretical basis for network marketing includes relation marketing and one-to-one marketing (Allen et al, 2001). The network provides two special advantages over traditional marketing means: *interactivity* and *ease of data collection*. Interactivity facilitates building relations among different parties and forming a community for relation marketing. Meanwhile, the ease of data collection helps understanding of consumer purchasing behavior, thus making possible the customized products and/or services of one-to-one marketing. Taking advantage of these two characteristics will be the key to the success of network marketing.

In (Hanson, 2000), the author emphasizes the fact that the products or services for customers need not be the best, but, they should fit customers' particular needs. In effect, this concept highlights the importance of relating marketing service to personalized needs of customers. In essence, personalization is to provide a customer with what is personally required by his/her needs. Similar ideas have been adopted and realized by Broadvision¹ in their personalized e-Self-Service for e-business. This service includes a learning mechanism that records the interactions of the customers in order to understand what the customers want. This information is then fed back to the system in order to provide better service the next time the customers log in.

In general, there are two ways to know what the customers want. One is by inferring from historical data based on his/her past purchasing behavior, while the other is by predicting their needs from other customers with similar habits (Balabanovic and Shoham, 1997; Yu, 1999). In either case, one has to acquire the customer behavior data via the network. In this regard, the most common method to analyze a consumer's need would be observing his/her browsing behavior on a web site (Tao and Muragh, 2000; Perkowitz and Etzioni, 2000), as every page transaction on a web site can be recorded in a log file.

In contrast, this paper takes a different approach, in that we analyze customers' preferences by collecting user profiles directly instead of inferring from historical logs. We believe that predicting user preferences by collecting personal profiles directly would be the most effective and precise way for decision making in network marketing. As a result, we propose a different approach to data collection.

2. Collecting Personal Profiles

Collecting a personal profile is the key step toward one-to-one marketing. Basic attributes in a personal profile for a person include age, sex, nationality, habits, purchasing power, etc. In a broader sense, the profile also includes a person's answers to various consumption questions that can reflect one's purchasing behavior. However, how much data can be used to describe a person and how much of the intended data can be collected explicitly? In order to acquire valid personal profiles, we have designed an interactive questionnaire interface, customized questions, and an intelligent problem filtering mechanism to improve their credibility and effectiveness.

On top of the personal profiles collected, an equally important issue concerning one-to-one marketing is the integration of profile data for long-term user data collection. Although each marketing survey geared to a particular marketing objective may have its particular set of attributes, many of these surveys share common attributes. In order to take advantage of the shared attributes, an integration process that avoids repetition in subsequent collecting is highly desirable. For instance, the Passport system² by Microsoft aims to provide, for each sign-on, an integrated view of the customers' data. In line with the design philosophy, we have proposed an interactive questionnaire mechanism that incorporates the idea of excluding redundant data collection. Meanwhile this design approach also integrates all personal attributes in a single profile database to suit different marketing objectives. With an analysis profile engine installed to suit a particular marketing objective, this integrated database can then be analyzed to produce more precise data for network-marketing decision making.

¹ BroadVision, <http://www.broadvision.com/>

² Microsoft Passport, <http://www.passport.com/>

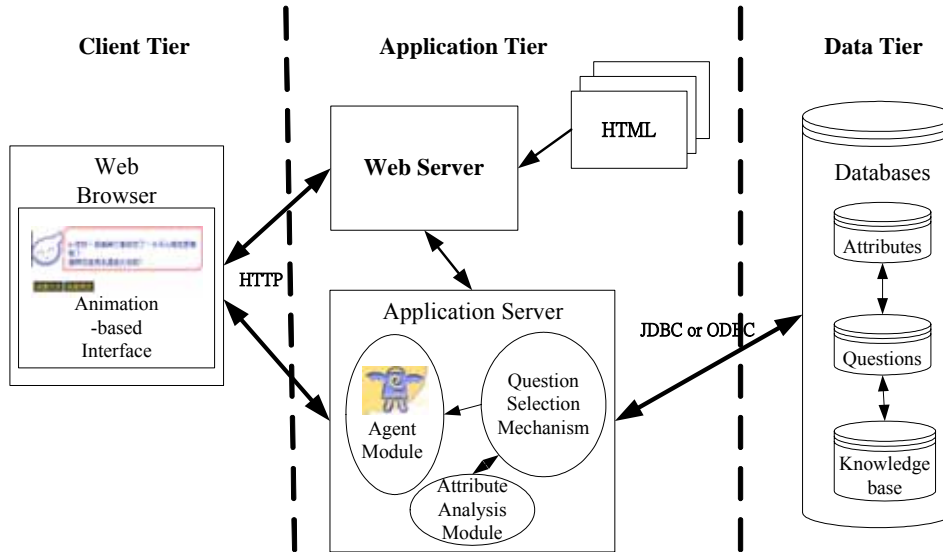


Fig. 1 System architecture of the interactive questionnaire mechanism

3. Contributions

There are four main techniques reported in this research. First, we have implemented an intelligent questionnaire mechanism with incremental and flexible questions to acquire reliable and timely personal profiles. Second, we have used data mining to analyze personal profile databases and feed back the results to the questionnaire mechanism in order to collect high quality profiles. Third, we have also designed and implemented questionnaire design tools with graphical user interfaces. These tools include a question editor, a profile editor, and a rule editor. Finally, we have also designed a profile analysis tool to assist the user to specify customers' network-marketing needs.

4. Paper Organization

In Section II, we will propose system architecture for the interactive questionnaire mechanism. We will present the core components of the system, question selection mechanisms, and attribute analysis methods. The focus will be on how to reflect the results of attribute analysis in the question selection mechanism in order to collect high quality profiles. In Section III, we will describe the platform of the interactive questionnaire mechanism. We will illustrate the design of various core components and authoring tools on a Java-based application server. In the fourth section, we will illustrate how the system works with a realistic example. Finally, we will conclude the paper with some future research directions.

II. INTERACTIVE QUESTIONNAIRE MECHANISM

1. System Architecture

Our interactive questionnaire mechanism uses a typical 3-tier architecture as shown in Fig. 1. The *data* tier is for information storage and management, while the *application* tier is for programming business logic. The *client* or *presentation* tier, on the other hand, provides user-friendly graphical user interfaces (GUI).

The database system in the data tier includes three databases: *the profile database*, *the question database*, and *the knowledge database*. The profile database stores personal attributes, while the question database stores all questions used in the questionnaire mechanism. The knowledge database stores information about the relations between questions or between questions and attributes.

The application layer includes a web server and an application server. The web server provides web pages while the application server provides business logic. The business logic in our application includes question selection mechanism, agent module, and profile analysis module. The question selection module selects and composes questions according to the rules that will be explained in later

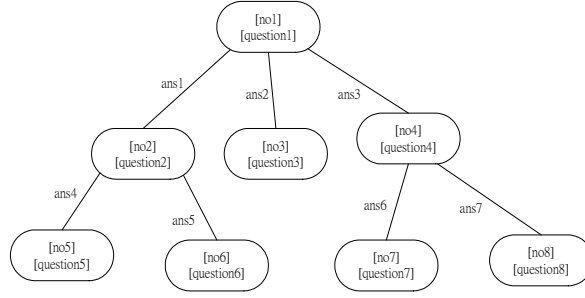


Fig. 2 The decision tree representation of a questionnaire

subsections. The agent module is in charge of composing questions for transmission and receiving answers to store them in the profile database. The profile analysis module uses data mining techniques to analyze the relations between attributes in the current profile database. The result is used in the question selection module in order to select the most relevant questions for a customized questionnaire.

The client tier uses a web-based graphical user interface to interact with the user. The questions are asked interactively with the help of attractive animations. The answers are then collected and sent back to the agent module for further processing.

2. Q/A Interface and Agent Module

In our interactive questionnaire mechanism, we use a decision-tree based method to present the questions. In order to increase the quality of the collected data, we only ask a few questions at a time in an incremental fashion. In other words, only one question is asked at a time and the next question will be asked based on the answer of the current question. The set of questions that will be asked at one time is collected in a questionnaire. For example, if a user answers “boy” upon the question of “Are you a boy or a girl?”, the next question might be “Have you finished your military service?”. In contrast, if the answer is “girl”, other questions related to women only will be asked.

On the server side, the agent module in the application tier needs to be able to compose questions organized by these rules. These questions are represented by a decision-tree structure, such as the example in Fig. 2. Every question in the questionnaire contains three parts: [no] represents the sequence number of the question, [question] represents the description of the problem itself, and [(ans1, no1), (ans2, no2), ...] includes a set of possible answers and their corresponding sequence numbers for the next question. The agent module sends the structured questionnaire to the client, and the client returns the collected answers back to the agent module. The agent module then maps the answers to their corresponding attributes in the profile database.

3. Question Selection Mechanism

In our system, the questions for a user are selected according to the priorities of the attributes in the user profile. The priority of an attribute for a user is composed of two parts: *base priority*, which is the same for a specific attribute for all users, and *individual priority* that is different for different users. These priorities are further defined as follows.

- (1). P_i^b : the base priority for an attribute A_i . For example, basic attributes such as name and gender may have higher base priorities than product-related attributes such as the brand of the cellular phone that a member is using.
- (2). ΔP_{ij}^u : the individual priority adjustment for the j th property of an attribute A_i of a user u . The properties of an attribute are defined by the system designers as well as the customers requesting the marketing survey. For example, an attribute A_i may have a negative adjustment ΔP_{ij}^u because the questions for this attribute were asked but not answered by the member. As another example, upon a market survey customer’s request and with our attribute analysis results, some attributes A_i (for example, brand name of cellular phone) are given positive priority adjustment ΔP_{ij}^u for certain target members. The “unanswered” status and a customer’s request are all ex-

amples of properties of an attribute.

- (3). The overall priority of an attribute A_i for a user u is defined as $P_i^u = p_i^b + \sum_j \Delta P_{ij}^u$. All priority

adjustments, possibly for different marketing purposes, are added to the base priority to form the overall priority.

The question selection mechanism first screens out the attributes that have not been filled in for a member u . Among these candidate attributes, the mechanism will select the one with the highest priority ($A_m | P_m^u = \max(P_i^u)$) and the corresponding questionnaire containing a question for the attribute.

We have tried to make the questionnaire short and interesting since a user is more likely to answer interactive short questions. However, it is still possible that a user refuses to release certain personal information. In order to avoid asking the same question repeatedly, the question selection mechanism will add an adjustment filter that contains a negative priority adjustment $\Delta P_{ij}^u < 0$ for an unanswered attribute A_i in order to reduce its priority. Consequently, the question for this attribute will not be selected again soon before other attributes with higher priorities are asked. Other adjustment filters resulting from attribute analysis will be explained in the next subsection.

4. Profile Analysis

Profile analysis aims to analyze the relations between attributes in order to predict the possible value of other unfilled attributes. In some data mining researches, association rules are used to analyze profile attributes (Srikant and Agrawal, 1996; Wijnsen and Meersmen, 1998; Han and Kamber, 2000). In this paper, we assume that the inputs of the profile analysis are categorical attributes only. For quantitative attributes, we discretize the value of an attribute into several ranges for discrete processing. However, the quantization method is out of the scope of this paper.

In profile analysis, we are concerned with which attribute values will lead to the target attribute value. For example, we would like to know who is willing to purchase product A. The analysis will first define “having purchased product A = yes” as the *target attribute value* and try to find the related attribute values for members who have already purchased product A. An attribute is called a *related attribute* for a target attribute value if most of the people having the target attribute value also have the same value for the related attribute. In other words, we treat the related attribute as the cause and the target attribute value as the effect so that the related attribute will result in the target attribute value. If a person has the related attribute value (cause) but has not purchase product A (the effect), then he/she will be our marketing subject.

We now define the concentration factor of an attribute for determining its relation with other attributes. For a given set of members, we would like to examine whether the value of each attribute concentrates on a specific one. If so, this common feature could be the cause of a target attribute value. We assume that all of our attributes are discrete so that the number of answers for an attribute is finite. For example, attribute Y has two possible answers (y_1, y_2), and attribute Z has three possible answers (z_1, z_2, z_3). Assume that in the current database we have 100 records that contain 75 records of $Y=y_1$ and 25 records of $Y=y_2$. Similarly, there are 50 records of $Z=z_1$, 30 records of $Z=z_2$, and 20 records of $Z=z_3$. It may look correct that $Y=y_1$ is more concentrated than $Z=z_1$ since it has more records. However, it is not fair for such a comparison since attribute Z has three options while attribute Y has only two. Therefore, a normalization processing on the results is necessary for making meaningful comparisons.

Assume that there are x members satisfying the target attribute value. We further assume that there are n_i options ($B_i = \{b_{i1}, \dots, b_{in_i}\}$) for attribute A_i and y_k members have selected attribute value $A_i = b_{ik}$, denoted by (A_i, b_{ik}) . We then define the *concentration factor*, C_{ik} , of an attribute value (A_i, b_{ik}) as follows.

$$C_{ik} = \frac{y_{ik} - (\frac{x}{n_i})}{x - (\frac{x}{n_i})} \times 100\% \quad (1)$$

The numerator of eq.(1) computes the difference between the number of members choosing a specific attribute value (A_i, b_{ik}) and the average number of members for each attribute value. The denominator is the possible largest value of numerator. In the extreme case, when all members have the

same value k for attribute i , the concentration factor for this attribute value will become 1. On the other hand, if only an average number of members (x/n_i) have the attribute value k , the concentration factor becomes 0. This formula is considered normalized because C_{ik} is always 1 and 0 for the above two extreme cases no matter the number of options an attribute could have. In some sense, the concentration factor is similar to the confidence measure defined for the associate rules in most data mining research (Han and Kamber, 2000) except for the normalization.

As an example, assume that the target attribute value is “cellular phone brand = NOKIA”, and there exist 386 members having this target attribute ($x=386$). In addition, assume that “age” is the attribute i under evaluation. We divide the common age range (0-100) into 10 intervals so that there are 10 options for this attribute ($n_i=10$). Suppose that the number of members whose age is in the range of 21 to 30 is 239 ($y_{ik}=239$). The concentration factor for this attribute value (age ranging from 21 to 30) is computed as

$$\frac{239 - \left(\frac{386}{10}\right)}{386 - \left(\frac{386}{10}\right)} \times 100\% = 57.69\%.$$

We define an attribute value $r_{ik} = (A_i, b_{ik})$ to be a “related attribute value” if its concentration factor C_{ik} is larger than some user-specified threshold C_{min} . The set of related attribute values is defined as $R = \{ r_{ik} \mid C_{ik} > C_{min} \}$. Although the attribute values in R are all related attribute values, they differ to some degree. In order to reflect the difference, we give each attribute value a weight defined as a monotonically increasing function of its concentration factor, $w_{ik} = h(C_{ik})$. For example, h is a linear function in our current implementation. In our interactive questionnaire mechanism, we will use w_{ik} to calculate how a member could satisfy the related attribute values R . Assume that the set of all possible attribute values for a member u is denoted as $A^u = \{r_{ik}\}$. The *satisfaction factor* S_R^u for a user u is defined as follows.

$$S_R^u = \frac{\sum_{i,k} (w_{ik} \mid r_{ik} \in A^u)}{\sum_i \max(w_{ik})} \quad (2)$$

For example, assume R consists of the following attribute values: (technology adorer, yes), $w=58$; (age ,21-30), $w=41$; (age, 31-40), $w=34$; (prone to be affected by advertisement, yes), $w=31$. Suppose the attributes of a member d0037 have the following values: (technology adorer, yes), (age, 21-30), (prone to be affected by advertisement, no). Then the satisfaction factor for this member can be computed as follows. $S_R^{d0037} = \frac{58 + 41}{58 + \max(41,34) + 31} \times 100\% = 76.15\%$.

The denominator of eq.(2) computes the sum of the weights with the ideal related attribute values. S_R^u basically defines how a user satisfies each of the related attribute values. If a member contains all the ideal related attribute values, his/her satisfaction factor will be 1; otherwise, S_R^u will be less than 1. We then can use this formula to find the members whose satisfaction factor S_R^u is greater than some minimal threshold S_{min} . For each qualified member u , we will add an extra positive priority adjustment, ΔP_{ij}^u , $\Delta P_{ij}^u = f(S_R^u) > 0$, as a filter for the target attribute A_i . This filter will be added into the overall priority as a priority adjustment for the target attribute in question selection as mentioned in the previous section. The filter can also be easily removed if the need for filling a target attribute disappears. Consequently, with the profile analysis and the question selection mechanism described above, we can select the members that are more likely to answer the target attribute question with the right value. The questions for this attribute will have a good chance to be asked when these members log in.

5. Overview on User Profile Updating

In this proposed framework, user profiles as represented by the collected questions answered by individual users are updated dynamically and individually to meet special marketing needs. Selection of target users that best fit the intended marketing interests is done by *profile analysis*, as delineated in Section II.4. At their next login session with our interactive questionnaire mechanism, the target users are then prompted with prioritized questions by the question selection mechanism, as described in Section II.3. Collectively, this subsection gives an overview regarding how the user profile updating process is executed.

file updating process is executed.

When the system is first started, all the users are prompted with questions without fixed order by the interactive questionnaire mechanism. Answers are collected. After sufficient information about the users is collected, there may be some marketing customer interested in the collected information of the whole user body, thus triggering *profile analysis* listed below:

Procedure *profile analysis* (snapshot of user profiles answered thus far by of all users; C_{min})

For all attribute index i

For all attribute values attribute value (A_i, b_{ik})

Computer *concentration factor*, C_{ik} , as follows.

$$C_{ik} = \frac{y_{ik} - (\frac{x}{n_i})}{x - (\frac{x}{n_i})} \times 100\% \quad (1)$$

Compute the set of related attribute values: $R = \{ r_{ik} \mid C_{ik} > C_{min} \}$

For all users u

Compute the associated *satisfaction factor* S_R^u as follows.

$$S_R^u = \frac{\sum_{i,k} (w_{ik} \mid r_{ik} \in A^u)}{\sum_i \max(w_{ik})} \quad (2)$$

For all target users u that satisfies the condition: $S_R^u > S_{min}$

For all attributes A_i

Update the associated priority adjustment ΔP_{ij}^u .

End Procedure

After the profile analysis is made, individual priority derived by the question selection algorithm will be invoked individually whenever a user u logs into the system. Individual priorities of different users differs from each other mainly on the accumulative priority adjustment that each individual user acquired when selected as a target in previous profile analysis. Unanswered questions in descending priority are then prompted in the interactive questionnaire mechanism. The form of the question selection is shown below.

Procedure *question selection* (user u)

For all unanswered questions i as an attribute A_i

Compute *individual priority*: $P_i^u = p_i^b + \sum_j \Delta P_{ij}^u$.

Sort unanswered questions in descending priority

End procedure

III. IMPLEMENTATION PLATFORM

1. Database Design for User Profile

The database in our system consists of three modules: *personal profile module*, *attribute and question module*, and *questionnaire and rule module*. The personal profile module is comprised of attribute values and personalized attribute priority adjustments. The attribute and question module stores the list of possible attributes and the questions for acquiring these attributes. An attribute could be asked in multiple questions. The questionnaire and rule module stores the sets of questions that will be asked as a unit. Different questionnaires could include identical questions presented in different orders or with different rules. For each question, we can set some rules as preconditions such that a question will be asked only if the precondition is true. By setting up these rules, we can ensure that conflicting or strange questions will not be asked by accident.

In addition, we have developed a customized database maintenance editor in order to ease the maintenance of the profile database. The editor allows a user to flexibly define the target attribute and its associated questions. Once the attributes and questions are defined, the interactive questionnaire mechanism will select the question at some point according to its priority. The maintenance editors

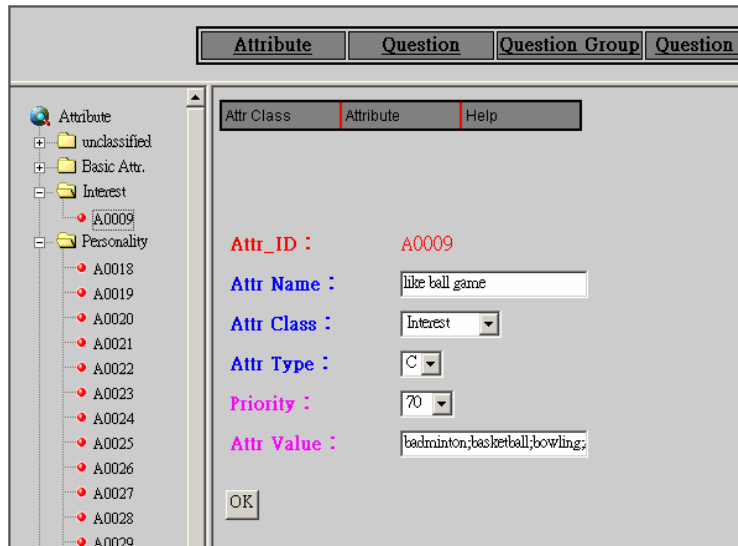


Fig. 3 The attribute editor

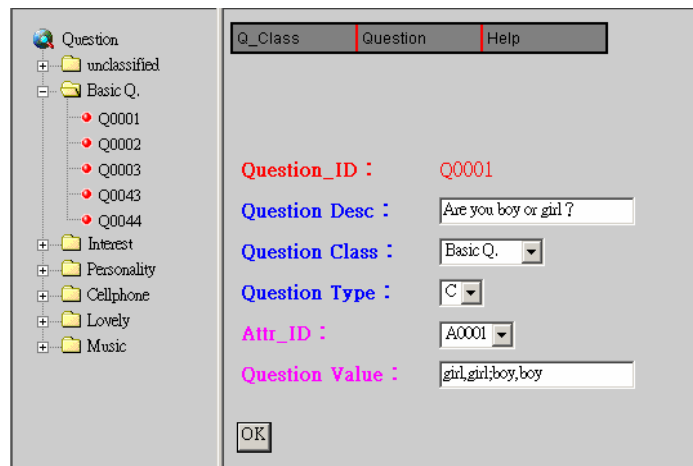


Fig. 4 The question editor

include an attribute editor, a question editor, a questionnaire editor and a question rule editor, as described below.

Attribute editor: as shown in Fig. 3, provides a user-friendly graphical interface for defining the attributes that will be collected. The editor can not only categorize the attributes for maintenance but also give each attribute individual priority adjustment. An attribute includes properties such as identification, name, category, type, and value.

Question editor: as shown in Fig. 4, provides an easy way for inputting questions and the mappings between answers and their corresponding attribute values. The definition of a question includes identification, description, type, corresponding attribute, possible answers and attribute values.

Questionnaire editor: as shown in Fig. 5, the editor provides an interface for selecting questions from the question database in order to compose a questionnaire with conditional rules between sequential questions. In other words, the question to be asked next could depend on the answer of the current question. The editor can allow a user to easily add questions to the questionnaire and specify their relations. The properties of a questionnaire include name, question id, possible answers, and their corresponding next questions.

Question rule editor: as shown in Fig. 6, the editor provides a graphical interface for editing question rules. These rules specify the conditions where a question cannot be asked. For example, if the gender of a member is “male”, then the questionnaire about “preference on cosmetics” should not be asked. The definition of a rule includes identification, questionnaire id, attribute id, Boolean condi-

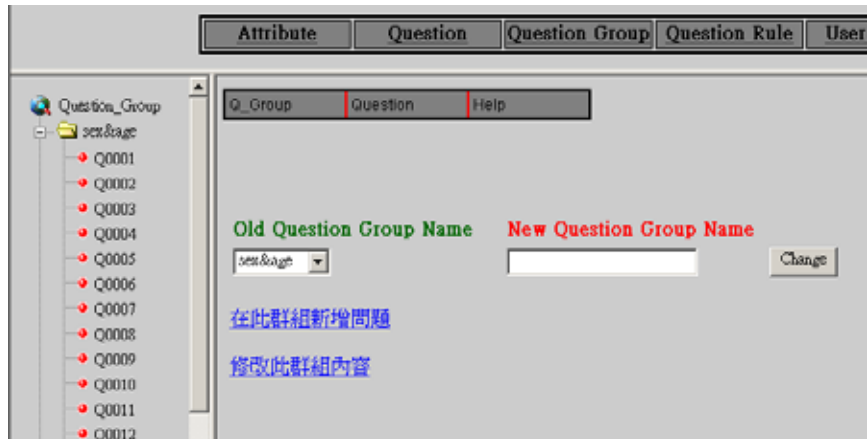


Fig. 5 The questionnaire editor

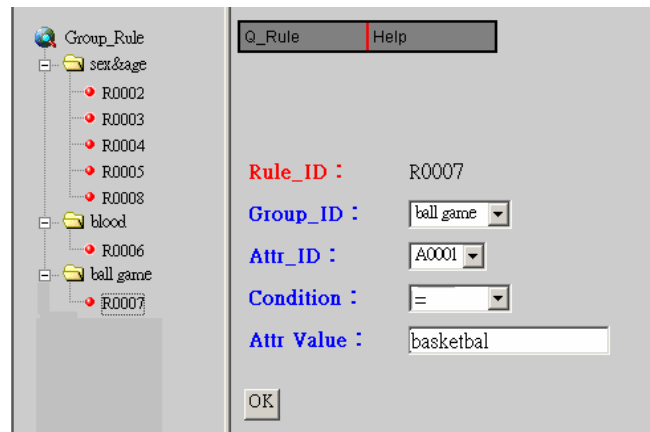


Fig. 6 The question rule editor

```
G0001&1@^Q0001@^Are you a boy or a girl ?
@^C@^girl**13#]boy**2&)2@^Q0002@^Really?...How old are you, guy ?
@^Q@^0:18**3#]19:24**6#]25:45**9#]46:100**12&)3@^Q0003@^Are you still in school ?
@^C@^No**5#]Yes Y **4&)4@^Q0004@^Really...Then, please study
hard!@^N@^null&)5@^Q0005@^Really...That was my mistake.@^N@^null&)6@^Q0006@^Are you serving in the army ? @^C@^No**8#]Yes Y
**7&)7@^Q0007@^Is it tough? Cheer up!@^N@^null&)8@^Q0008@^I
see..Thanks!@^N@^null&)9@^Q0009@^ Then where do you work?@^C@^Out of
work**11#]Yes Y**10&)10@^Q0010@^Please work hard@^N@^null&)11@^Q0011@^Then
you are doing nothing, right?@^N@^null&)12@^Q0012@^You are a bit
old.@^N@^null&)13@^Q0013@^Miss, How old are you?@^Q@^0:20**14#]21:30**15#]31:50**16&)14@^Q0014@^You
are so young.@^N@^null&)15@^Q0015@^Please enjoy the best moment in you
life.@^N@^null&)16@^Q0016@^Hey, this is the most charming age in your life.@^N@^null&)
```

Fig. 7 An example of encoded questionnaire

tion, and attribute value.

2. Java-Based Application Server

We have used the Java³ technologies in the application layer of our system. We use an Apache⁴

³ Java, <http://java.sun.com>

⁴ Apache, <http://apache.org>

```

While ( have any attribute in ATTR and not in the member`s PROFILE ) {
    total user_priority =  $\Sigma$  user_priority in this attribute;
    priority = base_peiority+total user_priority;
    if (priority > max_priority) {
        max_priority = priority;
        selected_attribute = this attribute;
    }
}
question=select question from QUESTION where attribute = selected_attribute;
q_group= select q_group from Q_GROUP where question = selected_question;

```

Fig. 8 The psuedocode of the questionnaire selection algorithm

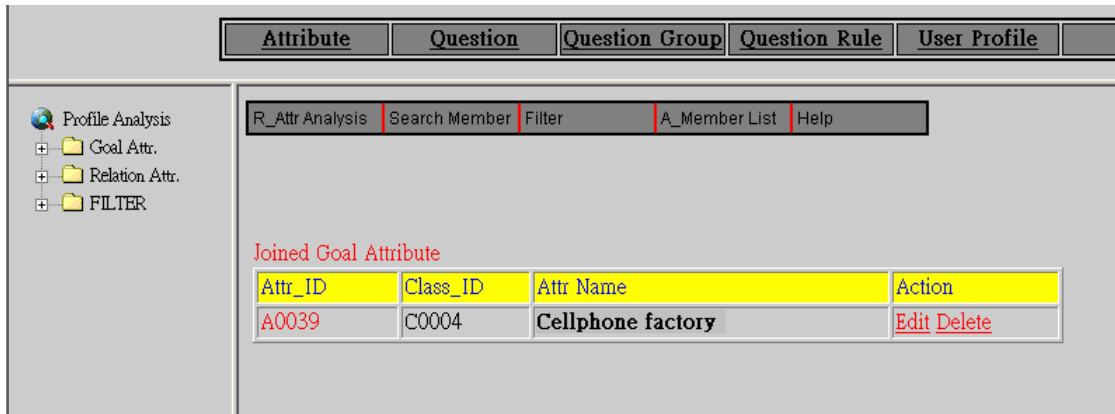


Fig. 9 The attribute analysis module

web server and a Java servlet⁵ engine to implement the business logic in the application layer. The application server consists of three program modules: agent module, questionnaire selection module, and profile analysis module.

The agent module is responsible for composing an interactive questionnaire selected by the questionnaire selection mechanism and transmitting the questionnaire to the client side. Each question item in an interactive questionnaire consists of three parts: question number[no], question description[question], possible answers and their corresponding next question id[(ans1, no1), (ans2, no2), ...]. Therefore, each question is represented by $Q\{[no],[question],[[ans1,no1),(ans2,no2),...]\}$ and a questionnaire consists of a set of encoded Q 's. An example output of the agent module is shown in Fig. 7.

The agent module sends the above questionnaire to the client side for interactive questioning via the HTTP protocol. In addition, the module handles the returned answers of the questionnaire by sending the values to appropriate attributes of the database in the data layer.

In addition, the questionnaire selection module selects the questionnaire with the highest attribute priority. The selection algorithm is shown in the pseudocode of Fig. 8. q_group is the questionnaire to be composed and sent to the client side for further processing.

The profile analysis module, as shown in Fig. 9, allows a customer to increase the priorities of certain questions. Candidate members will be prompted with the most effective questions, which could be different for different members.

3. Web-Based Graphical User Interface

At the client side, we have used a web-based graphical user interface to present the questionnaire. No other addition programs need to be installed in addition to having a standard web browser. We use the Java applet technologies with appropriate animations to interact with the users, as shown in Fig. 10. The applet is capable of parsing and presenting the encoded questionnaire containing a deci-

⁵ JServ, <http://java.apache.org>

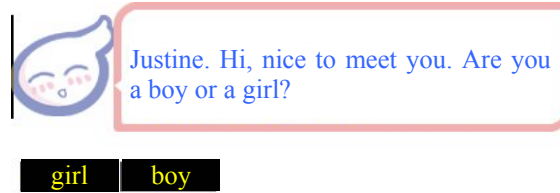


Fig. 10 Interactive questioning mechanism at the client side

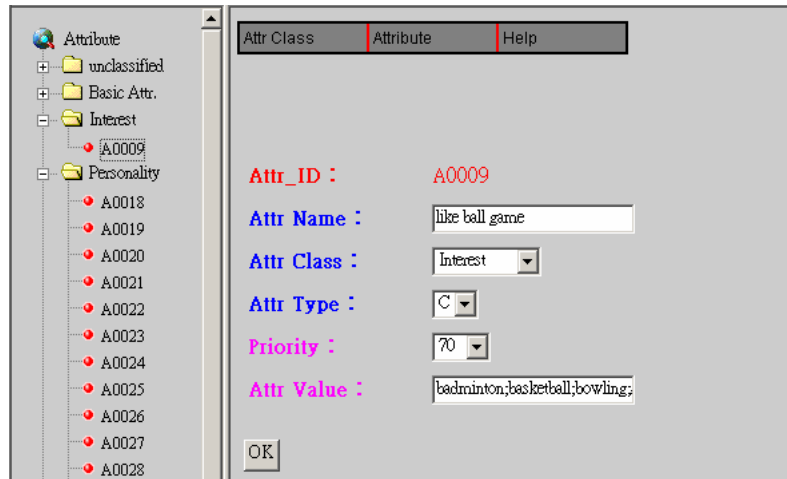


Fig. 11 Inputting attributes

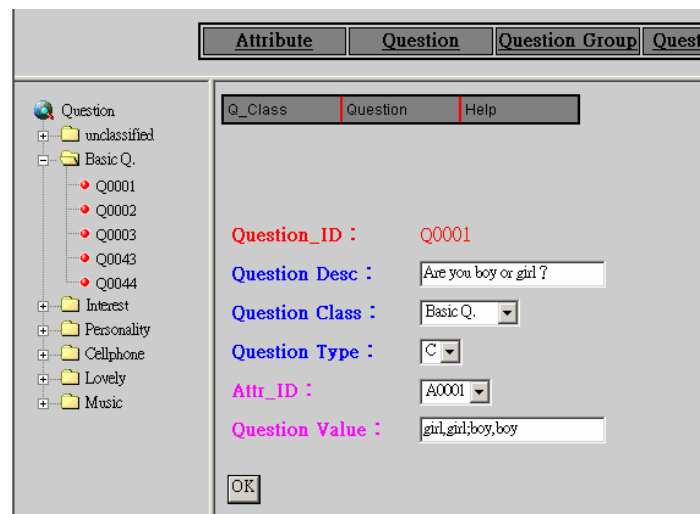


Fig. 12 Inputting a question

sion-tree type of mechanism. At the end of the questionnaire, the applet will return the collected answers to the agent module in the application layer.

IV. EXAMPLES

1. Questionnaire Design

By using the profile database maintenance editor, one can easily design a questionnaire that can be used immediately in the interactive interface. When designing such a questionnaire, we must first decide the attributes that we want to collect and input them with the attribute editor, as shown in Fig. 11. For example, we design an attribute with id as A0001, name as gender, type as basic, category as

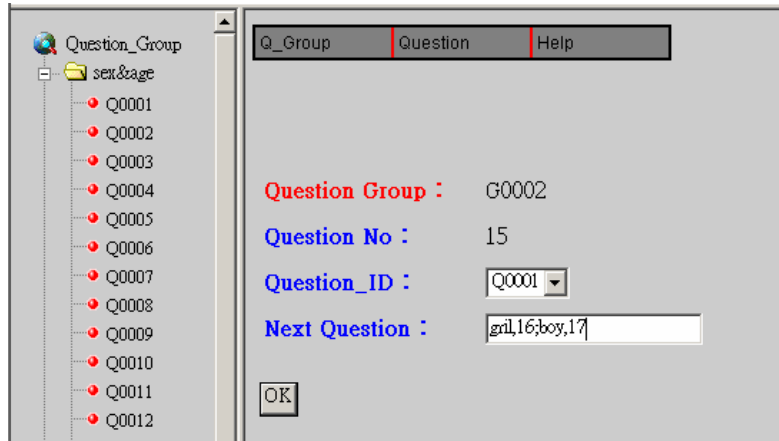


Fig. 13 Selecting questions to compose a questionnaire

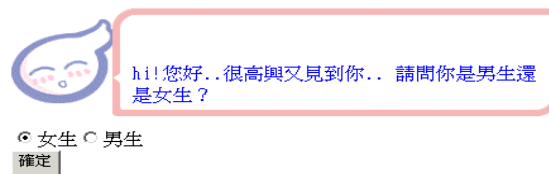


Fig. 14 Selecting questions to compose a questionnaire

Hi, nice to meet you. Are you a boy or a girl?

C (C: selective, Q: numerical, girl; boy; text input), priority as 90, and value as male or female.

Then we use the questionnaire editor to input questions as shown in Fig. 12. For example, the question id is “Q0001”. The question description is “Are you a boy or a girl?” The category for the question is “C”. The id of the attribute that this question tries to collect is “A0001”. The answers include “girl, female; boy, male” (answer, attribute value).

Once we have created all the questions, we can use the questionnaire editor to compose related questions into a structured questionnaire with rules, as shown in Fig. 13. For example, the questionnaire id is “G0001”. The question sequence number is “1”, and the question id is “Q0001.” The next question id is “girl, 13; boy, 2” (answer, next question sequence number).

2. Interactive Questioning

After the questioning interface is evoked by a user, the client applet will present a graphical interface with animations and questions such as the one shown in Fig. 14.

Suppose that the user selects “boy” for the question in Fig. 14. Then the following question will be chosen: “Great. Then how old are you?” Suppose that the user inputs “21”. Then the system continues to ask “Then, are you serving in the army?” Suppose that the user inputs “Yes” to the question. Finally, the system ends the conversation with the statement of “Is it tough? Cheer up!” On the other hand, if the user is a female and selects “girl”, then the following question will be asked: “Miss, how old are you?” Suppose that the user inputs “21” to the above question, then the system will make a final statement of “Enjoy the most valuable moment in your life.” to end the conversation. In short, the system will ask different questions for different answers from previous questions.

3. Attribute Value Analysis

The purpose of attribute value analysis is to analyze the relations between attributes in order to find the related attributes that will lead to the target attribute. Then we can find the members who are potentially more likely to fill in the target attribute value and increase the priorities of the target attributes for these members. The procedure consists of the following three steps. First, find the related attribute values for the target attribute value. Second, find the members who have the highest satisfaction factors. Third, increase the priority adjustment of the target attribute for these members.

In the first step, we assume that we try to help a customer E who is doing a marketing survey for

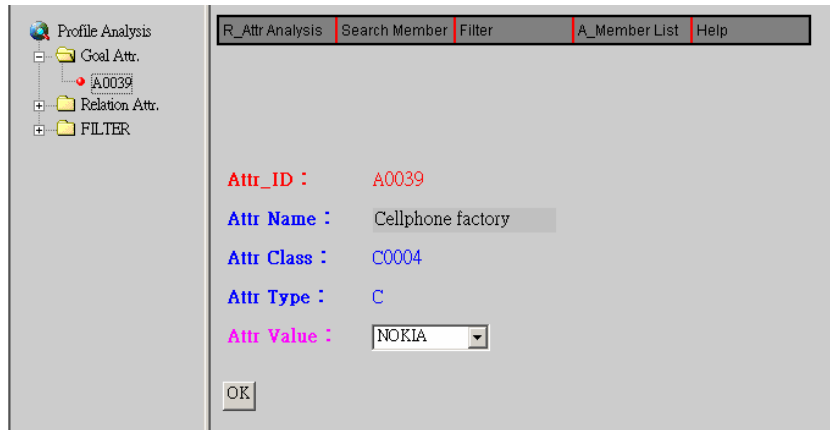


Fig. 15 Specifying a target attribute

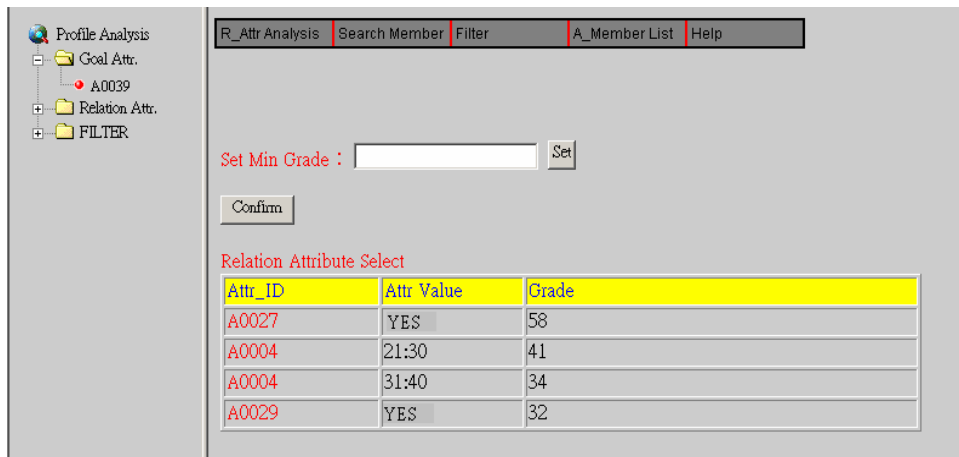


Fig. 16 Listing of related attribute values

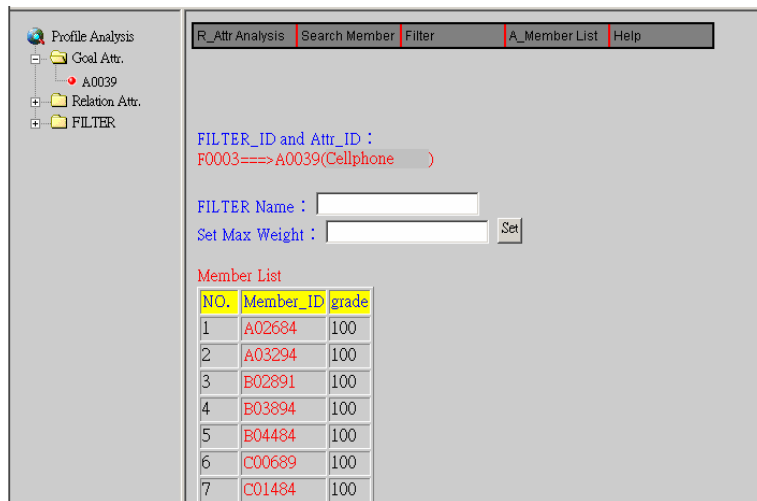


Fig. 17 List of members that are above the minimal satisfactory factor

its cellular phones. The target attribute (numbered A0039) is the brand of the cellular phone while the target attribute value is “ERICSSON,” as shown in Fig. 15.

Once we have set up the target attribute, we can use the system to find its related attributes, as shown in Fig. 16. Suppose that we set the minimal concentration factor to 30, then the following are a list of the related attribute values that the system finds: gender=male, weight=53; “easily affected by advertisement”=yes, weight=42; ages=31-40, weight=36; age=21-30, weight=33.

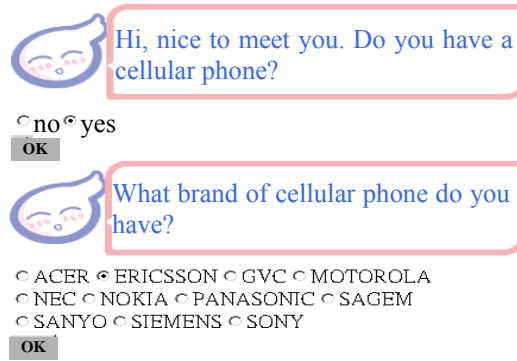


Fig. 18 Questionnaire for the target attribute

In the second step, we use the related attribute values to compute the satisfaction factor for each member. Assume that we have set the minimal satisfaction factor to 60, then we can obtain the list of members whose target attribute should be promoted, as shown in Fig. 17.

In the third step, we insert a filter for each of the priority adjustments for all potential members such that the questions for these target attributes can be asked earlier when the potential members log in, as shown in Fig. 18.

In addition, the system can collect different target attributes according to the priorities assigned to each one. If a member satisfies the conditions for more than one target attribute, then the system determines which of them will be asked first according to the overall computed priority. Therefore, the system allows several products to conduct their marketing surveys at the same time without conflicting with other products.

V. CONCLUSIONS AND FUTURE WORK

The objective of this paper is to propose a network-marketing platform capable of collecting personal attributes, designing customized questionnaires, and analyzing attribute relations. Through the development of an interactive questionnaire mechanism for a personalized network-marketing platform, the system can provide the decision maker with timely and effective marketing data.

The interactive questioning mechanism provides a convenient user profile collection tool for a network-marketing platform. The mechanism is effective because the attribute analysis helps locate members who are most likely to have the target attribute value. However, the profile analysis method in this paper is restricted to discrete attributes only. Extension to current results includes numerical attributes that need to be discretized into ranges in advance and descriptive attributes that take free text inputs in general. Therefore, our future research in complementing the proposed interactive questioning mechanism suggests at least the following directions.

First, we would like to determine the discretization for numerical attribute values automatically. Different applications may require different ways of discretization, while different ways of discretization may result in different effects on the collected data, as reported in the literature (Srikant and Agrawal, 1996). Therefore, it is highly desirable that the interactive questionnaire mechanism be capable of determining discretization at run time according to the contexts of different applications.

Second, we need to address efficiency and scalability problems especially when the profile database becomes large. According to our current database design, all attributes are stored in a monolithic arrangement. During the process of incremental data collection, the amount of attribute database and member database will proportionally increase, thus degrading the overall performance of our system. This is especially true, since we have used a more general definition for user profile. Therefore, the problem regarding how to periodically clean up these attributes in making room for future applications and the related database design would be another issue deserving further investigation.

Third, instead of using Java Applet technologies for front-end animation-based interface, we would like to try the possibility of using more advanced animation techniques to enhance the visual effect and increase a user's willingness to fill in a questionnaire on-line.

REFERENCES

1. Allen, C., Kania, D., and Yaeckel B., 2001, *One-to-One Web Marketing: Build a Relationship Marketing Strategy One Customer at a Time*, 2nd ed., John Wiley & Sons, New York, USA.
2. Han, J. and Kamber, M., 2000, *Data Mining: Concepts and Techniques*, Morgan Kaufmann; 1st ed., San Francisco, USA.
3. Hanson, W., 2000, *Principles of Internet Marketing*, South-Western College Publishing, USA.
4. Balabanovic, M., and Shoham, Y., 1997, "Content-Based, Collaborative Recommendation," *Communications of the ACM*, Vol. 40, No.3, pp. 66-72.
5. Yu, P. S., 1999, "Data Mining and Personalization Technologies," *Proceedings of the 6th International Conference on Database System for Advanced Applications*, Hsinchu, Taiwan, pp. 6-13.
6. Tao, F., and Muragh, F., 2000, "Towards Knowledge Discovery from WWW Log Data," *Proceedings of International Conference on Information Technology: Coding and Computing*, Las Vegas, USA, pp. 302-307.
7. Perkowitz, M., and Etzioni, O., 2000, "Adaptive Web Sites," *Communications of the ACM*, Vol. 43, No.8, pp. 152-158.
8. Srikant, R., and Agrawal, R., 1996, "Mining Quantitative Association Rules in Large Relational Tables," *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, ACM Press, Montreal, Canada, pp. 1-12.
9. Wijzen, J., and Meersmen, R., 1998, "On the Complexity of Mining Quantitative Association Rules," *Data Mining and Knowledge Discovery*, Vol. 2, No. 3, pp. 263-281.