Enhancing the IT service desk function through unobtrusive user profiling, personalization and stereotyping

A.Zaslavsky 1, C.Bartolini 2, A. Boulmakoul 3, O.Alahakoon 1, S.W.Loke 4, F.Burstein 1

1 Caulfield School of IT, Monash University, Australia
{arkady.zaslavsky, oshadi.alahakoon, frada.burstein}@infotech.monash.edu.au

2 HP Labs, Palo Alto, USA, claudio.bartolini@hp.com

3 HP Labs, Bristol, UK, abdel.boulmakoul@hp.com

4 School of Computer Science and Computer Engineering, La Trobe University, Australia, S.Loke@latrobe.edu.au
Outline

• Introduction and motivation
• eHermes – Personalization and profiling in e-commerce applications
• Integrating IT service management and eHermes personalization approach
• Related work and comparison
• Conclusions and future work
User modeling background

- **User modeling:**
  - construction of (typically computer-based) models of users' mental activities and behaviors, often used to make predictions about a system's usability or as a basis for interactive help systems (ei.cs.vt.edu/~cs5714/glossary.html)

- **User modeling (UM)** aims to make information systems really user-friendly, by adapting the behavior of the system to the needs of the individual (http://www.ionio.gr/~papatheodor/papers/Papatheodorou_formatted.pdf)

- **User personalization:**
  - **Personalization** (or personalisation) is tailoring a consumer product, electronic or written medium to a user based on personal details or characteristics they provide. (http://en.wikipedia.org/wiki/Personalization)
User models

• Collaborative
• Content-based
• Demographics-based
• Utility-based
• Knowledge-based

User profiling can also be defined as a process of identifying characteristics which are closely associated with a particular user or a target group of users, in other words, roles. It aims to identify either characteristics of individual users, or individuals who have the same or similar characteristics, thus profiling might be individual or group-based
Motivating scenario

• A typical interaction of a customer with the service desk begins with a service call.

• A call is any communication by a customer with the service desk, regardless of the method of communication (telephone, e-mail, voice-mail, and so on).

• Incidents are events, which are not part of the standard operation of a service, and could cause an interruption to or a reduction in the quality of service (QoS).
Motivating scenario (cont’d)

• A service request is a request for new or altered service.

• The types of service requests vary between organizations, but common ones include requests for change (RFCs), requests for information (RFIs), and service extensions.

• Service calls are handled by the initial support team, which is the team providing the very first line of support for processing incidents and service requests.
Personalization in IT service desk management

- Hasn’t received proper deserved attention
- Lack of adaptable, flexible interface
- Lack of tools
- Need for unobtrusive profiling in order to increase QoS, efficiency, effectiveness and user-friendliness
eHermes – Personalization and profiling in e-commerce

❖ Need of personalization to attract consumers to ecommerce websites.

❖ Personalized but less obtrusive interactions with consumers.
  • To minimize user interactions by asking fewer number of questions when searching for product needs

❖ Understanding users irrespective of the domain

❖ Single user model that can provide personalization across domains
Arkady Zaslavsky

Enhancing the IT service desk function through personalization ...

14th Workshop of the HP Software University Association (HPSUA), Garching/Munich, Germany, 8th – 11th July 2007
Main components of the system

There are 3 main components

1. The layered user model

2. Question graph – interactive, personalised question selection process, using the layered user model

3. Domain hierarchies to maintain interconnection of different layers of the user model.
User model, question graph and domain hierarchies

User needs personalized interaction in a selected domain

Registration

Layer 1
Layer 2
Layer 3

User Model

Domain Knowledge

User

Arkady Zaslavsky

Enhancing the IT service desk function through personalization

14th Workshop of the HP Software University Association (HPSUA), Garching/Munich, Germany, 8th – 11th July 2007
The Layered User model

- Considering the different user information required for behavior predictions, the single user profile we propose consists of three information layers.
  - L1 – Personal data/Demographics + behavior characteristics
  - L2 – Domain specific user behavior information
  - L3 – Product descriptive data collected during each transaction
- These layers make up the user profile which can be considered as the ‘heart’ of the proposed system.
- The profile stores and maintains information necessary for supporting all interactions and services to the registered users.
Arkady Zaslavsky
Enhancing the IT service desk function through personalization ...

14th Workshop of the HP Software University Association (HPSUA), Garching/Munich, Germany, 8th – 11th July 2007
Prototype system – First layer (L1) of the user model

This layer holds

- User’s personal information collected using the previous form
- User characteristics calculated based on above information

```
<?xml version="1.0" encoding="utf-8" ?>
<ProfileL1
 xmlns="http://tempuri.org/XMLSchema1.xsd">
 <PersonalInfo>
   User’s personal information collected using the previous form
 </PersonalInfo>
 <Characteristics>
   User characteristics calculated based on above information
 </Characteristics>
</ProfileL1>
```
Example of personal information stored

_&_<PersonalInfo>
  <Id>13</Id>
  <FirstName>Lio</FirstName>
  <LastName>Resnik</LastName>
  <DOB>20/Apr/1959</DOB>
  <Password>jjj</Password>
  <ReType>jjj</ReType>
  <Gender>F</Gender>
  <State>QLD</State>
  <Suburb>CAPTAIN CREEK</Suburb>
  <PostCode>3163</PostCode>
  <EmailAddress />
  <Industry>Agriculture/Chemicals/Forestry</Industry>
  <Occupation>Professional</Occupation>
  <WorkHours>3</WorkHours>
  <Family>Single/Bachelor</Family>
  <Education>Batchelor Degree</Education>
</PersonalInfo>
Example of calculated characteristic values – based on above information

- <Characteristics>
  <TimeSaver>0.43</TimeSaver>
  <Adventurer>0.19</Adventurer>
  <HealthConscious>0.58</HealthConscious>
  <FamilyPerson>0.25</FamilyPerson>
  <Socializing>0.21</Socializing>
  <PriceSensitive>0.3</PriceSensitive>
  <QualityConscious>0.12</QualityConscious>
  <Fun>0.49</Fun>
</Characteristics>
### Example: Layer 1

<table>
<thead>
<tr>
<th>IdentificationInfo</th>
<th>PersonallInfo</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Id&gt;30&lt;/Id&gt;</td>
<td>&lt;DOB&gt;10/Apr/1975&lt;/DOB&gt;</td>
</tr>
<tr>
<td>&lt;FirstName&gt;John&lt;/FirstName&gt;</td>
<td>&lt;Gender&gt;M&lt;/Gender&gt;</td>
</tr>
<tr>
<td>&lt;LastName&gt;Smith&lt;/LastName&gt;</td>
<td>&lt;Family&gt;Single/Bachelor&lt;/Family&gt;</td>
</tr>
<tr>
<td>&lt;Password&gt;jkkkll&lt;/Password&gt;</td>
<td>&lt;Education&gt;Diploma/Advanced Diploma&lt;/Education&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Industry&gt;Consumer Goods&lt;/Industry&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Occupation&gt;Trade person or related&lt;/Occupation&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;WorkHours&gt;21-40 hrs&lt;/WorkHours&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Income&gt;30K - 50K&lt;/Income&gt;</td>
</tr>
<tr>
<td>&lt;/IdentificationInfo&gt;</td>
<td>&lt;/PersonallInfo&gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;TimeSaver&gt;0.53&lt;/TimeSaver&gt;</td>
</tr>
<tr>
<td>&lt;PriceSensitive&gt;0.81&lt;/PriceSensitive&gt;</td>
</tr>
<tr>
<td>&lt;QualityConscious&gt;0.5&lt;/QualityConscious&gt;</td>
</tr>
<tr>
<td>&lt;Fun&gt;0.88&lt;/Fun&gt;</td>
</tr>
<tr>
<td>&lt;HealthConscious&gt;0.74&lt;/HealthConscious&gt;</td>
</tr>
<tr>
<td>&lt;FamilyPerson&gt;0.36&lt;/FamilyPerson&gt;</td>
</tr>
<tr>
<td>&lt;Socializing&gt;0.63&lt;/Socializing&gt;</td>
</tr>
<tr>
<td>&lt;Adventurer&gt;0.84&lt;/Adventurer&gt;</td>
</tr>
<tr>
<td>&lt;/Characteristics&gt;</td>
</tr>
</tbody>
</table>
Layer 2 of the Model (L2)

**Domain Features** \((d_i, f_j)\) – product descriptions that are important to consumers when deciding which product to purchase.

A given domain \(d_i\) has one or more **domain features** \(d_i, f_j\) where \((i, j = 1, 2, \ldots, n)\).
Domain features are collections of attributes

- Initially user’s preferred attributes are determined using the values calculated for each behavioral characteristic.

- One or more characteristics are relevant to each attribute, based on heuristics.

- Example of a heuristic:
  
  A less price sensitive person will have a higher chance of accepting a high cost restaurant.
Layer 2 holds user’s domain dependent behaviour

- User’s preferred attributes selected based on characteristics values
- A relevance value indicating the relevance of a given attribute to the user
- A confidence value associated with each feature
  - Indicate the system confidence of that value, depending reliability of the source acquired.
- These attribute values are used (combined with their relevance and confidence) to filter the user’s preferred products
  - hence reduce the questions directed to the user
Format of the details stored in the layer two

<?xml version="1.0" encoding="utf-8" ?>
<ProfileL2 xmlns="http://tempuri.org/ProfileL2.xsd">
  <Identification>
    <UserId></UserId>
    <DomainId></DomainId>
  </Identification>
  <Preferences>
    <Feature></Feature>
    <Attribute></Attribute>
    <Relevance></Relevance>
    <Confidence></Confidence>
  </Preferences>
</ProfileL2>
Example details stored in the layer two

```xml
<?xml version="1.0" standalone="yes" ?>
<ProfileL2 xmlns="http://tempuri.org/ProfileL2.xsd">
  <Identification>
    <UserId>30</UserId>
    <DomainId>1</DomainId>
  </Identification>
  <Preferences>
    <FeatureNum>2</FeatureNum>
    <Feature>
      <Attribute>163</Attribute>
      <Relevance>1</Relevance>
    </Feature>
    <Feature>
      <Attribute>165</Attribute>
      <Relevance>0.76</Relevance>
    </Feature>
    <Feature>
      <Attribute>167</Attribute>
      <Relevance>0</Relevance>
    </Feature>
    <Feature>
      <Attribute>169</Attribute>
      <Relevance>0</Relevance>
    </Feature>
  </Preferences>
  ...
  ...
  <Preferences>
    <FeatureNum>n</FeatureNum>
    <Feature>..</Feature>
  ...
</Preferences>
</ProfileL2>
```
Level 3 of the Model (L3)

- This layer consists of the transaction information during each user session.
- These are derived from the answers user provided regarding product attributes.
- This information is represented as attribute value pairs.
- Each domain feature is described using a set of attribute value pairs.

*Product Attributes and Values* \((<a_i,v_i>)\) – Each product is described by a set of attribute value pairs \(<a_i,v_i>\).
Information generalisation in layer 3 to layer 2

• Transaction information in layer 3 is generalised to form values for layer 2.

• Such generalisation mechanisms depend on the nature of the attribute values (symbolic attributes, numeric attributes etc.)

• With time, when the number of transactions increases, accuracy in 2\textsuperscript{nd} layer feature values become high

• Changing nature of user needs are taken in to account by making the user provide an initial query each time user visit the system.
Example user model of a user interacted eight times in four different domains

Level 1 – User’s personal information

Level 2 – Domain dependent user descriptions

Level 3 – Transaction history
Question Graph for Intelligent User Interaction

A CBR (Case Based Reasoning) approach is employed to question selection.

- The user interaction can be represented as a graph, where nodes contain questions.
- Each of these question nodes are dialog situations.
- An edge is an answer leading to another question /dialog situation (another node).
- To isolate the group of items required by the user, product attributes need to be constrained.
- To select the ideal product it is required to ask values for all the attributes from the user.
Next question is determined by the answer given to the previous question.
Unobtrusiveness

To achieve unobtrusiveness the number of questions directed to the user are need to be reduced. This is done as follows,

- Obtaining an initial query from the user with his/her most important attribute restrictions.

- Using his/her preferred attribute values in the second layer of the user model

- Once constraints are formed using above two steps, ask the minimum number of questions to whittle down the possible items
Algorithm

User selects attribute values as initial selection criteria

Add similarity values to avoid constraining similar items

Form a query for a parametric search

Retrieve items from the database

Count No_Of_Items

Calculate entropy to find first (%5) of the most discriminating attributes, for the selected items (Dis_Attri_Count) = N

Check L2 of the user model for selected attribute_value relevance

If (attribute_value is found && relevance =>2)

N = N-1

If N >0

Filter the selected items

If N = 0

Request user to provide a value for the first most discriminating attribute

Terminate search and move to presenting items

No

No

Yes

Yes
Product taxonomies and Domain Hierarchies

• All the domains usable with the user model are arranged in a hierarchy.

• When a new domain joins in, it is inserted to the correct position within the domain hierarchy.

• Domains are described using domain features.

• Domains down the hierarchy inherit the domain features of domains up hierarchy.

• Domain features are described by attributes.

• In product taxonomies, products are described using attributes.
Product taxonomies and Domain Hierarchies (cont’d)

- Each time a user selects a new domain, a new L2 layer is created.
- The new L2 hold the domain features/ user needs in the given domain.
- Depending on its position in the domain hierarchy some of the features are inherited from the upper layers of the hierarchy.
- Although there are three information layers in the user model, there are $n$ layers of L2 modules.
Multi-Agent architecture

- Since the profile building is meant to be used online, a multi-agent system is proposed for efficient and parallel handling of operations during the interactions.
- Each component in the system is handled by a dedicated agent.
- Employment of agents, also facilitate easy conversion to distributed environments where vendor sites are scattered over different servers.
- In such environments the system could be upgraded by providing mobility to these existing agents.
Interactions

A. User interaction with the Interface Agent (e.g. Product information gathering).

B. Interface Agent communicates with the Profile Instance Agent for information in the current profile for specific domain for the individual.

C. Demographic based characteristics of user buying behavior are passed on to the L2 Agent by filtering and tailoring to the specific domain, at the creation of L2 layer of the profile. Corresponding features are found using the mappings in domain descriptions data.

D. Domain specific user information in L2 is passed on to the Profile Instance Agent, who has to communicate with the Interaction Agent during the transaction.

E. This happens after and when several consumer behavioral trends have become visible across domains.

F. Individual transaction instances are generalized to layer 2 of the profile.

G. Answers to the question either obtained from the Profile Instance Agent or from the user, is prepared to be recorded in L3 layer of the profile.

H. Above prepared answers are stored as a new transaction (L3).

I. L2 Agent populates the L2 layer of the profile by obtaining the domain description data when a new domain is selected for the first time.

J. Cross domain features are identified using domain information data.
Various agents and services they provide

1. **L1 Agent** – L1 Agent generate the user’s common personality traits in human buying behavior (eg. Time saver, Price sensitivity etc).

2. **L2 Agent** – The L2 Agent is responsible of generating and maintaining of the second layer information. Initial values are generated acquiring information from the L1 layer and then maintained by generalized L3 transactions information.

3. **L3 Agent** – Maintains individual transaction instances, under separate domains.

4. **Profile Instance Agent** – The Profile Instance Agent is activated for each transaction to represent the current ‘image’ of the individual purchasing behavior and preference, in the particular domain.

5. **Interface Agent** – The Interface Agent manages the interaction between the user and the profile instance.

6. **Transaction Processing Agent** – The Transaction Processing Agent gathers user answers regarding product attributes, mapping them and uploading the L3 instance of the particular transaction.
Privacy Issues

- Since the user model stores user personal data it is necessary to ensure security of the information.
- The layered architecture supports locating each layer in different locations if preferred.
- For example storing the layer 1 (user demographics) with the user (with appropriate encryption) will make user feel safer rather than storing it in a server.
- The user model architecture supports a server based implementation.
- Since users are reluctant to provide their personal information: as required in the layer 1 (L1), an alternative of directly providing values for characteristics are allowed.
Proposed system architecture
Sources of user information

- User Inputs
- Profile database
- Stereotypes
- Web Search

User personalisation information gathering

Profile Repository

External Databases (Domain Specific)
Profile prototype for IT service management

Mr. Smith's Profile
- L1
- L2
- L3

Mr. Joan's Profile
- L1
- L2
- L3

Client Company X
- Company Roles

Software Problem 1
- Transaction 1

Software Problem 2
- Transaction 2

Software Problem 3
- Transaction 1
Related Work

- Most systems providing personalisation place the user model within the application.

- User modelling shell systems developed for generic user modelling tasks did not have the ability to reuse collected user information in different applications.

- Server based Doppelganger user modelling system introduced the idea of a single user model.

- Ms passport, liberty alliance are simple data stores which maintain user information (such as demographics) but do not have inferencing abilities.

- Later developed user modelling servers such as Personis reuse once acquired user information for more than single application.
Related work in each of the main components of the work

- The layered user model – is related to work on ‘Personis’ user modelling server.

- The user interface – related to entropy and similarity measured question selection strategies used in CBR systems.

- The knowledge engineering and knowledge base is related to Entrée system and the same data set used for Entrée is used in prototype building.

- The multi agent architecture and product hierarchies are related to the multi agent systems and general ontology used in SETA and PPG systems.