Changing machine decisions

Shapira Oz

Abstract

Introduction

User modeling has made considerable progress during the last decade, particularly in the last few years, user application expand their domain from the personal computer to smart phone, tablet computers and home devices (TV, cable modem, streamers…) <insert sentence> and change our live. today application become cross platform and cross devices , therefore the need to adaptive software is increase every day , software today are need to know every changes in their users , learn and analyze user changes in fast way through variety of information sources.

The classic user modeling approach consider the complex of the model itself, some researchers are working on creating <בכל מקום > for improving the basic design of user model systems <לתת דוגמאות למאמרים שעוסקים במבנה > in the aspect of data bases, high level design, deployment, software structure,ontology design, …<וכולי> they main mission is to contribute dynamic data base how needed in UM systems, but additional essential approach is the software adaption.

Adaption of software is the key for managing a dynamic application how reflects and respond to user behavior.

Therefore some we can catalog them to tree crude type:

1. Application with dynamic data base founded on anthology which “knows” all user stereotypes.
2. Adaptive application with learning abilities.
3. Combination of 1 & 2.

<להכניס שורת מעבר....>

According to Alfred Kobsa “User modeling research has spread into many disciplines which are concerned with the development of computer systems that are to be used by heterogeneous user populations. These fields include:

* 1. Human-Computer Interaction.
  2. Intelligent Interfaces.
  3. Adaptive Interfaces.
  4. Cognitive Engineering.
  5. Intelligent Information Retrieval.
  6. Intelligent Tutoring .
  7. Active and Passive Help Systems.
  8. Guidance Systems .
  9. Hypertext Systems and Expert Systems.”

[Alfred Kobsa “**User Modeling:Recent Work, Prospects and Hazards**”,1993]

In this lecture review will focus on the adaptive aspect, we examine if software can analyze itself by user behavior, change her code, structure, data base and follows.

**Know your user**

For creating adaptive software the first step to adopt the ability of “recognizes” users.

Recognition of user by application it’s the first step for adaptive software need overcome.

The meaning of recognition covers these fields:

* **User identification –** adaptive application need to identify her user according to the base identification parameters , according to Kobsa [“**User Modeling:Recent Work, Prospects and Hazards”,**1993] the identification process must fulfill tree tasks:
  + **User subgroup identification.** – Application must containing data base on users subgroups and to identify this subgroup from expected user population.
  + **Identification of key characteristics. –** Each subgroups has is unique characteristics (for example in music subgroup have characteristics as music type (rock,pop,juzz ….) ), the adaptive application need to draw relevant data for the identification process (without those characteristics computer system couldn’t identify the user )
  + ***Representation in (hierarchically ordered) stereotypes*** – The application-relevant characteristics of the identified user groups must be formalized in an appropriate representation system. The collection of all represented characteristics of a user subgroup is called a **stereotype** for this subgroup. The stereotype represent the order of characteristics by those stereotype , we can contribute to our application the ability of analyze user identification
* **User prior knowledge -** today most users had prior knowledge, some are first time users with minimum background knowledge while some are more experience, but even so adaptive application must obtain relevant past knowledge, filter it and analyze. This process is complex and not homogeneous, developer must consist his program to user knowledge, save relevant data and use it when the application need to.

**Arrange your data**

Since most the user model data is not hard coded, and it can change form user to user, the adaptive application must been establish generic data structure that can handled dynamic code changes, use different data bases and add new model characteristics.

This aspect has been research by many agents:

Shlomo Berkovsky ,Dominikus Heckman and Tsvi kuflik[“**Addressing challenges of Ubiquitous user model :between mediation and semantic integration”,** 2009] have been working on integrated system between semantic standardization of user model and build hybrid system - in their research they suggest to create internal semantic language combine with generic dynamic structure, with the ability to change according to language interrupter.

Other approach is to creating a collective software with the ability to collect data from different applications or for the web Francesca carmagnole [“**Handling semantic heterogeneity in interoperable distributed user models”,** 2009] insinuating to used semantic language between application and servers for data exchange.

k.vab der skuijs and G- j houben[**“automatic generation of semantic metadata as basis for user modeling and adapation”**,2009] take this approach even further more by creating frameworks with internal ontology base on semantic command language this frameworks will establish metadata for each object in the user model with the ability to read/write is metadata .

Federica cena and Roberto Furnanri[“**a model for feature base user model interoperability on the web”,2009**] suggest on using web data collector for finding requested data ,in their approach they suggest to used SOA (service oriented architecture) for creating dialog frameworks between application/servers Although this API approach is more healthy of using semantic language since it’s can spare misunderstand between applications , it’s hard to obtain foreign frameworks in each application.