

# Improving the Proactive Recommendation in Smart Home Environments: An Approach Based on Case Based Reasoning and BP-Neural Network

Gouttaya Nesrine, Belghini Naouar, Begdouri Ahlame, Zarghili Arslane

SIA Lab, Faculty of Science and Technology, Fez, Morocco

Email: [gouttaya.nesrine@gmail.com](mailto:gouttaya.nesrine@gmail.com), [naouar.belghini@usmba.ac.ma](mailto:naouar.belghini@usmba.ac.ma)

**Abstract**— Providing spontaneously personalized services to users, at anytime, anywhere and through any devices represent the main objective of pervasive computing. Smart home is an intelligent environment that can provide dozens or even hundreds of smart services. In this paper, we propose an approach to present spontaneously and continuously the most relevant services to the user in response to any significant change of his context. Our approach allows, firstly to assist proactively the user in the tasks of his/her daily life and secondly to help him/her to save energy in the smart home environment. The proposed approach is based on the use of context history information together with user profiling and machine learning techniques. Experimental results show that our approach can efficiently provide the most useful services to the user in a smart home environment.

**Index Terms**— Pervasive Computing, Proactivity, Pervasive Recommender Systems, Case Based Reasoning, Neural Networks.

## I. INTRODUCTION

Advances in computer sciences over the last decade have made digital electronics smart, cheaper and smaller. This, coupled with revolutions made in communication technology, has contributed to the development of sophisticated networked systems. Through this development, pervasive computing has emerged as part of our daily life and ubiquitous systems have raised the interest of many researchers [1].

Ubiquitous systems comprise a collection of devices embedded everywhere in the environment to make a wide range of systems capable to achieve greater functionalities. Context-aware systems, also called pervasive systems, are an emerging solution to achieve such task. The arrival of these systems will bring great opportunities to provide recommendation services to users. Indeed, they will be in charge of supervising the way users interact with the pervasive environment, providing a way to adapt the behavior of applications in order to satisfy user expectations. Therefore, access to these services must be done in a proactive manner. It requires the capacity of the pervasive system to perceive the current situation of the user, predict his needs and act

spontaneously by offering the most relevant/personalized services taking into account the context of use.

Pervasive systems can mainly be characterized by: i) The invisibility: They are integrated into everyday objects. ii) The context awareness: They can identify the environment and user characteristics in order to offer him the appropriate services. iii) Personalization: adapted to the user's needs and preferences. iv) Adaptive: have the ability to adapt to context changes. v) Proactive: act spontaneously to serve the user.

Home Area Networks can be considered as an important pervasive/recommender research area. It has received considerable attention from the computer industry in the past couple of years and many applications and projects have been developed [2]. The smarter home learns how people live their lives and offers automatic assistance. This might be to turn on/off lighting when the inhabitant enter or exit the home, or it might be to make sure that an Alzheimer patient always turns off the stove after cooking, etc.

Umar Saif presents in [3] an architecture of an embedded distributed operating system that offers a safe context-driven adaptation. The proposed operating system addresses challenge posed by ubiquitous systems using an XML-based directory that captures service and notifies the applications about changes in the device context to allow context-driven adaptation. A Dispatcher module has also been proposed to serve as context-aware call interface for the operating system, it allows to invoke the resource that best satisfies a requested service.

Mozer [4] proposed a neural network that predicts which zone in the smart home will become occupied within the next two seconds. The neural network used information upcoming from different sources like motion detectors, door status sensors, sound sensors, etc. The predictions are used to turn on the lights in the occupied zones before the user enters them. However, the approach supports only location prediction.

Authors in [5] have proposed a Smart Home Inhabitant Prediction algorithm to expect the next action of a user. The algorithm identifies matching sequences from the collected history and uses them to predict the next command. Gopalratnam and Cook present in [6] a system to predict arbitrary inhabitant-home interactions; they

predict which device the user will interact with next. The same authors also address the issue of predicting when the next device interaction will happen.

Katharina Rasch in [7] proposed two smart assistants for smart homes. The first assistant was a recommender system that suggests useful services for the user. The suggested services are tailored to the user's current situation, behaviors and preferences. The second assistant addressed the installation procedures. It reduces the need for manually describing device functionalities. As an alternative, users can simply plugin a new device and let the installation assistant run automatically.

This paper proposes an approach that enables smart home to become more context-aware. It recommends personalized actions to be performed for the user according to a current context (turning on lights when entering to home, turning off TV when inhabitant left the living room, turning on the air conditioning when the weather is cold, etc...). The proposed approach decides which services should be executed in the current situation and disables other unused services in other home locations.

The paper is organized as follows: Section 2 gives a brief overview about recommender systems. Section 3 introduces the Pervasive Recommender Systems. In section 4, we describe our proposed solution, followed by Section 5 which presents experimental results. The paper concludes with a discussion of future work.

## II. RECOMMENDER SYSTEMS

Recommender systems can be traced to the work by Herlocker and Konstan [8], who hypothesized that the inclusion of knowledge about the user's task into the recommendation algorithm in certain applications can lead to better recommendations. The general goal of a Recommender System aims to generate, for specific items, significant recommendations that might interest a user or collection of users. Recommender systems differ in the manner they analyze the collected data. Collaborative Filtering systems analyze only historical interactions; Content-based Filtering systems are based on profile attributes, while Hybrid techniques combine both collaborative and content based approaches.

### A. Collaborative Filtering systems

Collaborative Filtering refers to the technique of recommending items to users, based on what similar users (with similar tastes) have liked in the past. They are commonly divided into neighborhood based and model-based approaches. In the first method, a subset of users is chosen based on their similarity to the active user, and a weighted combination of their ratings is used to produce predictions for the current user. In contrast, model-based approaches induce predictive models based on the past ratings of all users [9].

### B. Content based filtering systems

In the context of recommender systems, content based filtering refers to content-based recommenders, which

provide recommendations by comparing representations of content describing an item to representations of a current content.

These approaches take into consideration all users and all items, where predictions are made without regard to the specifics of individual users or items. However, one can make a better personalized recommendation by knowing more about a user, such as demographic information, or about an item, such as the producer of a film[10].

Several researches have treated this problem as an Information Retrieval task, where the content associated with a user's preferences is treated as a query. An alternative to Information Retrieval approaches is to treat recommending as a classification task, where each example represents the content of an item, and a user's past ratings are used as labels for these instances. In the domain of demotic, many classification algorithms have been used for content-based recommending, including naives bayes, decision trees, and neural networks.

### C. Hybrid filtering systems

Hybrid approaches comes to allow both content-based and collaborative filtering methods to produce separate ranked lists of recommendations, and then merge their results to produce a final list. Several hybrid approaches are based on Collaborative Filtering, but also maintain a content-based profile for each user. These content-based profiles are used to find similar users.

## III. PERVASIVE RECOMMENDER SYSTEMS

### A. Presentation

Currently, the main challenge of pervasive systems is to capture and model the user's intention, to assist him in his daily life by providing appropriate services at the right time and without "direct" intervention from him. Indeed, the intelligent environment must be no longer just to "react" to user commands but rather "acting". However; in the current pervasive systems, the user still expresses his needs and preferences using voice or gestures, and the intelligent environment responds with appropriate actions.

Approaches that use contextual information in the recommendation process attempt to model and learn contextual user preferences. These recommender systems are built on knowledge of contextual user preferences and typically deal with data records of the form <user, context, service, rating>. Each record captures how much a user liked a particular service in a specific context.

In our vision, a Pervasive Recommender System should have the characteristics that we define by:

- **Context Awareness**

Traditional recommender systems operate in a two-dimensional space (User, Service). However, the use of multiple services, by the user, depends on the context. Many definitions have been proposed in the literature to define context and context-awareness. However, most researchers agree with the definition of Dey [11] he describes the context as:

“Any information that can be used to characterize the situation of entities (whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves.”

Pervasive Recommender systems must operate on Pervasive space (User, Service, Context) and propose services based on the user’s context.

• **Proactivity**

In addition to the ability of context awareness, pervasive recommender systems need to be proactive. In other words, they must be able to provide, spontaneously, personalized services to the user in order to minimize his/her explicit interaction with the intelligent environment.

• **Anticipation**

Pervasive Recommender systems must be able to anticipate future contexts of use and future user needs and preferences in these new contextual situations in order to serve the user continuously (in an uninterrupted manner).

• **Scalability**

Pervasive Recommender systems must be also able to evolve over time in order to support any changes that may arise in user’s context or/an in user’s needs and preferences in terms of intelligent services.

*B. Case based pervasive recommender systems*

Case based reasoning (CBR) is a problem solving approach, based on the reuse by analogy from past experiences called cases. A case consists of two parts: a problem and its solution. All the experiences form a cases base.

Fig. 1 show the CBR's cycle. Its steps are: the "Elaborate", the "Retrieve", the "Reuse", the "Revise" and the "Retain" phases.

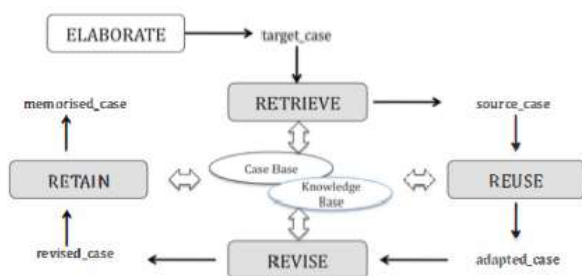


Fig. 1. The CBR's cycle [12]

The process starts when the application encounters a problem. The elaborate phase consists to build the target case from the problem. The retrieve phase consists to recuperate from the cases base the most similar cases to the target case (candidate cases). The similarity measures are defined according to the domain knowledge of the application. The reuse (adaptation) phase aims to adapt the candidate cases returned by the previous step to propose a solution to the original problem. The solution can be constructed either by transformation or by direct transfer, that is to say, by a direct copy of the recuperated solution. The revise phase consists of a review of the proposed solution by experts from the field. When the

proposed solution is evaluated as correct, the new case composed of the description of the problem and the proposed solution is retained (The retain phase) in the cases base for future use.

In the context of smart home, few studies have used the CBR method to automate service recommendation to residents, among these works we mention:

[13] Proposes a case based approach to provide a set of automated services in smart homes. It supports scenarios depicting a smart home environment, such as: air conditioning and lighting on/off, TV channel selection). The work by [14] applies case-based-reasoning using neural networks for smart homes, especially eldercare, health-care and emergency solutions.

IV. THE PROPOSED APPROACH

*A. Principle*

Our approach aims to integrate to the pervasive recommender systems, the ability to predict the most useful services to be offered to the user in future contextual situations and this by exploiting the context history. This latter is extracted from past interactions between the user and Pervasive recommender systems.

To implement our approach, we:

- Build the Contextual User Profile Database (CUPDB) from the raw cases base which contains all the contextual user interactions with the pervasive recommender system. The CUPDB is obtained by aggregation cases that have the same context [cf.parag. IV.C.2].
- Then we use this CUPDB to allow the learning process of the back propagation algorithm [15] which we use to train the neural network [cf.parag. IV.C.3].
- Finally, we use case based reasoning method to allow: firstly, the prediction of the most relevant list of services to propose to user in future context situation. Secondly, we integrate to the CBR cycle new procedures to save energy into the smart home environment. Finally, to memorize the new user’s experiences into the raw cases base.

*B. Contextual User Profile modeling*

*1) User experience*

A case within CBR meaning is traditionally composed of a description of the problem and its solution. In our approach, we define a user experience as a case under CBR meaning, which the problem is the user's context and the solution is one preferred service by the user in this context. We define a user experience by:

$$\text{User experience} = \langle \text{user context}, \text{user preference} \rangle$$

Where:

- **User context** represent all context parameters that may influence the user preferences in relation to an intelligent service. A context parameter ( $P_i$ ) is a parameter which represents fully or partially a context information. In our approach, we use the model value-attribute [16] to represent parameters of the context.

The user context is represented as follows:

$$User\ Context = \{P_i / i= 1, \dots, n \}$$

Where, n is the number of parameters that define a relevant context for an intelligent service.

- **User preference:** We define a user preference with a set of descriptions of an intelligent service including: what a user intends to do. To illustrate that, let consider a user who wishes to launch the TV in the living room. The preference express that the user wants to put the "TV" ON. Consequently, a user preference can be presented by the two parameters: The service identifier and the state of the service (ON/OFF).

In general manner, we represent a user preference by:

$$User\ preference = \{Sd_j / j= 1, \dots, m \}$$

Where, m is the number of parameters that define a set of descriptors, for this preference, in relation to an intelligent service.

2) *User Contextual Profile*

A user contextual profile is a set of preferences that characterize a user or a group of users in a particular context. Besides, in terms of services, the user preferences are naturally affected by its context. These preferences differ from one context to another. For example, a user may prefer to listen to the radio in the morning while driving his car and watching TV in the evening. So the preferences of a user are strongly related to his context and his profile. In fact, we assume that profile and context are two inseparable elements and a

profile did not make sense if it is not associated to its context.

As a result, we define the contextual user profile of a user by his *set* of preferences in a given context and we set it by the couple:

$$User\ Contextual\ Profile = (User\ Context, List\ of\ user's\ preferences)$$

Where,

$$List\ of\ user\ preference = \{Pref_k \quad ; k= 1, \dots, t \}$$

And k is the number of user's preferred services in a given context.

C. *Contextual User Profile database construction and learning*

1) *Objective*

The aim of the contextual user profile database that we propose is to facilitate prediction of the most relevant services to the user in a given context. To achieve this, we go through the following steps (fig.2):

- Raw cases base preprocessing,
- Preprocessed cases storage in the preprocessed cases base,
- Construction of the Contextual User Profile database, by aggregation cases that have the same context values in their part "user context".
- Storage of user's contextual profile in a database that we call Contextual User Profile Database (CUPDB).

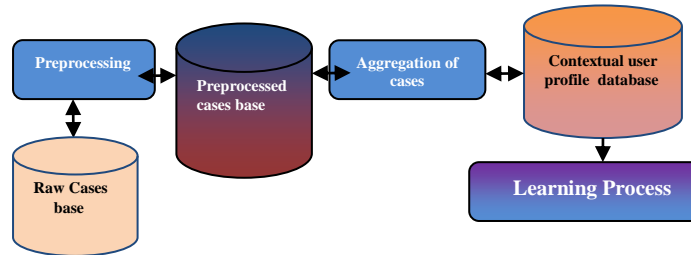


Fig. 2. Contextual User Profile Database construction

2) *Raw cases base preprocessing*

A raw case is a user experience. The preprocessing phase is the process of detecting and eliminating duplicate user experiences.

3) *Contextual user profil contruction*

To build the contextual user profile, we proceed by the aggregation of preprocessed cases in which the part of the user's context is similar.

Generally, we calculate the Contextual User Profile (X) by the following formula:

$$X_i = (\{P_i\}, \{Pref_j\}) = \sum_{j=1}^{j=q} (\{p_i\}, pref_j) \quad (1)$$

Where:

- l is the number of the extracted user contextual profile
- q is the set of user's preferred services in the context represented by the set of context parameters {Pi}.

We note that this Contextual User Profile Database is periodically updated (regenerated) in order to track any eventual changes in user's preferences over time.

4) *The learning process*

After computation of contextual user profiles X<sub>1</sub>, back propagation algorithm was used to train the neural network. The learning process is as follow:

- Randomly initialize weights and bias of the network.
- Compute the hidden layer neuron using sigmoid activation function:  
$$Y=f(\sum X \cdot w + w_0) \quad (2)$$
- Compute the output layer neuron using sigmoid activation function:  
$$Z=f(\sum Y \cdot v + v_0) \quad (3)$$
- Calculate errors of the output and hidden layers.
- Adjust weights:

$$w_{ji}^{(n)} = w_{ji}^{(n-1)} + \eta \delta_j^{(n)} y_i^{(n)} + \alpha \Delta_{ji}^{(n-1)} \quad (4)$$

With:

$$\delta_j^{(n)} = \begin{cases} e_j^{(n)} y_j^{(n)} [1 - y_j^{(n)}] & \text{If } j \in \text{output layer} \\ y_j^{(n)} [1 - y_j^{(n)}] \sum_k \delta_k^{(n)} w_{kj}^{(n)} & \text{If } j \in \text{hidden layer} \end{cases} \quad (5)$$

- Calculate the total error of the network.
- Repeat these steps until minimizing the network total error.

Where, X the vector of input layer neurons, Y is the vector of the hidden layer neurons, and Z represents the output layer neurons. w is the weight matrix between the

input and the hidden layer.  $w_0$  is the bias of the hidden layer neurons. v is the weight matrix connecting the hidden and the output layers, and  $v_0$  is the bias of the output layer neurons.  $\eta$  and  $\alpha$  are learning rate and momentum factor.

#### D. Case based reasoning cycle of our approach

To allow automatic service recommendation in pervasive environments we are based on the CBR method. We summarize the whole CBR cycle of our approach in fig. 3:

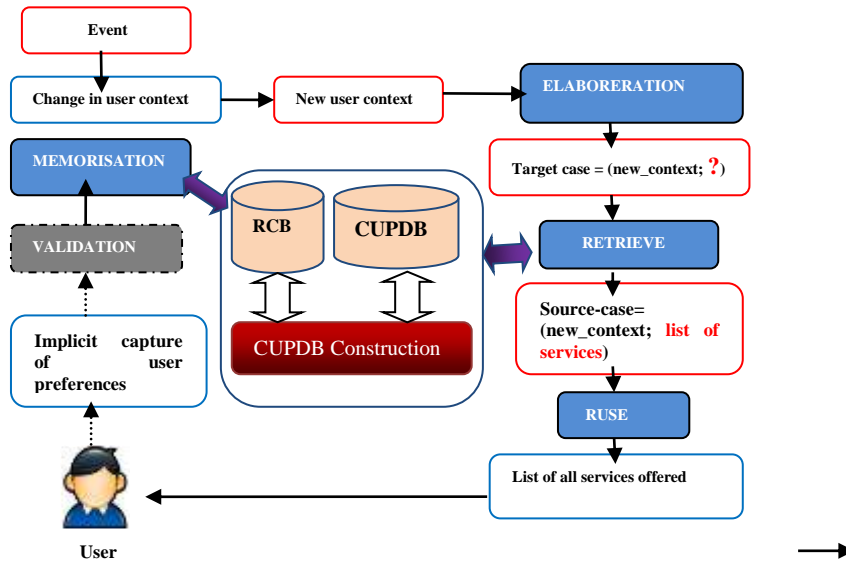


Fig. 3. CBR cycle of our approach

#### 1) Elaboration Phase

When a relevant context for services recommendation to user is met by the pervasive recommender system, a target case is constructed by the elaboration phase, including in its part problems the current context of the user and his part solution is unknown. This part solution must be the list of services that the system must offer the user in this context.

#### 2) Retrieve phase

The target case built by the elaboration phase, constitute the input of the retrieve phase. This phase should return a list of services based on learned inputs : (context, services).

So, the incoming target case is considered as new input "X" and fed into the neural network. Outputs are calculated and the output "Z" with maximum value is selected.

$$Y_i = f(\sum X \cdot w + w_0) \quad (6)$$

$$Z_i = f(\sum Y \cdot v + v_0) \quad (7)$$

#### 3) Reuse phase

In our CBR cycle, the retrieve phase produces the services list to recommend in the current context of the user.

This list contains a set of service to enable or disable as needed by the user. For example, if in the current context the user like watching TV in the dark. The list should contain the services {TV ON , Light OFF}.

This list is transferred to the actuators of the smart home to enable or disable services mentioned in the list.

But, in the intelligent home there is another concern is economization of energy: in this sense we thought to operate (execute) services recommended by the pervasive recommender services in the current context of the user and to disable the rest of smart home services and this to save the energy.

Disabling the rest of the services of the house should be done if there's no one in other parts of the home.

We accomplish this operation in CBR cycle based on the rules that we call energy management rules in the smart home.

With:

- $S = \{s_i / i=1, \dots, n\}$  The set of all intelligent services of a smart home.
  - $CXT = \{cxt_j, j=1, \dots, m\}$  all user contexts in a smart home.
  - Current-context is the current context of the user.
  - $U_{cxt_j} = \{u_j / j=1, \dots, k\}$  the set of service to recommend to the user in the context<sub>j</sub>. with  $u_j$  belongs to S.
- Energy management rules in the smart home:

“If current-context =  $cxt_j$  and For each  $cxt$  belongs CXT-  $cxt_j$  and If user’s-presence = ‘no’ Then For each  $s$  belongs to S-  $U_{cxt_j}$  put state = OFF”.

4) Memorization phase

Once the set of services is provided, it comes the role of the memorization phase which observes the reaction of the user, which can accept all of the services offered or to select another set of services.

Memorization phase analysis of the reaction of the user and product a new case to be recorded directly in the raw cases base. Each case contains in its part "user context" the current context of the user and its solution part contains the description of a particular service from the set of the offered services.

The memorization phase is done without any explicit validation by the user. This implicit validation phase of the CBR cycle is a key point of our approach to meet our objective to increase the proactivity of pervasive recommendation systems.

V. EXPERIMENTATION

In this section, we introduce the way of utilizing the proposed approach. First, original data are generated by sensors, and as in classical CBR system, phases varies from case elaboration to case memorization (fig.1). The application can be capable to automatically launch, for the user, the appropriate services according to his current

context in any context of use and disables other unused services.

The information context in smart home environment can be defined using several parameters. In this study, we define the context of the user profile by:

User Context = {P1= Time, P2= Localization, P3= temperature}

Where attributes:

- P1: represents the current time of the day {1-6, 6-8, 8-10, 10-12,12-13, 13-14, 14-16, 16-18, 18-20, 20-22, 22-01}
- P2: indicates the current location of the user {kitchen, living room, bedroom, dining room}
- and P3 indicates the daytime temperatures {cold, warm, hot}

We define a smart home service by two parameters: The service identifier and the state of the service (ON/OFF).

Where:

- Service-id: represents the name of a specific smart service: {Light, TV, air-conditioner ,etc...}
- State represents the state of the service : {ON,OFF}

The proposed system activates or deactivates the appropriate candidate solutions (TV, radio, air-conditioner, stores, and lighting) according to a given context.

The following table (table 1) shows some considered scenarios:

Table 1. some scenarios of trained data

Training Database					
User context			service		classes
Time	Localization	temperature	Services-id	state	
20-22h	living room	warm	Light, TV	On	C0
20-22h	living room	cold	Light, Tv,air-conditioner	On	C13
7-8h	bedroom	warm	stores	On	C4
20-22h	Dining room	cold	TV, air-conditioner	On	C1
20-22h	Dining room	warm	TV	On	C3
>1h	bedroom	warm	Light, stores, air-conditioner	Off	C12
Example of new contexts					
Time	localization	temperature	Desired action	Obtained result	
23h	bedroom	warm	C12(Light, stores, air-conditioner, off)	C[12]=0.9766893941905033,	
00h	living room	warm	C12(Light, stores, air-conditioner, off)	C[12]=0.9924555636118803,	
22	living room	cold	C13 (Light, Tv,air-conditioner)	C[13]=0.9859041344639088,	

Experimental results show that the system responds well with a prediction rate that reaches 97%.

VI. CONCLUSION

Hopefully to have a computing structure vanished into the environment; the aim of this paper was to incorporate the Contextual User Profile in the automatic recommendation process of smart home environment.

The general idea is to facilitate user interaction with some context aware applications by determining, on behalf of the user, the application to be performed. The

proposed solution extracts user’s preferences based on his previous interactions with the system and launches automatically, in a specific context, the appropriate services and disables other unused services inside the smart home environment.

The experimental evaluation was performed using simulated contexts and user preferences.

It is obvious that the gap is large between laboratory demonstration and scenarios available on a large scale. In our future works, additional scenarios will be considered to support various user specifications and additional methods will be developed to address the problem of missing and noisy sensor data.

REFERENCES

- [1] Weiser M. The computer for the 21st century. On Scientific American, 1991, 265(3): 94-104.
- [2] Barlow J, Gann D, Venables T. Digital Futures: Making Homes smarter, 1999, Convery, Publisher: Chartered Institute of Housing book.
- [3] Umar S. Architectures for ubiquitous systems. On Technical Report N °527, 2002, university of Cambridge,
- [4] Mozer M C. Lessons from an adaptive home. On Smart Environnements, 2005: 271–294.
- [5] Das S, Cook D, Battacharya A, Heierman E, Lin T. The role of prediction algorithms in the MavHome smart home architecture. IEEE Trans. On Wireless Communications, 2002,9(6):77–84.
- [6] Gopalratnam K, Cook D J. Online sequential prediction via incremental parsing: The Active LeZi algorithm. IEEE Trans. On Intelligent Systems, 2007,22(1):52–58.
- [7] Katharina R. Smart assistants for smart homes. Phd Thesis, 2013,Stockholm, Sweden.
- [8] Herlocker J L, Konstan J A. Content-independent task-focused recommendation. IEEE Internet Computing, 2001,5(6) :40–47.
- [9] Breese J S, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, July 1998, Madison, WI.
- [10] Melville P, Mooney R, Nagarajan R. Content-boosted collaborative filtering for improved recommendations. Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, Alberta, 2002: 187–192.
- [11] Dey A K, Abowd G D. Towards a better understanding of context and context-awareness. Proceedings of the Workshop on the What, Who, Where, When and How of Context-Awareness. 2000, ACM Press, New York .
- [12] Aamodt A, Plaza E. Case-Based Reasoning: Foundational Issues, Methodological Variations, and Systems Approaches. On AI Communications Journal,1994, 7(1):39-59.
- [13] Tinghuai M, Kim Y, Qiang M, Tang M., Zhou, W. Context-Aware Implementation based on CBR for Smart Home. On Wireless and Mobile Computing, Networking and Communications (WiMob'2005), IEEE International Conference, 2005: 112 – 115.
- [14] Leake D, Maguitman A, Reichherzer T. Cases, Context, and Comfort: Opportunities for Case-Based Reasoning in Smart Homes. Lecture Notes in Computer Science, 2006 4008(1):109-131.
- [15] Bryson A E, Ho J. Applied optimal control, Blaisdell publishing Co,1969.
- [16] Schilit B, Adams N, Want R. Context-aware computing applications. IEEE Workshop on Mobile Computing Systems and Applications. 1994, Washington, DC, USA: 85–90.

Authors' Profiles



**Nesrine Gouttaya** was born in Taounate, Morocco in December 1980. He received her Ph.D. degree in 2014 from Sidi Mohamed Ben Abdellah University, Fez, Morocco.

His research interests include Pervasive Computing, SmartThings and Human-Computer Interaction. His contact address is: gouttaya.nesrine@gmail.com



**Naouar Belghini** is a doctor of computer sciences from Sidi Mohamed Ben Abdellah University (Fez-Morocco); she received her Ph.D. degree in 2013 in the field of biometric authentication using facial data. Her research interests include machine learning, pattern recognition,

neural networks, etc.



**Ahlame begdouri** is a professor in the department of computer science, head of the VIA research group at the SIA lab "système Intelligents et Applications " of the Faculty of Science and Technology of Fez-University of Sidi Mohamed Ben Abdellah. She received her PhD on the topic of "Quality of services of multimedia traffic over best effort networks" from the Faculty of Science of Rabat-Morocco on 1999. Her current research intersets concern context aware adaptation, ambient intelligence and mobile learning. Prof. Begdouri is the coordinator of an FP7 project called MoICT(2010-2014) of IEEE Moroccan section, Member of the local office of IEEE Moroccan section , Member of IEEE Moroccan Education Chapter, Co-founder and member of the E-NGN research group since 2008,Associate member of the Moroccan FP7-ICT-NCP unit , Program committee chair international conference (WNGN'08, NGNS'09, CIST'11, CIST'12). Her contact address is : BP2202,Route d'Imouzzar, Faculty of Science and Technology of Fez. E-mail:abegdouri@gmail.com



**Arsalane Zarghili** is a Doctor of Science from Sidi Mohamed Ben Abdellah University (Fez-Morocco). He received his Ph.D. in 2001 and joined the same University in 2002 as Professor at the computer science department of the Faculty of Science and Technology (FST).

In 2007 he was head of the computer sciences department and chair of the Software Quality Master in the FST-Fez. He lectures Programming, Distributed, compilation and Information processing, for both undergraduate and master levels. In 2008 he obtained his HDR in information processing. In 2011, he is the co-founder and the head of the Laboratory of Intelligent Systems and Applications in the FST of Fez. He is a member of the steering committee of the department of computer sciences and was a member of the faculty board. In 2011 he is the chair of Master Intelligent Systems and Networks. He is also IEEE member since 2011. His main research is about pattern recognition, image indexing and retrieval systems in cultural heritage, biometric, etc.

**How to cite this paper:** Gouttaya Nesrine, Belghini Naouar, Begdouri Ahlame, Zarghili Arslane, "Improving the Proactive Recommendation in Smart Home Environments: An Approach Based on Case Based Reasoning and BP-Neural Network", International Journal of Intelligent Systems and Applications (IJISA), vol.7, no.7, pp.29-35, 2015. DOI: 10.5815/ijisa.2015.07.04