Towards an Affective Aware Home

Benţa Kuderna-Iulian, Cremene Marcel and Todica Valeriu

Comm.Dept.,Technical University of Cluj-Napoca, Romania
{Iulian.Benta, Marcel.Cremene, Valeriu.Todica}@com.utcluj.ro

Abstract. The nowadays smart homes run predefined rules, but the user's desired behaviour for a smart home varies, as his/her needs change over time. To edit the initial rules is a difficult task for a usual user. We propose a control mechanism that allows the system to learn the new behaviour preferences without editing the rules, but responding emotionally to the system's decisions. In order to capture the emotion reaction we use FaceReader, a tool for facial analyses, adapting it to read three valence levels that work as positive, negative or neutral feedback. The results in training a MLP neural network to learn the preferred behaviour from the user's emotional reaction are discussed. Ontology is used in order to describe the context.

Keywords: Smart Home, Affective Computing, Context Awareness, Ontology, Neural Networks.

1   Introduction

Editing the smart homes rules is difficult for the user because of the complexity that comes with the use of different sensors and actuator driving to a large number of combinations for the rules to include [1][2]. Moreover in [2] the authors notice that "rule-based reasoning is not flexible and can not adapt to changing circumstances".

Speech or motor impaired find difficult to give vocal or physical commands in order to control the behaviour of a smart home but they may facially display short time affective responses (emotions) that could be used as feedback. Taking as reference the normal neutral state, a positive emotion will mean approval and a negative emotion will mean disapproval of the system's decision, if expressed immediately after it.

Scenario. Maria is an old and speech impaired person. She is invited to her friend Laura that has a smart home. As Maria is a welcomed guess, the system will authorize her to personalize its behaviour. One of the home rule closes the blinds when the outside light has the same intensity as inside. Maria likes to look outside the window and so, when the first decision of the system to close the blinds is triggered (at sunset, for instance), she will display immediately (in the next minute) a negative emotion (i.e. anger) showing her disapproval. The smart home will learn (after repeating it a few time, if needed) the new Maria's preference.

Technical Issues. We have in mind the following questions to answer to:

1. How should we represent the Affective Aware Home's knowledge?

Approach. We propose a control mechanism that allows the system to learn the new behaviour preferences without editing the rules, but responding emotionally to the system's decisions. In order to capture the emotion reaction we use FaceReader, a tool for facial analyses, adapting it to read three valence levels that work as positive, negative or neutral feedback. The results in training a MLP neural network to learn the preferred behaviour from the user's emotional reaction are discussed. Ontology is used in order to describe the context.

Paper outline.The rest of the paper is organized as follows. In section 2 we overview the existing solution regarding aforementioned questions. Then in section 3 we present the principle of the affective control loop mechanism. The implementation details of the Affective Aware Home are explained in section 4. In the next section the results in training a MLP neural network to learn the preferred behaviour from the user's emotional reaction are discussed. In the last section we conclude out work and present the future work.

2   Related Work

The use ontology in context modelling because it is independent from any programming language, supports formal representation of the context [3][4], allows for knowledge distribution and reuse, logical context reasoning (consistency check, subsumption reasoning, implicit knowledge inference) [5], has expressing power (i.e. OWL has cardinality constraints), hierarchical organization, use standards for efficient reasoning, abstract programming and interoperability [6]. By using reasoning mechanisms the context can be augmented, enriched and synthesized [7]. Moreover it solves heterogeneity, ambiguity, quality and validity of the context data [8].

The user data is usually considered as a part of the context and can also be ontologically modelled [9], including details on her/his affective states [10].

In [11] the authors review the existing context related preference representation and propose a scored based solution. They assign a score to each preference possibility, consisting in a real value in the [0, 1] interval or a predefined value (veto, indifferent, mandatory, error situation). If a context C, and an associated variable set v are present, the score will be the function *score(p.s,C,v)*, where *p.s* is the scoring expression, otherwise the score is indifferent. In this model the context elements are considered distinct, without any relation between them.

An ontological representation of the preferences is presented in [12]. It models ontologically the relations between the context elements and the preferences. The *Preferences* class has relations with all the main classes (*Time*, *Agent*, *Location*, *Activity*). The preference can be positive or negative indicating an appropriate or inappropriate choice for a resource, environment or operation. This model uses a

probability to set the preference priority, but has only two values to express the relation between the context and the service (desired behaviour).

Another solution [13] uses Bayes RN-Meta-networks, organized in multi-layers. It is the only one we found to support online preference discovery mechanism in context awareness. The mechanism consists in updating the preference model for each user if the system's decision was disproved by at least one user. The preference model update is done by calculation the distribution probability for each user and then propagating the values to the next meta-network layers. Its main issue is that the prior probabilities need to be initially calculated by a human that is difficult for a large number of context elements. The main advantage of this model is that it supports online preference update.

In the article [14] the author presents an associative network between context and application. Each context element could be associated with all N applications for a user. The association relation was modelled by a variable weight w that indicated the connexion strength between the context element and the application, thus given the weight matrix and a certain context, one may predict the application a user will chose. Extending this idea, the weights could store the user's preferences, but still this solution lacks the advantages of ontological modelling.

The neural networks are used in [15] to describe weighted relations between the context elements (responding to: who, where, when, how) and the context elements (responding to how), the services and service parameters. They use MLPs (Multi Layer Perceptrons) with one hidden layer. This solution allows modifying the relations' weights, but the authors explore only the offline neural network training and do not use ontologies in context modelling.

Table 1. A comparison between different preference representation and update solutions.

Solution	Ontological Context	Context-service relation	Online update
CtxPrefScore'06 [11]	-	score ([0, 1])	-
OWLPref'05 [12]	+	ontological(appropriate/not)	-
Bayes Meta-Net'06 [13]	-	probabilistic	++
NNAssoc'05 [14]	-	association network weights	-
UPM'05 [15]	-	MLP weights	+

We may notice in Table 1 that only one solution adopted an ontological context modeling, has only two values to express the relation between the context and the desired behaviour. There are different approaches for context-service (behaviour) relation which allow for a more or less fine grained expression of the preferences. In the last column, we notice that even if the MLP solution [15] supports weight update based on a back propagation algorithm it does not explore the online training from the user's feedback. The Bayes Meta Network solution [13] is the nearest to meet our online updatable preferences objective, but in this case they do not use ontologies and need a prior probabilities calculation.

Regarding the use of emotional response for learning the desired behaviour, the article [16] presents a reinforcement learning mechanism where a social robot learns from rewards and punishments expressed by positive (happy) and negative (fear)

emotions. A reinforcement learning mechanism implies to give feedback for a set of tasks, but our objective is to have a simpler loop with immediate response. We also searched for a more general emotion valence assessment tool, explained in detail in section 4.3 where we justify the choice for the FaceReader[17].

3 The Principle of Affective Control Loop Mechanism

We made the following decisions for representing the Affective Aware Home's knowledge:
    1. To use ontology for context and service representation
    2. To represent the relation context-service, the preferred behaviour as weights, stored in the ontology
    3. The preference update will be according to the user's affective reaction (emotions) to the system's decisions
    In principle we consider the context C, composed by context elements in relation with each other, a service vector S, and a weight vector w, that records the preferred behaviour, that is the service to choose when the context C is present and a current affective state Ψ of the user U.

Preference representation. We argue that storing the preference in neural network weights is better then in Bayes RN Meta-networks like in [13] because:
    1. The neural network allows initial training by an example training set, comparing to a mandatory prior probability calculations, simplifying the work at this stage.
    2. If rules or Bayesian approach would be used a full description of the behaviour should be given (all combinations of context values and desired behaviour), a neural network can run with a few training examples if any, due to its generalization capability, and adjust online.
    3. The neural network have the ability to generalize from a given set of examples.
    Representing the preferences in ontology is motivated by:
    1. The ontology supports the distribution and reuse of the once learned preference in other applications with the same context elements and services or similar (when increasing or decreasing one or more context elements or services)
    2. The neural network is to become dynamically reconfigurable (we may change its parameters on runtime: the number of hidden layers, neurons on each layer, activation function type, learning rate for each layer neurons)
    The actual part that the ontology that contains the representation of the neural network is beyond the purpose of this article, as in this first implementation we saved the neural network parameters values in a file.

The Affective Control Loop Mechanism. We propose to replace the rule based decision mechanism with a neural network that learns from the user's affective feedback the new preferred behaviour in order to respond to the user's new needs.
    In order to estimate the current affective state we should use a software tool that analyses a person's facial features and asses the current basic emotion.
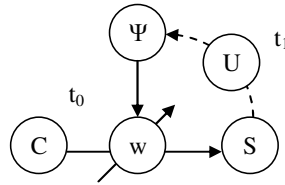
Fig. 1. The principle of preference update based on the user's affective response

The mechanism works as explained below (see Fig.1):

1. At t_0 the system will choose a service for the present context by feeding forward in the randomly or prior trained (with values from that user's behaviour history in similar conditions) neural network.

2. This decision for a service s at t_0 will determine a user reaction in the next time interval t_1. From this reaction, we are interested only in the valence of the emotion: positive (meaning acceptance) or negative (denial).

3. The acceptance or denial will determine the adequate weight w modification. Then the cycle repeats from 1. This way the system adapts itself in successive steps.

4 The Affective Aware Home Implementation

We developed and tested an agent based architecture sharing a two layered ontology similar to the one presented in [4]. We used the Phidgets [18] sensors, RFID and motors platforms to read the room and outside light, to identify the user and to open/close the blinds. As a detailed description of the system is beyond the scope of this paper, we will focus on the parts that implement the affective awareness.

The Affective Knowledge Representation. We added in the context ontology the concept State as in [10], but, as we were interested by the valence representation for the current state, we defined the subclass CurrentState and for it the valence property with three possible values (positive, negative and neutral) as depicted in Fig.2:
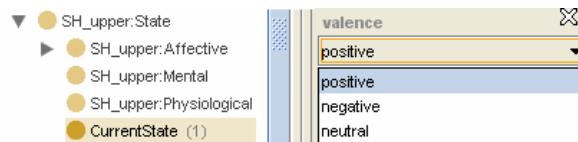

Fig. 2. Fragment from the *SH_lower* ontology illustrating a *CurrentState* individual (*left*) and its valence datatype property with the three possible values (*right*)

Modelling Preferences in a MLP Neural Network. At this stage we implemented our multilayer perceptron (MLP) using a public Java neural network API [19] that saves the network parameters in a file.

The entrance of the MLP had two inputs, the room light (LightSensor1) and the outside light (LightSensor0), with three possible values (low, medium and high) updated into the ontology by the sensor agent:

```
light_indoor=sensorMap.getSensorById("LightSensor1").getS
ensorValue();
```

There is just one output of the neural network, the blinds status (on/off) that has to be set up in the ontology once a decision is taken:

```
deviceMap.setDeviceStatus("Blind_2","ON");
```

The initial training set is generated by using a predefined rule for test purposes only: each time outside is brighter then inside, the blinds will open, otherwise they will be closed, a total of 9 input combinations, with normalized input and output values (-0.5 for low, 0.0 for medium, 0.5 for high; -0.5 for closed, 0.5 for open). In a real case, historical data could be used (input, desired output pairs) for this training.

After repeating several simulations the optimum MLP parameter configuration was 6 neurons in the hidden layer and 50000 epochs, or an equivalent of 0.05 training error rate.

We may compare this with the equivalent Bayes RN Meta-networks [20] solution where we have an important increase of prior probabilities with the number of inputs. In our scenario we would have to complete $3^2*2^1=18$ combinations, but adding a binary value input (authorized/not authorized user) the number of prior probabilities would double: 36. So, an exponential grows. Moreover the presence of two users demands for one more layer, resulting that for n users n+1 Bayesian layers are needed. As a consequence $3*36 = 108$ values need to be computed.

The complexity of the Bayes RN Meta-networks [18] is:

$$O(N*p*q^{\alpha}+q^{\alpha*N*p})\qquad\qquad(1)$$

Where N is the number of users, p is the user's probability to be in a certain location, q the number of service values or possible actions, α is a value proportional with the number of context elements multiplied by the possible values for that element. For the given example the complexity would be O $(1*1*2^6+2^{6*1*1})$ =O (128).

In the neural network case we reduce the complexity to:

$$O(e*q).\qquad\qquad(2)$$

Where e is the number of context elements, q the number of service values or possible actions, so we have O $(2*2)$ =O (4). That reduces the complexity 32 times.

Adapting the FaceReader. In order to know the affective reaction we used the FaceReader [17] software that analyse a person's facial features and asses the current basic emotion. We improved the current state valence calculation allowing for three levels (positive, negative, neutral) estimation by analysing the FaceReader's outputs in time after interpolating to solve the missing samples issue and empirically establishing particular thresholds between the levels for one user only: maximum instantaneous values for happy>=0.4, angry>=0.74, disgust>=0.65 and mean values

for one minute (5 samples, one each 200 milliseconds) happy>=0.47, angry>=0.78, disgust>=0.82.

5 Results and Discussions

We ran 100 simulations for two inputs (outside light, room light) with three levels (low, medium, high) and one output (on/off blinds) and varying the neuron number in the hidden layer (1-10) and the number of training examples (1-9) to answer to what is the minimum number of examples needed to relearn the new preference, after an initial behaviour training. The best configuration was 6 neurons in the hidden layer, demanding for a minimum of 3 examples out of 9 possibilities to overlap 80% of correct behaviour for a new behaviour, increasing to 90% after 6 examples. The duration of the retraining is 780 milliseconds, less then 1 second the time we considered a person needs to express an emotion.

We had pilot experiments with the real setup in our research lab (sensor platform, ontology, blind motor, a web camera capturing the user's face for the FaceReader to assess the emotions) for one person, in which the prior trained neural network replaced the 9 behaviour rule set. In almost all test cases the trained rule was successfully replaced. We managed to have one successful retraining with the new preferred behaviour with just 3 examples, but we should do some more adjustments to the system to work fine: increase the number of random training examples to 6, reduce the time window for expressing the emotions and increasing the valence accuracy.

6 Conclusions and Future Work

We proposed and tested a new behaviour control mechanism for smart homes, based on the user's affective reactions to the system's decisions. This allows preference discovery, storage and use for responding to ever changing user needs. At this stage the preferences are stored as neural network weights in a file, but we envision storing them in an ontological representation.

In order to know the affective reaction we used the FaceReader software that analyse a person's facial features and asses the current basic emotion. We improved the current state valence calculation allowing for three levels estimation by analysing the FaceReader's outputs in time and empirically establishing thresholds between the levels.

The next work should answer to the following question: how does the number of user given training examples for preference learning evolve once we increase the number of context elements.

We are also preparing a method to determine more accurately the valence (nine levels) and thus allowing neural network learning rate variation in accordance with the user's level of acceptance or denial for the Affective Aware Home's decision.

References

1. Nishigaki, K., Yasumoto, K., Shibata, N., Ito, M., Higashino, T.: Framework and rule-based language for facilitating context-aware computing using information appliances. In: First International Workshop on Services and Infrastructure for the Ubiquitous and Mobile Internet, pp. 345--351 (2005)
2. Guan, D., Yuan, W., Cho, S. J., Gavrilov, A., Lee, Y.-K., Lee, S.: Devising a Context Selection-Based Reasoning Engine for Context-Aware Ubiquitous Computing Middleware, LNCS, vol. 4611, pp. 849--857, Springer (2007)
3. Gu, T., Pung, H.K., Zhang, D.: A Service-Oriented Middleware for Building Context-Aware Services. Elsevier JNCA, vol. 28, issue 1, pp. 1--18 (2005)
4. Wang, X., Zhang, D., Gu, T., Pung, H. K.: Ontology Based Context Modeling and Reasoning using OWL. In: Proceedings of Workshop on Context Modeling and Reasoning, Orlando (2004)
5. Yau, S. S., Liu, J.: Hierarchical Situation Modeling and Reasoning for Pervasive Computing. In: Proceedings of the Fourth IEEE Workshop on Software Technologies For Future Embedded and Ubiquitous Systems, and the Second international Workshop on Collaborative Computing, integration, and Assurance, Washington (2006)
6. Ejigu, D., Scuturici, M., Brunie, L.: An Ontology-Based Approach to Context Modeling and Reasoning in Pervasive Computing. In: Proceedings of the Fifth IEEE international Conference on Pervasive Computing and Communications Workshops, Washington (2007)
7. Buriano, L. Marchetti, M. Carmagnola, F., Cena, F., Gena, C., Torre,I.: The Role of Ontologies in Context-Aware Recommender Systems. In: IEEE International Conference on Mobile Data Management, pp. 80 (2006)
8. Krummenacher, R., Strang, T.: Ontology-Based Context Modeling. In: 3rd Workshop on Context Awareness for Proactive Systems, Guildford (2007)
9. Heckmann, D.: Ubiquitous User Modeling, PhD. Thesis, Saarbrucken (2005)
10. Benta, K-I. Rarau, A., Cremene, M.: Ontology Based Affective Context Representation. In: Euro American Association on Telematics and Information Systems, Faro (2007)
11. Henricksen, K., Indulska, J., Rakotonirainy, A.: Using context and preferences to implement self-adapting pervasive computing applications, Software-practice & Experience, vol. 36, issue 11-12, pp. 1307--1330, Ed. John Wiley & Sons Ltd (2006)
12. Anh, K., Lee, Y.-K., Lee, S.:OWL-Based User Preference and Behavior Routine Ontology for Ubiquitous System. pp. 1615--1622, LNCS, vol.3761, Springer (2005)
13. Hasan, Md. K., Anh, K., Mehedy, L., Lee, Y.K., Lee, S.: Conflict Resolution and Preference Learning in Ubiquitous Environment. In: The 2006 International Conference on Intelligent Computing, LNAI, vol. 4114 0355, Kunming Yunnan Province (2006)
14. Flanagan, J.: Context awareness in a mobile device: Ontologies versus unsupervised/supervised learning. In: Proceedings of International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning, pp. 167--170, Espoo (2005)
15. Suh, Y., Kang, D., Woo, W.: Context-based User Profile Management for Personalized Services, ubiComp, workshop ubiPCMM (2005)
16. Broekens, J.: Emotion and Reinforcement: Affective Facial Expressions Facilitate Robot Learning. LNAI, vol. 4451, pp. 113--132, Springer (2007)
17. Noldus Information Technology, FaceReader™, http://www.noldus.com/human-behavior-research/products/facereader
18. Phidgets - USB sensing and control, http://www.phidgets.com/
19. Feed Forward Neural Networks v2.0, http://aydingurel.brinkster.net/neural/
20. Anh, K.: User Preference Learning in Context-aware Computing. Master thesis, Department of Computer Engineering, Faculty of Graduate School of Kyung Hee University, Korea (2005)