Novelty Detection in User Behavioural Models within Ambient Assisted Living Applications: An Experimental Evaluation


Abstract— Current approaches to networked robot systems (or ecology of robots and sensors) in ambient assisted living applications (AAL) rely on pre-programmed models of the environment and do not evolve to address novel states of the environment. Envisaged as part of a robotic ecology in an AAL environment to provide different services based on the events and user activities, a Markov based approach to establishing a user behavioural model through the use of a cognitive memory module is presented in this paper. Upon detecting changes in the normal user behavioural pattern, the ecology tries to adapt its response to these changes in an intelligent manner. The approach is evaluated with physical robots and an experimental evaluation is presented in this paper. A major challenge associated with data storage in a sensor rich environment is the expanding memory requirements. In order to address this, a bio-inspired data retention strategy is also proposed. These contributions can enable a robotic ecology to adapt to evolving environmental states while efficiently managing the memory footprint.

I. INTRODUCTION

Smart home environments are emerging rapidly as sensor rich systems. These systems require substantial computation to extract high level knowledge and understanding from low level sensory information, so as to enable appropriate decisions to be made regarding the state of the environment, i.e., the ecology. The main objectives of introducing intelligence into a smart home environment are to identify various events and automatically activate suitable responses based on their degrees of importance [1]. The intelligence comes from the adaptive behaviour of the overall ecology as per the requirements of the user. Different aspects of smart home environments have been reported in the literature [2,3]. These include an intelligent just-in-time Activity of Daily Living (ADL) assistance provision within an integrated system architecture [4], a home monitoring system for elderly-care application [2], and a context aware system for smart home applications [3,5].

The RUBICON (Robotic UBiquitous COgnitive Network) EU FP7 project [6], which this work is part of, aims at creating a self-sustaining, self-organizing, learning and goal-oriented robotic ecology. It consists of a network of heterogeneous computational nodes interfaced with sensors, effectors and mobile robots which are used to accumulate environmental information; process it and then act accordingly. There are four technical layers (see Figure 1) within this ecology, namely: learning, cognitive, control and communication. The learning layer addresses sensory information for event classification. The cognitive layer, using the event information, focuses on knowledge development that accurately reflects the dynamics of inhabitant’s behaviour and supports their daily activities by setting goals which the control layer then achieves by employing robots within the ecology. The communication layer is responsible for data transmission among the layers. The work presented in this paper focuses on implementing a novelty detection approach to support the cognitive role within the ecology and empirically evaluates the novelty detection aspect in a test-bed setup.

Novelty detection can be defined as the process of identifying interesting new stimuli that are different from anything known before [7,8]. In this sense, novelty detection can be seen as a form of selective learning which treats any experience which falls outside of those seen during training as novel. A number of novelty detection methods have been proposed in the literature, mainly focussing on detecting anomalies and outliers, i.e. identifying patterns that do not conform to expected behaviour [7,8,9,10]. Typically for these problems there are substantial data about the normal classes but very little data displaying the novel features that should be detected. Hence, it is essential to learn a model of the normal dataset and then attempt to detect deviations from this model. Within an Ambient Assisted Living (AAL) application, novelty detection serves to build up a picture of an inhabitant’s expected behaviour, in this case their natural daily routine. Deviations from this routine, once formulated, are considered novel and cause the system to act on this new information by attempting previously known but unused actions which can be performed by a mobile robot.

Figure 1: RUBICON high level architecture
As with most data accumulative systems, there is the potential for memory bloat with the ever increasing amount of data collected and processed. This is addressed through the development of a bio-inspired data retention strategy for efficient memory management. A key assumption in most biological models is the existence of a reinforcement signal which is presumed to mediate up- and downward changes in synaptic efficiency. Hence the neuromodulator dopamine, which plausibly forms part of the neurochemical substrate of reinforcement signals [11] is simulated as a moderator for data retention.

The remainder of this paper is organised as follows. Section II briefly discusses an overview of the cognitive architecture and the role of novelty detection within it. Section III introduces a memory management system which incorporates a biologically inspired forgetting factor to discard information that is infrequent or not experienced for a period of time. In Section IV, the method of constructing a model of the normal dataset using a Markov based approach is introduced. The test-bed used is then presented in Section V along with its evaluation. Finally, Section VI summarises the paper with a discussion and conclusion.

II. COGNITIVE OVERVIEW

The objective of the RUBICON Cognitive Layer is to implement a cognitive model which reflects the dynamics of the ecology. There are three core modules in the Cognitive Layer: (i) a cognitive memory module, (ii) a cognitive reasoning module and (iii) a cognitive decisions module. The cognitive memory is responsible for holding current and historical states of the RUBICON ecology as perceived and processed by the Learning, Control and Cognitive layers. The reasoning module reasons across current event data from the Learning Layer to determine the present state of the RUBICON ecology, while the decisions module decides what goals to set in response to the reasoning output, taking into account recent and longer term history. An overview of this architecture is shown in Figure 2.

![High level cognitive architecture](image)

Both the reasoning and decisions modules incorporate a self-organizing fuzzy neural network (SOFNN) as a learning component [12,13], the rules of which explore the relationship between the inputs and the desired reasoning/decision outputs. The decision outputs are presented in the form of goals to be carried out by the Control Layer.

Novelty detection is proposed to add a further feature of adaptation to the cognitive system by allowing it to learn what is expected in terms of activities from the Learning Layer (via the Cognitive memory) and act on any information that is deemed to be a deviation from the model of the normal dataset. In this way, the aim is for the system to attempt a previously unused goal on detection of such a deviation. However to enable this adaptive behaviour, activities should be continuously monitored and memorised to identify deviations, which in a sensor rich environment such as RUBICON can result in memory bloat; this is addressed using a memory management strategy described in Section III.

III. MEMORY MANAGEMENT

The novelty detection algorithm is designed to accumulate user behavioural information continually which is used to constitute a daily model and, if required, a weekly model. The daily model is representative of patterns that are repeated every day whilst the weekly model is more specific to a particular day within the week. One challenge associated with the cognitive memory module is how to determine when a change in user routine becomes incorporated into the reasoning system. The point at which the state is embedded in the model is decided via a process similar to a form of cognitive habituation of neural responses to novel information versus repeated information [14]. The result of this process is a habituation value ($H_n$) which is associated with each state discovered and it is the deciding factor not only for the state addition to the model but its deletion as well. $H_n$ is introduced with a default value when a new state is first discovered, with each state being a unique snapshot of activated events coming from the Learning Layer (explained further in Section IV). It is then manipulated via two separate models which control the growth or decay of the value, dependent on how long (in terms of days) since the the state has or has not been seen.

A. Growth Model

The growth model, stemming from the ‘habituation of neural responses’ approach [14], can be represented via a trapezoidal function, shown in Figure 3, which depicts the current Habituation Value ($H_n$) plotted against a dopamine boost parameter ($D_b$).

![Habitual Growth Function](image)

Dopamine is a very important neuromodulator which has been shown to be an integral part of the learning process [15,16,17], playing parts in both behavioural control and decision making. Hence, within the growth model dopamine...
is used as a parameter to modify the habituation value based on the current conditions. The purpose of $D_b$ is to act as an additive to $H_c$ rather than a direct influence. This allows for a parameterised growth level and can be shown by:

$$H_n = D_bK + H_c$$

where $K$ ($=0.4$) is a constant equal to the maximum boost required for the model. A newly discovered state is then influenced in the following way: if a new state is initiated, the habitual response is a slow incline to such times as the state is considered ‘novel’. At this point the state is added to the generalised daily routine within the system and the relevant information is passed to the decision module. Further encounters serve to solidify the state’s presence within the model, though at the same time its habitual response starts to reduce. This is representative of novelty starting to wear off with frequent instances of the same experience. After several repeated encounters growth levels start to decline, representing a natural decline in novelty [14]. When $H_c$ reaches a level which is considered saturated (i.e. $H_c = 1$), $D_b$ no longer has any effect. This follows a pattern similar to biological processes where the level of novelty is habituated [14,16].

$D_b$ can be represented by the following equation:

$$f(D_b) = \begin{cases} 0 & H_c \leq a_x \\ \beta_x(H_c - a_x) & a_x < H_c < a_s \\ \frac{(\beta_x - \beta_s)(H_c - a_s)}{(a_s - a_x)} + \beta_s & a_s \leq H_c \leq a_3 \\ \frac{(\beta_x - \beta_3)(H_c - a_3)}{(a_3 - a_s)} + \beta_3 & a_3 < H_c < a_4 \\ \frac{(\beta_4 - \beta_3)(H_c - a_4)}{(a_4 - a_3)} + \beta_4 & a_4 \leq H_c \leq a_5 \\ \frac{(\beta_5 - \beta_4)(H_c - a_5)}{(a_5 - a_4)} + \beta_5 & a_5 < H_c < a_6 \\ \beta_6(a_x - H_c) & a_6 \leq H_c \leq a_7 \\ 0 & a_7 < H_c \end{cases}$$

where $H_c$ is the current habituation value, $a_x$ and $\beta_x$ are configurable options (See Figure 3 for example values) which control the transitional incline/decline points of the growth levels for the model.

B. Decay Model

The state will remain at a saturated level for as long as it is encountered on a daily/weekly basis. During periods when the state is not encountered, it will enter into a decay mode. In keeping with the natural way humans forget things through a process known as synaptic decay [18], the decay model is introduced to constitute a ‘forgetting factor’ in the system, to constantly free up any space from states which have not been witnessed for several days. Typically, the process can be represented by an exponential decay curve, though in this case a biphasic exponential decay curve (shown in Figure 4) is used to define a two-part process which is more representative of both short-term and long-term memory. The initial degradation of the state’s value within the system represents the time leading up to its removal from the model. Thus, the time from the transitional point represents the time leading up to the state’s deletion from the complete system. The exponential curves are signified by:

$$H_n = H_c\left\{\begin{array}{ll} e^{k_1t} & t \leq t_c \\ e^{k_1t+c} & t > t_c \end{array}\right.$$  

where $t$ is the period of time that the state has been absent from the user’s routine (measured in days or weeks depending on the model), $k_1$ and $k_2$ represent two different exponential rate constants which follow first order kinetics and $t_c$ is the transition that occurs between them (for the graph $t_c = 3, k_1 = -0.25, k_2 = -1.75$).

C. Combined Model

The system combines the growth and decay models and decides which model should be utilised for a state as follows:

$$f(H_n) = \begin{cases} D_bK + H_c & x = 1 \\ H_c\left\{\begin{array}{ll} e^{k_1t} & t \leq t_c \\ e^{k_1t+c} & t > t_c \end{array}\right. & x = 0 \end{cases}$$

where $x$ is the activity of the current state under review for that day (1 = Active, 0 = Inactive). The current settings of these two models combined insures that memories (states) that were learned but not completely forgotten will be relearned much faster than completely new experiences (new states). This particular approach to remembering and forgetting in novelty detection contributes to two aspects. In terms of hardware, system memory becomes managed and controllable, therefore continual learning is possible without the threat of overloading the system. Secondly, the methodology represents simplified human behavioural learning in a phenomenological manner therefore reducing the computational overhead required.

IV. MODEL OF THE NORMAL DATASET

A Markov-based approach is adopted to model the generalised daily routine of the user, as this is suited to both novelty detection [9,19,20,21] and modelling sequences of events [22] (or observations) that occur one after the other. Events in this case are the user activities that are communicated to the Cognitive Layer from the Learning Layer with each state (event) of the model being an individual activity or collection of activities. Consider the simplified example, in Figure 5, of a representative user routine:
As described in Section II, a Markov system is in state $i$, there is a fixed probability $P_{ij}$ of it going into state $j$ in the next time step. $P_{ij}$ can alternatively be referred to as the transition probability and can be denoted by:

$$P_{ij} = Pr(X_n = j|X_{n-1} = i)$$

The total probability of a path is the product of the probabilities on the arcs that make up the path so for the example shown in Figure 5, these are:

$$P(h, e, c, r, s) = P(h, e, c, w, s) = 0.4$$
$$P(h, cl, c, r, s) = P(h, cl, c, w, s) = 0.1$$

A transitional matrix (Table I) is then derived using the collation of these probabilities and, in essence, this matrix represents the standard or generalised user model as a basis for detecting novel changes.

Using the information presented in Table I, it can be shown that the most probable chain of events can be represented by Figure 6.

Using the information presented in Table I, it can be shown that the most probable chain of events can be represented by Figure 6.

Figure 6: Most probable path for example user routine

Anything significantly different or outside of this routine can be considered novel resulting in the system generating information that aids the decision process. To handle scenarios where two or more events are occurring simultaneously, the model employs the use of ‘super-states’ [23] which consider each of the simultaneous events as sub-states, thus simplifying the model. For example, each state is created on the fly when it is first encountered by the system though it is not added to the model until this experience is repeated on several occasions (as described in Section II) thus excluding occasional deviations from the routine which are not consistently sustained.

On detection of any of the three main modalities of change: addition of a new state (event) to the model, deletion of a state from the model and the re-ordering of the most probable path, the system is prompted into a type of exploratory mode where it attempts an alternate goal from the Cognitive Layer’s list of potential goals, particularly those which have yet to be activated. If a repeatable change occurs on a daily basis it adapts the generalised daily model, otherwise it updates or creates a weekly model pertaining to the specific day that the change has occurred. The system then sends the goal to the Control Layer and awaits feedback as to whether the goal has been successful or not. Positive feedback in this scenario results in the goal being permanently activated for the recorded behavioural change of the user. If the feedback is negative, the goal is rejected.

V. PERFORMANCE EVALUATION

A. Test-bed

To evaluate this concept of novelty detection, a test-bed simulating an AAL environment was utilised (Figure 7) within the Intelligent Systems Research Centre (ISRC) at the University of Ulster (UU). In this scenario, UU focused on a situation where a change in a user’s behavioural patterns would be detected. This presented itself in the form of a night where the user would avail of a takeout option (pizza delivery) in place of his/her normal food cooking activity. The focus of this scenario is to demonstrate that the RUBICON ecology can repurpose its instruments to set new goals based on information that it has perceived. Should the goal be successful, the new goal becomes a permanent feature of the ecology.

The task involved setting the scene of a users’ daily routine from Monday through to Thursday as a base (normal) model and then to detect changes from this pattern. To do this, a synthesised routine was created which utilised the events and goals listed in Table II.

Using the information presented in Table I, it can be shown that the most probable chain of events can be represented by Figure 6.

Table I: TRANSITIONAL MATRIX FOR EXAMPLE USER ROUTINE

<table>
<thead>
<tr>
<th></th>
<th>home</th>
<th>clean</th>
<th>exercise</th>
<th>cook</th>
<th>relax</th>
<th>wash dishes</th>
<th>sleep</th>
</tr>
</thead>
<tbody>
<tr>
<td>home</td>
<td>0</td>
<td>0.2</td>
<td>0.8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>clean</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>exercise</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cook</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>relax</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>wash dishes</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>sleep</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 7: AAL environment test-bed

The task involved setting the scene of a users’ daily routine from Monday through to Thursday as a base (normal) model and then to detect changes from this pattern. To do this, a synthesised routine was created which utilised the events and goals listed in Table II.

Table II: List of events/goals used in the test-bed scenario (columns are unrelated)

<table>
<thead>
<tr>
<th>Events</th>
<th>Active Goals</th>
<th>Inactive Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>User leaves</td>
<td>Attend door</td>
<td>Bring item to user</td>
</tr>
<tr>
<td>Door bell</td>
<td>Bring drink</td>
<td></td>
</tr>
<tr>
<td>Item: Letter</td>
<td>Attend drip</td>
<td></td>
</tr>
<tr>
<td>Item: Pizza</td>
<td>Attend fire</td>
<td></td>
</tr>
<tr>
<td>User arrives</td>
<td>Bring item to table</td>
<td></td>
</tr>
</tbody>
</table>
Events within the first three days culminate in the creation of a Markov chain that is representative of the user routine for a generalised day. This formulation precedes the novelty detection system activating its exploration mode. Upon a persistent change to this routine, the novelty detection system will firstly modify the chain to be more representative of the activity and secondly, activate the exploration mode which should attempt a new goal based on the current snapshot of the RUBICON world.

For initiation of the generalised model, the reasoning and decision modules are trained to assign the ‘attend door’ goal on the occasion of the doorbell ringing. The PR2 is configured to approach the door and extend its gripper in anticipation of a delivery as a default action. If no item is given, it simply returns to its starting location (bedroom). If on the other hand, an item is placed within its gripper its directive, upon item identification, is to deliver the item to the hall table (‘Bring item to table’ goal) regardless of the item type. The item identity is presented to the novelty detection system in the form of an event and thus will eventually become part of the routine.

A PR2 robot [24] was selected for use in the evaluation. To aid in this demonstration, UU employed the use of the PR2 robot’s vision system. The capability to visually differentiate between objects at the door upon collection is provided using Speeded Up Robust Features (SURF) and a k-Nearest Neighbour (kNN) classifier (see Figure 8). In a real AAL environment with many objects to be identified, more complex classifiers such as support vector machine (SVM) or neural networks may be used.

The routine instigated by the user followed a simplified pattern of events which were repeated (simulated) over several days (Mon – Thurs). These consisted of:

- The user leaving in the morning to go to work.
- Whilst at work, the post was delivered and the robot was expected to attend to the door, retrieve the post and place on the hall table
- In the evening, the user would return, prepare his evening meal, wash up and then relax in the living room before going to bed.

This pattern continues for a long time until at some stage the user decides to order a pizza instead of cooking a meal. This occurs on a Friday and so the routine changed slightly in that the user ordered a pizza in lieu of preparing a meal and ate it whilst watching TV in the living room. This is the change in the user behaviour to which the ecology should respond.

The RUBICON system is designed to work with sensors embedded in the environment for the identification of events. For simplicity, within the UU test-bed, a VICON motion tracking system (partly labelled in Figure 7) is utilised as a method of inferring user location and current events. With the VICON identifiable markers placed on various items within the environment, including the user, the capability to track the user’s location with respect to the kitchen, kitchen table and food items becomes possible. It is therefore conceivable to infer that the user is eating at the kitchen table whilst all three markers are within the same location. Similarly, other events such as the user relaxing on the sofa can be inferred using the same principles.

B. Results

For the initial postal delivery, the RUBICON system responds as expected and the ‘letter’ is delivered to the hall table on each occasion. Within the first two instances of the routine, the live events are introduced to the system but not yet learned. Only after a repeated number of occurrences, in this case three, does the system formulate a pattern of events. On the third occurrence, the system learnt the pattern and a generalised daily model was created, which is shown in Figure 9.

The system has now a baseline with which to compare deviations. On Friday when such a deviation occurred, in the form of a pizza delivery, the PR2 delivered the pizza to the hall table, as this is its default directive. The user had to collect the item and return to the living room to eat it. On the third consecutive encounter of the pizza delivery over a three week period (simulated), a specific Friday model was created (Figure 10) distinct form the daily model and the system was prompted into exploration mode due to the new sequence of events now present.

A new behavioural model, in this case the “Friday” model, triggers the RUBICON system to attempt a new goal. For the phone, relax and doorbell events, nothing is done as they are not item related events. However, for the ‘Item: Pizza’ event, the system correlated this information with the only inactive goal ‘Bring item to user’. On presenting a goal to the Control Layer, the system awaits feedback from the Control Layer before the goal is considered learned. If positive feedback is received, the system adapts to this new change in user behaviour and the goal is made permanent and associated with a pizza delivery scenario. On all future occurrences, the RUBICON systems now correlates the ‘Item: Pizza’ event with the ‘Bring item to user’ goal as part of its reasoning and decisions process.
VI. DISCUSSION AND CONCLUSION

In this paper, an approach to model user behavioural patterns in an AAL application using Markov Decision Processes (MDPs) has been proposed. This is supported by a data retention strategy based on human behavioural learning principles to efficiently manage memory within the system. Within the experimentally evaluated scenario, the creation of the daily model which represents a user’s day-to-day routine is demonstrated and then used to show the initial creation of one instance of a weekly model which pertains only to Friday.

Evaluation shows that a heuristic approach, both compact and memory efficient, which encapsulates a user’s routine does provide a good basis for detecting change. When such a change happens, but only on Fridays (in this example), the system learns at a slower pace in terms of weeks rather than days. The importance of this is that the system can learn to distinguish between a daily routine and a weekly routine, i.e. it learns that the pizza delivery happens only on Fridays so it creates a new specific model pertaining to Fridays only. Further changes to the Friday model can only take effect after a persistent change on a weekly basis. The value of this persistence means that should the user deviate from a routine once, i.e., visitors call round or he/she decides to go out for a night, these temporary deviations do not alter the models. This affords the behavioural detection system a level of robustness in being able to cope with spontaneous acts of the user.

As this is a simplified scenario to capture and act on simple changes in user behaviour, implications of possible deviations over longer periods (i.e. holidays) are not considered. It must also be noted that the system is limited to unique sequential states in that revisiting a certain state may cause a negative impact on the model performance. For example, if a user were to relax on coming home, get up to prepare a meal and return to relaxing, the model could (providing both states were identical) contain an infinite loop of ‘relax’ $\leftrightarrow$ ‘prepare food’. To prevent this, future work will involve the implementation of a Markov model with memory to handle more complex user behavioural scenarios by increasing the complexity of each state within it.

Finally, with respect to future work, both the goals and memory management strategy will be expanded. In terms of goals being sent to the control layer, parameterisation and classification of goals would permit a larger range of goals for the system to choose from (i.e. item based, interaction based, service based etc…). This would require significant work in developing a method of selection based on the current state of the environment which would stem from the existing SOFNN approach detailed in [12]. With respect to memory management, by utilising the growth model, the system learns over time which reduces the accumulation of information that is not a repeatable pattern. This is expandable in terms of its parameterised (currently fixed) learning and forgetting rates which could later incorporate a dynamic rate that may be more representative in biological terms.

ACKNOWLEDGMENT

This work is partially supported by the EU FP7 RUBICON project (contract no. 269914) – www.fp7rubicon.eu.

REFERENCES