Memory Formation, Consolidation, and Forgetting in Learning Agents

Budhitama Subagdja,
Wenwen Wang, Ah-Hwee Tan
School of Computer Engineering,
Nanyang Technological University
Nanyang Avenue, Singapore 639798
{budhitama,wa0003en,asahtan}@ntu.edu.sg

Yuan-Sin Tan, Loo-Nin Teow
DSO National Laboratories,
20 Science Park Drive, Singapore 118230
{tyuansin,tloonin}@dso.org.sg

ABSTRACT

Memory enables past experiences to be remembered and acquired as useful knowledge to support decision making, especially when perception and computational resources are limited. This paper presents a neuropsychological-inspired dual memory model for agents, consisting of an episodic memory that records the agent's experience in real time and a semantic memory that captures factual knowledge through a parallel consolidation process. In addition, the model incorporates a natural forgetting mechanism that prevents memory overloading by removing transient memory traces. Our experimental study based on a real-time first-person-shooter video game has indicated that the memory consolidation and forgetting processes are not only able to extract valuable knowledge and regulate the memory capacity, but they can mutually improve the effectiveness of learning the knowledge for the given task in hand. Interestingly, a moderate level of forgetting may even improve the task performance rather than disadvantaging it. We suggest that the interplay between rapid memory formation, consolidation, and forgetting processes points to a practical and effective approach for learning agents to acquire and maintain useful knowledge from experiences in a scalable manner.

Categories and Subject Descriptors

I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*

General Terms

Algorithms, Design, Experimentation, Theory

Keywords

Memory, Forgetting, Adaptive Resonance Theory

1. INTRODUCTION

Memory plays a key role in reasoning and decision making by providing past relevant episodes to improve learnt knowledge [10]. An agent with very limited or partial observability can construct a complete picture about its task environment by remembering all relevant information from its memory. Some agent architectures have incorporated declarative memory systems to support different aspects of the agent's performance. For example, [10] and [6] demonstrate the use of episodic memory to improve task performance and survivability of agents in simulated environments. [2, 8, 1, 7] also show how memory can improve the realism or human-likeliness of virtual agents. Most of these architectures consider declarative memory as a flexible information storage that can perfectly store and accurately retrieve information.

Although episodic memory can provide useful information about previous experiences at any moment of time, a significant amount of computational resources may still be needed to process specific items in memory to support reasoning and decision making. As the tasks and the environment become more complex, it is often impossible to make use of all the stored information necessary to make the right decision. For example, in a real-time environment, an agent may not have enough time to search and recall all relevant information to support its decision while it is performing some tasks. On the other hand, the agent may need to obtain enough information to acquire more compact, generalized, and efficient knowledge structure through a particular learning algorithm in order to make a timely and appropriate decision. Furthermore, most existing memory architectures for agents assume that all stored information is always relevant and consistent despite its limited capacity and possible erroneous inputs. The limitation of memory to keep all relevant and useful information is an important practical issue for learning agents but still seldom concerned.

According to neuropsychology, human memory systems have been known to be as non-unitary and partitioned into different types. Long-term declarative memory has been divided into episodic memory enabling one to remember personal experiences in specific manner and semantic memory that stores concepts, rules, and general facts [15]. It has also been considered that a consolidation process gradually transfers the specific items from episodic memory into general facts and rules in semantic memory [15, 3]. Episodic memory (particularly located in hippocampal area in the brain) and semantic memory (distributed across neo-cortical area in the brain) are anatomically separated but interconnected. It is suggested that these complementary memory systems prevent interferences of new information to old memories [9]. The consolidation between the two memory systems also implies that some forgetting processes may reg-

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

ulate and shape the memorized items to optimize performance as if the brain minds about the usefulness of memory traces rather than fidelity [12, 17]. It may be the case that perfect memory retrieval would burden the brain with too much details at the expense of remembering useful information.

In this paper, we propose a computational model of such a dual memory system based on a composition of fusion Adaptive Resonance Theory (ART) neural networks [14], which inherently serve learning, categorization, and recall operations. The dual memory model consists of an episodic memory that records agent's experience in real time and a semantic memory that captures factual knowledge relevant to the environment. The memory system is designed in such a way that the agent can store, recall, and playback experienced episodes in a sequential manner beyond individual momentary events. The availability of both the episodic and semantic memory modules enables a resource-bounded agent to focus on the situation and tasks in hand by capturing the episodic experience on-the-fly into a temporary memory store and deferring the resource-intensive learning of factual knowledge to a later stage through a memory consolidation process from episodic to semantic memory. In addition, the use of episodic memory as a buffer allows an agent to select and iterate through the relevant past cases as many times as needed at a later time, based on the updated awareness of the tasks and environment. This naturally leads to more effective learning comparing with learning the knowledge in an online manner while performing the task. To prevent the episodic memory from overloading, we further present a forgetting mechanism that associates each memory category with a time-decaying memory strength and removing unimportant traces or categories from the model.

The proposed dual-memory model has been evaluated using a real-time first-person-shooter video game called Unreal Tournament to support a non-player-character (NPC) agent to learn from experiences and improve performance. Surprisingly, we find that the memory model not only improves the task performance but in some cases, a moderate level of forgetting even results in more effective learning. Further examinations on the effects of forgetting show that selecting and pruning erroneous and outdated patterns promotes more efficient and robust learning.

The rest of the paper is organized as follows. Section 2 reviews related works in modelling declarative memory for autonomous agents and discusses some computational and neuropsychological accounts of consolidation and forgetting processes. The paper continues to describe the proposed memory model in Section 3. The section also formulates and analyses the characteristic of the memory model. Section 4 describes the implementation and the experiments of the memory system as parts of the non-player character agent in Unreal Tournament. The last section concludes the paper with some future directions.

2. RELATED WORK

Various types of declarative memory have been devised in recent years as parts of agent architectures. Most of them are developed as unitary systems, serving mainly the functionality of episodic memory to store linearly ordered traces of experiences or log records. This kind of sequential records has been demonstrated to optimize the agent's task performance [10] and improve the survivability of artificial life beings in virtual environment [6]. In the sequential traces model of episodic memory, specific mechanisms to search, retrieve, and recall the appropriate memory items are necessary to support performance. Each entry in episodic memory may also refer to other cognitive traits like procedural knowledge (in SOAR) [10] or emotions [1] to enhance the capability of memory operations and support complex behaviour.

To improve realism and interactivity, episodic memory entries can be associated with temporal relationships. For example, each entry can be associated with temporal weights to produce the effect of recency [8, 7] or generating temporal granurality [2]. This temporal association can emulate forgetting in which some details of memory item are suppressed according to time. However, these architectures still exclude memory consolidation processes to transfer and reorganize the contents of memory, nor consider the forgetting as beneficial to the overall performance.

On the other hand, it has been known that information memorized in the brain are subject to consolidation to make more general experience-independent forms of knowledge [3]. Memory traces from past experiences are played back before performing actions in similar or relevant contexts [11, 5]. Meanwhile, forgetting may take place removing or suppressing less useful memory items [17]. This forgetting dynamics of memory has been recognized as computationally beneficial to reduce inconsistency and the complexity of reasoning by discarding irrelevant information [16]. In multiagent systems, forgetting can also be useful for resolving conflicts between agents [4].

Our work in this paper also explores the role of episodic memory in agents. However, the main focus is to look at the memory consolidation process to extract useful knowledge and the discard of irrelevant information through forgetting. Instead of just applying episodic memory as a unitary flexible storage system, we make dual episodic-semantic memory systems working together to acquire useful knowledge through the interplay of consolidation and forgetting processes.

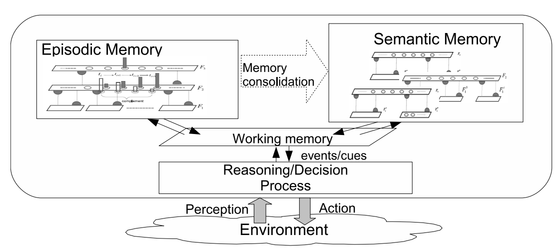

Figure 1: The Episodic-Semantic Memory

3. THE DUAL MEMORY MODEL

The proposed memory model is considered as a part of the reasoning system of an agent architecture (Figure 1). It can be assumed that at each point in time, a snapshot of an agent's situation and perception can be encoded as an individual event and held temporarily in working memory. Episodic memory automatically stores and organizes the events in a sequential order into cognitive units of episodes, which are then periodically transferred to semantic memory.

In general, the memory system goes through different stages of operation as follows:

- Events captured in working memory are continuously stored in episodic memory.
- Periodically, traces in episodic memory are readout to working memory triggering learning in semantic memory.
- Memory items in episodic and semantic memory can any time be recalled based on certain memory cues through a process of pattern completion or reconstruction.
- Rarely accessed items in episodic memory will be removed (forgotten).

The remaining parts of this section first formulate the notion of events and episodes before presenting the details of the model and process as follows.

3.1 Events and Episodes

An *event* is a snapshot of perceived experience at one moment in time which can be defined as a collection or a tuple of attributes.

Definition 1. An event ε is a tuple reflecting a moment of experience such that $\varepsilon = (\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_k)$. Each attribute $\mathbf{v}_i$ is defined as a tuple such that $\mathbf{v}_i = (v_1^i, v_2^i, .., v_l^i)$ and v_j^i is a normalized real value $v_j^i \in [0, 1]$.

By normalizing the value of v_j^i, an event can represent a proposition with a binary truth at the extreme values (0 or 1) or a certain degree in between (fuzzy values). To recall a stored event, an *event cue* can be expressed with the same tuple structure. However, some elements of the cue may be left unspecified and the recall operation would reconstruct the target event. Figure 2(a) shows an example of an event representation in the Unreal Tournament domain used in our experiment (explained later in detail).

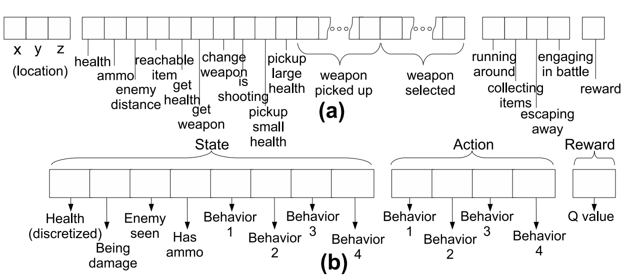

Figure 2: (a) Event Encoding; (b) Input (Output) representation of *RLBot* NPC in the experiment

An *episode*, on the other hand, can be defined as a finite list of events collected in a temporal order.

Definition 2. An episode E is a sequence of events such that $\mathrm{E} = [\varepsilon_{t_0}, \varepsilon_{t_1}, ..., \varepsilon_{t_n}]$, where t_j denotes the relative time point wherein the event ε_{t_j} occurs.

In contrast to the event structure, which always has the same length of tuple, the length of the episode sequence may vary.

3.2 Building Blocks

The proposed memory model is based on fusion Adaptive Resonance Theory (ART) [14], which applies a myriad of learning paradigms to recognize and learn an incoming stream of input patterns across multiple channels in real time. It employs a bi-directional process of categorization and prediction to find the best matching category (resonance). It also learns continuously by updating the weights of neural connections at the end of each search cycle. ART may also grow dynamically by allocating a new category if no match can be found. This type of neural network is chosen as the building block of our memory model as it enables continuous formation of memory with adjustable vigilance of categorization to control the growth of the network and the level of generalization. Specifically, a fusion ART model can be defined as follows:

Definition 3. Suppose F_1^k and F_2 are the kth input (output) field and the category field of fusion ART (Figure 3) respectively for $k = 1, .., n$. Let $\mathbf{x}^k$ denote the F_1^k activity vector and $\mathbf{w}_j^k$ denote the weight vector associating kth field with jth node in F_2. F_1^k is associated with choice parameter $\alpha^k > 0$, learning rate $\beta^k \in [0, 1]$, contribution parameter $\gamma^k \in [0, 1]$, and vigilance parameter $\rho^k \in [0, 1]$.

Definition 4. Choice function T_j returns the activation value of category j such that:

$$T_j = \sum_{k=1}^{n} \gamma^k \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{\alpha^k + |\mathbf{w}_j^k|} \tag{1}$$

where the fuzzy AND operation $\wedge$ is defined by $(\mathbf{p} \wedge \mathbf{q})_i \equiv \min(p_i, q_i)$, and the norm $|.|$ is defined by $|\mathbf{p}| \equiv \sum_i p_i$ for vectors $\mathbf{p}$ and $\mathbf{q}$.

Definition 5. Template matching m_j^k is the matching value or similarity of category j with the input $\mathbf{x}^k$ such that:

$$m_j^k = \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{|\mathbf{x}^k|} \tag{2}$$

A category J of F_2 field is in *resonance condition* if and only if:

$$T_J = \max\left\{T_j : \forall k, m_j^k \geq \rho^k, \text{for all } F_2 \text{ category } j\right\} \tag{3}$$

Definition 6. Given the selected category J, *Template learning* modifies the weights associated with J such that:

$$\mathbf{w}_J^{k(\text{new})} = (1 - \beta^k)\mathbf{w}_J^{k(\text{old})} + \beta^k(\mathbf{x}^k \wedge \mathbf{w}_J^{k(\text{old})}) \tag{4}$$

The corresponding weight vector of the chosen F_2 node J can be readout into the input field F_1^k such that $\mathbf{x}^{k(\text{new})} = \mathbf{w}_J^k$.

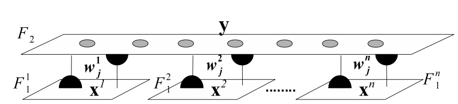

Figure 3: Fusion ART Neural Network

If no existing F_2 category can be found in resonance condition with the current input, a new category is recruited to represent the current input pattern. This implies that the ART network can grow to accommodate the incoming stream of different input patterns. The growth rate of the categories depends on how much the incoming patterns differ from one another and is adjustable through adjusting the vigilance parameters (ρ^k). Lower vigilance may tolerate differences more than the higher one and hence lead to a slower growth.

3.3 Episodic Memory

From *Definition* 1, it is clear that an event corresponds directly to the input vector representation in the fusion ART model. Specifically, an event ε^k is encoded into an input vector $\mathbf{x}^k$ to be learnt by fusion ART. On the other hand, based on *Definition* 2 an episode corresponds to the sequence of selected categories (events) collected in a temporal order. An episodic memory model can be built to store events and episodes by combining two fusion ARTs: one for storing events and the other for episodes. Figure 4 shows that events are represented and processed by the ART network between F_1^k and F_2 fields. In this case, an event can be learnt and retrieved by selecting a matching category in F_2 and readout the pattern back to F_1^k. In a word, each F_2 category j represents a single event.

To capture episodes, another layer of ART gets input from categories selected in F_2 from the other network. Let $\mathbf{y} = (y_0, y_1, .., y_m)$ denote the F_2 activity vector. If J is the F_2 category currently selected by the resonance search, then $y_J = 1$ and $y_j^{(new)} = y_j^{(old)}(1-\tau)$ for all F_2 category $j \neq J$ and $\tau \in (0, 1)$.

Lemma 1. *Consider the F_2 activity vector $\mathbf{y}$ as described above. If $y_J = 1$ for J is the currently selected category and $y_j^{(new)} = y_j^{(old)}(1-\tau)$ for all $j \neq J$ with $\tau \in (0, 1)$, then $\mathbf{y}$ reflects the relative order of category selection in F_2 such that $y_{j_t} > y_{j_{t-1}} > y_{j_{t-2}} > ..y_{j_{t-m}}$ for j_t is the node selected at relative time point t.*

Proof. Given $0 < (1-\tau) < 1$, it is clear that $1 > (1-\tau) > (1-\tau)^2 > .. > (1-\tau)^m$. Since $y_{j_t} = 1$ for any category j_t and from $y_j^{(new)} = y_j^{(old)}(1-\tau)$ it follows that $y_{j_{t-m}} = (1-\tau)^m$, consequently $y_{j_t} > y_{j_{t-1}} > y_{j_{t-2}} > .. > y_{j_{t-m}}$. □

The list of events with their relative order expressed as $\mathbf{y}$ vector becomes the input to the upper ART network (between F_2 and F_3 fields) to be learnt as an episode. Based on the bi-directional activation and matching process in ART, a category I representing an episode is selected in F_3 such that

$$T_i = \frac{|\mathbf{y} \wedge \mathbf{w}_i|}{|\mathbf{w}_i|},\ m_i = \frac{|\mathbf{y} \wedge \mathbf{w}_i|}{|\mathbf{y}|}, \text{and}$$

$$T_I = \max\left\{T_i,\ m_i \geq \rho_2, \text{for all}\ F_3\ \text{node}\ i\right\}.$$

Parameters α, γ, and the field index k are omitted as the upper network only has a single input field (F_2). Given the selected category I, learning takes place such that $\mathbf{w}_I^{(\text{new})} = (1-\beta_2)\mathbf{w}_J^{(\text{old})} + \beta_2(\mathbf{y} \wedge \mathbf{w}_I^{(\text{old})})$. ρ_2 and β_2 are the vigilance and learning rate parameters respectively of the field F_2.

If consecutive events are received in F_1^k, a pattern of their sequence can be formed in F_2 as vector $\mathbf{y}$ and can be used as an input to select category I in F_3. In this case, the input patterns act as memory cues. To reproduce the original sequence of events of the episode, two stages of readout

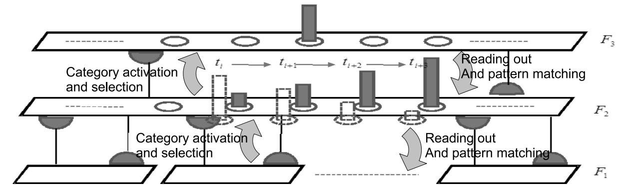

Figure 4: The Episodic Memory Model

operation are conducted by firstly reading out the sequential pattern of the episode into vector $\mathbf{y}$ and secondly an F_2 category J is selected such that $T_J = \max(\overline{y}_j : \overline{y}_j = 1 - y_j, \text{for all}\ F_2\ \text{category}\ j)$ and readout to the corresponding F_1^k before reset to zero or $y_J = 0$. The readout cycles continue until $\mathbf{y} = \mathbf{0}$.

3.4 Semantic Memory

Different from episodic memory, we view that the semantic memory is not unitary, with different fusion ARTs representing different structure of knowledge. In contrast to episodic memory, each entry in semantic memory generalizes similar inputs into the same category rather than as separate entries. Each input field of a semantic memory represents a property or an attribute of a concept. The generalization can be achieved by lowering the vigilance parameter ρ^k so that slightly different input patterns will still activate the same category. The value of an attribute can be paired, as described below, so that the ART learning can generalize the value as a range of values.

Let $\mathbf{I}^k$ be the input vector for $F_1^k, I_i^k \in [0, 1]$. $\mathbf{I}^k$ is augmented with $\bar{\mathbf{I}}^k$ such that $\bar{I}_i^k = 1 - I_i^k$. The activity vector $\mathbf{x}^k$ of F_1^k thus augments the input vector $\mathbf{I}^k$ with its complement $\bar{\mathbf{I}}^k$ which are learnt as a $\mathbf{w}_j^k$. Let $(w_{ij}^k, \overline{w}_{ij}^k)$ be the corresponding pair of $\mathbf{w}_j^k$. The value of the connection becomes less specified when $w_{ij}^k \neq 1 - \overline{w}_{ij}^k$.

Lemma 2. *For the pair $(w_{ij}^k, \overline{w}_{ij}^k)$ of $\mathbf{w}_j^k$ described above, if $w_{ij}^k \neq 1 - \overline{w}_{ij}^k$, any corresponding complemented input pair $(I_{ij}^k, \overline{I}_{ij}^k)$ will have a maximum matching value or always in resonance as long as $w_{ij}^k \geq I_{ij}^k \geq 1 - \overline{w}_{ij}^k$.*

Proof. Let the pair $(x_{ij}^k, \overline{x}_{ij}^k) \equiv (I_{ij}^k, \overline{I}_{ij}^k)$. It is clear that if $x_{ij}^k \leq 1 - \overline{w}_{ij}^k$ then $\overline{x}_{ij}^k \leq \overline{w}_{ij}^k$. Thus, $(x_{ij}^k, \overline{x}_{ij}^k) \wedge (w_{ij}^k, \overline{w}_{ij}^k) = (x_{ij}^k, \overline{x}_{ij}^k)$ such that *template matching* $m_j^k = \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{|\mathbf{x}^k|} = \frac{|\mathbf{x}^k}{|\mathbf{x}^k|} = 1$ □

It can be considered that a stored value is unspecified if the values of the corresponding complementary pair $(w_{ij}^k, \overline{w}_{ij}^k)$ are equal.

Similar to episodic memory, the content of semantic memory can be retrieved by pattern completion based on memory cues. Figure 5 illustrates various types of semantic memory. A single fusion ART may consist of domain specific associative rules (e.g a set of association between a certain object and its location in the environment, a set of rule associating the effectiveness of a certain weapon and the distance to the opponent) or generic causal relations associating a particular type of event to another that follows. These types of semantic knowledge can be derived by exposing the played back items from the episodic memory to the input of the semantic memory using a lower vigilance parameter ρ^k and a smaller learning rate β^k such that similar instances may gradually be clustered together regardless of their order.

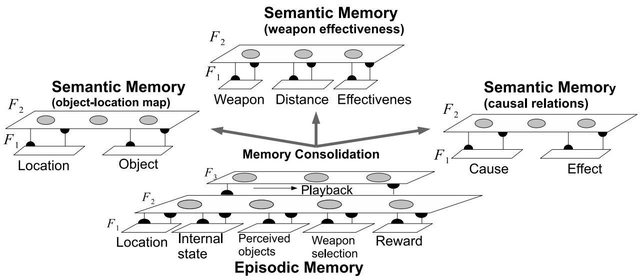

Figure 5: Different types of Semantic Memory and the Memory Consolidation Process

3.5 Memory Consolidation

Knowledge can be transferred from episodic memory by playing back stored episodes or reading out category nodes in F_3 field to each corresponding input fields. The readout events are then passed to the working memory, one at a time, to be shared by other memory modules. Depending on the kind of semantic structure and the domain problem, different subsets of items in working memory are connected to different input fields in semantic memory.

Definition 7. Let $\mathbf{O}$ be the vectors in F_1 fields read out from the selected category in episodic memory as a result of the playback process. Function $\mathcal{M} : \mathcal{P} \rightarrow \Theta$ maps the read out vectors from episodic memory into the corresponding input vectors of semantic memory, wherein $\mathcal{P}$ and Θ are the vector space of the input fields of episodic memory and semantic memory respectively.

During the episodic memory playback, $\mathbf{x}' = \mathcal{M}(\mathbf{O})$ becomes the input vector to learn in semantic memory. The played back vectors can be the results of the top-down readout process or it can be the results of a retrieval operation based on some memory cues. If $\mathbf{x}' = \mathcal{M}(\varepsilon)$, it can be said that semantic memory works standalone by learning directly from the received events ε bypassing episodic memory. A standalone version of semantic memory is also presented in this paper as one configuration to compare with the dual memory model in our experiment (explained later).

3.6 Forgetting

Forgetting is a mechanism to free up memory space in episodic memory by removing unnecessary items. Intuitively, a memory item can be considered unnecessary or obsolete if it is rarely accessed or recalled for long. The forgetting mechanism in the proposed model is applied to episodic memory for both the event layer (F_2) and the episode layer (F_3).

Definition 8. Given a category j (representing either an event or an episode) in a fusion ART structure, a memory strength $\mathcal{S}_j^t$ reflects how often category j is selected such that:

$$\mathcal{S}_j^t = \begin{cases} \mathcal{S}_j^{t-1} + (1 - \mathcal{S}_j^{t-1})r_s & \text{if } j \text{ is selected at time t} \\ \mathcal{S}_j^{t-1}(1 - \delta_s) & \text{otherwise} \end{cases} \tag{5}$$

where $r_s \in [0, 1]$ and $\delta_s \in [0, 1]$ are reinforcement and decay rate parameters respectively. A category j with $\mathcal{S}_j^t < \theta_s$ will be removed or pruned from memory including all the associated connections. If j is a new allocated category, it is assigned with an initial strength $\mathcal{S}_j^{init}$.

4. CASE STUDY

The episodic-semantic memory system is implemented and embedded into an autonomous non-player character (NPC) in a first-person-shooter video game called Unreal Tournament (UT). Our objectives are to test if the proposed memory model can produce useful knowledge for the agent and whether the forgetting process may sacrifice the agent's performance. The scenario of the game used in the experiment is "Deathmatch". The objective of each agent is to kill as many opponents as possible and to avoid being killed by others. In the game, two (or more) NPCs are running around and shooting each other. They can collect objects in the environment, like health or medical kit to increase its strength and different types of weapon and ammunition for shooting. The battle simulation in UT game is a suitable platform for evaluating memory tasks. Besides complex spatial maps and terrains, different objects and situations in the game may have some intricate relationships that should be memorized and remembered in non-trivial ways.

4.1 Weapon Learning Task

In the first experiment, we task the agent to learn the relationship between the type of weapon and its effectiveness to kill given the distance of the opponent agent. We compare the performance of different agents with reinforcement learning and the dual episodic-semantic memory.

4.1.1 The Baseline Agents for Comparison

All agents that we evaluate in the experiment play against an NPC agent called *AdvanceBot* that behaves according to hardcoded rules. There are four different hardcoded behavior modes in *AdvanceBot*: (1) Running around behaviour, in which the agent runs around exploring the environment randomly; (2) Collecting items behavior, in which the agent goes around and picks up collectible items; (3) Escaping from the battle situation, in which the agent turns and runs away from the opponent; (4) Engaging in battle, in which the agent approaches its opponent and shoots to kill it. *AdvanceBot* always chooses one of the four behaviors based on a set of predefined rules.

Under the battle engagement behavior, the agent also always tries to select the best weapon available for shooting. The weapon selection rules are based on some heuristics optimized for a certain environment map used in the game.

As a performance comparison, another agent (named *RL-Bot*) is made to employ the same set of behaviors but its selection is conducted dynamically based on a fusion ART neural network conducting reinforcement learning algorithm. The state, action, and reward vectors in Figure 2(b) correspond to the input fields in a fusion ART network of *RL-Bot*. The behavior pattern in the state vector represents the behavior (1 to 4) currently selected. The action vector indicates the next behavior to be selected. Based on the state field input and the reward cue (set to the maximum), the network searches the best match category node and reads out the output to the action field indicating the behavior type to be selected. The network then receives feedbacks in terms of the new state and reward (if any). The network learns by updating the weighted connections according to the feedback received and applying temporal difference methods [13] to update the reward field if the immediate reward is absent. The agent receives the reward signal (positive or negative) whenever it kills or is killed by another agent. In contrast to *AdvanceBot*, *RLBot* chooses an available weapon randomly in the battle engagement behavior. Another agent called *RLBot++* is also used to employ the same reinforcement learning model as *RLBot* but select the weapon based on the optimized predefined rules just like in *AdvanceBot*.

4.1.2 Episodic-Semantic Memory Based Agent

The proposed model is embedded in an agent with the same architecture as *RLBot*, but with the episodic and semantic memory modules running concurrently. The episodic memory captures episodes based on the event information in the working memory. An event from the UT game is encoded as a vector shown in Figure 2(a). There are four input fields in episodic memory for location, state, selected behavior, and the reward received. In the experiment, the vigilance of all input fields (ρ_e) and the F_2 field (ρ_s) are set to 1.0 and 0.9 respectively so that it tends to always store distinct events and episodes in response to the incoming events. At a certain period of time, the contents of the episodic memory is played back by reading out the events to the working memory. The reinstatement occurs in the period between different battles wherein one agent has just been killed and started to respawn in another place. The semantic memory then acquires the knowledge by learning from the recalled events. In the experiment, only one type of semantic memory about weapon effectiveness is learnt given the distance towards the enemy. Whenever the value of the reward field in the event vector is large enough to be considered as a successful killing (0.5 is the threshold), the values of weapon selected, opponent distance, and reward (or the effectiveness to kill) fields are fed and learnt by the semantic memory.

Figure 5 shows the fusion ART network of the semantic memory for weapon effectiveness used in the experiment. The network has three input fields: the Weapon field representing the identity of the weapon (F_1^a); the Distance field representing the distance between the agent and its opponent at the time of shooting (F_1^b); and the Effectiveness field representing the chance to kill the enemy (F_1^c). In the experiment, the vigilance of the Weapon (ρ^a), Distance (ρ^b), and Effectiveness (ρ^c) fields are 1.0, 0.9, and 0.8 respectively. The learning rate β^a, β^b, and β^c are 1.0, 0.1, and 0.2 respectively. The agent reasoning system can use the knowledge in the semantic memory by providing the current distance to the opponent while setting up the effectiveness to maximum (the greatest chance of killing) as memory cues. The retrieved values support the agent to decide which weapon to select during the battle. If the cue is not recognized, a random weapon is selected.

As a comparison, we also implement an agent with a standalone version of semantic memory as mentioned in the previous section. The agent, called *AssocBot*, learn the weapon selection knowledge by directly associating weapon and enemy distance from the incoming events without consolidating episodic memory or $\mathbf{x}' = \mathcal{M}(\varepsilon)$. This direct semantic memory only learn the event whenever the NPC shot hits the opponent.

Table 1: Sample Rules Learnt in Semantic Memory

`IF distance is not so far [1800 2099]`
`THEN ASSAULT_RIFLE effectiveness 0.07`
`IF distance is very near [300 599]`
`THEN ASSAULT_RIFLE effectiveness 0.048`
`IF distance is extremely near [0 299]`
`THEN SHOCK_RIFLE effectiveness 0.946`
`IF distance is very near [300 599]`
`THEN ROCKET_LAUNCHER effectiveness 0.932`

weapon range categorization: `extremely near:0-299;`
`very near:300-599;near:600-899;medium near:900-1199;`
`not so near:1200-1499;midrange:1500-1799;not so far:1800-2099;`
`medium far:2100-2399;far:2400-2699;very far:2700-2999;`
`exremely far:3000 or more`

Table 1 illustrates sample learnt rules of weapon effectiveness translated into symbolic forms. Each rule corresponds to a category node in F_2 layer of the semantic memory. The generalization employed using Fuzzy operators makes it possible to represent the rule with a range of values like the rule antecedents shown above. Table 1 also shows the symbolic categorization of the distance range for interpreting the rules. The experiment also uses forgetting in episodic memory with $\mathcal{S}_j^{init}$, threshold (θ_s), and reinforcement rate (r_s) set to 0.5, 0.0001, and 0.5 respectively. To evaluate the effect of forgetting, different decay rates (δ_s) in the events field F_2 are used: 0 (no forgetting), 0.005, 0.01, and 0.02.

4.1.3 Results

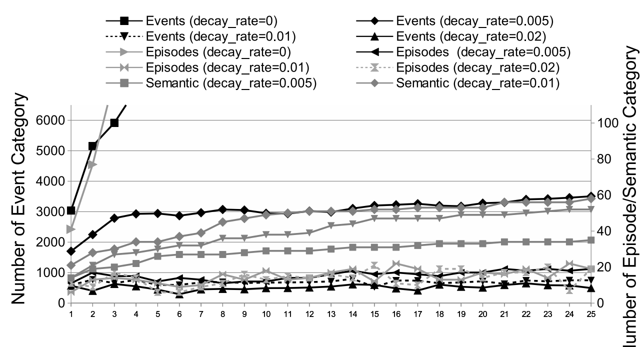

Figure 6: Memory usage for events, episodes, and transferred semantic knowledge with different forgetting decay rate during the game play

Experiments are conducted by letting *RLBot*, *RLBot++* and the memory-based *RLBot* (called *MemBot*) with different forgetting decay rates (δ_s=0.005, δ_s=0.01, and δ_s=0.02) to individually play against *AdvanceBot*. In addition, the direct semantic-memory-based *RLBot* (*AssocBot*) is also put to the test. For practical reason, *MemBot* without forgetting (δ_s=0) is excluded from performance comparison as the program overloads the system memory soon after the game starts causing the system to halt and the agent refrains from playing. A single experiment run consists of 25 games or trials, which is counted whenever the agent kills or is killed by another agent.

Figure 6 shows the memory size taken up in the episodic memory (in terms of the number of nodes in F_2 and F_3 of a *MemBot*) and the number of nodes created in the semantic memory with different δ_s in F_2 sampled from a single run against *AdvanceBot*. Without forgetting ($\delta_s = 0$), the memory space is taken up rapidly into its limit after about three trials. In contrast, the forgetting mechanism can make the memory size converge and stabilize at certain points. Hence the agent can always perform and learn continuously. It is clearly shown that the larger the decay rate, the smaller number of categories is produced in episodic memory. Interestingly, a low semantic memory decay rate (e.g δ_s=0.005) creates lesser categories comparing with those obtained with higher rates (e.g δ_s=0.01 and δ_s=0.02).

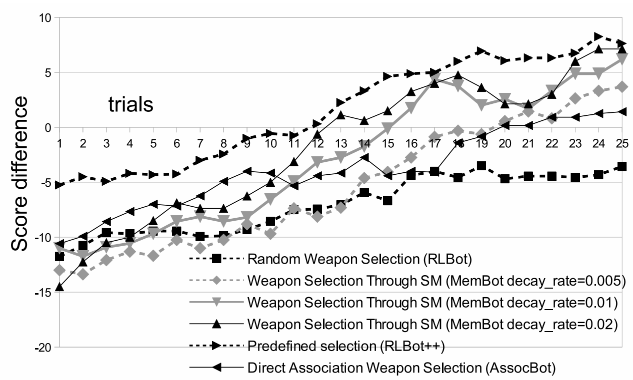

Figure 7: Performance of *RLBot*, *RLBot++*, *MemBot*, *AssocBot* over 25 trials

Figure 7 plots the performance of *RLBot*, *RLBot++*, *MemBot*, *AssocBot* with different δ_s in terms of game score differences against *AdvanceBot* averaged over four independent runs. It shows that incorporating the proposed episodic and semantic memory model improves the learning which results in a much better performance than using the reinforcement learning alone (with random weapon selection). This indicates that the semantic memory can learn useful knowledge about weapon selection. It is also shown that although *AssocBot* can learn and improve better than RLBot, it is still marginally inferior than MemBot which applies consolidation but with the smallest forgetting decay rate ($\delta_s = 0.005$). It indicates that the consolidation can be advantageous for a learning agent. The reason could be that, with consolidation, the same categories might be activated or selected in semantic more than once as the result of the playing back episodic memory. This reactivation of certain categories shapes and reinforces knowledge in semantic memory, whereas learning the semantic directly without consolidation may produce over-generalization.

Surprisingly, the results also indicate that with a higher forgetting rate (e.g δ_s=0.01 and δ_s=0.02), the performance and learning efficiency of MemBot are better than those obtained with the smaller one (δ_s=0.005) and can eventually reach the performance using the optimized rules model. In other words, forgetting less important things faster can make learning better. One explanation of this beneficial effect of forgetting is that events in the UT game related to weapon use are noisy and full of inconsistencies. Thanks to forgetting, events that could impair the consolidated knowledge are filtered out before being generalized in semantic memory. The semantic memory would thus end up with the appropriate generalization and some specific but necessary information.

4.2 Noisy Event Recognition Task

To validate our explanation about the significance of forgetting in noisy and inconsistent environment, we conduct another test to see how forgetting contributes to the retrieval accuracy in a situation where some noise distribution is introduced to the input events. The test is conducted off-line based on the recorded events of selected UT game sessions. The forgetting test is applied to episodic memory only.

4.2.1 Testing Configuration

The episodic memory model described above is made to learn from different sets of events recorded from a UT game session with the same event structure applied in the above experiment. The original set of events consists of 77350 events and 1000 game sessions. We assume that, the memory learns a sequence of events as a single episode at the end of each session. To simulate the noisy environment, different sets of events are generated by introducing two different rates of noises following a Gaussian distribution. Two sets of events are used with 5% and 10% noise rates.

Episodic memory learns each noisy data set to generate the representation of events and episodes in memory. The evaluation is conducted to obtain the retrieval accuracy by measuring the difference of the selected episodes with the ones selected by episodic memory that has learnt the original (without noise) data set. It should be noted that the accuracy is not measured based on recall or reconstruction of the original episodes but only based on the recognition of the presented episodes.

The retrieval cues used in the evaluation are one-fifth portions of the target episode taken from the original data set (without noise). We compare the retrieval accuracy of both episodic memory configurations that learn different data sets with and without forgetting. The parameters of event level forgetting used in the experiment are as follow: initial confidence $\mathcal{S}_j^{init} = 0.5$, decay factor $\delta_s = 10^{-4}$, reinforcement parameter $r_s = 0.5$, and threshold $\theta_s = 0.1$. For episode level forgetting: the parameter values are initial confidence $\mathcal{S}_j^{init} = 0.5$, decay factor $\delta_s = 0.008$, reinforcement parameter $r_s = 0.5$, and threshold $\theta_s = 0.1$. The vigilance parameters ρ for events and episodes level are 0.5 and 0.95 respectively.

4.2.2 Results

Figure 8 clearly shows that, in a noisy condition, forgetting helps episodic memory to retrieve the correct episodes despite some learnt events and episodes have been discarded from memory. In both 5% and 10% level of noises, the performance with forgetting is always superior to the configurations without forgetting. The results support our hypothesis that forgetting can contribute to the overall performance of the agent when information perceived from the environment are noisy and full of inconsistencies.

5. CONCLUSION

We have presented an explicit dual memory model for agents by integrating two separate modules of episodic and semantic memory. The stored contents of episodic memory can be recalled to derive abstract knowledge and general

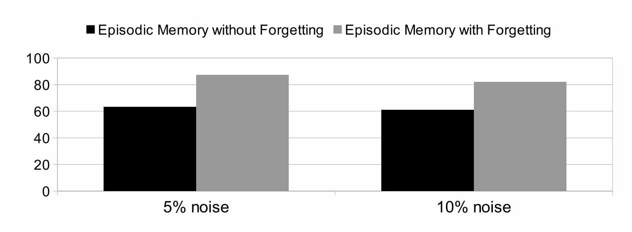

Figure 8: Accuracies in retrieval with and without forgetting in noisy environment

facts, that in turn are transferred to more permanent forms in semantic memory. The episodic and semantic memory modules are realized with fusion ART (adaptive resonance theory) neural networks as two independent but connected networks operating in different paces of learning. In line with theories and findings in neuropsychology, the asymmetrical rate of learning with some periodical consolidation between the two enables the acquisition of useful knowledge without risking to loose prior entries. A forgetting mechanism is also applied to regulate the size of memory by removing insignificant entries.

Our experiments confirm that an explicit episodic-semantic memory model can improve the agent learning and performance by acquiring useful knowledge for the task at hand through memory consolidation, relieving the agent from continuously reasoning and processing the information for learning. It is also demonstrated that the forgetting regulates the memory size while the performance is still improving. Moreover, the experiment shows faster forgetting can result in better learning. This indicates that the forgetting can successfully filter insignificant entries while maintaining the useful ones. The findings can inspire the exploration of forgetting as a useful feature of intelligent agents and machine learning systems in general.

In the future, we shall extend our model to learn more useful and general purpose semantic structures from episodic memory. We shall also extend our study to look at how episodic memory model may contribute directly to the performance of the agent rather than just as a transient structure. The forgetting mechanism can also be extended by applying different variations of memory strength functions to include task-related aspects like rewards, risks, or penalties. This may reveal the potential of the dual episodic-semantic model as effective memory systems that continuously and mutually process, learn, and forget information.

6. ACKNOWLEDGEMENTS

This research is supported by the DSO National Laboratories under research grant DSOCL11258.

7. REFERENCES

[1] C. Becker-Asano and I. Wachsmuth. Affective computing with primary and secondary emotions in a virtual human. *Autonomous Agents and Multi-Agent Systems*, 20(1):32–49, 2010.

[2] C. Brom, T. Korenko, and J. Lukavsky. How do place and objects combine? what-where? memory for human-like agents. *Intelligent Virtual Agents*, LNCS 5773, pages 42–48. Springer Berlin, 2009.

[3] M. A. Conway, J. M. Gardiner, T. J. Perfect, S. J. Anderson, and G. M. Cohen. Changes in memory awareness during learning: The acquisition of knowledge by psychology undergraduates. *Journal of Experimental Psychology*, 126(4):393–413, 1997.

[4] E. Erdem and F. Paolo. Forgetting actions in domain descriptions. In *Proceedings of the 22nd national conference on Artificial Intelligence (AAAI 2007)*, volume 1, pages 409–414. AAAI Press, 2007.

[5] A. S. Gupta, M. A. A. van der Meer, D. S. Touretzky, and A. D. Redish. Hippocampal replay is not a simple function of experience. *Neuron*, 65(5):695–705, 2010.

[6] W. C. Ho, K. Dautenhahn, and C. L. Nehaniv. Computational memory architectures for autobiographic agents interacting in a complex virtual environment: A working model. *Connection Science*, 20(1):21–65, 2008.

[7] Z. Kasap, M. B. Moussa, P. Chaudhuri, and N. Magnenat-Thalmann. Making them remember–emotional virtual characters with memory. *IEEE Computer Graphics and Applications*, 29(2):20–29, 2009.

[8] M. Y. Lim, R. Aylett, W. C. Ho, S. Enz, and P. Vargas. A socially-aware memory for companion agents. *Intelligent Virtual Agents*, LNCS 5773, pages 20–26. Springer Berlin, 2009.

[9] J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning memory. *Psychological Review*, 102(3):419–457, 1995.

[10] A. M. Nuxoll and J. E. Laird. Extending cognitive architecture with episodic memory. In *Proceedings of the 22nd national conference on Artificial Intelligence (AAAI 2007)* Volume 2, pages 1560–1565. AAAI Press, 2007.

[11] C. Ranganath, A. P. Yonelinas, M. X. Cohen, C. J. Dy, S. M. Tom, and M. D'Esposito. Dissociable correlates of recollection and familiarity within the medial temporal lobes. *Neuropsychologia*, 42(1):2–13, 2004.

[12] A. J. Silva and S. A. Josselyn. Cognitive neuroscience: The molecule of forgetfulness. *Nature*, 418:929–930, 2002.

[13] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, 1998.

[14] A.-H. Tan, G. A. Carpenter, and S. Grossberg. Intelligence Through Interaction: Towards A Unified Theory for Learning. In *International Symposium on Neural Networks (ISNN 2007)*, LNCS 4491, pages 1098–1107, Springer Berlin 2007.

[15] E. Tulving. *Elements of episodic memory.* Oxford University Press, 1983.

[16] K. Wang, A. Sattar, and K. Su. A theory of forgetting in logic programming. In *Proceedings of the 20th national conference on Artificial intelligence (AAAI 2005)*, volume 2, pages 682–687. AAAI Press, 2005.

[17] J. T. Wixted. The psychology and neuroscience of forgetting. *Annual Review of Psychology*, 55:235–269, 2004.