MEMORY IN EMBODIED AGENTS

TECHNICAL FIELD

[0001] Embodiments described herein relate to the field of artificial intelligence, and systems and methods for implementing and using Memory in Embodied Agents. More particularly, but not exclusively, embodiments described herein relate to unsupervised learning.

BACKGROUND ART

[0002] A goal of Artificial Intelligence (AI) is to build computer systems with similar capabilities to humans, including human-like learning and memory. Most contemporary machine learning techniques rely on "offline" learning, wherein AI systems are provided with prepared and cleaned data to learn on, limited to a specific domain. An outstanding challenge in the prior art remains in creating AI systems which experience objects and events in the world in a human-like way and learn from embodied interaction. By virtue of their embodiment and sensorimotor feedback loops with their environment, such AI agents may influence and guide their own learning. Such agents would make sense of streams of multimodal data from the world and retain information in a meaningful and useful way. A further outstanding challenge is to create a flexible AI Embodied Agent which can both learn from its own experience, as well as have its memories authored or altered by an external source (such as a human user). Hierarchical Temporal Memory (HTM) is an approach to replicating human memory which is based on a computational structure with multiple registers as analogues to cortical layers. HTM is configured to replicate patches of cerebral cortex. Nonetheless, HTM fails to provide Memory in Embodied Agents which allows Embodied Agents to learn and develop in real-time from sensorimotor experience.

SUMMARY

[0002a] In a first aspect, the present disclosure provides a method for training a SOM, including a plurality of Neurons, each Neuron associated with a weight vector, and a training record; including the steps of: receiving an input vector; determining if the input vector is "new"; if the input vector is not new: selecting a first Winning Neuron, favouring higher similarity between the Input Vector and the Winning Neuron, and modifying the weight vector of the first Winning Neuron towards the input vector; if the input vector is new: selecting a second Winning Neuron, favouring neurons with lower training records, and modifying the weight vector of the second Winning Neuron towards the input vector.

[0002b] In a second aspect, the present disclosure provides a method for targeted forgetting in a SOM including a plurality of Neurons, each Neuron associated with a weight vector, including the steps of: creating a Reset Mask comprising a plurality of Mask Values, each Mask Value associated with a Neuron of the SOM; applying a Reset Function to each Neuron of the SOM associated with a non-zero Mask

Value, wherein the Reset Function includes: a Forgetting Component, for resetting the Neuron's weights to an untrained state; and a Mask Component for modifying the output of the Reset Function as a function of the Forgetting Component and the Mask Value; modifying SOM Neuron weight vectors according to the output of the Reset Function.

BRIEF DESCRIPTION OF DRAWINGS

Figure 1: a schematic diagram of a CDZ architecture.

Figure 2: an ASOM.

Figure 3: eligibility signals for different modalities.

Figure 4: how an Eligibility Trace creates an Eligibility Window.

Figure 5: the phases of a learning event.

Figure 6: a user interface for setting the eligibility of different modalities.

Figure 7: a display of ASOM training.

Figure 8: a query viewer Input Field.

Figure 9: a user interface for specifying query patterns.

Figure 10: a display of LTM and STM.

Figure 11: a Working Memory System (WM System).

CLAIMS

1. A method for training a SOM, including a plurality of Neurons, each Neuron associated with a weight vector, and a training record; including the steps of:
   receiving an input vector;
   determining if the input vector is “new”;
   if the input vector is not new:
   selecting a first Winning Neuron, favouring higher similarity between the Input Vector and the Winning Neuron, and modifying the weight vector of the first Winning Neuron towards the input vector;
   if the input vector is new:
   selecting a second Winning Neuron, favouring neurons with lower training records, and modifying the weight vector of the second Winning Neuron towards the input vector.
2. The method of claim 1 wherein determining if an input vector is new includes the steps of:
   determining a first Winning Neuron, favouring higher similarity the Input Vector and the Winning Neuron;
   determining a match quality between the input vector and the first Winning Neuron;
   determining the input vector as new if the match quality is below a match quality threshold.
3. The method of claim 2 wherein match quality is determined as the activation of the first Winning Neuron in response to the Input Vector.
4. The method of claim 2 wherein if the input vector is not new, adjusting the learning frequency of the SOM according to the activity of the first Winning Neuron.
5. The method of claim 2 wherein selecting a second Winning Neuron, includes favouring neurons with lower training records, and higher similarity between the Input Vector and the Winning Neuron.
6. The method of claim 5 wherein favouring neurons with lower training records is achieved by applying an Activation Mask comprises Mask Values inversely proportional to the amount of training received by their respective Neurons.
7. The method of claim 1 wherein the training record is a weight and the value of the training record is proportional to the amount of training the training record’s associated Neuron has received.
8. The method of claim 2 including the step of adding random noise to the activation map of the SOM to the input vector, prior to selecting a second Winning Neuron.
9. The method of claim 2 wherein the training record of Neurons are configured to decay with SOM training and/or time.
10. A method for targeted forgetting in a SOM including a plurality of Neurons, each Neuron associated with a weight vector, including the steps of:

creating a Reset Mask comprising a plurality of Mask Values, each Mask Value associated with a Neuron of the SOM;

applying a Reset Function to each Neuron of the SOM associated with a non-zero Mask Value, wherein the Reset Function includes:

a Forgetting Component, for resetting the Neuron's weights to an untrained state; and

a Mask Component for modifying the output of the Reset Function as a function of the Forgetting Component and the Mask Value;

modifying SOM Neuron weight vectors according to the output of the Reset Function.

11. The method of claim 10 wherein the Forgetting Component creates random noise.

12. The method of claim 10 for targeted forgetting of memories associated with an Input Vector wherein the step of creating the Reset Mask includes the steps of, for each Mask Value of the Reset Mask:
determining a similarity between the Input Vector and the Mask Value's associated Neuron,
determining if the similarity is above a Reset Threshold;
if the similarity is above the Reset Threshold, setting the Mask Value to 1; otherwise if the similarity is not above the Reset Threshold, setting the Mask Value to 0.

13. The method of claim 10 wherein the SOM is a probabilistic SOM wherein an activity of each Neuron is determined using an Activation Function mapping a distance to a probability space and weights of Neurons are updated according to Neuron activity.

14. The method of claim 13 for targeted forgetting of memories associated with an Input Vector, including the steps of: providing the input vector as input to the SOM; and setting the Reset Mask according to the activation map of the SOM.

15. The method of claim 10 wherein each neuron is associated with a training record.

16. The method of claim 15 for targeted forgetting of infrequent memories, wherein the Reset Mask is set to the inverse of the training record.

17. The method of claim 15 for targeted forgetting of memories older than a Forgetting Threshold including the step of:
eroding the training record of each neuron over time; and
setting the Reset Mask to the inverse of the training record.

18. The method of claim 15 wherein the step of creating the Reset Mask includes the steps of, for each Mask Value of the Reset Mask:
determining if the Training Record of the Mask Value's associated Neuron is below a Reset Threshold;
if the Training Record is below the Reset Threshold, setting the Mask Value to 1; otherwise if the Training Record is not above the Reset Threshold, setting the Mask Value to 0.