

Evolution of metamemory based on self-reference to own memory in artificial neural network with neuromodulation

Yusuke Yamato (✉ yamato@alife.cs.i.nagoya-u.ac.jp)

Graduate School of Informatics, Nagoya University

Reiji Suzuki

Graduate School of Informatics, Nagoya University

Takaya Arita

Graduate School of Informatics, Nagoya University

Research Article

Keywords: metamemory, memory, neural, networks, evolved

Posted Date: November 8th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1034049/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Evolution of metamemory based on self-reference to own memory in artificial neural network with neuromodulation

Yusuke Yamato^{1,*}, Reiji Suzuki¹, and Takaya Arita¹

¹Graduate School of Informatics, Nagoya University, Nagoya, Japan

*yamato@alife.cs.i.nagoya-u.ac.jp

ABSTRACT

The ability that is people's self-monitoring and controlling of their memory processes is called metamemory. It has been regarded as truly unique characteristics of human memory, and has been studied widely as a component of metacognition in cognitive psychology. We aim to evolve artificial neural networks with neuromodulation, that have a metamemory function. Our constructive approach is based on the repetition of evolutionary experiments, analysis of the evolved networks, and refinement of the measure, to reduce the gap between the functional properties of behavior and human subjective reports of phenomenal experience. In this paper, we show the evolution of neural networks that have metamemory function based on the self-reference of memory, and analysis of the evolved mechanism of metamemory. We also discuss a similarity between the structure of the evolved neural network and the metamemory model defined by Nelson and Narens.

Introduction

Metamemory refers to people's self-monitoring and self-control of their own memory and its processes, and has been studied widely as a component of metacognition in cognitive psychology¹. While it had been regarded as truly unique characteristics of human memory², testing metacognition or metamemory in animals has become a challenging topic. Furthermore, recently, constructive studies using artificial life methods (e.g., evolutionary computation and artificial neural networks) are also becoming another methodology. Metacognition is not a behavioral phenomenon and thus difficult to analyze in general. Therefore, comparative studies based on these methodologies, each of which has its own pros and cons, will become increasingly important in elucidating metacognition.

When we use animal subjects and investigate whether they have metamemory ability, we have to rely heavily on behavioristic paradigms as it is not possible to use questionnaires as in the case of experiments with human subjects. Hampton devised a *delayed matching-to-sample* (DMTS) paradigm³. The idea is as follows: Animals able to distinguish between the presence and absence of their own memory should improve accuracy in memory tests if allowed to decline the tests when they have forgotten. They should also decline tests by selecting the escape option more frequently if memory is attenuated experimentally (e.g., by adding forgetting time). Hampton successfully showed that one of two macaque monkeys examined met these criteria. As a constructive approach, Sudo and others⁴ used DMTS paradigm and showed that computational agents controlled by artificial neural networks could evolve metamemory ability. Their success partly depended on the usage of a neuromodulation technique⁵ by which a specific type of neurons, *modulatory neurons*, can dynamically alter the plasticity (learning rate) of the connections of the neurons they project to.

However, the validity of the methods to detect metamemory in animals has been actively and controversially discussed. According to Hampton, it is important that the mechanism of metamemory is distinguished into public and private⁶. The former is an adaptive cognitive control that is based upon the use of publicly available information, and the latter is a cognitive control that is contingent on the privileged access the subject has to their own cognitive states. Hampton pointed out that metamemory detection cannot be said to have been achieved unless the availability of external cues is eliminated by focusing on the types of cues used in metamemory tasks in animal experiments. At the same time, Hampton claimed that his experiments³ had detected the metamemory which is based on internal cues with high possibility.

Contrary to Hampton's claim, other researchers presented other interpretations that were not dependent on internal cues. For example, DMTS-based tests might be passed if the subject selected the escape option when the input pattern seemed difficult to remember, or simply complicated⁷. Furthermore, some researchers suggest that metamemory tasks like DMTS do not necessarily require metamemory ability even when the judgment is based on internal cues. For example, taking a strict stance on the definition of the word, metacognition, Carruthers and Crystals distinguished between first-order explanations and metacognition, and claim that alternative explanations based not on metamemory but on first-order explanations are possible for

the results of many animal experiments for detecting metacognition including Hampton's experiment^{8,9}. Following this theory, metacognition is truly cognition about cognition, and should be formed as second-order representation.

How is this problem considered in human metacognition in general? Kornell¹⁰ summarized that humans cannot directly evaluate internal memory strength, and instead, they make certainty judgments by drawing inferences from many cues they can evaluate (e.g., familiarity and ease of processing).

In light of these arguments, there is no doubt that the results of Hampton's DMTS experiments alone are insufficient to make a substantial contribution to the discussion of metamemory. This is due to the fact that there is a complex gap between behavior or function of metamemory and the mechanisms that govern them. This is where the constructive approach comes into play in the sense that we could observe and analyze all the circuits in the evolved neural networks in any way we like.

Following Sudo and others' study⁴, Yamato and others made a similar evolutionary model, and successfully found that there were two types of mechanisms in the evolved agents, either of which made them show the metamemory ability from the perspective of the DMTS paradigm¹¹. However, if we consider the above arguments in animal experiments, it would be difficult to claim that one of the type of the evolved mechanisms have had a metamemory ability. Specifically, this type had a strategy which tends to decline the test when the input pattern was difficult to remember or recognize independent of the state of its neural network, exactly as Call pointed out⁷. Such a strategy is purely public in Hampton's classification. On the other hand, the other type was implemented by detecting the deterioration of memory content (forgetting) but was not achieved by self-referencing its memory state. Instead, it was implemented by some kind of spurious relationship between its memory state and selection circuit of declining. More precisely, the metamemory judgment in this type of mechanisms is based on whether there is a change in its network meaning that a big disturbance of cognitive activity has occurred in the network. This is a first-order explanation of an internal physiological state. This mechanism might roughly correspond to the memory flag hypothesis¹² that a subject uses an indicator for the presence of memory as a metacognitive response while not being aware of the content of the memory. Although the judgment based on this mechanism can be regarded as metamemory from this perspective, this mechanism does not meet the criterion of metamemory in the narrow sense⁸.

To make a substantial contribution to the discussion of the mechanisms of metamemory, this study, as a constructive approach, further aims to evolve artificial networks with a metamemory ability in a narrow sense, in other words, which is based on the self-reference of the memory state. For this purpose, we make two changes to the method for the evolution of neural networks¹³. Specifically, we add Gaussian noise to the input values of all neurons in the neural networks so that robustness is required for the abilities of agents. We also provide a constant value to each of the pattern input neurons during the choice phase to explicitly indicate that the agent is in that phase. This paper successfully shows, by conducting a detailed analysis of the evolved neural networks, that the evolved neural network meets the criterion of metamemory in the narrow sense, presented, for example, by Carruthers⁸ and finally situates the results in Nelson and Narens' conceptual formulation of metamemory^{14,15}.

Model

Task

Fig. 1 shows an overview of the task based on DMTS paradigm used in⁴. Firstly, an agent receives a target pattern composed of 5 binary digits, which is randomly selected from 00001, 00010, 00100, 01000 and 10000 in the study phase. The delay phase follows, in which the agent receives 00000 as a distractor pattern a predefined number of times defined by equation (1), which will affect the degree of uncertainty in its memory.

$$N_{delay} = \lfloor \frac{-1}{\lambda \times \ln(R)} \rfloor + 1, \quad (1)$$

where λ is a parameter related to the shape of the distribution, and R is a uniform random number from $[0, 1]$.

Then, with a probability of $2/3$, the choice phase starts. During the phase, the agent receives a signal meaning that it is in that phase and a constant value (CV) which is given to each input neuron as an input pattern. The constant value is expected to be used for self-reference of a memory. One output from the agent in the range $[-1, 1]$ will be interpreted as an intention to decline or take the trial when it is more than $1/3$. In this case, the agent receives a small reward (0.3). Otherwise if the value equals to or is more than $1/3$, it will be interpreted as an intention to take the test. On the other hand, with a probability of $1/3$, the choice phase is skipped as a compulsory condition.

In the test phase, the agent receives all patterns one by one in random order, and an output ranged in $[-1, 1]$ is interpreted as a response for each pattern. Specifically, when the response goes over $1/3$ for the first time, the corresponding input pattern will be interpreted as the selection of the agent and the task ends. If it matches the target pattern presented in the study phase, the agent is rewarded with a large reward (1.0). Otherwise, if it does not match the target or all responses do not go over the value of $1/3$, it is rewarded with nothing (0.0).

In addition, we use an unsolvable condition, in which agents do not receive one of the 5 patterns but receive a distractor pattern 00000 in the study phase. In this condition, they receive the large reward (1.0) only if it selects the decline option. We expect this addition accelerates the evolution of the ability to select the option.

Metamemory criteria

The condition for an agent to have metamemory is that it meets one of the behavioristic paradigms for metamemory (e.g., DMTS paradigm) without depending on external cues. On the other hand, there are multiple mechanisms that meet this condition as we discussed in the previous section. To distinguish them, we define the following criteria from the viewpoint of introspection on the memory¹¹.

- Criterion 1 (C1)

The agent satisfies one of the behavioristic paradigms for metamemory, but not by changing the behavior according mainly to particular stimuli configurations.

- Criterion 2 (C2)

C1 is met, but is based on the self-reference on some part of the stored information regarding the stimuli input.

However, this is not enough to distinguish whether metamemory is based on second-order representations or not. Therefore, we add the following criterion.

- Criterion 3 (C3)

C2 is met, but is based on judgments based primarily on internally constructed second-order representations.

Neural network

The neural network of each agent is composed of several standard neurons including 7 input and 2 output neurons, and modulatory neurons (described in the next subsection) as shown in Fig. 2. Among 7 inputs, one input neuron gets a signal indicating whether it is in the choice phase (1) or not (0). Another is a bias neuron that receives a constant input value (1.0). Each input of the other 5 neurons gets one of 5 digits of an input pattern, respectively.

The topology of the network evolves while keeping the number of neurons not more than 16, including standard and modulatory neurons, but excluding input neurons. The network is inputted the same patterns sequentially for 4 times in the choice phase and 3 times in the other phases. Repetition of input is essential when the evolved neural networks have recurrent connections. Each value of all neurons in all phases is slightly modified by adding a suitable amount of Gaussian noise (input neurons: $\mu = 0.0$ and $\sigma = 0.1$, other neurons: $\mu = 0.0$ and $\sigma = 0.0001$). The reason for adding the noise is to prevent the evolution of agents that are not robust. We regard the lack of robustness as the cause of which the evolved agents met C1 but not C2¹³.

Neuromodulation

We use neural networks with neuromodulation to evolve metamemory ability because it is suggested that neuromodulation has an advantage in the evolution of higher cognitive functions. Specifically, Arnold and others¹⁶ suggested selection for learning and selection for second-order learning as the causal factors driving the emergence of innate and acquired forms of representation, respectively, and successfully showed that cognitive map which is one of the mental representations is able to evolve through secondary learning based on second-order modulation. The neural network in our model also contains modulatory neurons in addition to standard neurons. Modulatory neurons affect the learning rate of the connection weights of target neurons and dynamically change it as shown in Fig. 3. In particular, the output of the modulatory neuron m_i modulates the learning rate of the update rule of the connection weight by using it as a modulatory signal instead of directly affecting an activation signal a_i . They are computed by equations (2) and (3).

$$a_i = \sum_{j \in Std} w_{ji} \cdot o_j, \quad (2)$$

$$m_i = \sum_{j \in Mod} w_{ji} \cdot o_j, \quad (3)$$

where w_{ji} is a connection weight from a presynaptic neuron j to a postsynaptic neuron i . Std and Mod are the sets of

standard and modulatory neurons connected to the neuron i , respectively. o_j is an output of the neuron j and is computed as $o_j = \tanh(a_j)$. We regard that the neuron is positively activated if its output value is positive (or close to 1) and negatively activated if it is negative (or close to -1). The connection weight from a neuron j to a neuron i is updated by equation (4), which is based on an extension of Hebb's rule called Extended Hebbian rule¹⁷.

$$\Delta w_{ji} = \tanh(m_i) \cdot \eta \cdot (A o_j o_i + B o_j + C o_i + D), \quad (4)$$

where o_j and o_i are the outputs of a presynaptic neuron j and a postsynaptic neuron i , respectively. η , A , B , C and D are also genetic parameters. Thus, the update rule can represent various types of synaptic updating via the evolution of these parameters.

Evolutionary algorithm

We use a variant of a genetic algorithm. The setting of the algorithm is basically the same as the one proposed in⁴. Each agent has a matrix of real-valued connection weights (one axis corresponds to presynaptic neurons while the other to postsynaptic neurons) and the type of each neuron (standard or modulatory) for the decision of the structure, and five parameters (η, A, B, C, D) for the update rule equation (4), as a genome. Each connection weight w_{ij} and η are ranged in $[-100, 100]$ while A, \dots, D in $[-1, 1]$. These parameters in genotype are converted to each weight w_{ji} in the phenotype by equation (5) and A, \dots, D in the phenotype by equation (6) except for η .

$$w_{ji} = \begin{cases} 0 & (|w_{ji}^3| < 0.1) \\ 10 \cdot w_{ji}^3 & (otherwise), \end{cases} \quad (5)$$

$$p = \begin{cases} 0 & (|p^3| < 0.1) \\ p^3 & (otherwise), \end{cases} \quad (6)$$

where $p \in \{A, B, C, D\}$. The total score of each agent obtained by performing the task of each is defined as its fitness. The genetic operators are conducted as follows. Individuals are stored in an array, and are divided into consecutive segments of size 5 (with random segmentation offset at each generation). The best individual of each segment becomes a parent, and generates 5 children for its segment by repeating crossover with a probability of 0.1 or copying itself otherwise. When crossover happens, a partner individual is randomly selected from the all individuals in the population. Random integers r and c are selected from $[1, N]$, and two matrices are generated by exchanging the sub matrices of the parents that are composed by i, j elements with i and j less than or equal to r and c , respectively. A uniform crossover is performed on the parameters for the plasticity rule. Then, as a mutation operator, with a probability of 0.1, Gaussian noise ($\mu = 0.0$, $\sigma = 0.3$) is added to each of connection weights and the parameters for the plasticity rule except η , while Gaussian noise ($\mu = 0.0$, $\sigma = 3.0$) is added to η . Finally, insertion, deletion and duplication of each neuron are independently performed with probabilities of 0.04, 0.06 and 0.02, respectively. When insertion happens, the weight of the added neuron is randomly set in the range $[-1, 1]$ and the type of the neuron (standard or regulatory) is randomly set. These processes constitute a generation and are repeated G times.

Experiment and Analysis

We performed 10 evolutionary trials using the parameters: $N = 300$, $G = 1000$, $\lambda = 0.7$ and $CV = 0.25$. During the first 100 generations, the agents have to jump to the test phase by skipping the choice phase every time. This setting was meant to guide the evolution in the initial stage. In subsequent generations, the tasks with the unsolvable condition were performed $U = 50$ times randomly among $T = 300$ times of task execution. The neural network of each agent was initialized by setting random values within corresponding possible ranges every when a task started.

We successfully found neural networks by analysis that satisfy C3 in 1 trial among 10 trials. Fig. 4 shows the behavior of the evolved agents for each delay time in the delay phase. This individual has a higher accuracy of the selection condition than that in the forcing condition. We can also observe the increase in the avoidance rate with the increase in the delay. This result is reasonable if we consider the agent has a metamemory function, and was also observed in the behavior of the monkey in Hampton's experiment³.

Table 1 shows the average values of accuracy with a time delay of 40 in forced tests, chosen tests, and declined tests (assuming that they had taken the test despite declining). The latter two are shown with the probabilities of taking and declining in parentheses, respectively. There is a certain difference in accuracy between forced and chosen tests, and between chosen and

declined tests, independent of input patterns. This reveals that the agent tended to avoid the trials which it could not answer correctly. The behavior of this agent satisfies C1 because the behavior was not simply based on the response for particular input stimuli configurations.

We analyzed how the network had solved the metamemory task by observing temporal changes in neuronal activity and connection weights in the network. Fig. 5 shows the network which was analyzed above. As shown in the yellow area of Fig. 5, we found that the network has second-order modulator structure in which two modulatory neurons in md_0 and md_1 (in light blue) modulated plasticity of the learning of some connections from the standard neurons to another modulatory neuron md_2. This structure played an important role in memorizing a received pattern, declining the test and answering the test.

Fig. 6a shows the mechanism of memorizing a received pattern. The network reflects and keeps a structure of a target pattern which it received in the study phase by corresponding the sign of the connection weights to the value of each bit of the pattern (0 or 1). This ability is made by md_0 and md_1 maximally strengthening/weakening each connection weight between the corresponding input neuron from bit_0 to bit_4 and md_2 (hereinafter referred to as *core modulatory neuron*) by modulating the plasticity of the learning of some connections which connect to md_2. During the delay phase, the structure of the memorized pattern is probably protected because the learning of the connection weights is impeded by the absolute value of the sum of the modulatory signals which cancel each other out. However, the absolute value becomes larger than the small value if md_0 and md_1 are disturbed by much noise. In this case, the learning is promoted, which can make the agent forget the memorized pattern.

Fig. 6b shows the mechanism of declining the test. The core modulatory neuron can manage the behaviors of declining and answering by its activity state. It judges the presence of the memory of the presented pattern by the sign of the sum of the inputs coming through each connection (each weight of which should reflect the presented pattern) between the input neurons and the core modulatory neuron. In detail, the network judges that the memory exists when the sign of the sum of inputs is plus. Otherwise, it judges that the memory does not exist. If and only if the agent forgets the memorized pattern, the core modulatory neuron receives the negative sum of the inputs and activates negatively in the choice phase. As a result, the core modulatory neuron changes the plasticity of some connections and the result of the learning makes the declining neuron activate positively. Thus, the network successfully declines the test. We also found that the mechanism of pattern selection in the test phase was also based on the monitoring of memory state by the core modulatory neuron. Thus, our experiments indicate that, in conclusion, that these networks can work correctly based on self-reference, in other words, they satisfy C2.

The evolved network is clearly separated between the memory part where the pattern is reflected and the part where modulates some connection weights by referring to the memory state. We can find that the flow of information based on this structure may meet Carruthers and Crystal's criterion of metamemory. Firstly, the memory part which is constructed by some connections receives a constant value and outputs some value, in Choice phase. This output value is able to be represented as "the memory state". Secondly, the core modulatory neuron activates positively or negatively by receiving the output value from the memory part and changes the plasticity of some connections based on its activation state. The activation state of the core modulatory neuron corresponds to whether the sum of the value of the memory state is positive or negative. We can consider it a judgment whether the memory state exists or not. Where the state of the connections formed through changing plasticity of connections by the core modulatory neuron is considered the representation of whether the network remembers "the memory state" or not. This representation corresponds to knowledge about cognition of memory, exactly metamemory. In summary, these networks satisfy C3.

This structure is composed of second-order plasticity through second-order modulation as described above. This fact suggests that second-order plasticity plays an important role in the realization of metamemory ability in a narrow sense. If a network has only first-order plasticity, it would have behavior based on first-order explanations. For example, we consider an alternative network in which a declining neuron connects negatively to answering neurons, as shown in Fig. 7. At first glance, the network has declining and answering abilities based on the results of memory state referencing as the evolved network, but they are not based on the representation of the judgment, which is second-order representation, based on the memory state reference. The metamemory of the alternative network does not correspond to C3 while satisfying C2.

Finally, we would like to point out that this structure which governs the above series of the process corresponds well to the metacognition structure and of Nelson and Narens's model (Fig. 8), which is often used to explain the processes of multiple aspects of metacognition integrally^{14,15}. Specifically, as shown in Fig. 9, the memory part corresponds to the object-level, the part which observes the memory state and the part which modulates the connection weights based on the core modulatory neuron corresponds to the meta-level, and the activation state of the core modulatory neuron corresponds to a model of the object level inside the meta-level. Such structural correspondence with the very abstract model is unique in at least some of the networks that satisfy C3 and might be advantageous for the universal response of various metamemory judgments because it includes the processes of multiple aspects of metacognition.

Conclusion

Our purpose is to make a substantial contribution to the discussion of the mechanisms of metamemory. For this purpose, this paper conducted evolutionary experiments using the constructive methodology of the previous studies^{11,13}, analyzed the metamemory mechanism of the evolved neural network in detail, and clarified the significance of the evolved metamemory. When discussing its significance, we used the criteria (C1, C2, and C3) for metamemory, which can exclude the solutions that had been criticized in the discussion on the experiments based on the delayed matching-to-sample (DMTS) paradigm that had been often used for detecting metacognition in non-human animals.

We successfully found an agent which met C3, and investigated the evolved neural network of the agent by analyzing its behavior, structure, and neural dynamics. Firstly, we found that the agent shows the behavior like a monkey that was claimed to have an ability of metamemory in Hampton's experiment, and also that the behavior is not a reflex behavior for particular input stimuli configurations. We then observed temporal changes in the state of the network, and showed that the network has a structure in which two modulatory neurons modulate some connections from the standard neurons to another modulatory neuron. We showed that this structure plays an important role in memory and metamemory ability as follows: In the study phase, the network reflects and keeps the structure of the received pattern in each connection weight between the corresponding input neuron and the modulatory neuron by the coupling effect of two modulatory neurons. Moreover, in Choice phase, the core modulatory neuron modulates the plasticity of some part of connections which connect to the standard neurons for output based on a monitoring result of the memory state. This modulation allows the network to appropriately decline or answer the test. A series of these processes realize metamemory based on self-reference. We finally discussed a good correspondence between the evolved network structure and Nelson and Narens's formulation, which is an essential feature of the networks that meet C3.

We believe that our study contributes to the understanding of human metamemory and realization of artificial consciousness. As a first step toward this goal, we are focusing on the evolution of co-representation by using collaborative problems by several agents based on an extension of DMTS paradigm.

Data availability

The source code and the data used in the current study are available from https://github.com/yamatakeru/Evolurionary_experiment_with_DMTS.

References

1. Dunlosky, J. & Thiede, K. W. Metamemory. in d. reisberg (ed.). *The Oxf. handbook cognitive psychology* 283–298 (2013).
2. Tulving, E. & Madigan, S. A. Memory and verbal learning. *Annu. review psychology* **21**, 437–484 (1970).
3. Hampton, R. R. Rhesus monkeys know when they remember. *Proc. Natl. Acad. Sci.* **98**, 5359–5362 (2001).
4. Sudo, M., Suzuki, R. & Arita, T. Can agents with neuromodulation know when they remember? *Proc. 19th Int'l Symp. on Artif. Life Robotics* 330–334 (2014).
5. Soltoggio, A., Bullinaria, J. A., Mattiussi, C., Dürr, P. & Floreano, D. Evolutionary advantages of neuromodulated plasticity in dynamic, reward-based scenarios. *Proc. 11th international conference on artificial life (Alife XI)* 569–576 (2008).
6. Hampton, R. R. Multiple demonstrations of metacognition in nonhumans: Converging evidence or multiple mechanisms? *Comp. cognition & behavior reviews* **4**, 17 (2009).
7. Call, J. Do apes know that they could be wrong? *Animal cognition* **13**, 689–700 (2010).
8. Carruthers, P. Meta-cognition in animals: a skeptical look. *Mind & Lang.* **23**, 58–89 (2008).
9. Crystal, J. D. & Foote, A. L. Metacognition in animals. *Comp. cognition & behavior reviews* **4**, 1–16 (2009).
10. Kornell, N. Where is the "meta" in animal metacognition? *J. Comp. Psychol.* **128**, 143–149, DOI: [10.1037/a0033444](https://doi.org/10.1037/a0033444) (2014).
11. Yamato, Y., Suzuki, R. & Arita, T. Creating metamemory by evolving neural network with neuromodulation. *Proc. 24th Int'l Symp. on Artif. Life Robotics* 85–90 (2019).
12. Hampton, R. R., Zivin, A. & Murray, E. A. Rhesus monkeys (macaca mulatta) discriminate between knowing and not knowing and collect information as needed before acting. *Animal cognition* **7**, 239–246 (2004).
13. Yamato, Y., Suzuki, R. & Arita, T. Evolution of metamemory ability by artificial neural networks with neuromodulation. *Proc. 2019 Conf. on Artif. Life* 461–462 (2019).
14. Nelson, T. O. Metamemory: A Theoretical Framework and New Findings. *Psychol. Learn. Motiv. - Adv. Res. Theory* **26**, 125–173, DOI: [10.1016/S0079-7421\(08\)60053-5](https://doi.org/10.1016/S0079-7421(08)60053-5) (1990).

15. Nelson, T. & Narens, L. Why Investigate Metacognition? *Metacognition* DOI: [10.7551/mitpress/4561.003.0003](https://doi.org/10.7551/mitpress/4561.003.0003) (1994).
16. Arnold, S., Suzuki, R. & Arita, T. Selection for representation in higher-order adaptation. *Minds Mach.* **25**, 73–95 (2015).
17. Mouret, J.-B. & Tonelli, P. Artificial evolution of plastic neural networks: a few key concepts. In *Growing adaptive machines*, 251–261 (Springer, 2014).

Acknowledgements

This work was supported in part by JSPS/MEXT KAKENHI: JP17H06383 in #4903 (Evolinguistics).

Author contributions statement

Y.Y. developed the experiment's programs, performed the experiment, analyzed the data and created the figures. All authors conceived the study, discussed the experiment's results and wrote the manuscript.

Competing interests

The authors declare no competing interests.

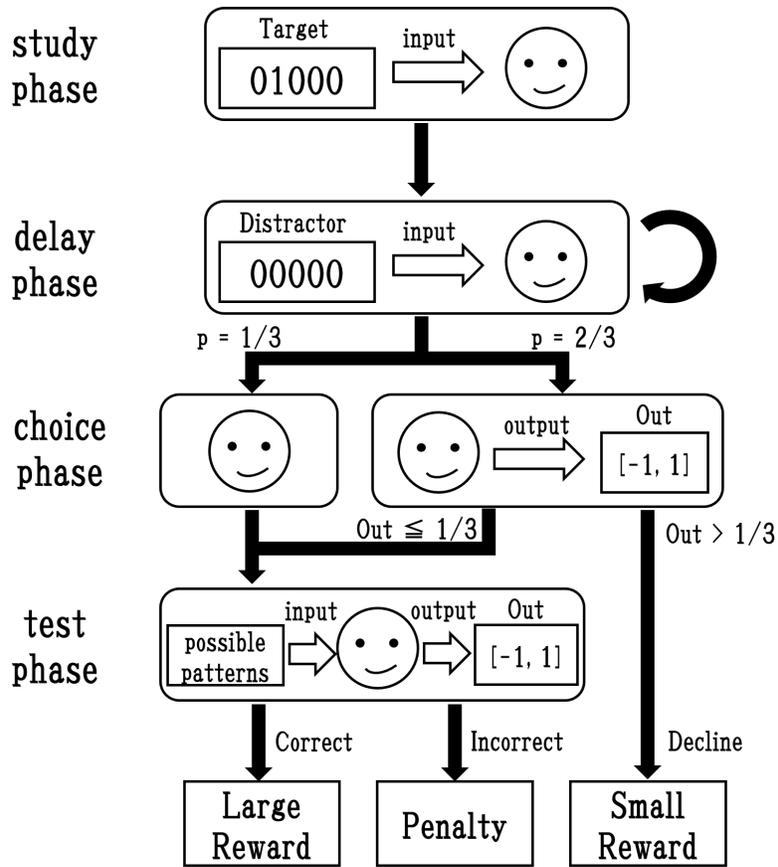


Figure 1. The delayed match-to-sample task which is introduced the escape option.

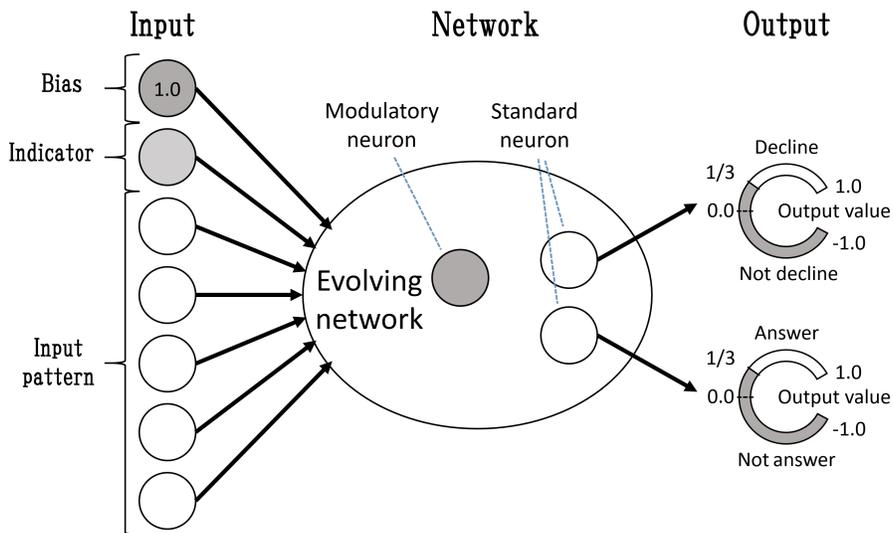


Figure 2. Inputs and outputs of a neural network.

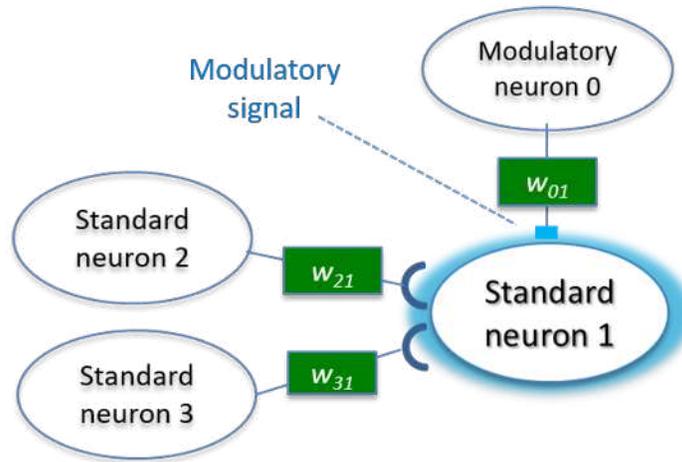


Figure 3. The overview of the neuromodulation.

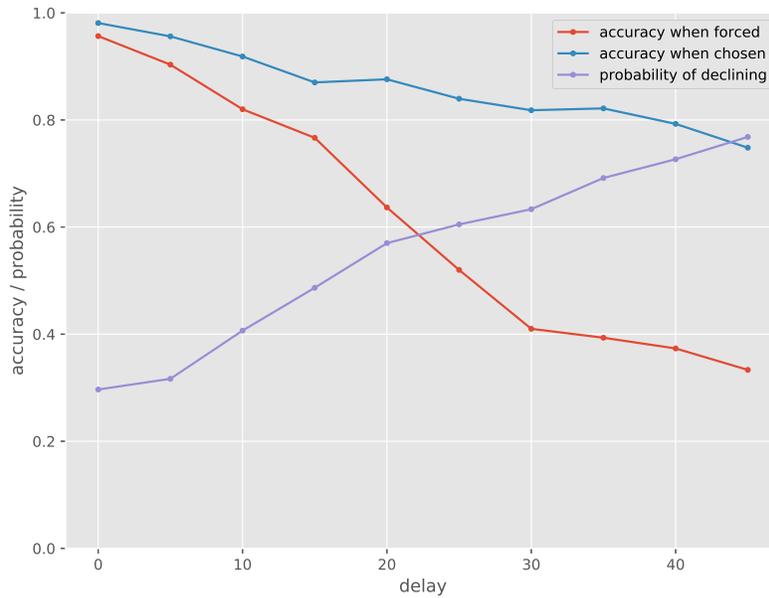


Figure 4. The behavior of the evolved agent for each delay time.

Input patterns	Accuracy in forced tests	Accuracy in chosen tests (probability of taking)	Accuracy in declined tests (probability of declining)
00001	0.361	0.835 (0.270)	0.175 (0.730)
00010	0.354	0.803 (0.266)	0.216 (0.734)
00100	0.340	0.764 (0.240)	0.205 (0.760)
01000	0.361	0.819 (0.258)	0.183 (0.742)
10000	0.347	0.816 (0.272)	0.206 (0.728)

Table 1. Accuracy for each input pattern in each case ($delay = 40$).

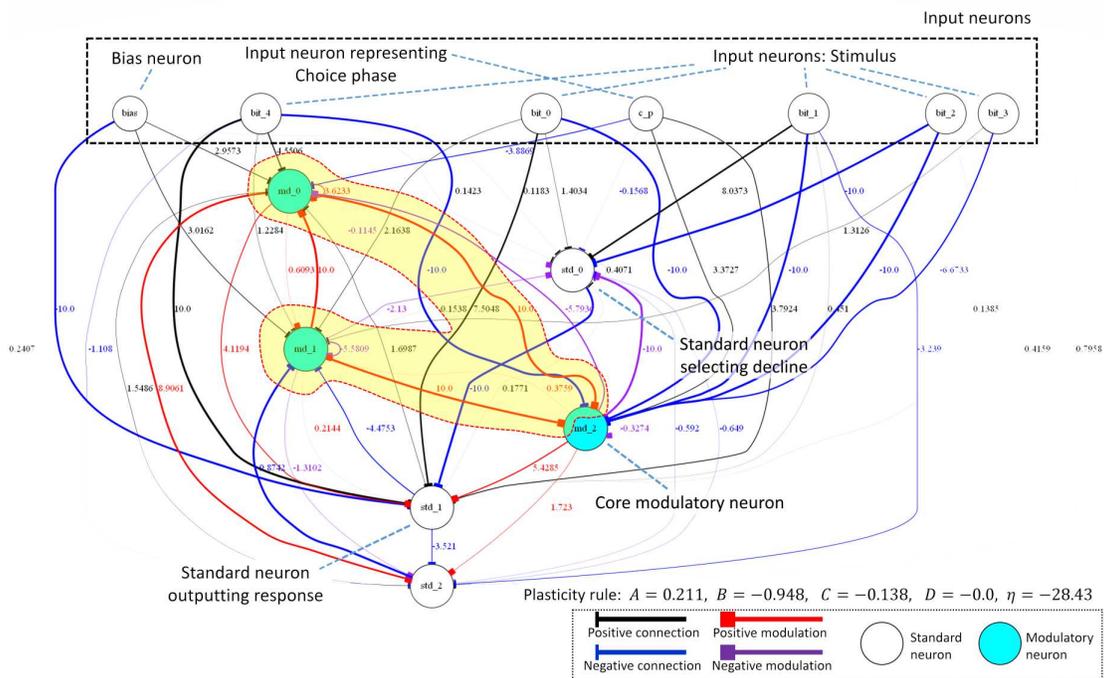
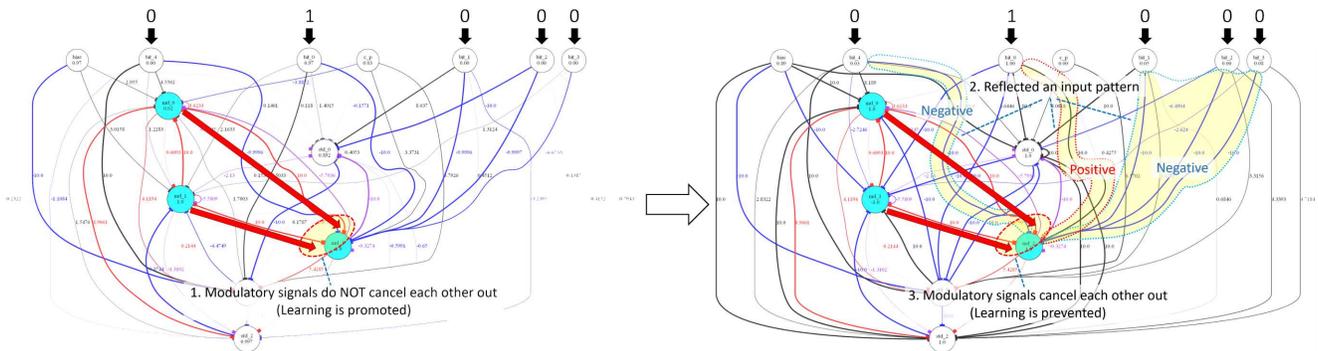
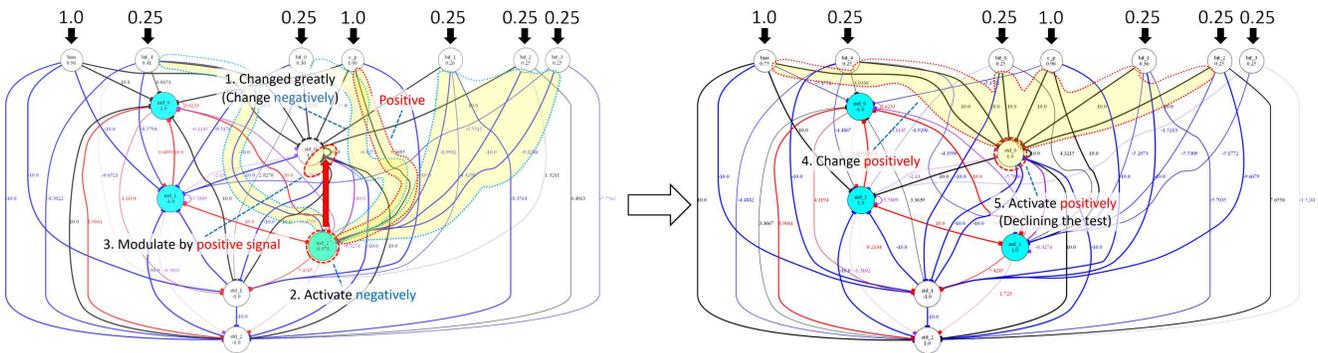


Figure 5. The structure of the evolved neural network. The width of each line represents the weight of a connection.



(a) The mechanism of memorizing and keeping a target pattern in the study phase. 1) Learning of connection weights is promoted because the modulatory signal from md_0 to md_2 and that from md_1 to md_2 do not cancel each other out. 2) As a result, signs of some connections reflect a structure which is composed of 0 and 1 of an input pattern. 3) Thereafter, the modulatory signals cancel each other out and learning is prevented. In other words, the reflected information is protected.



(b) The mechanism of declining the test in the choice phase. 1) If the connections in which an input pattern is memorized change greatly, md_2 (core modulatory neuron) receives the inputs in which the sum is negative from bit_0, ..., bit_4 and c_p (where c_p acts as a bias in order to judge whether the agent forgets or not). 2) As a result, md_2 activates negatively, and 3) the plasticity of the learning of some connections which connect to std_0 (the declining neuron) is modulated by the positive signal from md_2. 4) As a result, the connections which connect to std_0 change positively, and 5) std_0 activates positively and the network declines the test.

Figure 6. The mechanism by which the network makes memorizing and keeping a target pattern (00001), and declining the test. A red arrow represents a modulation. Each event occurs in numerical order.

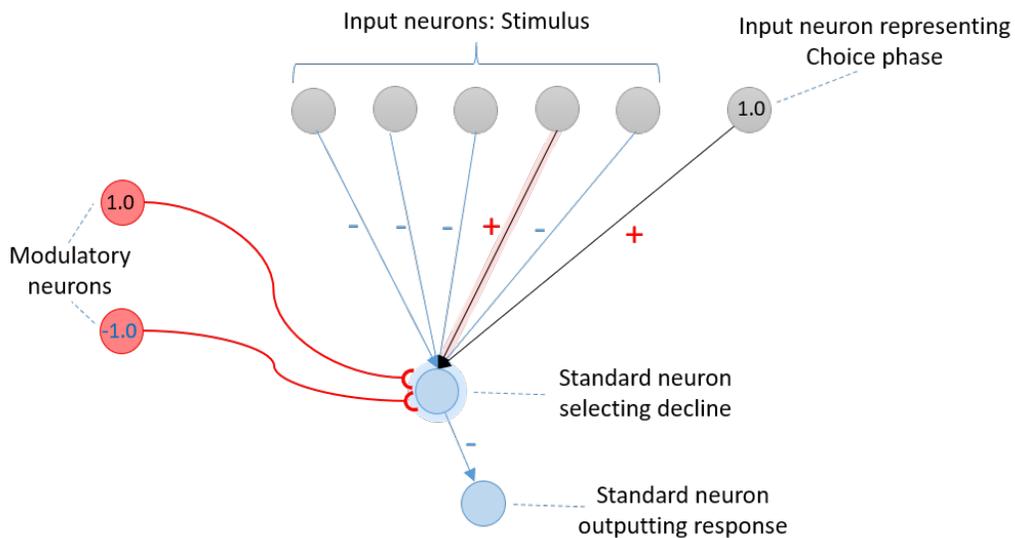


Figure 7. The main structure of an alternative first-order plasticity network which is capable of behaving like the evolved network. The network makes a metamemory judgment by directly using the result of a judgment whether the memory state exists or not without forming a representation.

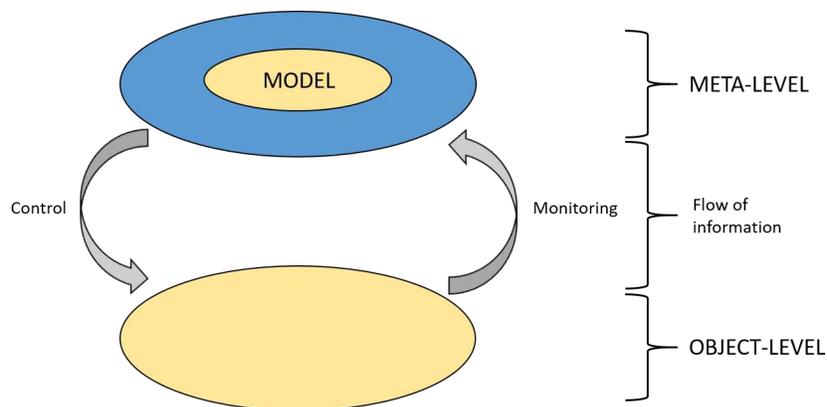


Figure 8. Nelson and Narens's formulation of a meta-level / object-level theoretical mechanism consisting of two structures (meta-level and object-level) and two relations in terms of the direction of the flow of information between the two levels. The meta-level contains an imperfect model of the object-level.

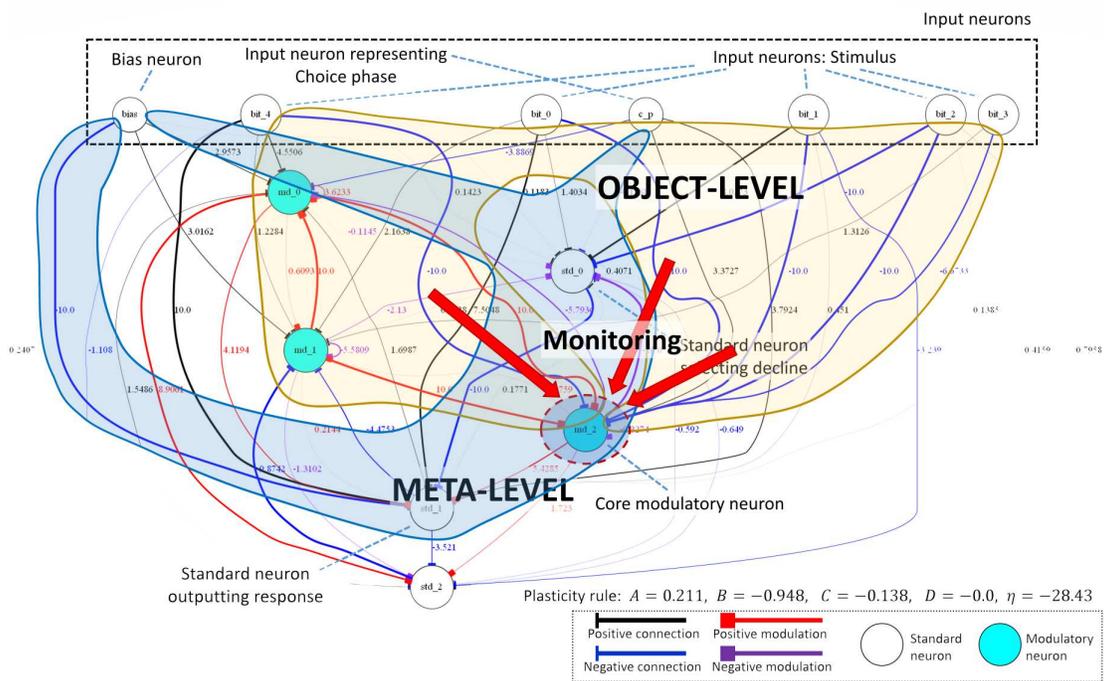


Figure 9. The internal structure of the evolved neural network and its correspondence to the structure of Nelson and Narens's formulation.