A Brain-Inspired, Logic-based Sequence Learning Model*

Bowen Xu

Department of Computer and Information, Temple University, 1801 N Broad St, Philadelphia, 19122, PA, USA.

Contributing authors: bowen.xu@temple.edu;

*This is the manuscript updated in June 2024.

abstract

Background. Sequence learning is an essential aspect of intelligence. In Artificial Intelligence, sequence *prediction* task is usually used to test a sequence learning model.

Methods. In this paper, a model of sequence learning, which is interpretable through Non-Axiomatic Logic, is designed and tested. The learning mechanism is composed of three steps, hypothesizing, revising, and recycling, which enable the model to work under the *Assumption of Insufficient Knowledge and Resources*.

Results. Synthetic datasets for sequence *prediction* task are generated to test the capacity of the model. The results show that the model works well within different levels of difficulty, reaching the theoretically highest accuracy. In addition, since the model adopts *concept-centered* representation, it does not suffer from *catastrophic forgetting*, and the practical results also support this property.

Conclusion. This paper shows the potential of learning sequences in a logical way.

keywords. Sequence Learning, Non-Axiomatic Logic, Brain-inspired, Mini-column

1 Introduction

Sequence leaning (sometimes known as *sequential learning*, *serial order learning*, *etc.*) refers to acquiring the proper ordering of *events* or stimuli [1, 2]. It is the foundation of many learning processes for an intelligent agent to interact with the world, such as sensorimotor process, natural language acquisition, *etc.*

In Cognitive Science, *Serial Reaction-Time* task was widely used for measuring subjects' performance of sequence learning [3], where given some repeated sequences of stimuli, subjects' reaction time decreases with time goes by. While in Artificial Intelligence (AI), people usually measure the anticipation accuracy of a sequence learning model. There are several types of tasks in AI to evaluate a sequence learning model, including sequence *prediction*, *generation*, *recognition*, and *decision making* [2]. Various of approaches for sequence learning are proposed, including Markovian approaches [4], recurrent neural network [5], *etc.* Neural networks, such as Transformer [6], have gained huge progress in natural language processing, which could be viewed as a special case of sequence learning task. There are some biologically plausible models, among which an intriguing one is Hierarchical Temporal Memory (HTM): through modeling neocortical column, the HTM model can memorize frequently occurring sequences as long as each *event* can be converted to Sparse Distributed Representation (SDR) [7], though how to deal with uncertainty is still a challenge for the HTM model.

Explainability is an important issue on AI security, and a major criticism of neural networks is their lack of explainability: the models are black or grey boxes, and developers are hard to understand what is going on and how to fix it when unexpected behaviors occur. It can be argued that this issue can be addressed if a model follows a logic, in other words, a model is interpretable if it is described through symbolic or *logical representation*. Among various candidates besides the well-known First-Order Predicated Logic (FOPL) and Expert Systems, there is a promising model of intelligent reasoning, Non-Axiomatic Logic (NAL) [8], which is able to deal with uncertainty and has proposed a solution of the *symbol grounding* problem [9, 10]. In NAL, there are some logical rules for temporal inference[11], *e.g.*, *deduction*, *induction*, *etc.* However, how to extract temporal patterns from sequences remains a hard problem with this logical representation.

Highly inspired by HTM, the biologically-constrained model, and NAL, the logic for modeling intelligence, in this paper, a brain-inspired model of sequence learning is proposed. The previous research [12, 13] suggested that *mini-column* is a widely distributed structure in the Neocortex. To model this structure, in HTM, a collection of mini-columns represents an event, and a neuron in a mini-column corresponds to a certain context. With the same intuition, the model proposed here has the structure of *mini-column* [14] but adopts *concept-centered representation* (see Sec. 2.1.2) instead of distributed representation: each *event* corresponds to a single *mini-column*. In this paper, the model can be interpreted by NAL: a link between two neurons is interpreted as a statement of temporal implication/equivalence with a *truth-value*. The strength of a link is modified via *temporal induction*, and future events are anticipated by *temporal deduction*. A *mini-column* corresponds to a *concept* in NAL, and a neuron's being activated corresponds to partial *meanings* of the *concept* being recalled. Due to the properties of NAL [8, 15], the model is naturally capable of handling uncertainty, and the model's behaviors and internals are fully understandable by human beings.

The model is tested on *prediction* tasks, where the input is a list of events, and the model is expected to predict future events. The list is assumed to have no beginning and no end (though in practice, usually there has to be a start-point), so that it is impossible for the model to memorize all the contents. With this assumption, the learning procedure should be *online* and *life-long* [16]. An example of input is "$(..., \$, A, B, C, D, \$, \$, X, B, C, Y, \$, ...)$", where "\$" denotes a random *event*, while characters denote different types of events. It is noted that the types of *events* are not predetermined before a system is initialized but dynamically constructed by the model. In this example, there are two prototypes of sequences, "(A, B, C, D)" and "(X, B, C, Y)", meaning that sequence "(A, B, C)" is always followed by event D, but by observing only "(B, C)" either D or Y is probable to occur immediately. In Sec. 3, the lengths and the number of prototypes vary in several cases, in order to test the capacity of the model. In the meanwhile, *catastrophic forgetting* [17] is a difficult problem in models with distributed representation (*e.g.*, in neural networks). The qualitative results show that the model proposed does not suffer from *catastrophic forgetting*.

2 Methods

This paper presents a novel biologically plausible sequence learning model with a theoretical foundation distinct from mainstream deep learning approaches. One motivation for developing this new model is the lack of interpretability in deep learning. In contrast, the model is based on a logical framework. Traditional logic has faced challenges with symbol grounding [9], making it difficult to apply in learning from experience. However, over the past four decades, a new type of logic, Non-Axiomatic Logic (NAL) [8], has been developed, which combines the strengths of both symbolic and connectionist approaches, ensuring both interpretability and adaptability. Despite its potential, NAL has not been widely applied in AI research and has primarily developed within a small community. This paper attempts to apply NAL to sequence learning.

Inspired by the structure of the brain, a "conceptual network" is designed and implemented for sequence learning based on the logic. According to a theory [18], the Neocortex consists of many *cortical columns*, each comprising numerous *mini-columns*, which are columnar structures made up of neurons. In this paper each *mini-column* is modeled as a *concept*, and each neuron within a *mini-column* as a *contextual concept* that represents the *concept* in a specific context. To illustrate this, consider the word "bank" in English, which has different meanings depending on contexts: (1) "To save money, I go to the bank every month." (2) "On the bank of the river locates a restaurant." The word "bank" corresponds to a *mini-column* (*i.e.*, a *concept*) in the brain. In each context, "bank" activates different neurons within the *mini-column*, representing different *contextual concepts*. The connections between *contextual concepts* represent predictive relationships. For instance, given sequences "(A, B, C, D)" and "(X, B, C, Y)", when *concept* C is activated, the following *concept* to be anticipated depends on whether C is in the context of "ABC" or "XBC". In the two contexts, C specifically activates either $C^{(1)}$ or $C^{(2)}$, where the connection from $C^{(1)}$ to $D^{(1)}$ predicts *concept* D, and the connection from $C^{(2)}$ to $Y^{(1)}$ predicts *concept* Y.

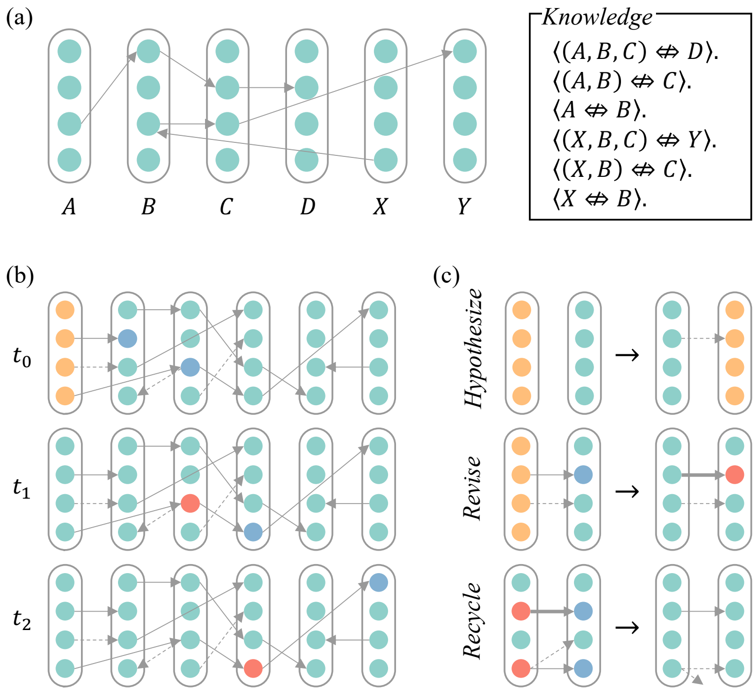


Fig. 1: Model diagram. (a) An example of the learned network. There are six *concepts*, A through D, X, and Y. Each *concept* is represented as a *column* that contains multiple *nodes*. There are *links* between *nodes*. Multiple *links* constitute a *chain*, representing a group of knowledge. For instance, *chain* "$(A^{(3)}, B^{(1)}, C^{(2)}, D^{(2)})$" represents three beliefs, "$\langle(A, B, C) \not\Leftrightarrow D\rangle$.", "$\langle(A, B) \not\Leftrightarrow C\rangle$.", and "$\langle A \not\Leftrightarrow B\rangle$.". (b) An example of inference procedure. It shows the internals of the model at three consecutive time-steps. At t_0, *concept* X is activated. Since there is no context, all the *nodes* are activated. It anticipates $A^{(2)}$ and $B^{(3)}$ to occur for the next time-step. *Node* $A^{(2)}$ is not anticipated even if there is a *link* from $X^{(3)}$ to $A^{(3)}$, because the *truth-value* (*i.e.*, the strength of the *link*) is too low. At t_1, *concept* B is activated. *Node* $B^{(3)}$ is activated due to the anticipation, while the other *nodes* in B remain silent. In the meantime, $C^{(3)}$ is anticipated. Similarly, at t_2, $C^{(3)}$ is activated, and $Y^{(1)}$ is anticipated. (c) The learning mechanism. When there is no *link* between two *concepts*, some *links* are built as *hypotheses* (see Sec. 2.2.1). When one or two *nodes* at both ends of a *link* are activated, the *truth-value* in the *link* is revised according to distinct situations (see Sec. 2.2.2). When the number of *links* exceeds a threshold, one or some of them are deleted (see Sec. 2.2.3).

Based on this intuition, the network structure shown in Fig. 1a is adopted. In this network, each column represents a *concept*, and each node within a column represents the *concept* in a specific context, referred to as a *contextual-concept*. Connections between nodes represent *predictive relationships* between *contextual concepts*. As a series of *events* are input, *concepts* are sequentially activated. Ideally, if an event sequence is highly frequent, a chain will form in the *conceptual network*, representing the memorized sequence. When a part of this sequence appears, a relevant node in the chain is predicted. If the prediction is correct, the node is activated, and the network continues to predict the next node in the chain.

It is assumed that the input events form an infinite list, meaning that learning from the list repeatedly is not allowed. Instead, the model modifies the network locally with each event input in real-time, including structural changes (*i.e.*, creating or deleting connections) and adjusting connection strengths. Specifically, as previously mentioned, connections represent predictive relationships, denoted in NAL as "$(A \not\Leftrightarrow B).\langle f, c\rangle$," meaning both B always appears after A and A before B, with a *truth value* of $\langle f, c\rangle$. Here, f represents the *correctness* of the prediction, and c indicates *confidence* in this *correctness*. The *truth value* represents connection strength. When a new connection is established, *confidence* c is initially low, suggesting the predictive relationship is a *hypothesis*. If event B follows event A as anticipated, the system accumulates *positive evidence*, increasing both f and c. If B does not follow A, *negative evidence* accumulates, decreasing f while increasing c, indicating greater *confidence* that A is not followed by B. This process aligns with the Hebbian rule and spike-timing-dependent plasticity (STDP) learning mechanisms in neuroscience [19], despite differences in computational details. Due to the *Assumption of Insufficient Knowledge and Resources* [20], the number of a concept's predictive relationships must be limited, thus, less valuable connections are forgotten by the system. Connections with low f values are hardly worthwhile for adapting to the environment, as unlikely events are not a concern. This summarizes the self-organization of the *conceptual network*, involving *hypothesizing*, *revising*, and *recycling*. The following sub-sections formalize this process, and Sec. 3 presents the test results of the model, validating preliminarily its practical effectiveness.

2.1 Representation

The correspondence between neurons and concepts has been a longstanding topic of discussion in neuroscience and brain science, with the most famous example being the so-called "grandmother cell" – the idea that a concept is encoded by a group of neurons. This paper also aims to establish a unification of structure and principle, positing that the same model can be interpreted and represented from both neural and logical perspectives, with an intrinsic correspondence between the two. From this viewpoint, neural network and *conceptual network* are "two sides of the same coin," rather than one being constructed upon the other or one emerging from the other.

2.1.1 Neural Representation

Previous research in neuroscience has discovered the presence of columnar arrangements of neurons in the brain, known as mini-columns [14]. In a brain-inspired model, the activation of these neurons is context-dependent [7]. The proposed model adopts a similar intuition. Additionally, research on spiking neural networks has shown that neurons can exist in multiple states, including *resting* (or *non-active*), *depolarized* (or *predictive*), and activated (or *active*) states[1] [19]. In this model, a neuron represents an event, with the *depolarized* state indicating an anticipation of the event and the *activated* state indicating the event has been observed.

When an event occurs, the corresponding *mini-column* processes it, considering all possible sequences the event could belong to. The question then arises: which sequence does the current event correspond to? In other words, which neuron within the *mini-column* should be activated, given that a neuron represents a component of a specific sequence? Without context, all neurons would be activated, suggesting the system speculates all sequences occur simultaneously, even though in reality, only one sequence occurs at a time. The system refines its guesses through subsequent observations, gradually eliminating incorrect anticipations until only one remains, meaning that only one neuron in a *mini-column* is activated. This refinement process primarily relies on testing anticipations. When a neuron is activated, its connected neurons enter a *depolarized* state, supposing the connection strength is sufficient. When a *mini-column* is activated, neurons in the *depolarized* state are preferentially activated, while neurons in the *resting* state are inhibited. This *lateral inhibition* phenomenon is also observed in neuroscience [19]. Since only one event occurs at a time, most anticipations are incorrect, and only the correct anticipation is retained. Consequently, most *depolarized* neurons return to the *resting* state, while a few (often only one) *depolarized* neurons become *activated*, reducing multiple superposed sequences into one. This mechanism acts like a "filter," gradually retaining the correct sequence being observed.

Formally, suppose that the ith neuron in *mini-column* c, in which there are n_c *neurons*, is denoted by $N_c^{(i)}$. The rule of activating *mini-column* c is shown in Eq. 1,

$$A_c^{(i)} = \begin{cases} 1 \text{ if} & \forall j \in \{1, ..., n_c\}, \hat{A}_c^{(j)} = 0 \\ 1 \text{ if} & \hat{A}_c^{(i)} = 1 \\ 0 \text{ if} & \hat{A}_c^{(i)} = 0, \text{and } \exists j \in \{1, ..., n_c\}, \hat{A}_c^{(j)} = 1 \end{cases} \tag{1}$$

where $A_c^{(i)}$ indicates the *active*/*resting* state of neuron $N_c^{(i)}$ ($A_c^{(i)} = 1$ if in *active* state, and $A_c^{(i)} = 0$ if in *resting* state), while $\hat{A}_c^{(i)}$ indicates the *depolarized* state of neuron $N_c^{(i)}$ ($\hat{A}_c^{(i)} = 1$ if in *depolarized* state, otherwise, $\hat{A}_c^{(i)} = 0$).

Neurons are connected by synapses. Each synapse is plastic, meaning that its strength is changeable. Some learning rules regarding synapse were proposed, such as *Hebbian* learning, STDP, [19]*etc.* If the strength is greater than a threshold (denoted

[1] An elaborate model of spiking neuron with other states is much more complex, but here is the simplified spiking neuron which contains necessary parts for the sequence learning model.

as θ here), pre-synaptic neuron's activation would lead to post-synaptic neuron's activation, otherwise, the synapse is a *potential* connection waiting for being strengthened. The depolarization procedure is expressed by Eq. 2,

$$\hat{A}_{c_1}^{(i),t} = \begin{cases} 1 \text{ if } A_{c_2}^{(j),t-1} = 1 \text{ and } W_{c_1(i)}^{c_2(j)} > \theta \\ 0 \text{ otherwise} \end{cases} \tag{2}$$

where $\hat{A}_{c_1}^{(i),t}$ indicates the depolarized state of neuron $N_{c_1}^{(i)}$ at time-step t, $A_{c_2}^{(j),t-1} = 1$ denotes the active state of neuron $N_{c_2}^{(j)}$ at time-step $t-1$, and $W_{c_1(i)}^{c_2(j)}$ is the strength of the synapse connecting neuron $N_{c_1}^{(i)}$ to neuron $N_{c_2}^{(j)}$.

The rule of modifying synaptic strength is not explicitly presented here. Generally speaking, it is similar to *Hebbian* rule: a synapse is strengthened if its pre-synaptic and post-synaptic neurons are activated simultaneously, and is weakened if only one of the neurons is activated in a short duration. The learning rule in this paper is a variant of the *Hebbian* rule (see Sec. 2.1.2 and Sec. 2.2).

Different from the HTM theory [7], in which an event is represented by *sparse distributed representation*[2] [7] , the model in this paper adopts *concept-centered* representation (see 2.1.2), so that the model can work in a human-understandable way.

2.1.2 Logical Representation

As mentioned above, this paper attempts to establish a correspondence between neurons and *concepts*. Specifically, the *neuronal network* and the *conceptual network* are describable using different knowledge-representations but fundamentally identical. The *conceptual network* discussed here is similar to traditional "semantic networks" and "knowledge graphs" but is essentially different. In traditional knowledge graphs, each node represents a concept, and edges represent relationships between concepts, akin to this model. However, traditional knowledge graphs are confronted with the symbol grounding problem and are often arbitrarily organized. Non-Axiomatic Logic (NAL) provides the theoretical foundation for the *conceptual network* described in this paper, offering a normative framework for the network's self-organization process.

More specifically, a *mini-column* corresponds to a *concept*, which can have different meanings in different contexts. In this model, we use *contextual concepts* to capture this nuance. When a *concept* is activated, all *contextual concepts* are considered, to determine which specific meaning is relevant in the current context. Without relevant context, all *contextual concepts* are activated, suggesting that all relavant sequences occur simultaneously and predicting which *contextual concepts* will be activated next. Similar to the *neural representation*, if certain *contextual concepts* within a *concept* are anticipated, they are preferentially activated, while unanticipated *contextual concepts* are inhibited.

The prediction and activation of *contextual concepts* are governed by NAL. A connection between *contextual concepts* corresponds to a *predictive equivalence statement* in NAL: "A is of predictive equivalence to B" indicates a causal relationship between

[2] Briefly speaking, a *sparse distributed representation* in HTM is a binary vector with a little amount of elements to be 1 and the others to be 0.

A and B. The strength of the connection is represented by the *truth value* of this *statement*, which does not decay over time but is updated as evidence accumulates. Meanwhile, the activation state of *contextual concepts* is also represented by *truth values*. When a *contextual concept* A is activated, *frequency* and *confidence* in its *truth value* are high, with *confidence* decaying over time. In this model, *confidence* decays to zero after one time step, meaning the influence of a prediction does not last long. Considering a *predictive equivalence statement*, when A is activated, B is predicted with its *truth value* calculated based on the *deduction rule* of NAL. If A or B is activated, the *induction rule* of NAL is applied to accumulate evidence. The following provide a more formalized description.

The schematic diagram of the representation approach is shown in Fig. 1a. A *column* is interpreted as a *concept*. Within each column, there are several *nodes*. A *node* is interpreted as a *contextual concept* that is comprised of a *truth-value* and relationships with other *contextual concepts*. "A (*contextual*) *concept* is activated" means that the system is perceiving or feeling something at a time. For example, when seeing a red flower, a *concept* which represents the red flower raises up, in other words, it feels the red flower. *Truth-value* of a (*contextual*) *concept*, which represents the extent of the system's perceiving or feeling, is composed of two parts, *frequency* (denoted as f) and *confidence* (denoted as c), represented by a two-dimensional tuple $\langle f; c\rangle$. *Frequency* measures ratio of *positive evidence* among all observations, and *confidence* reflects the impact of future evidence[3]. An *event*, in this sense, is not what occurs outside the mind but the subjective experience of the occurrence.

In NAL, the temporal relations between two *concepts* E_1 and E_2 include *predictive implication* "$\langle E_1 \mathrel{/\!\!\!\Rightarrow} E_2\rangle$", *retrospective implication* "$\langle E_2 \mathrel{\backslash\!\!\!\Rightarrow} E_1\rangle$", and *predictive equivalence* "$\langle E_1 \mathrel{/\!\!\!\Leftrightarrow} E_2\rangle$". A sequence of *events* can be represented as "$(E_1, E_2, ..., E_n)$".

As shown in Fig. 1a, a chain of nodes represents multiple beliefs simultaneously. For example , "$\langle A^{(1)} \mathrel{/\!\!\!\Leftrightarrow} B^{(1)}\rangle$", "$\langle (A^{(1)}, B^{(1)}) \mathrel{/\!\!\!\Leftrightarrow} C^{(3)}\rangle$", and "$\langle (A^{(1)}, B^{(1)}, C^{(3)}) \mathrel{/\!\!\!\Leftrightarrow} D^{(4)}\rangle$" shares the same chain.

Given two *events* $E_1.\ \langle f_1; c_1\rangle$ and $E_2.\ \langle f_2; c_2\rangle$, and their corresponding occurrence time t_1 and t_2 such that $t_2 < t_1$, the *temporal induction rules* in NAL includes

$$\{E_1\ \langle f_1; c_1\rangle, E_2.\ \langle f_2; c_2\rangle\} \vdash E_2 \mathrel{/\!\!\!\Rightarrow} E_1\ \langle F_{ind}\rangle \quad (3)$$

$$\{E_1\ \langle f_1; c_1\rangle, E_2.\ \langle f_2; c_2\rangle\} \vdash E_1 \mathrel{\backslash\!\!\!\Rightarrow} E_2\ \langle F'_{ind}\rangle \quad (4)$$

$$\{E_1\ \langle f_1; c_1\rangle, E_2.\ \langle f_2; c_2\rangle\} \vdash E_2 \mathrel{/\!\!\!\Leftrightarrow} E_1\ \langle F_{com}\rangle \quad (5)$$

where F_{ind} and F_{com} are the *induction functions* which map the truth-values of premises to that of conclusion[4].

For each *event*, the *truth value* is constant (*e.g.*, $\langle 1.0; 0.9\rangle$) in this paper, though they could be revised dynamically in future work. The *truth-value* of an anticipation

[3]In NAL, there is no "absolute truth", and the truth of a judgement is evaluated by the evidence the system has observed. Suppose there are w^+ pieces of positive evidence and w^- negative evidence, then the total amount of evidence is $w = w^+ + w^-$. *Frequency* is measured by $f = w^+/w$, while *confidence* is measured by $c = w/(w + k)$, where k is a constant.

[4]In $\langle F_{ind}\rangle$, $w^+ = f_1 f_2 c_1 c_2$ and $w = f_2 c_1 c_2$; In $\langle F'_{ind}\rangle$, $w^+ = f_1 f_2 c_1 c_2$ and $w = f_1 c_1 c_2$,; in $\langle F_{com}\rangle$, $w^+ = f_1 f_2 c_1 c_2$, $w = (1-(1-f_1)(1-f_2))c_1 c_2$. *Frequency* and *confidence* are then calculated by $f = w^+/w$ and $c = w/w + k$.

can be derived by the *temporal deduction rule* in NAL, *i.e.*,

$$\{E_1\ \langle f_1; c_1\rangle, E_1 \not\Leftrightarrow E_2.\ \langle f_2; c_2\rangle\} \vdash E_2\ \langle F_{ded}\rangle \tag{6}$$

where F_{ded} is the *deduction function*[5]. When a *concept* is activated, that is, a corresponding *event* "$E.\langle f; c\rangle$" occurs, all the *contextual concepts* "$E^{(i)}\langle f^{(i)}; c^{(i)}\rangle$" ($i \in 1, ..., n$, where n is the number of *contextual concepts* of *concept* E) are observed as a fact represented by a *truth-value*, so that the *revision* rule is applied to merge the two *truth-values* of fact and anticipation. The *revision* rule in NAL is

$$E\langle f_1; c_1\rangle, E\langle f_2; c_2\rangle \vdash E\langle F_{rev}\rangle \tag{7}$$

where F_{rev} is *revision function*[6]. It is implied that an anticipated *event* has higher $confidence$ when it actually occurs.

When a *concept* is activated, which *contextual concept* to be activated depends on the *expectations* of the *truth values*. In NAL, the *expectation* of *statement* "$S\langle f; c\rangle$" is

$$e(S) = F_{exp}(f, c) = c(f - 0.5) + 0.5 \tag{8}$$

There exists a threshold ζ, such that the *expectation* of an occurring but not anticipated *event*, or an anticipated but not occurring *event*, is less than ζ, while the *expectation* of an occurring and anticipated *event* is greater than ζ. A *contextual concept* is activated when its *expectation* is greater than ζ, or when all the *expectations* of *contextual concepts* in a *concept* are less than ζ, *i.e.*,

$$A_c^{(i)} = \begin{cases} 1 \text{ if} & \forall j \in \{1, ..., n_c\}, e(E_c^{(j)}) < \zeta \\ 1 \text{ if} & e(E_c^{(j)}) > \zeta \\ 0 \text{ if} & e(E_c^{(i)}) < \zeta, \text{and } \exists j \in \{1, ..., n_c\}, e(E_c^{(j)}) > \zeta \end{cases} \tag{9}$$

The procedure of *temporal induction* for *statement* "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_1}^{(j)}$" happens only when $A_{c_1}^{(i)} = 1$ or $A_{c_2}^{(j)} = 1$.

Although predictive implication ("$\not\Rightarrow$") and retrospective implication ("$\not\Rightarrow$") are also important, as a start point, the model in this paper exploits merely predictive equivalence ("$\not\Leftrightarrow$") for learning and inference.

2.1.3 Graph Representation

We have seen in Sec. 2.1.1 and Sec. 2.1.2 that the model can not only be explained as a neuronal network, but also be interpreted by a logic. However, to better illustrate it, I have to use a more abstract representation (which is called *Graph Representation*[7]) to avoid conceptual ambiguity in the description, mainly by using the formal language of *Graph Theory*. As shown in Fig. 1, the basic elements are *node* and *link* (*a.k.a.* vertex and directed edge in *Graph Theory*). A *column* (as hyper-vertex) is a collection

[5] In $\langle F_{ded}\rangle$, $f = f_1 f_2$ and $c = f_1 f_2 c_1 c_2$

[6] In $\langle F_{rev}\rangle$, $w^+ = w_1^+ + w_2^+$, $w^- = w_1^- + w_2^-$, and $w = w^+ + w^-$

[7] This might not be a suitable term, but I was not able to find a better one.

of *nodes*. Each *link*'s weight is adjustable. Each *node* has three states, *activation*, *non-activation*, and *pre-activation*, each of which is represented by a pair of real numbers (*i.e.*, *truth-value* in Sec. 2.1.2) ranging from 0 to 1. The states of real number can be binarized by a threshold (see Eq. 9).

The correspondence among the terms in the three representations is shown in Tab. 1.

Graph Repr.	Neural Repr.	Logical Repr.
node	neuron	contextual-concept
column	mini-column	concept
link	synapse	temporal statement (*e.g.*, "$\langle A \not\Rightarrow B\rangle$")
link-weight	synaptic strength	truth-value (*abbr.*, t.-v.)
weight adjustment	synaptic plasticity	temporal-induction and revision
activation	active state	event with t.-v. "$(1.0; 0.9)$"*1
pre-activation	depolarized state	anticipation with t.-v. "$(1.0; 0.9)$"*1
non-activation	resting state	event with t.-v. "$(1.0; 0.1)$"*2

*1 Here in the truth-value, *frequency* and *confidence* are both very high, though the concrete values does not have to be the same as $(1.0; 0.9)$.

*2 Here in the truth-value, *confidence* is very high but *frequency* does not matter, and the concrete values does not have to be the same as $(1.0; 0.1)$.

Table 1: The correspondence of terms among the three representations: *Neural Representation*, *Logical Representation*, and *Graph Representation*.

2.2 Sequence Learning

The challenge is how to construct the *links* given a series of *events*. First, there is no way to fully-connect among all nodes (meaning that a node has links connected to all other nodes). Facing an endless list of events, the number columns cannot be pre-determined (thus, new columns should be able to be built up dynamically), and the number of links would explode as the number of columns increases with fully-connection. Due to the Assumption of Insufficient Knowledge and Resources (AIKR) [20], the number of links connected to or from a node should not exceed constant (though it could be either large or small), consequently, there has to be a certain mechanism through which new *links* are created with old *links* to be recycled. In Sec. 2.1.2, the logic rules of temporal *induction* has been introduced, however, when to do induction and to revise the link remains to be answered in the following.

2.2.1 Hypothesizing

Initially, there are no *nodes* and no *links* in the network. Whenever an *event* occurs, the corresponding *column* is constructed if there does not exist one. Each *node* in a *column* has no links at the beginning of its creation. When two *columns* are activated in succession, two sets of nodes are activated correspondingly. Suppose a set of nodes $\mathcal{N}_1$ in column C_1 and a set of nodes $\mathcal{N}_2$ in column C_2 are activated, then one node $E_{c_1}^{(i)}$

is picked out from $\mathcal{N}_1$, and another one $E_{c_2}^{(j)}$ from $\mathcal{N}_2$, a new link is created connecting from $E_{c_1}^{(i)}$ to $E_{c_2}^{(j)}$ if there does not exist one. Since the initial weight of the link, represented by truth-value, is very weak, *i.e.*, the *confidence* is low (*e.g.*, $c = 0.1$). The link, represented by "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)} \langle 1.0; 0.1 \rangle$", in this sense is what we usually mean by *hypothesis*.

When picking out a node for hypothesizing from a set, which one to pick? Let us consider the *meaning* of a node. A node is activated given a context of events, thus, intuitively, a node means a concept in a certain context. Ideally, there should be at most one *pre-link* pointing into it and at most one *post-link* pointing out from it, and there should at least one link possessed by it. In this case, the node would be activated if and only if one certain context of events occur. For example, a node $B^{(1)}$ is activated only when the sequence "(A, B, C, D)" occurs, and $B^{(1)}$ exactly identifies the event B in that context, rather than the B in "(X, B, C, Y)". However, due to AIKR, the number of nodes should be a constant, so that a node has to serve for multiple different contexts. We can say the meaning of a node is *clear* or *unambiguous*, *either* if it has a post-link with strength much greater than other post-links, and a pre-link with strength much greater than other pre-links, *or* if one of its pre-link and post-link is much greater than its other links. To pick out one for hypothesizing, the overall principle is to avoid as far as possible to do harm to the *clear* meaning of a node. The concrete strategy in this paper is to pick out a node with the lowest *utility*. Here, the utility of node $E_c^{(i)}$ is defined as

$$u(E_c^{(i)}) = 1 - (1 - u_1)(1 - u_2) \tag{10}$$

where

$$\begin{aligned} u_1 &= \begin{cases} 0 & \text{, if } \mathcal{L}_{pre}(E_c^{(i)}) = \varnothing \\ \max\limits_{\forall L \in \mathcal{L}_{pre}(E_c^{(i)})} e(L) & \text{, otherwise} \end{cases} \quad \text{, and} \\ u_2 &= \begin{cases} 0 & \text{, if } \mathcal{L}_{post}(E_c^{(i)}) = \varnothing \\ \max\limits_{\forall L \in \mathcal{L}_{post}(E_c^{(i)})} e(L) & \text{, otherwise} \end{cases} \end{aligned} \tag{11}$$

where $\mathcal{L}_{pre}(E_c^{(i)})$ and $\mathcal{L}_{post}(E_c^{(i)})$ are the sets of node $E_c^{(i)}$'s pre-links and post-links correspondingly, and $e(L)$ is the *expectation* of the truth-value of link L (see Eq. 8). Thus, if a node has a much clear meaning, it tends not to be picked out. Though the side effect is that a node, which has ambiguous meaning but has a link with strong strength, is also inclined to be selected, it seems not an issue in practice.

New *hypotheses* are constantly come up with, though they do not lead to strong conclusion until enough evidences are collected. A too weak hypothesis like "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)} \langle 1.0; 0.1 \rangle$" leads to non-activation of its consequent $E_{c_2}^{(j)}$, according to Eq. 9, in this sense, a hypothesis is a *potential* link between two nodes. This potential link is similar to a synapse with a low strength. Only when the strength is greater than a threshold, the synapse can be viewed as truly connected and can transit signals between two neurons. Nevertheless, a link can be strengthened whether it is strong or weak, so that a *hypothesis* has chance to become stronger and cause the activation of its consequent given the activation of its antecedent.

2.2.2 Revising

Whenever a column is activated, one link is picked out for revising. The general principle to enhance the link is that is the most probable to become conclusive. Specifically, the selected link L is

$$L = \underset{\forall L \in \mathcal{L}_{pre}(E_c^{(i)}), \forall i \in \{1,...,n_c\}}{\text{argmax}} e(L) \tag{12}$$

meaning that it picks out a link with the maximal *expectation* from all the pre-links of all the nodes within a column. As a result, a link represented by statement "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}$" is selected for revising.

Given two nodes $E_{c_1}^{(i)}$ and $E_{c_2}^{(j)}$ which are concerned on, the temporal induction rule is applied to revise the truth-value of statements including "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}$", according to Eq. 3 and Eq. 7. The difference in the learning procedure is that some negative evidences are obtained if an anticipated event does not occur. Specifically, node $E_{c_2}^{(j)}$ (as consequent) is in the state of *pre-activation* at time-step t (*i.e.*, $\hat{A}_{c_2}^{(i),t} = 1$) if node $E_{c_1}^{(i)}$ (as antecedent) is in the state of *activation* at time-step $t-1$ (*i.e.*, $A_{c_1}^{(i),t-1} = 1$), and the expectation of statement "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}$" is greater than a threshold θ, *i.e.*,

$$\hat{A}_{c_2}^{(i),t} = \begin{cases} 1 & \text{, if } A_{c_1}^{(i),t-1} = 1, \text{ and } e(E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}) > \theta \\ 0 & \text{, otherwise} \end{cases} \tag{13}$$

An event usually may have multiple causes and effects, however, in the sequence learning model, It is possible to build *causal chains* in which each event has at most only one cause and at most one effect in a certain context. Therefore, in the case of $\exists j \in \{1, ..., n_c\}, e(E_c^{(j)}) > \zeta$ (*i.e.*, the current event has a certain context, such that one or some of nodes within a column are activated but not all), if node $E_{c_2}^{(j)}$ is anticipated ($\hat{A}_{c_2}^{(i),t} = 1$) but not activated ($A_{c_2}^{(i),t} = 0$), then some negative evidences of $E_{c_2}^{(j)}$ are collected. Similarly, when node $E_{c_2}^{(j)}$ is activated, all of its possible causes are paid attention to. If an antecedent $E_{c_1}^{(i)}$ is not activated before $E_{c_2}^{(j)}$, given statement "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}$", then some negative evidences of $E_{c_2}^{(j)}$ are also collected. Otherwise, the two nodes are activated in succession, temporal induction can be applied directly. [8]

There is also a punishment in the case $\forall j \in \{1, ..., n_c\}, e(E_c^{(j)}) < \zeta$ (*i.e.*, all the nodes within a column are activate without being anticipated first). If the whole column is activated, a bunch of anticipations would occur. For those anticipations which are not verified later, a slight amount of evidences are collected by $w^- = |\{e(L) > \theta, \forall L \in \mathcal{L}_{post}(E_{c_1}^{(i)})\}| + b$, where the first term denotes the number of $E_{c_1}^{(i)}$'s post-links each of whose *expectation* is greater than threshold θ, and the second term b is a constant (*e.g.*, $b = 40$); here, threshold θ and constant b are hyper-parameters of the model. When $b = \infty$, it means no penalty for this case.

[8] In practice, A simplified but equivalent implementation is adopted. If nodes $E_{c_1}^{(i)}$ and $E_{c_2}^{(j)}$ are activated in succession, then an amount of positive evidences, $w^+ = p^+$, are collected for statement "$E_{c_1}^{(i)} \not\Leftrightarrow E_{c_2}^{(j)}$". However, if only one of the nodes is activated, then some negative evidences , $w^- = p^-$ are collected. Here constants p^+ and p^- are hyper-parameters of the model, typically $p^+ = p^- = 1$.

2.2.3 Recycling

Again due to AIKR, the number of links regarding a node should not exceed a certain threshold, otherwise, some of the links should be dropped. This is related to the forgetting process of memory.

Pre-links $\mathcal{L}_{pre}(E_{c_1}^{(i)})$ and *post-links* $\mathcal{L}_{post}(E_{c_1}^{(i)})$ of node $E_{c_1}^{(i)}$ are stored in a priority queue, sorted by *utility*. In the current design, *utility* of a *link* is determined by the *expectation* of its *truth-value*. When the number of *link* n_L in the priority queue is greater than a certain threshold ξ (*e.g.*, $\xi = 100$), the exceeding part is recycled, *i.e.*, $(n_L - \xi)$ links with the lowest priorities are deleted.

Utility probably not only depends on the *expectation* of a *link*'s *truth-value*, but also some other factors. For example, a link may be reserved in a short period after it is newly created, though its *expectation* is much less than some long-standing links (see *Future Work* for more discussion in Sec. 4).

3 Results

Since the tasks, sequence *prediction* and sequence *generalization*, are equivalent to each other, while the sequence *recognition* task can be reduced to *prediction*[2], the test-cases in this paper only involves sequence *prediction*. In the meanwhile, sequence *decision making* task [2] can be considered as associated with much more complex procedures of intelligence, thus, *decision making* is not considered in this paper, though the model proposed in this paper is the foundation of further work.

The model is tested on some synthetic datasets. Different from typical approach for evaluating machine learning methods, there is no explicit division between training set and test set here, since it is considered that the learning process is *online* and *life-long*: each sample observed by an agent is not only a training sample but also a test one, and current experience is not necessary to be similar to the past, that is to say, no stable distribution of data is assumed.

The data for tests are manufactured in the following way. Suppose there are n_r types of events; each event is labeled by a *term*, such as characters like "*A*", "*B*", "*X*", and strings like "*e0347*", "*e1001*", which identifies the type of an *event*. Whatever a *term* looks like for human developers, it is just the name of a *concept* inside the system, a *concept* whose meaning merely depends on its acquired (rather than predetermined) relations with other *concepts*, as suggested in Non-Axiomatic Logic (NAL) [8]. A dataset in this paper is a list of *events*, which contains sequences like "$(..., A, B, C, D, ...)$", "$(..., X, B, C, Y, ...)$", and so on. Some *events* are determined by their predecessors, for example, in a given situation, *event* "B" is always followed by "C" but comes after either "A" or "X"; event "D" follows "(A, B, C)", but given merely "(B, C)", either "D" or "Y" is expected to occur. Besides, other events are randomly generated, leading to the whole list of events unpredictable to some extent.

With this form of input data, three aspects are considered for evaluating the model, *capacity* (see Sec. 3.1), *catastrophic forgetting* (see Sec. 3.2), and *capability* (see Sec. 3.3), though the capability aspect is analyzed only in theory.

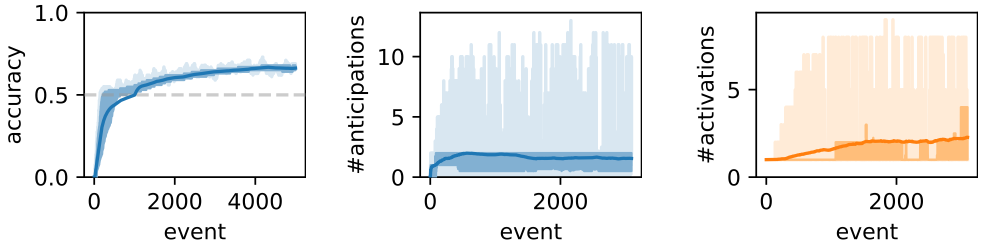
3.1 Capacity Tests

Evaluating the capacity of the model is related to two factors, the number of sequences and the length of a sequence that is expected to be recognized. In a test, datasets are generated, within the prototype of "($\$, ..., \$, E_1, ..., E_m, \$, ..., \$$)", where $E_1, ..., E_m$ are deterministic events which keep the same in every sample of the prototype, while "\$" is the unpredictable variable, which varies from sample to sample; The dataset's parameter m denotes the number of deterministic events for each sequence. There are p pieces of prototypes of sequences to be generated. When an event occurs, the model anticipates some events to occur for the next step. If an event is anticipated and occurs immediately, then we can say the event is correctly anticipated. The proportion of the number of truly anticipated events within a certain past period (*e.g.*, the past 100 time-steps) is the anticipation accuracy of the current time-step.

Firstly, a simple case is tested. Suppose each event is named by a single character (from A to Z), so that there are 26 possible types of events for the model. The dataset contains two prototypes of sequences "($\$, \$, A, B, C, D, E, \$$)" and "($\$, \$, X, B, C, D, Y, \$$)", where "\$" denotes a random event. In this case, $m = 5$ and $p = 2$), and only 50% of the events are deterministic and can be predicted very well. The test results are shown in Fig. 2. Figure 2a shows the accuracy of anticipation as time goes by. At each time-step, there could be multiple anticipations, and Fig. 2b shows the number of events anticipated by the model – ideally, there should be only one anticipated event if the system is pretty sure what context it observes; multiple anticipated events implies that the system retains the possibility of several contexts. We can see that around 2 events are anticipated on average for each time-step. Figure 2c shows the number of activated nodes in the model – generally speaking, a node's activation means a certain context is recognized by the model. The fewer nodes are active, the clearer the context is. It shows in Fig. 2c, there are around 2 nodes activated on average for each time-step.

Secondly, the model is tested with different options of length m and the number of prototypes p, and even the different numbers of types of events. The proportions of unpredictable events in the datasets are all 50%. As shown in Fig. 3, the model has proper anticipations on future events. With $m = 5$ and $p = 5$ (see Fig. 3a), as well as $m = 14$ and $p = 20$ (see Fig. 3d), the anticipation accuracy in either cases is greater than 50%, exceeding the theoretically highest accuracy (the same as that shown in Fig. 2a). It is speculated for two reasons: for the random part of the dataset, there is still a chance to get the correct answer by random selection; second, the model learns some patterns from the random events. The number of anticipated nodes and that of active nodes are both no more than 2 in each of the two cases.

A probably simpler setting for the model is that the number of types of events is much greater than 26. The test results are shown in Fig. 3g-3i, where the number of types n_r is 1000. We can see that the the accuracy is closely around 50%, and the number of anticipated nodes and that of active nodes are both around 1. The model performs better in this setting than the previous ones, because in the previous tests, one type of event most probably engages in multiple prototypes of sequences, so that the model may be confused; while in this test, the types of events are much greater, so that one type of event get a higher chance to be involved in a single context, consequently, it is much easier to memorize and distinguish different patterns for the model.



(a) (b) (c)

Fig. 2: Capacity-Test results for the simple case, where the prototypes of sequences are "($, $, A, B, C, D, E, $)" and "($, $, X, B, C, D, Y, $)", where "$" denotes a random event. (a) The accuracy of anticipation. (b) The number of anticipations. (c) The number of active nodes.

3.2 Catastrophic Forgetting Tests

The issue of *catastrophic forgetting* (also known as *catastrophic interference*) was proposed by McCloskey and Cohen in 1989 [17], pointing out that distributed representation of connectionist networks (*a.k.a. Deep Neural Networks* nowadays) have a non-desirable property that modifications on new data interfere the memory for old data, leading to forgetting large amount of the previous experience. Some modern research (*e.g.*, [21]) over the years still tried to solve this issue.

Since the model in this paper adopts *concept-centered* representation (see Sec. 2.1.2), this annoying property seems probably not to occur in theory. However, due to relatively insufficient resources of memory and computation [20] assumed in this paper, the model has to remember something new and forget something old, thus, acquiring new knowledge is possible to interfere old one, and the extent of interference should be evaluated (at least qualitatively if not quantitatively), to see whether it is catastrophic.

The test results of catastrophic-forgetting is shown in Fig. 4 and Fig. 5. With 26 types of events (in Fig. 4) or 1000 types (in Fig. 5), in an episode, 20 prototypes of sequences with length 14 for each are generated. The patterns vary across different episodes. After seeing three episodes one by one, the model encounters the previous episodes repeatedly. If there existed catastrophic forgetting in the model, then we would have seen the anticipation accuracy fell down significantly when seeing an episode with the same patterns once again. However, that does not happen in Fig. 4 and Fig. 5. Therefore, qualitatively speaking, the model does not suffer from catastrophic forgetting.

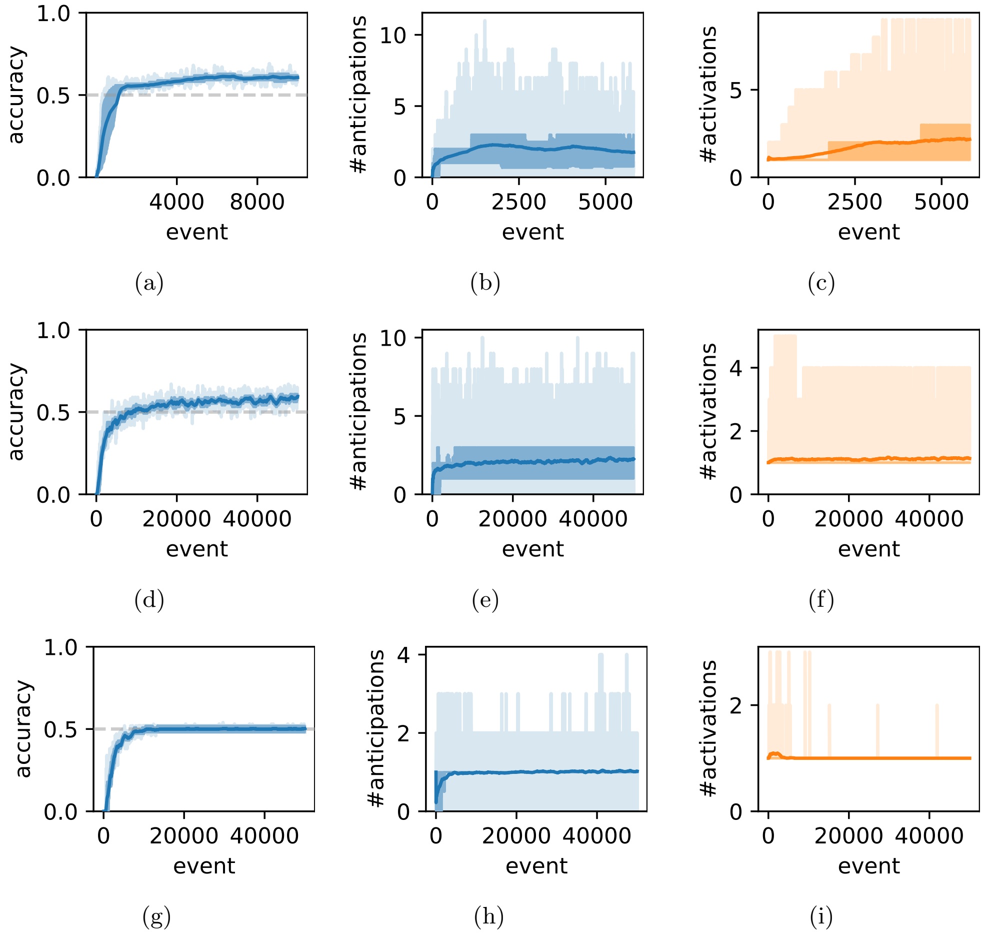


Fig. 3: Capacity-Test results with different options of length m and the number of prototypes p, and event the different numbers of types of evens. (a), (b), and (c) are test results with $m = 5$, $p = 5$, and 26 types of events. (d), (e), and (f) are test results with $m = 14$, $p = 20$, and 26 types of events. (g), (h), and (i) are test results with $m = 14$, $p = 20$, and 1000 types of events.

3.3 Capability Analysis

The bound of the model's capability needs to be clarified. First, the model is not an AGI system, although it can be considered as a first step on modeling the complex unity of intelligence. Second, the model focuses on an aspect of intelligence, *i.e.*, *sequence learning*. Third, in the current design, the model can only deal with the situation where merely one single event appears at a certain time-step; the case where multiple events appear simultaneously is not the target in this paper. Forth, the time interval of any pair of events is a constant; this assumption on the interval enables the

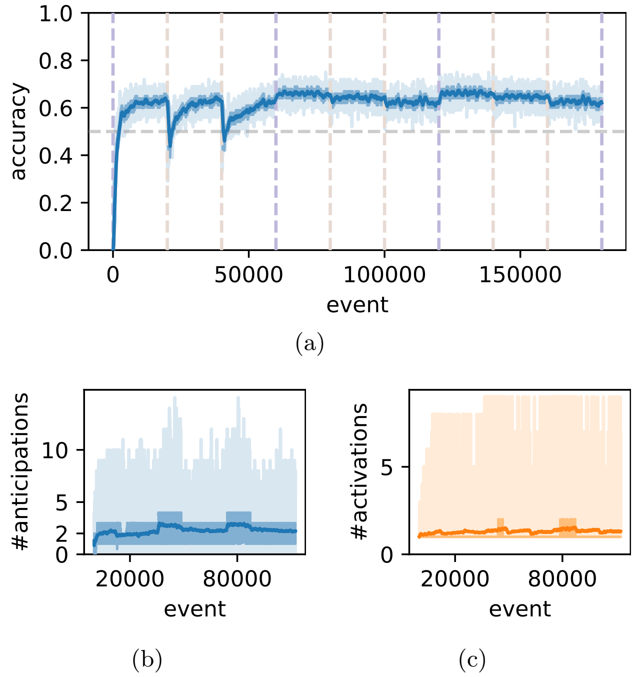


Fig. 4: Results of catastrophic forgetting test when the number of concepts $n_r = 26$ (see Sec. 3.2 for more details).

model to deal with some situations where the order of events matters but time interval does not; an example of this kind of situations is natural language processing. Part of future work is to expand the capability of the model (see *Future Works* in Sec. 4).

Inside the bound, the model is capable of learning patterns from an endless list of events. In the meanwhile, the model is enabled by the power of Non-Axiomatic Logic to handle uncertainty. This property (*i.e.*, being able to handle uncertainty) is directly derived in theory, thus, no test is needed to prove that in practice, and exploiting the property is more related to applications of the model.

4 Discussion

Implications. In this part, endeavors are made to establish wide connections to other research. Firstly, This paper proves the potential of learning patterns of sequences based on a logic, the problem which was addressed well by purely statistical models (*e.g.*, Hidden Markov Models [4]), neurodynamics models (e.g., HTM [7], spiking neural networks [22]), and neural networks (*e.g.*, RNN [5]). The model is fully interpretable by a logic, *i.e.*, Non-Axiomatic Logic [8], as a result, human developers are capable of explaining the system's behaviors by recording, in a human-understandable way, and checking its internal activities. It provides an alternative

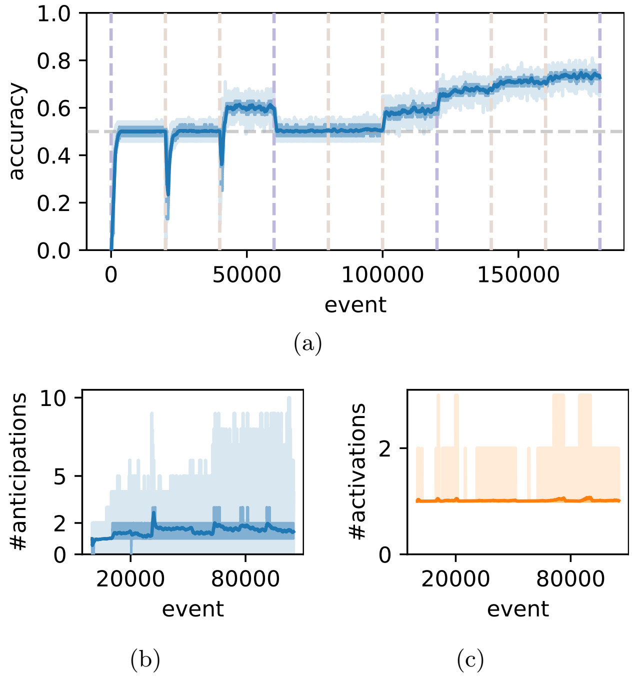


Fig. 5: Results of catastrophic forgetting test when the number of concepts $n_r = 1000$ (see Sec. 3.2 for more details).

besides well-performed but inexplicable models, especially neural networks that are widely criticized as black boxes.

Secondly, how concepts are structured and represented in the brain has been extensively investigated for a long time in AI, Psychology, and Philosophy (*e.g.*, [23–27]). This paper provides a potential neuronal basis of logic and concepts. There are two possibilities of how logic emerges from the human brain, one is that neural networks might learn something which is called logic, the other is that neural activities could be interpreted as logic. Evidently, this work supports the latter one, though it does not negate the former one. As shown in Sec. 2.1 and Tab. 1, the model can be illustrated in both two ways, a neural one and a logical one. The correspondence between membrane voltage of a neuron and truth-value of a statement is not discussed deliberately, though I guess there would emerge some valuable work on this issue.

Thirdly, from the learning mechanism proposed in this paper, the following principle can be summarized:

> *Computational resources tend to converge toward knowledge with lower levels of uncertainty.*

Specifically, in the model, a *link* with higher *truth-value* gets a greater chance to be enhanced. This principle is not a novel idea. It could be a guidance for designing AI systems, and it is also observed in biological systems: In neuroscience and neuronal dynamics, the well-known *winner-take-all* rule [19, 28, 29] shares the same intuition. In psychology, Piaget's theory suggests that new information input to a subject is incorporated into already existing knowledge[30]. In other words, the existing knowledge will be allocated computational resources to give a meaning to the content. Earlier, in ancient China, as it is said in *Tao Te Ching*[9], "The *Way of Nature* reduces excess and replenishes deficiency. By contrast, the *Way of Humans* is to reduce the deficient and supply the excessive." [10] The learning process of the proposed model exactly follows the *Way of Humans*.

Comparisons. The previous works are valuable, but there are still some differences from this model. The practical performances of various models are not compared in this paper, majorly due to two reasons. First, of course, the model proposed here is a preliminary one and is not powerful enough: at least a hierarchy should have been learned by the model to deal with some complex situations. Thus, it does not make much sense to apply it to some complex tasks, such as natural language processing. Second, there are some different theoretical assumptions in this paper. It assumes that the types of *events* are unknown to a system before it is initialized, as a result, corresponding representations should be generated in the run-time. In typical Hidden Markov Models (HMMs) [4], the types of *events* should be pre-specified and cannot be changed when a system starts running: specifically, the shape of the state transition matrix and the size of the observations set should be predetermined, so that new events cannot be handled by HMMs; in contrast, facing new events, new concepts are constructed in the model proposed. In neural networks, an event is represented by a vector, and theoretically there is no restriction of handling new events (assuming there is no fully-connected network in the system, since the events that the system can handle are defined by the output layer of that fully-connected network). Nonetheless, it usually assumes that data are known satisfactorily to a system, so that the system can see the the whole data set repeatedly. In this work, the assumption is that the data set is endless, thus, the model has to learn in real time. Besides, the interpretability of this model is an attractive property, especially compared to neural networks. The performance of the model proposed is similar to HTM [7], though in HTM adopts quite different theoretical foundations. There are some advantages of distributed representations in HTM, for example, robustness to noise and damage. In contrast, by adopting the *concept-centered* representation, uncertainty can be represented naturally.

Limitations. The model is capable of distinguishing the patterns with the same head and tail but different middle parts, *e.g.*, "(A, B, C, D)" and "(X, B, C, Y)", however, I do not think it learns a hierarchical structure from a sequence. A pattern is implicitly stored in the memory, in the form of *chain*, rather than *tree* in computer science. "Chunk" is the basis of building a hierarchy [1], and a *chain* could be a hint or heuristic to form a chunk. As suggested in previous works (*e.g.*, [31] and [31]), forming hierarchical representations of sequences benefits for accessing and self-repairing learned sequences. The model proposed suffers from the problem of memorizing a long sequential pattern: a long *chain* is broken into smaller ones in the learning procedure,

[9] A reference translation – Lao-Tzu, Addiss, S., Lombardo, S., Watson, B.: Tao Te Ching, copyright 1993 edn. Hackett Publishers, Indianapolis (1993).

[10] In Chinese – 《道德经》云："天之道，损有余而补不足；人之道则不然，损不足以奉有余。"

as a result, some events in a long sequence cannot be anticipated correctly. This problem, I believe, can be solved by imposing hierarchical structure to the model – *Nodes* in a *chain* are combined as a *chunk* (i.e., a *compound* in NAL [8]), which simultaneously serves as a *column* and is processed in the same way as in this paper.

Another limitation of the model is the lack of the capability to retrospect past events, as a critical feature in natural language processing. Consider the "bank" example from Sec. 2: when determining the meaning of "bank," we not only rely on the preceding context but also infer its meaning based on the subsequent context. For instance, encountering the word "river" allows us to retroactively infer that "bank" refers to a riverbank. The current model does not yet incorporate such a bidirectional inference mechanism, although its absence is not apparent in simple sequence prediction examples.

The capabilities of the proposed *conceptual network* are also limited by the lack of evidence that it addresses the problem of *representation learning*. In neural networks, *representation learning* involves compressing large amounts of data into shorter vectors, which is crucial for tasks such as image perception. Sequence learning and representation learning are fundamentally different problems. In sequence learning, it is usually assumed that the representations of events are distinguishable, and the learning objective is to establish the sequential relationships among events. Nonetheless, *conceptual networks* might be able to handle the problem of representation learning. The relationship between input data and learned representations in neural networks can be viewed as a combinatorial relationship between sets of *concepts*. However, the mechanisms for representation learning within *conceptual networks* remain to be explored.

Future Works. There is a great deal of work to be done in the future, related to either improving the sequence learning model *per se* or modeling other intelligence phenomena upon the model proposed.

On the one hand, apart from introducing the hierarchical structure mentioned above, the major improvement might involve the competition among links. In the current design, links compete with each other based on its *utility*, however, *utility* of a link should have depend on several factors besides its *expectation*, such as time that a link is established (more specifically, a link built in the recent time tends not to be forgot, even if its *expectation* is low), the goal of a system (therefore, there should be a top-down interaction with the model), and so on. In addition, predictive implication and retrospective implication have not been considered much in the current design, and the learning mechanism could be modified to improve the efficiency of learning. Certainly, the current design of the model is not perfect enough. For example, in Fig. 3b, I hope the number of anticipations should be close to 1, meaning that the system clearly know the context it locates in. The current result shows that there are around 2 possible contexts, and the system cannot determine which is correct exactly, though it knows to some extent.

On the other hand, sequence learning could be the basis of sensorimotor learning, where the input is high-dimensional data (typically, 2-D image), and an agent perceives its environment by glimpsing different places (*a.k.a.* eye-movement). How to built a interpretable sensorimotor learning model seems a big challenge. Also, multiple events might occur simultaneously, and how to deal with concurrent events deserves further research.

5 Conclusion

In this paper, a brain-inspired model of sequence learning is proposed. The model exploits Non-Axiomatic Logic (NAL) [8] as the basis of representation, inference, and learning. The model is brain-inspired since it mimics the *mini-column* structure in the Neocortex [12–14].

Even if the model is inspired by human brain, in AI systems, it still needs the answer why it should be of the structure like that, rather than a trivial answer as "because the brain looks like that". The reason why to use the *mini-column* structure is that it represents a *concept* with distinct meanings under different contexts. A *neuron* in a *mini-column* is activated because part of the meanings of the corresponding *concept* is recalled. *Neurons* are connected as a *chain*, representing *concepts* organized as a sequence. Due to the *Assumption of Insufficient Knowledge and Resources* (AIKR) [20], the total number of links should not exceed a constant, thus, a balance between memorizing and forgetting does matter; there is no way to memorize the whole dataset (to do *offline learning* [16]) as well as all possible sequences, in the mean while, the time complexity should be a constant when handling each *event* input to the system. Based on this view, the learning mechanism in Sec. 2.2 is designed, comprising of three steps: *hypothesizing*, *revising*, and *recycling* – The *hypothesizing* and *recycling* procedures are responsible for resources allocation, to guarantee that the model satisfies AIKR. In the *revising* procedure, candidate *links* are picked out for *temporal induction* and *revision* that are logical rules in NAL. To predict future *events*, the *temporal deduction rule* is applied to generate anticipations. The model can be converted to *Narsese*, the formal language of NAL, so that the model is fully interpretable, explainable, and even trust-worthy.

The dataset for test is assumed to be an endless list of events, meaning that theoretically there is no explicit head or tail of the list; thus, the model has to do the so called *online learning* [16] and work in *real-time* in a sense. The dataset is generated synthetically, with 50% predictable events and 50% random events. For the predictable part, a certain number (denoted as p) of prototypes of sequences are generated; each of the prototype has a certain length (denoted as m). To test the capacity of the model, with different ms and ps, the model is asked to make anticipations on next events. The correctness of anticipations is plotted in Fig. 2 and Fig. 3, showing that the model performs well in several settings with different requirements for capacity, achieving or exceeding 50% accuracy, which approximately is the highest accuracy in theory. In addition, another task, *a.k.a.* continual learning [21], is used to test whether the model suffers from *catastrophic forgetting* [17], which is a long-standing problem in models with distributed representation, such as neural networks. The problem of *catastrophic forgetting* does not occur in the model proposed, as shown in Fig. 4 and Fig. 5. This is because the model exploits logical representation (or concept-centered representation, see 2.1.2), through which modifying one *concept* or its relevant connections does not intervene other irrelevant *concepts*, though forgetting is inevitable due to insufficient resources.

To conclude, this paper demonstrates the potential of learning sequential patterns in a logical way, though there is some interesting work for further researching.

Declarations

Data Availability. The source code is available at "https://github.com/bowen-xu/SeL-NAL".

Acknowledgments. I thank those who reviewed this article for their suggestions; especially, I discussed a lot with my advisor, Dr. Pei Wang[11], on the idea and the work proposed in this paper, and I appreciate his comments and advice.

[11] https://cis.temple.edu/~pwang/

Compliance with Ethical Standards

Funding. (Not applicable)

Conflict of Interest. The author of this article declares that he has no conflict of interest.

Ethical approval. This article does not contain any studies with human participants performed by any of the authors.

Informed consent. (Not applicable)

References

[1] Conway, C.M.: Sequential learning. In: Seel, N.M. (ed.) Encyclopedia of the Sciences of Learning Sequential Learning, pp. 3047–3050. Springer, LLC, 233 Spring Street, New York, NY 10013, USA (2012)

[2] Sun, R., Giles, C.L.: Sequence learning: From recognition and prediction to sequential decision making. IEEE Intelligent Systems **16**(4), 67–70 (2001)

[3] Clegg, B.A., DiGirolamo, G.J., Keele, S.W.: Sequence learning. Trends in Cognitive Sciences **2**(8), 275–281 (1998) https://doi.org/10.1016/S1364-6613(98)01202-9

[4] Rabiner, L., Juang, B.: An introduction to hidden markov models. IEEE ASSP Magazine **3**(1), 4–16 (1986) https://doi.org/10.1109/MASSP.1986.1165342

[5] Medsker, L.R., Jain, L.: Recurrent neural networks. Design and Applications **5**, 64–67 (2001)

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[7] Hawkins, J., Ahmad, S.: Why neurons have thousands of synapses, a theory of sequence memory in neocortex. Frontiers in neural circuits, 23 (2016)

[8] Wang, P.: Non-axiomatic Logic: A Model of Intelligent Reasoning. World Scientific, Singapore (2013)

[9] Harnad, S.: The symbol grounding problem. Physica D: Nonlinear Phenomena **42**(1-3), 335–346 (1990)

[10] Wang, P.: Experience-grounded semantics: a theory for intelligent systems. Cognitive Systems Research **6**(4), 282–302 (2005)

[11] Wang, P., Hammer, P.: Issues in temporal and causal inference. In: Artificial General Intelligence: 8th International Conference, AGI 2015, AGI 2015, Berlin, Germany, July 22-25, 2015, Proceedings 8, pp. 208–217 (2015). Springer

[12] Wallace, M.N., Zobay, O., Hardman, E., Thompson, Z., Dobbs, P., Chakrabarti, L., Palmer, A.R.: The large numbers of minicolumns in the primary visual cortex of humans, chimpanzees and gorillas are related to high visual acuity. Frontiers in Neuroanatomy **16** (2022) https://doi.org/10.3389/fnana.2022.1034264

[13] Jones, E.G.: Microcolumns in the cerebral cortex. Proceedings of the National Academy of Sciences of the United States of America **97**(10), 5019–5021 (2000)

[14] Buxhoeveden, D.P., Casanova, M.F.: The minicolumn hypothesis in neuroscience.

Brain **125**(5), 935–951 (2002)

[15] Wang, P.: Confidence as higher-order uncertainty. In: Proceedings of the Second International Symposium on Imprecise Probabilities and Their Applications, Ithaca, New York, pp. 352–361 (2001)

[16] Hoi, S.C., Sahoo, D., Lu, J., Zhao, P.: Online learning: A comprehensive survey. Neurocomputing **459**, 249–289 (2021)

[17] McCloskey, M., Cohen, N.J.: Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In: Bower, G.H. (ed.) Psychology of Learning and Motivation vol. 24, pp. 109–165. Academic Press, New York, NY, USA (1989)

[18] Mountcastle, V.B.: The columnar organization of the neocortex. Brain **120 (Pt 4)**, 701–722 (1997) https://doi.org/10.1093/brain/120.4.701

[19] Gerstner, W., Kistler, W.M., Naud, R., Paninski, L.: Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition. Cambridge University Press, Cambridge (2014)

[20] Wang, P.: On defining artificial intelligence. Journal of Artificial General Intelligence **10**(2), 1–37 (2019)

[21] Zeng, G., Chen, Y., Cui, B., Yu, S.: Continual learning of context-dependent processing in neural networks. Nature Machine Intelligence **1**(8), 364–372 (2019)

[22] Liang, Q., Zeng, Y., Xu, B.: Temporal-sequential learning with a brain-inspired spiking neural network and its application to musical memory. Frontiers in Computational Neuroscience **14**, 51 (2020)

[23] Blouw, P., Solodkin, E., Thagard, P., Eliasmith, C.: Concepts as Semantic Pointers: A Framework and Computational Model. Cognitive Science **40**(5), 1128–1162 (2016) https://doi.org/10.1111/cogs.12265

[24] Tenenbaum, J.B., Kemp, C., Griffiths, T.L., Goodman, N.D.: How to Grow a Mind: Statistics, Structure, and Abstraction. Science **331**(6022), 1279–1285 (2011) https://doi.org/10.1126/science.1192788

[25] Lin, E.L., Murphy, G.L.: Effects of background knowledge on object categorization and part detection. Journal of Experimental Psychology: Human Perception and Performance **23**(4), 1153–1169 (1997) https://doi.org/10.1037/0096-1523.23.4.1153

[26] Fodor, J.A.: Concepts: Where Cognitive Science Went Wrong. Oxford University Press, New York, NY, USA (1998)

[27] Murphy, G.: The Big Book of Concepts. MIT Press, Cambridge, Massachusetts

London, England (2004)

[28] Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. Nature Neuroscience **2**(11), 1019–1025 (1999)

[29] Lee, D.K., Itti, L., Koch, C., Braun, J.: Attention activates winner-take-all competition among visual filters. Nature Neuroscience **2**(4), 375–381 (1999)

[30] Müller, U., Ten Eycke, K., Baker, L.: Piaget's Theory of Intelligence. In: Goldstein, S., Princiotta, D., Naglieri, J.A. (eds.) Handbook of Intelligence: Evolutionary Theory, Historical Perspective, and Current Concepts, pp. 137–151. Springer, New York, NY (2015)

[31] Lashley, K.S.: The problem of serial order in behavior. In: Cerebral Mechanisms in Behavior; the Hixon Symposium, pp. 112–146. Wiley, Oxford, England (1951)