A Model of Sequential Learning based on Non-Axiomatic Logic

Bowen Xu

Department of Computer and Information Sciences,
Temple University, Philadelphia, USA
bowen.xu@temple.edu

Abstract. Sequential learning is a fundamental function of an intelligent agent. This technical report introduces a model of sequential learning, which is interpretable through Non-Axiomatic Logic. The learning procedure includes three steps, hypothesizing, revising, and recycling, and can work under the Assumption of Insufficient Knowledge and Resources. Although there are limitations for the current design, the model has been proven effective in some simple cases.

Keywords: Sequential Learning · Non-Axiomatic Logic · Brain-inspired

1 Introduction

Sequential leaning, which is of paramount importance for an intelligent agent to interact with the world, refers to acquiring the proper ordering of *events* or stimuli[2]. It is the foundation of many learning processes, such as sensorimotor learning, natural language acquisition, and so on.

Some successful modern approaches addressed this issue. For example, Recurrent Neural Network[7], Transformer[8], and their variants have gained huge progress in natural language processing and computer vision. Through modeling neocortical column, the Hierarchical Temporal Memory (HTM) approach can memorize frequently occurring sequences as long as each *event* can be converted to Sparse Distributed Representation (SDR) [4]. Interpretability is an important aspect of AI security. The major issue of these neural approaches is their lack of interpretability: the models are black or grey boxes, and developers is hard to understand what is going on and how to fix it when unexpected behaviors occur. A sequence of *events* can also be represented in a logical way, through which a sequential learning model would be interpretable. In Non-Axiomatic Logic [9], there are some logical rules for temporal inference[12], including deduction, induction, *etc*. However, how to extract temporal patterns from sequences is still a challenge under this logical representation.

Before an agent could acquire the capability to predict an *event* occurring in the far future, we believe it should firstly be able to learn some basic patterns of sequences in which *events* occur during a relatively short term. Although under the context of Artificial General Intelligence (AGI), the issue goes far beyond learning a sequence, we might as well focus on sequential learning temporarily.

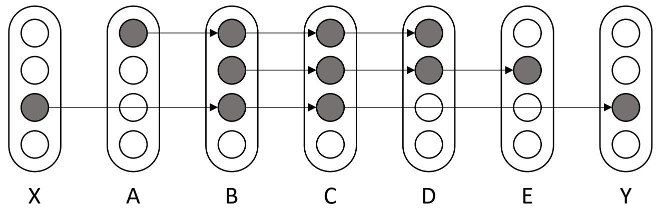


Fig. 1. Schematic diagram of the sequential learning model

2 Model

2.1 Representation

We use a *Narsese representation* as the major formal language for describing the model. Simultaneously, the model could also be described through *graph* as well as *neural representation*.

In sequential learning, a representation should be highly contextual. In HTM theory, a representation under a certain context is modeled by *mini-column*[1] of neocortex – a collection of neurons in several mini-columns constitute a distributed representation, *i.e.*, the whole collection represents a single object, while each component, within the collection, is involved in several objects' representations[5]. However, despite of the biological-plausibility, there seems to be no strong reason why SDR is necessary for intelligence. At the same time, how to deal with uncertainty is a challenge in this model[4]. In principle, a collection of *neurons* in SDR is equivalent to a *concept* in NAL. We believe the most critical intuition in HTM's *temporal memory* model is that a *column* is a collection of representations under multiple contexts, and each *neuron* in a column is a representation under a certain context. It is natural to think if we could use a single *node*, instead of multiple *nodes*, as a representation, simultaneously preserving the intuition of *temporal memory* in HTM. From another perspective, each *concept* in NARS (Non-Axiomatic Reasoning System)[3], an AGI system based on NAL, is *weakly contextual*, meaning that what *concept* to be activated is determined by the overall status of the system, while it is not determined directly by what is activated at present. As a result, we agree that NARS is good enough at modeling consciousness[11], however, it still needs to be improved for sequential learning. By exploiting the logic part of NARS, namely NAL, a representation can work with uncertainty, and new representations can be derived via well justified logical rules, promising the interpretability of the model.

In the following, we will introduce the sequential learning model proposed in this paper. The schematic diagram of the representation approach is shown in Fig. 1. A *column* is interpreted as a *concept*. Within each column, there are several *nodes*. A *node* is interpreted as a *task* that is comprised of a *statement*, a *budget*, and a *truth-value*. The *statement* is the identity of the task, meaning that the agent is perceiving or feeling something at a certain time. For example, when seeing a red flower, a *task* which represents the red flower raises up. The *truth-value* represents the extent of the agent's perceiving or feeling. A *node*, as a *task*, is especially an *event* since its *truth-value* is time-dependent. The *budget* represents the extent of computation resources allocated to the *task*; it is highly related to the attention of the agent. The event, in this sense, is not what occurring outside the mind but the subjective understanding of the occurrence. Even though a single *concept* corresponds to multiple *events*, under a certain context, usually there should be only one or very few *events* to activate, so that only part of *meaning* of the concept is utilized.

There could be a directed *link* between two *node*s. A *link* represents temporal relations, including *predictive implication* ($/\!\!\!\!\Rightarrow$), *retrospective implication*($\backslash\!\!\!\!\Rightarrow$), and *predictive equivalence* ($/\!\!\!\!\Leftrightarrow$) in NAL [9]. the relations between two *events* E_1 and E_2 involves three *prediction*s, "$\langle E_1 /\!\!\!\!\Rightarrow E_2\rangle$.", "$\langle E_2 \backslash\!\!\!\!\Rightarrow E_1\rangle$.", and "$\langle E_1 /\!\!\!\!\Leftrightarrow E_2\rangle$.". If *event* E_1 occurs and there is a *prediction* "$\langle E_1 /\!\!\!\!\Rightarrow E_2\rangle$.", then E_2 is anticipated to occur, and we call E_2 an *anticipation*. In this way, the model can be interpreted through the formal language of NAL, *Narsese*. On the other hand, the model can also be interpreted using a "neural language". A *node* is interpreted as a *neuron*, while a *link* is interpreted as a *synapse*. If *event* E is an *anticipation*, we call *neuron* E is *pre-active*. A *mini-column* is *active* if and only if any of its *neuron* is active, and so as the *pre-active* case. If a *mini-column* is *activated* when none of its *neuron* is *pre-active*, then all the *neurons* are *activated*, otherwise only those *pre-active neurons* are *activated*. By this representation, the *meaning* of a *concept* is highly dependent of previous occurring *events*, *i.e.*, we say a *concept* is *strongly contextual*.

2.2 Sequential Learning

Now the challenge is how to construct the links given a series of *events*. Due to the Assumption of Insufficient Knowledge and Resources (AIKR) [10], there is no way to store all the *events* and simultaneously handling all potentially possible links within limited time. In the model, when a stimulus is input, there are three steps to response. First, some hypothetical links would be generated; second, the link strengths are revised; finally, those useless links are recycled since the resource is limited.

The current design of the learning mechanism is described as the following.

Revising: Given two *nodes* E_1 and E_2 between which there is a *link*, there are three cases when revising the *truth-value*s of the *prediction*s within a *link*:

- E_1 is *active* after which E_2 is *active*. A positive evidence is provided to *prediction*s "$\langle E_1 /\!\!\!\!\Rightarrow E_2\rangle$." as well as "$\langle E_2 \backslash\!\!\!\!\Rightarrow E_1\rangle$.".
- E_1 is *active* after which E_2 is not *active*. A negative evidence is provided to *prediction* "$\langle E_2 /\!\!\!\!\Rightarrow E_1\rangle$."
- E_1 is not *active* after which E_2 is *active*. A negative evidence is provided to *prediction* "$\langle E_2 \backslash\!\!\!\!\Rightarrow E_1\rangle$.".

The other probable case "E_1 is not *active* after which E_2 is not *active*" makes no sense because no evidence can be provided to the predictions.

Given new evidences, the *revision rule* [9] is applied to the corresponding *predictions*, and the *deduction rule* [9] is applied to generate *anticipations*.

Hypothesizing: Before links could be revised, they should be constructed first through the hypothesizing process. We focused on two *columns*, C_1 and C_2 activated in succession:

- If some but not all *nodes* of C_1 are *active*, and so as C_2, then skip the hypothesizing process since there have been some hypotheses for revising.
- If some but not all *nodes* (denoted as N_1) of C_1 are *active*, and all *nodes* of C_2 are *active*, then randomly pick up some *nodes* (denoted as N_2) of C_2, and build links between N_1 and N_2.
- If all *nodes* of C_1 are *active*, and some but not all *nodes* (denoted as N_2) of C_2 are *active*, then randomly pick up some *nodes* (denoted as N_1) of C_1, and build links between N_1 and N_2.
- If all *nodes* of C_1 and C_2 are *active*, then randomly pick up some *nodes* (denoted as N_1) of C_1, and some *nodes* (denoted as N_2) of C_2, and build links between N_1 and N_2.

For each time, the total number of new links is restricted to a certain value.

Recycling: Due to AIKR, the number of links within a column should not exceed a certain threshold, otherwise, some of the links should be dropped. This refers to the forgetting process of memory. The *priority* in *budget* determines the extent of a link tending to be recycled, and the *quality* in *budget* is related to long-term memory, *i.e.*, the link with high *truth-value* leading to high *quality* tends to be preserved. A new link usually has relatively high *priority*, which gradually decays to the *quality*. In this way, the balance between long-term and short-term memory is achieved.

The *link*s between *concept*s are revised through the *Revising* procedure which is well justified. Since developers are able to figure out what is going on inside the model and trace-back any unexpected behaviors, the model is *interpretable* in this sense.

3 Experiment

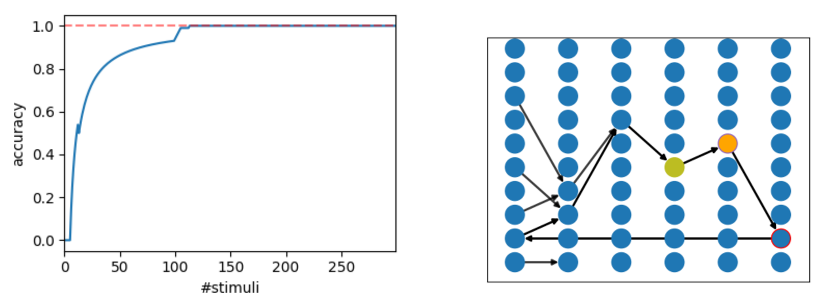
A sequence of characters is input into the model, and the prediction accuracy is measured and compared to the theoretically highest one.

Setting 1: The simplest setting is $[C_1, C_2, .., C_m, C_1, C_2, ..., C_m, ..., C_1, C_2, ..., C_m]$, *i.e.*, the sub-sequence $[C_1, C_2, ..., C_m]$ is repeated for many times, where C_i is a constant character. For example, the sub-sequence could be $[A, B, C]$ if $m = 3$. In this case, the expected highest accuracy of prediction is 100%. The result with $m = 6$ is shown in Fig. 2.

Setting 2: A more complex setting is that the sub-sequence is $[V_1, C_2, .., C_{m-1}, V_m]$ where V_1 and V_m are variable characters rather than constant. The variable characters V_1 and V_m are sampled for k times, so that k sub-sequences are generated. The sub-sequences are repeated for n times, and for each time one of the k sub-sequences is selected. For example, two samples $[A, B, C, D]$ and $[X, B, C, Y]$ are generated when $m = 4$, and the final sequence is the combination of these two, such as $[A, B, C, D, X, B, C, Y, X, B, C, Y, ...]$. Given the context $[A, B, C]$, character D should be predicted, while given the context $[X, B, C]$, character Y should be predicted. If the context $[B, C]$ is given, both characters D and Y are predicted. In this case, the expected highest accuracy of prediction is 87.5%, since only $[B, C, D]$ and $[B, C, Y]$ are totally predictable, while A and X can be predicted with 50% accuracy. The result with $m = 6$ is shown in Fig. 2(a).

Setting 3: Further, some anonymous characters could be added into the sequence, so that the uncertainty increases. In this case, the sub-sequence is $[V_1, C_2, ..., C_{m-1}, V_m, \diamond_1, ..., \diamond_p]$, where $\diamond_i$ denotes a random character, which means it is sampled from any characters when generating the final sequence. An example is $[A, B, C, D, J, U, X, B, C, Y, W, L, A, B, C, D, S, C, ...]$ when $m = 4$ and $p = 2$. In this case, the expected highest accuracy of prediction is 50%, since only $[B, C, D]$ and $[B, C, Y]$ are totally predicable, while the other half of the characters are unpredictable. The result with $m = 6$ and $p = 4$ is shown in Fig. 2(b).

We can see that under the experimental settings, the model can always achieve the best performance. However, there are still some limitations due to the design of learning mechanism. For example, as shown in Fig. 4, given various lengths and the numbers of sequences, the model with current design is weakly scalable, and it still leaves room for improvement.



(a) (b)

Fig. 2. The accuracy (a) and learned network (b) under setting 1

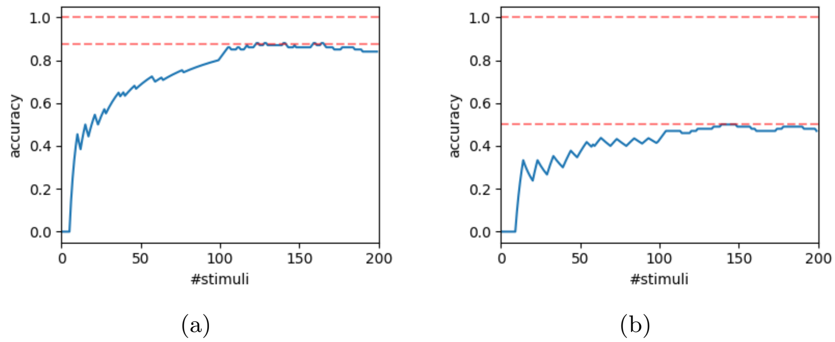


Fig. 3. The accuracy under setting 2 (a) and setting 3 (b)

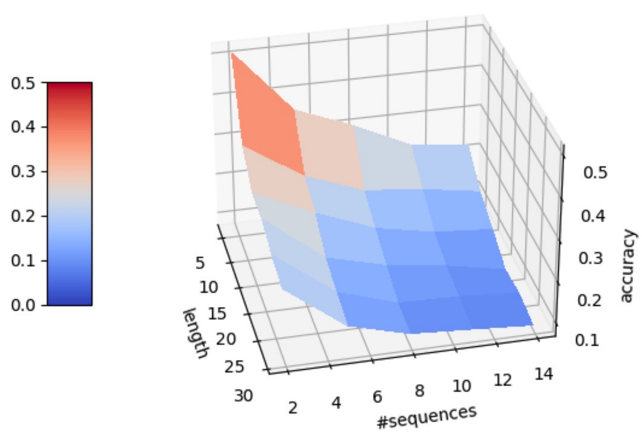


Fig. 4. Accuracy with various lengths and numbers of sequences

4 Summary

In this paper, an interpretable model of sequential learning is proposed, with a representation that uptakes the features of Non-Axiomatic Logic and a brain-inspired approach. The learning procedure of the model involves three steps, hypothesizing, revising, and recycling, and it can work under the Assumption of Insufficient Knowledge and Resources (AIKR) [10]. The model is tested with some slightly contrived problem, and it proves practically to be effective in the simple cases. However, the current design is far from perfect, and there is still some issues, especially scalability, with the model. Nevertheless, we can see the potential of the model which deserves further research.

There are also some interesting issues derived from current research. One is the connection between *principle* and *structure*: the model can be described in both two ways, neural and logical, and there might be some potential connection between *temporal induction* in NAL and synapse learning mechanisms, *e.g.*, STDP[6], in spiking neural network. Another is the potential of exploiting quantum computing to enhance the model. A *column* in the model has the meaning of multiple possibilities within different contexts, so that a *column* can be interpreted as a quantum superposition state[13]. This perspective might lead to some interesting work.

References

1. Buxhoeveden, D.P., Casanova, M.F.: The minicolumn hypothesis in neuroscience. Brain **125**(5), 935–951 (2002)
2. Conway, C.M.: Sequential learning. In: Seel, N.M. (ed.) Encyclopedia of the Sciences of Learning Sequential Learning, pp. 3047–3050. Springer Science & Business Media, LLC, 233 Spring Street, New York, NY 10013, USA (2012)
3. Hammer, P., Lofthouse, T., Wang, P.: The opennars implementation of the non-axiomatic reasoning system. In: Artificial General Intelligence: 9th International Conference, AGI 2016, New York, NY, USA, July 16-19, 2016, Proceedings 9. pp. 160–170. Springer (2016)
4. Hawkins, J., Ahmad, S.: Why neurons have thousands of synapses, a theory of sequence memory in neocortex. Frontiers in neural circuits p. 23 (2016)
5. Hawkins, J., Ahmad, S., Purdy, S., Lavin, A.: Biological and machine intelligence (bami). Initial online release 0.4 (2016)
6. Markram, H., Gerstner, W., Sjöström, P.J.: Spike-timing-dependent plasticity: a comprehensive overview. Frontiers in synaptic neuroscience **4**, 2 (2012)
7. Medsker, L.R., Jain, L.: Recurrent neural networks. Design and Applications **5**, 64–67 (2001)
8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
9. Wang, P.: Non-axiomatic logic: A model of intelligent reasoning. World Scientific (2013)
10. Wang, P.: On defining artificial intelligence. Journal of Artificial General Intelligence **10**(2), 1–37 (2019)

11. Wang, P.: A constructive explanation of consciousness. Journal of Artificial Intelligence and Consciousness **7**(02), 257–275 (2020)
12. 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. Springer (2015)
13. Yanofsky, N.S., Mannucci, M.A.: Quantum computing for computer scientists. Cambridge University Press (2008)