# Universal Learning Machine – Principle, Method, and Engineering Model Contributions to ICIS 2018

Chuyu Xiong

# Universal Learning Machine – Principle, Method and Engineering Model Contributions to ICIS 2018

Chuyu Xiong ⋆

Independent Researcher
4 Merry Lane, Jericho, NY 11753, USA
chuyux99@gmail.com

**Abstract.** Universal learning machine is a computing system that can automatically learn any computational task. Universal learning is very important research topic, where many disciplines cross. Here we discuss the principles, methods and engineering models of universal learning machine. X-form is the central concept and tool, which is introduced by examining objective and subjective patterns in details. We also discuss conceiving space and governing space, data sufficiency, learning strategies and methods, and engineering model.

**Keywords:** Universal learning machine, subjective pattern, X-form, conceiving space, governing space, learning dynamics, data sufficiency, squeeze to higher abstraction, embed to parameter space

## 1 Introduction

Universal learning machine is a computing system that could automatically learn any computational task without special presetting or manual intervention, if we provide it with sufficient data to drive the learning. But, why do we need universal learning machine?

A computer is a machine that processes information, and an intelligent computer is a machine that can process information more flexibly. Thus, it is natural to ask the question: Where does a computer's ability to process information come from? Currently, the ability mostly comes from human programming. This is surely effective. However, as the well-known AlphaGo demonstrated, we can not manually program a Go software to win over human players, but we can build a system that can acquire a Go program that wins over all human players by learning from inputing data. This fact and many other similar facts tell us that we are coming to the end of manual programming and we need to make computing systems that can learn from its input data and acquire information processing ability automatically.

⋆ Great thanks for whole heart support of my wife. Thanks for Internet and research contents contributers to Internet.

However, not all computing systems that can learn from its input are universal learning machine. For example, a Go program, which can learn playing Go game, is an effective learning machine, but not a universal learning machine. It can only learn playing Go, nothing else. Also, it needs a lot of manual presetting and manual intervention.

Machine learning is a hot research field at present, but there is no established principle and method for universal learning machine and universal learning theory. The current theory of machine learning, such as deep learning, focusing on certain special fields, is not good enough to explain the results it achieved, let alone guide universal learning. In short, we need to have a theoretical framework for universal learning. Thus, we started studies on universal learning [2]. We have found that, in order to achieve universal learning, computing system must have a specific structure, namely, X-form. Actually, by using X-form, we can understand better about what deep learning is doing [3].

Here we discuss X-form, conceiving space, governing space, primary consciousness, learning dynamics, data sufficiency, learning strategies, engineering model. We show that, in theory, universal learning machine can be made.

Remark, this article is the English and short version of our article in Chinese [5]. Due to restriction to size, many details are skipped, please check [5] [2]

## 2 Universal Learning Machine

We start with the definition of the learning machine.

**Definition 1 (Learning Machine).** *: An N-M information processing unit (IPU) having an input space (N bits), an output space (M bits), and a processor. An IPU processes the input information through the processor and outputs it. If an IPU modify its ability its own processor during the information processing, according to a set of simple and fixed rules, this IPU is called a learning machine.*

The phrase "a set of simple and fixed rules" is used to excludes human intervention. Without loss of generality, we can make discussion easier by only considering N-1 learning machine. Refer to [2].

**Definition 2 (Universal Learning Machine).** *For the learning machine $M$, assume that its current processor is $P_0$, and $P_1$ is another processor. If we have a data sequence $T$, and when we applied $T$ to $M$, eventually, the processor changes from $P_0$ to $P_1$, then we say that $M$ can learn $P_1$ starting from $P_0$. If for any given pair of processors $P_0$ and $P_1$, $M$ can learn $P_1$ starting from $P_0$, we say $M$ is a universal learning machine.*

Simply say, universal learning machine is a learning machine that can learn any information processing starting from any existing information processing.

Inside the learning machine, obviously, there is some kind of learning mechanism. More specifically, this learning mechanism captures the information embedded in the data sequence and uses that information to modify its own information processing. However, we need to distinguish between two situations:

1) the learning mechanism can only modify the information processing, and the learning mechanism itself is not being modified; 2) the learning mechanism itself is also being modified. But how can you exactly describe these two situations? Actually, we can do so by using data sequence.

**Definition 3 (Level 1 Learning Machine).** *Suppose $M$ is a universal learning machine. For any given pair of processors $P_0$ and $P_1$, by definition, there is at least one data sequence $T$, so that the learning machine learns $P_1$ starting from $P_0$. If the data sequence $T$ depends only on $P_0$ and $P_1$, and does not depend on any historical state of $M$, we call $M$ as a level 1 universal learning machine.*

By the same idea, we can also define level 0, which is not modifiable, and a level 2, whose learning mechanism is modifiable, etc.

Learning machines differ from most computing systems, instead of treating data by each bit, they treat data by pattern. Pattern is very important for the learning machine, and everything about the learning machine is related to pattern. Therefore, we need to understand patterns. But let's first explain what are spatial pattern and temporal pattern. We can briefly say that spatial pattern is independent of the time factor of the input (such as the order, the previous input, etc.), only related to the input itself, and temporal pattern is also related to the time factor of the input. The spatial pattern is relatively simpler. In this article, we will focus on the spatial patterns, and not discuss temporal pattern. Of course, we are very clear that temporal pattern is crucial. We will discuss it in later articles.

## 3 Subjective Pattern and X-form

### 3.1 Objective Pattern

Objective pattern is completely independent from the learning machine.

**Definition 4 (Base Pattern Space).** *N-dim base pattern space, written as $PS_N^0$ (PS stands for pattern space), is an N-dim binary vector space:*

$$PS_N^0 = \{b = (b_1, b_2, \ldots, b_N) | \quad b_k \in \{0, 1\}, k = 1, \ldots, N\}$$

Each element in $PS_N^0$ is a base pattern. $PS_N^0$ has a total $2^N$ base patterns. When $N$ is not very small, this is very huge. In fact, this hugeness is the root of the richness and difficulty of universal learning machine.

We skip details of objective pattern. We just state here: base patterns are starting point, most patterns are not base patterns; an objective pattern is a set of base patterns; operations on objective patterns are identical to set operations.

### 3.2 Subjective Pattern

The key is that pattern is not only related to the bits presented in the input data, but also how the learning machine perceives the input. For example, consider a simple 4-1 learning machine, which input space is 4-dim. When the input is (1,1,1,0), the learning machine can perceive that it is just a base pattern, but it can also perceive the input as that two base patterns (1,1,0,0) and (1,0,1,0) appear together, or other more complicated situations [2].

This simple fact and many more similar facts strongly tell us: Objective pattern alone is not good enough to understand a learning machine. Subjective pattern is more important. We must understand how learning machine subjectively perceive pattern.

**Definition 5 (Perception Bit).** *For the learning machine $M$, if there is a bit $pb$ inside it, which behaves in this way: there is a set $B \subset PB_N^0$, so for any $b \in B$, $pb = 1$, and for any $b \notin B$, $pb = 0$, then we call this bit $pb$ a processing bit of $M$. If $M$ has more than one such bit, i.e. there is a set: $\{pb_j|\ j = 1, \ldots, L\}$, where each $pb_j$ is a processing bit, we call this set as a processing bit set of $M$, or just processing bits, or perception bits.*

Since in a learning machine, processing is strongly related to its perception to input, we also use name perception bit. The perception bits reflect how a learning machine processes information and are its important structural elements. The perception bits of a learning machine can be quite complicated and not easy to see. However, we know that it does exist.

**Theorem 1 (Perception Bits Exist).** *For any learning machine $M$, its perception bits $\{pb_j|\ j = 1, \ldots, L\}$ exist and are not empty.*

We skip proof here. For a learning machine $M$, if $b$ and $b'$ are two different base patterns, i.e. two different N-dim binary vectors, then $M$ may have the same or different perception bits for $b$ and $b'$. If the perception bits are same, then $b$ and $b'$ are perceived same by $M$, even $b$ and $b'$ might be very different bit-wise. If $M$ perceives $b$ and $b'$ different, then $M$ must have different perception bits (at least one perception bit has a different behavior) for them. In this way, we have the following definition.

**Definition 6 (Base Pattern Perceived).** *Suppose $M$ is a learning machine, $\{pb_j|\ j = 1, \ldots, L\}$ are its perception bits, if for 2 base patterns $b_1$ and $b_2$, the values of their perception bits are $(pv_1^1, pv_2^1, \ldots, pv_L^1)$ and $(pv_1^2, pv_2^2, \ldots, pv_L^2)$, respectively, and for at least one $k$, $1 \le k \le L$, we have $pv_k^1 = pv_k^2 = 1$, we say that $M$ subjectively perceive $b_1$ and $b_2$ the same, at the peception bit $pb_k$.*

That is to say, for the two base patterns, if they are different at all perception bits, the learning machine cannot perceive them as same. However, if at least one perception bit, the value is the same, $M$ may subjectively perceive them to be the same. Of course, $M$ can also be subjectively perceive them to be different.

**Definition 7 (Pattern Perceived).** *Suppose $M$ is a learning machine, $\{pb_j|\ j = 1, \ldots, L\}$ are its perception bits. Then if $p$ is a set of base patterns, and at one perception bit $pb_k$, $1 \le k \le L$, any base pattern in $p$ will take the same value at $pb_k$, we say that $M$ subjectively perceives $p$ as one pattern at $pb_k$. That is to say, $p$ (a set of base patterns) is one subjective pattern for $M$.*

Note that in the definition, all the base patterns in $p$ need only behave the same at one perception bit. This is the minimum requirement. Of course this requirement can be higher. For example, to requiring to behave the same at all perception bits. These requirements all are subjective.

In a learning machine perception bits reflect how the learning machine perceives and processes pattern. Based on perception bits, we can understand subjective pattern. If the two base patterns behave the same at one perception bit, then they can be subjectively perceived identical. For any objective pattern, i.e. a set of base patterns $p$, if all of base pattern in $p$ behave the same at one perception bit, $p$ can be subjectively perceived as the same. This is where objective pattern connects to subjective pattern.

For a learning machine, the perception bits is changing with learning, their behavior changes, and their number changes.

Due to size, we skip details of operations of subjective patterns. But, they are super important. There are 3 subjective pattern operations ("$\neg$", "$+$", "$\cdot$"). Not like operations of objective pattern, which require no modifications inside the learning machine, operations of subjective pattern need to modify learning machine, often by adding perception bits and modify the behaviors.

### 3.3 X-form

There are 3 subjective operations ("$\neg$", "$+$", "$\cdot$") over subjective patterns. Their results are also subjective patterns. Naturally, we ask: if we apply the 3 subjective operations in succession, what will happen? Clearly, we will get an algebraic expression. But, what exactly is this algebraic expression? What should we do? What it can bring to us?

**Definition 8 (X-form as Algebraic Expression).** *Suppose $g = \{p_1, p_2, \ldots, p_K\}$ is a set of variables, and we repeatedly apply 3 operators "$\neg$", "$+$", "$\cdot$" over $p_1, p_2, \ldots, p_K$, we then construct an algebraic expression $E$, we then call $E$ as an X-form on $g$, which is written: $E = E(g) = E(p_1, p_2, \ldots, p_K)$.*

But, we can view such algebraic expression as subjective pattern (under certain conditions):

**Definition 9 (X-form as Subjective Pattern).** *Suppose $g = \{p_1, p_2, \ldots, p_K\}$ is a set of subjective pattern, and $E = E(g) = E(p_1, p_2, \ldots, p_K)$ is one X-form on $g$ (as algebraic expression). If all operations in $E$ indeed have support, this expression $E$ is a new subjective pattern.*

Further, such algebraic expression can be viewed as information processing:

**Definition 10 (X-form as Information Processor).** *Assuming $M$ is a learning machine, $g = \{p_1, p_2, \ldots, p_K\}$ is a set of subjective patterns subjectively perceived by $M$, and $E = E(g)$ is a X-form on $g$ (as algebraic expression), then $E(g)$ is an information processor that processes information like this: when a basic schema $p$ is put into $M$, since $M$ perceives this pattern, the subjective patterns $p_1, p_2, \ldots, p_K$ forms a set of boolean variables, still written as: $p_1, p_2, \ldots, p_K$, and* *when this set of boolean variables is applied to $E$, the value of $E$ is the output of the processor, and it is written as: $E(g)(p)$.*

So, we now understand the meaning of X-form, in several steps. Why do we call as X-form? These expressions are mathematical forms and have rich contents that we still have many unknowns. So following the tradition, we use X-form to name it. Theorem below connects objective pattern, subjective pattern and X-form. We skip the proof here.

**Theorem 2 (Objective and Subjective Pattern, and X-form).** *Suppose $M$ is an N-1 learning machine. For any objective pattern $p_o$ (i.e. a subset in $PS_N^0$), we can find a set of base pattern $g = \{b_1, b_2, \ldots, b_K\}$, and one X-form $E$ on $g$, $E = E(g) = E(b_1, b_2, \ldots, b_K)$, so that $M$ perceive any base pattern in $p_o$ as $E$, and we write as $p_o = E(g)$. We say $p_o$ is expressed by X-form $E(g)$. Moreover, we can also require that the number of base patterns less than $N$ (i.e. $K < N$).*

According to the definition of the X-form, it is easy to see that if operations ("+" or "·") are used to join several X-forms, a new X-form is formed. Similarly, it is easy to see, if we do a suitable segmentation on an X-form, the segmented portion is also an X-form.

**Definition 11 (Sub-form of X-form).** *Suppose $E$ is an X-form, so $E$ is an algebraic expression on $g = \{p_1, p_2, \ldots, p_K\}$, where $g = \{p_1, p_2, \ldots, p_K\}$ is a set of variables, $E$ is an algebraic expression (with 3 subjective operations): $E = E(g) = E(p_1, p_2, \ldots, p_K)$. Then a sub-form of $E$, $E_s$, is an algebraic expression on $g_s$, where $g_s = \{b_{s_1}, \ldots, b_{s_J}\}$, $J \leq K$ is a subset of $g$, and $E_s = E_s(g_s) = E_s(b_{s_1}, \ldots, b_{s_J})$, and $E_s$ is an proper sub-expression of $E$.*

By definition, the sub-form is also an X-form. We skip discussions on properties of X-form. We just point out: an X-form can be thought as one information processor, and vice versa, a information processor can be expressed by at least one X-form. Importantly, we note, a processor could be expressed by many different X-forms. This simple fact is crucial.

### 3.4 Discussions about X-form

X-form is introduced in the process to understand subjective pattern. It turns out, X-form is a very good tool for this purpose. We can see several aspects of X-form. First, it is an algebraic expression, so algebraic operations can be done, which are relatively easy. Second, the X-form is associated with the perception bits of learning machine. Thus, we could not to change X-form arbitrarily, and changes of X-form requires modification of perception of learning machine. This two-sided nature makes X-form particularly good for learning machines. On the one hand, easy algebraic operations on X-form gives the learning machine ability to do imagination and various reasoning. On the other hand, these operations, must link with perception bit and be specifically placed inside the learning machine. So, the imagination and reasoning is ready to be realized. And, X-form tells how to realize itself. This is very perfect.

As a comparison, we can see usual artificial neural network (ANN). ANN indeed is one information processor. But, it is very hard to do explicit algebraic operations on ANN, and it is hard to understand how perception of pattern is done inside ANN. It is very hard to see inside structure, and very hard to write down what is going on. Another comparison, we can see the motion of concepts with category theory or similar, where several operations can be performed on concepts. While such operations are good for reasoning, it is hard to see how it is realized. X-form can do all these easily.

X-form itself can be viewed as a Boolean logical statement, so the X-form naturally is associated with classical logical reasoning within learning machine. However, more types of reasoning can be done, including non-classical reasoning, e.g., various non-classical, probabilistic, fuzzy, etc. The work in these areas has yet to be unfolded.

In short, the X-form is an algebraic expression, subjective pattern and information processor, which can be easily understood, manipulated, and realized. It is network-like, but much easier to handle than popular neural network. The learning is accommodated by X-forms, yes, many X-forms. Surely, X-form is a mathematical object that deserves a lot of researches.

The X-form is the core concept of our discussion, and play very crucial role in a learning machine. Thus we would like to explicitly propose a conjecture for X-form.

*Conjecture 1 (X-form).* : Any computing system, as long as it has flexible learning capabilities, has at least one X-form inside it, and learning is accommodated by the X-form.

## 4 Conceiving, Governing and Primary Consciousness

It is clear that X-form is the most important part of an universal learning machine, which is doing information processing, perceiving patterns, and accommodating learning. Naturally, we think that there will be many X-forms inside an universal learning machine, which forms conceiving space.

**Definition 12 (Conceiving Space).** *The space formed by all X-forms in a learning machine $M$ is called the conceiving space of $M$.*

Learning is done on X-forms in conceiving space. But, what is promoting, regulating and implementing, governing those changes of X-forms? We intrdoduc the concept of governing space.

**Definition 13 (Governing Space).** *Inside a learning machine $M$, all mechanisms that promot, regulate, control the changes of X-forms inside conceiving space is called as governing space of the learning machine.*

So what are inside the governing space? First, there are learning strategies and learning methods, which we will discuss in later sections. However, we need something deeper. A learning machine needs subjectivity to become effective. We would use the term *primary consciousness* for this subjectivity (see [9], where it is well described). In governing space, there is the primary consciousness that is working to guide activities of learning machine. If a learning machine has no primary consciousness, it could not be universal.

However, even for primary, there is difference in capability. It is important to understand minimal primary consciousness, which is a threshold below that the learning machine cannot be universal. We think that the minimal primary consciousness should be the capability of sufficient awareness and correct response to: 1) change of base pattern of input and feedback. 2) its own perception of such change, and 3) adaption to this change.

In section 2, we discussed level of learning. By using conceiving space and governing space, it is very easy and clear to distinguish different level of learning.

## 5 Learning Dynamics and Data Sufficiency

In previous discussion, we have made clear that inside learning machine the information processing and pattern perception is done by X-form, and learning is developing and selecting a better X-form. We now turn to how to develop and select better X-form.

Learning is a motion of X-form, moving from one X form to another. Or, learning is the dynamics in the conceiving space. So let's discuss this dynamics. We only consider the N-1 learning machine $M$ below.

Suppose $M$ is an N-1 learning machine. The typical learning process is:
1. Set an objective pattern: $p_o \subset PS_N^0$;
2. Select a sample set $S_{in} \subset p_o$, usually $S_{in}$ is a much smaller set than $p_o$. But in extreme cases, $S_{in} = p_o$;
3. Select another sample set $S_{out} \subset p_o^c$, i.e. all elements in $S_{out}$ are not in $p_o$. The set $S_{out}$ is usually much smaller than $p_o^c$, but in extreme cases, $S_{out} = p_o^c$ may appear.
4. Use the data in $S_{in}$ and $S_{out}$ to form the sample data sequence $b_i, i = 1, 2, \ldots$, $b_i \in S_{in}$ or $b_i \in S_{out}$.
5. However, feedback data (also known as labeled data) is also required. Data $b_i \in S_{in}$ or $b_i \in S_{out}$ could be labeled, or not labeled. Both cases are allowed, and also allowed sometime labeled, something not labeled. Thus, the data sequence is: $(b_i, o_i), i = 1, 2, \ldots$, if $b_i \in S_{in}$, $o_i$ is 1 or $\varnothing$, if $b_i \in S_{out}$, $o_i$ is 0 or $\varnothing$. Here $\varnothing$ stands for empty to indicate data is not labeled at this time.
6. Apply the data sequence continuously into $M$. There is no limit on how to enter, how long to enter, input frequency, repeat input, which parts to enter, etc.

Suppose the data sequence is $\{(b_i, o_i) \mid i = 1, 2, \ldots\}$, and the original X-form is $E_0$. We can consider the learning dynamics. We then can write down the learning in equations.

Starting from $E_0$, the first step would be:

$$E_1 = LM(E_0, b_1, o_1) \tag{1}$$

Here $LM$ stands for learning method specifying how X-form changes. Note that we can write above way of function because we assume that the learning machine follows "a set of simple and fixed rules", without this, it is problematic to write the above equation.

This process continues, the data is entered in order, at the $k$ step, we have:

$$E_k = LM(E_{k-1}, b_k, o_k) = LM(E_0, b_1, o_1, b_2, o_2, \ldots, b_k, o_k), k = 1, 2, \ldots \tag{2}$$

We should note, the equation 2 is very similar to equation in mathematical dynamics. Actually, they are dynamics of learning. We can say, the phase space is the conceiving space, and dynamics is written in learning methods, strategies and primary consciousness. Of course, to make these statements mathematically precise, there are much more works to be done.

Under the drive of data sequence, we hope eventually the X-form $E_k$ would become what we want. There are several factors. One is data sequence, another is "smartness" of learning method. Here, we discuss data sequence. We need to know what data are enough. This leads to data sufficiency. Fortunately, the X-form itself gives a good description of enough data. Using X-form, we can define data sufficiency.

**Definition 14 (Data Sufficient to Support).** *Suppose $E$ is an X-form, and assume that there is a set of base patterns $D$, $D = \{b_j \mid j = 1, \ldots J\}$, if for any $E$ subform of $F$, there is at least one data $b_j, 1 \le j \le J$ so that $F(b_j) = 0$, but $E(b_j) = 1$, we say that the data set $D$ is sufficient to support X-form $E$.*

**Definition 15 (Data Sufficient to Bound).** *Suppose $E$ is an X-form, and assume that there is a set of base patterns $D$ $D = \{b_j \mid j = 1, \ldots J\}$, if for any X-form $F$ that covers $E$ (ie $E$ is a sub-form of $F$), then there at least is a data $b_j, 1 \le j \le J$, so that $E(b_j) = 0$ and $F(b_j) = 1$, we say that the data set $D$ is sufficient to bound X-form $E$.*

We can see the meaning of data sufficiency: If for each sub-form of $E$, we have a data $b_j$ that makes it in the supporting set of $E$, but not in the supporting set of this sub-form, the data set is sufficient to support. Also, if for each X-form with $E$ as a sub-form, we have a data $b_j$ outside the supporting set of $E$, but in the supporting set of the X-form, the data set is sufficient to bound.

## 6 Learning Strategies

In equation 2, LM is actually formed by learning strategy and method. Strategy defines the framework while method defines more detailed part. There are multiple strategies and methods for different situations and requirements. We discuss some here. Due to restriction on size, we have to skip many details. Please refer to [5].

**Strategy 1 - Embed X-forms into Parameter Space**
The basic idea of this strategy is this: set a real Euclidean space $\mathbb{R}^U$, $U$ is a huge integer, and then divide $\mathbb{R}^U$ into $L$ mutually disjointed areas, then attach an X-form to each area, thus embedding $L$ X-forms. Then introduce a dynamics in $\mathbb{R}^U$, which is equivalent to introducing dynamics on $L$ X-forms. Dynamics in real European space is a very mature branch of mathematics, and we have a lot of tools to deal with this dynamics. In this way, we transform the learning dynamics into the dynamics in a real-European space.

More precisely, we can write down as below. Suppose $\mathbb{R}^U$ is a real Euclidean space, $U$ is a huge integer, and then assume that we have somehow cut $\mathbb{R}^U$ to $L$ pieces:

$$\mathbb{R}^U = \bigcup_{i=1}^{L} V_i, \quad V_i \cap V_{i'} = \varnothing \quad \forall i \neq i'$$

Also, we preset $L$ X-forms $E_i, i = 1, 2, \ldots, L$, and attach each $E_i$ to $V_i$. Thus, for any point $x \in \mathbb{R}^U$, if $x \in V_i$, then at this point $x$, the corresponding X-form is $E_i$. We can write this X-form as $E_x = E_i$. Then, using following loss function:

$$Lo(x) = \sum_{j=1}^{J} (E_x(b_j) - o_j)^2, \quad \forall x \in \mathbb{R}^U$$

where $E_x$ is the X-form corresponding to $x$. With these setting, we have theorem.

**Theorem 3 (Embed into Parameter Space).** *Suppose the X-form that we want is $E^*$, and assume that $E^*$ is among $\{E_1, E_2, \ldots E_L\}$. More, suppose the data sequence $D$ is sufficient to support and sufficient to bound $E^*$. Further assume that the dynamics in is: seeking loss function $Lo$ take the minimum value. Then, if $Lo(x_{k^*})$ reaches the minimum at some $k^*$, the X-form, $E_{k^*}$ is the X-form we want to learn, i.e. $E^* = E_{k^*}$ .*

We have some comments on this strategy: 1) Such a setting indeed is an effective learning machine. Deep learning is in fact using this strategy [3]. 2) Of course, the previous settings are ideal and are only for ease of discussion. However, deep learning fully demonstrates that this learning strategy and method is very effective to many situations. 3) This strategy is much more general then usual deep learning. But, it could not achieve universal learning machine.

**Strategy 2 - Squeeze X-form to Higher Abstraction from Inside and Outside**
This strategy can be summarized as: 1) Check the input space to see if necessary to include the input and whether necessary to exclude the input. 2) Squeeze current X-form to a higher level of abstraction and more generalization. 3) Select the best X-form from conceiving space.

The data requirements for this strategy are: sufficient to support and sufficient to bound the desired X-form, and all data is required to be labeled. So in data sequence $D = \{(b_j, o_j), j = 1, 2, \ldots\}$, all $o_j$ are not empty. This strategy also has requirements for the ability of the learning machine.

**Definition 16 (Capability 1 - Squeeze X-forms to higher abstractions).** *Could squeeze the current X-form $E$ into another X-form $E'$ with higher abstraction and more generalization. More precisely, squeeze will do the following: Suppose $E$ is the X-form on $g$, $E = E(g)$, where $g$ is a set of base patterns $g = \{b_1, b_2, \ldots, b_K\}$, then the squeezed X-form $E' = E'(g')$ should satisfy: 1. $g' \in g$, 2. $B \in B'$, where $B$ is the supporting set for $E$, and $B'$ is the supporting set for $E'$. If could find such an X-form, put it in conceiving space, otherwise take no action.*

**Theorem 4 (Squeeze from both inside and outside).** *Suppose a learning machine $M$ uses Strategy 2 to learn, and $M$ has the ability 1. Assume further that the data sequence for driving learning is $D = \{(b_j, o_j), j = 1, 2, \ldots\}$, and $D$ is sufficient to support and bound the X-form $E^*$. Then, starting from the empty X-form, if the data in $D$ is fully input into $M$ and repeat long enough (without missing any data, and can repeat), then $M$ will eventually learn $E^*$.*

This theorem show that with Strategy 2, a learning machine with Capability 1, can become a universal learning machine.

**Strategy 3 - Squeeze X-form to Higher Abstractions from Inside** Strategy 3 is: compressing the X-form from internal to high abstraction. We summarize this strategy as: 1) Check the input space to see if you need to include this input (a basic schema). 2) Compress the current X-form to a higher level of abstraction and more generalization, but not excessive. 3) Select the best X-form from the thought space. Just like strategy 2, strategy 3 can achieve universal learning machine.

**Strategy 4 - Squeeze X-form to Higher Abstractions, Unsupervised** Both Strategy 4 and Strategy 3 squeeze the X-form from inside to high abstraction. However, Strategy 4 uses unsupervised data (at least partially), this is very desirable. But, using unsupervised data, we need to have a stronger capability. Just like strategy 2, strategy 4 can achieve universal learning machine as well.

Now we know universal learning machine is achievable, at least in theory. Although the above strategies are more on theoretical side and there is some distance to engineering, the theoretical result is important. We can pursue to realize universal learning machine without worry.

## 7 Engineering Model

Engineering models to realize universal learning machine must follow certain common principles. One major part of a universal learning machine is its conceiving space. There are more than one ways to realize X-forms. For example, combination of inner product and nonlinear function could form X-forms [3]. A well-designed conceiving space should have properties below. 1) Should have enough expressive power for all X-forms. 2) Should have very efficient way to manipulate X-forms, for various activities, including connection, decomposition, combination, add on and decrease, logical reasoning, probability rationale, and so on. 3) Should have a clean structure, so that it can be easily explained and cooperated with with other parts of the learning machine, especially with the governing space.

Inside governing space, primary consciousness should be established first. Governing space should integrate the 5 categories naturally: 1) logical reasoning and deduction, 2) connectionism, 3) Bayesian methods, 4) analogy, 5) evolution [6]. Specially, logical reasoning and probability rationale should support each other. Further, governing space can have a AGI module working inside it, such as NARS or Logic pack [8] [10]. X-form is good for all of these.

We have done some specific implementation of the engineering model, e.g. patent application [4].

## References

1. Chuyu Xiong. Discussion on Mechanical Learning and Learning Machine, arxiv.org, 2016.
http://arxiv.org/pdf/1602.00198.pdf
2. Chuyu Xiong. Descriptions of Objectives and Processes of Mechanical Learning, arxiv.org, 2017.
http://arxiv.org/abs/1706.00066.pdf
3. Chuyu Xiong. What Really is Deep Learning Doing, arxiv.org, 2017.
http://arxiv.org/abs/1711.03577.pdf
4. Chuyu Xiong. Chinese patent application Application # 201710298481.2 (in Chinese)
5. Chuyu Xiong. Principle, Method and Engineering Model of Universal Learning Machine, researchage.net, 2018, (in Chinese)
https://www.researchgate.net/publication/323915356
6. Pedro Domingos. The Master Algorithm, Talks at Google.
https://plus.google.com/117039636053462680924/posts/RxnFUqbbFRc
7. L. Valiant. A theory of the learnable. Communications of the ACM, 27, 1984.
http://web.mit.edu/6.435/www/Valiant84.pdf
8. Pei Wang. A Logical Model of Intelligence An introduction to NARS, https://sites.google.com/site/narswang/home/nars-introduction?pli=1
9. ZHONG Yixin. Mechanism-based artificial intelligence theory[J]. CAAI transactions on intelligent systems, 2018, 13(1): 218. (in Chinese)
10. HE Huacan. Universal logic theory: logical foundation of mechanism-based artificial intelligence theory[J]. CAAI transactions on intelligent systems, 2018, 13(1): 1936. (in Chinese)
11. WANG Peizhuang. Factor space-mathematical basis of mechanism based artificial intelligence theory[J]. CAAI transactions on intelligent systems, 2018, 13(1): 3754. (in Chinese)