An Artificial Intelligence Model that Combines Spatial and Temporal Perception

Jianglong Nan and **Fintan Costello**
School of Computer Science and Informatics
University College Dublin
Belfield, Dublin 4, Ireland
jianglong.nan@ucdconnect.ie, fintan.costello@ucd.ie

Abstract

This paper proposes a continuous-time machine learning model that learns the chronological relationships and the intervals between events, stores and organises the learnt knowledge in different levels of abstraction in a network, and makes predictions about future events. The acquired knowledge is represented in a categorisation-like manner, in which events are categorised into categories of different levels. This inherently facilitates the categorisation of static items and leads to a general approach to both spatial and temporal perception. The paper presents the approach and a demonstration showing how it works.

Introduction

The general goal of artificial intelligence requires the intelligent agent to learn and acquire the knowledge not only from the data set that represents a static environment, but also from a dynamic world in which some events may always happen after others and the intervals between them may follow some patterns. Tackling both the spatial and the temporal aspects of the problem has attracted the interest of other researchers as well (Sur09).

Intelligence arises first from getting to know the similarities and differences between different items. This has been well studied by psychologists in the area of categorisation and concept formation. However, in the very wide range of models presented in the literature of this area (Ros78) (Ham95) (Nos86) (Kru92), the items are presented as a set of features in a static way, in which each presentation is a single discrete event, independent of and unrelated to all other events.

The shortcomings of this static view of concept formation lie in two aspects. Firstly, it lacks the ability to deal with irregular dynamic events that last a period of time, which, we think, can also be handled in a similar way to the static items. And secondly, human beings and animals actually always perceive an environment that continuously changes. What is missing in the static view is how the dynamic perception can be transformed into the static items.

An experiment described in (Car02) showed that the vision of our eyes is actually very small. What is usually thought to be observed at once by the eyes is actually perceived piece by piece and step by step. To our nervous system, even the perception of a static world is a multi-step observation process that spans over a period of time, which needs to be converted to or treated as an integrated item somehow to facilitate the intelligence of a higher level.

On the other hand, the learning of time-spanning events has been studied independently of concept formation. In Pavlov's studies (Pav28), dogs were trained or conditioned by being presented with a conditioning stimulus (e.g. the ringing of a bell) that was followed, after a certain controlled interval, by an unconditioned stimulus (e.g. the presentation of food). The critical observation in these studies was that the timing of a trained dog's conditioned response (e.g. salivation) depended on the interval between the conditioned and unconditioned stimuli (between the ringing of the bell and the presentation of food). The longer the interval between the bell and the presentation of food, the longer the interval between the bell and the start of salivation: the shorter the interval between the bell and food, the shorter the interval between the bell and salivation. Skinner (Ski38) similarly found that the timing of operant response in trained pigeons was also dictated by the intervals between reinforcement.

However, unlike the studies of categorisation, all these studies focused only on one or two specific events and the corresponding intervals, rather than the relationships between various events and the acquisition of knowledge. This paper presents a machine learning approach based on the argument that time-spanning events can be considered in the same way as the concepts that are learnt and categorised in the traditional categorisation perspective.

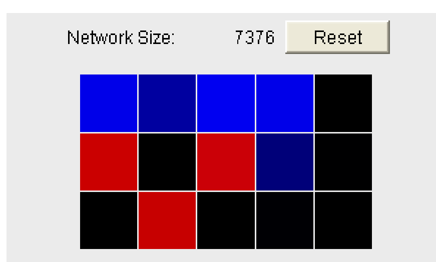

Figure 1: Demonstration of our Machine Learning Approach at `http://csserver.ucd.ie/~jlongnan/agi.html`. We invite readers to play with this demo before reading on.

An agent-environment perception is taken. And the model is designed to serve the purpose of making predictions of what will happen in the environment.

The remainder of the paper is organised as follows. First, we present an overview of the model. Then, we go through different aspects of the model in detail. Finally, we present a demonstration as illustrated in Figure 1 that shows how the model works.

Overview

Basically, the model is based on a network of knowledge. Each node of the network is a rule in the form of

If something happens, then something (else) will happen, in some time.

which includes three elements: an *antecedent*, a *consequence*, and an *interval*. Note that the antecedent and the consequence can be the same thing.

Such a node can also be considered as an individual *event* in which

First something happens, and then after some time, something (else) happens.

And this event, represented by an individual node in the network, can take the role of the antecedent or the consequence of some other rules, i.e. nodes.

In this way, nodes are connected with each other. And although a node consists of only one antecedent and one consequence, it can represent a long series if it is on a higher level and thus contains a lot of other lower level nodes indirectly. Ultimately, every node refers to a set of *sensory inputs*. Despite that we can create as many nodes as needed, we do have a fixed set of sensory inputs for any given agent in this model. These sensory inputs are supposed to receive pulse-like stimuli to perceive the changes of the environment. And the nodes not only define sets of sensory inputs, but also define the patterns and rhythms in which they get stimulated. Figure 2 shows an example. Note that the interval between the antecedent and the consequence is defined to be the interval between the first stimulus of the antecedent and the first stimulus of the consequence.

Unlike the Temporal Causal Networks described by Bouzid and Ligeza (BL00), the causality of not only the individual inputs but also specific sequences of inputs is represented in this network.

This knowledge representation is designed to allow the predictions of what will happen in the environment to be made in a distributed manner. Each node, having observed that its antecedent has happened or is happening, makes a prediction that its consequence will happen.

Both the concrete experiences and the abstractions are represented by and handled through the network nodes in exactly the same way. For a particular series of perceived stimuli, in addition to the node representing the whole event, different nodes covering different aspects of the event are also created. The latter is considered the abstraction.

Less abstract nodes refer to more abstract ones. For example, in Figure 2, `A` refers to `B` and `D`. `A` represents that if `S1`, `S2` and `S3` get stimulated in this particular pattern then

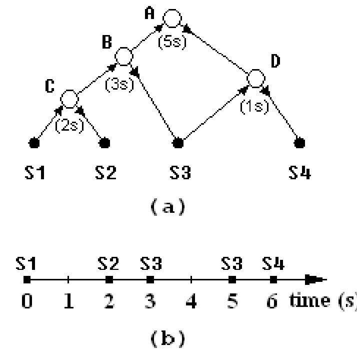

Figure 2: An Example of the Network: **(a)** Four nodes, `A`, `B`, `C`, and `D`, denoted by the circles are formed based on four sensory inputs, `S1`, `S2`, `S3`, and `S4`, which are denoted by the dots. The intervals are written in the brackets below the nodes. Arrows are used to point the antecedent of a node to the node and also the node to its consequence. The antecedent and consequence of a node are always drawn below it. **(b)** The series represented by node `A` happens first with `S1` getting stimulated. Assuming that it is at time 0, `S2` then gets stimulated at time 2s. `S3` gets stimulated twice at time 3s and 5s respectively. And finally `S4` gets stimulated at time 6s. As mentioned previously, the interval between `B` and `D` is defined to be the interval between the time when `S1` gets stimulated and the time when `S3` gets stimulated for the second time as the antecedent of `D`.

`S3` and `S4` will get stimulated in another pattern. It is considered less abstract than `B` and `D` because it contains more details. A part of `A`, e.g. `B`, may be the useful part while the rest may be just the trivial details. In this case, `A` can be considered as an instance of category `B`. A node can be referred to by multiple other nodes. This leads to a semi-hierarchical network, in which the knowledge represented by the nodes may arbitrarily overlap with one another and no strict tree structure is constructed. The nodes can be viewed not only as different events, but also as the categories they fall into. A more abstract node can be considered as the category of all the less abstract nodes that refer to it. And in this way, the network can be considered both as a collection of events and as the categorisation of these events.

This model deals with categorisation from a perspective of perception. The perception is categorised based on the similarity between different experiences.[1] For example, *my car as observed this morning* and *my car as observed this evening* are considered as two different instances or items. Both of them are represented by a node and they may both refer to a more abstract one representing *my car*. In other words, they are categorised as *my car*. Meanwhile, *my car*

[1] Although we do believe the things that are categorised together by human beings must, from a perspective of perception, share something in common, e.g., hearing other people call those things the same name, this paper does not discuss whether this principle is appropriate or not. The proposed prediction-oriented model inherently deals with categorisation this way.

may refer to and thus be categorised into a more abstract node, *car*. Unlike the traditional categorisation models such as (Ros78) and (Kru92), this model does not differentiate between the items and the categories they fall into. In the above example, *my car* is not only an item but also a category.

The generalisation happens when some nodes are considered more reliable than others. The nodes that are more likely to make the right predictions and more reliable in reflecting what really is going on in the environment dominate the others.

The following sections focus on the three major aspects of the model, which are *learning*, *recognition*, and *prediction* respectively.

Learning

Event

Unlike most conventional machine learning models, the sensory system of this model is designed to receive pulse-like stimuli input instead of maintain a set of state variables that can be retrieved freely. The intelligent agent possesses an array of sensory inputs. An *event* is simply one or more sensory inputs getting stimulated in some pattern, in which an individual sensory input may be stimulated more than once.

For the sake of discussion, an event is considered to be a *sub-event* of another if all its stimuli are also presented in the other, and the intervals between them are the same. Equivalently, the latter is called a *super-event* of the former.

There are two types of sub-events. A *segment* sub-event is a continuous fragment of its super-event. It contains every stimulus in this fragment. While a *non-segment* sub-event is a concatenation of more than one continuous segments, with missing stimuli from its super-event in between. Figure 3 illustrates this.

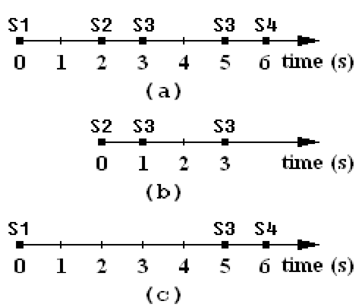

Figure 3: Segment Sub-events and Non-segment Sub-events: **(a)** The event denoted by node `A` in Figure 2, in which the sensory inputs `S1`, `S2`, `S3` and `S4` get stimulated in a specific tempo. **(b)** A segment sub-event of it. **(c)** A non-segment sub-event of it.

Memory

As described previously, network nodes are formed to denote the memories of the perceived events in the form of antecedent-consequence pairs with intervals, for future use.

Simple nodes, which usually form in the early stage of a simulation, have sensory inputs as both their antecedents and consequences. They represent simple events that consist of only two stimuli to the sensory inputs. Meanwhile, nodes that represent long and complicated events can be formed based on existing nodes.

Concrete Experience

When an event is perceived, i.e. some sensory inputs are stimulated in some pattern, the event itself and all its sub-events will be stored, assuming none of them has ever been perceived before.

Take the event shown in Figure 4 `(a)` for example. First, as shown in Figure 4 `(b)`, a node for the first two stimuli is formed based directly on the sensory inputs (Node `A1`). Next, a node for the first three stimuli is formed based on the first node and the third sensory input (Node `A2`). The rest of the event stimuli is all included this way. And finally the node for the whole event is formed (Node `A`).

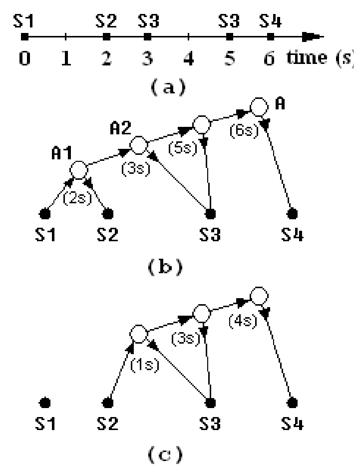

Figure 4: Nodes Formed for Segment Sub-events

But that's not all. Only the nodes for the segment sub-events that start with the first stimulus have been formed. In addition to that, the nodes for the segment sub-events starting with the second stimulus of the event are formed in a similar way as shown in Figure 4 `(c)`. And the nodes for the segment sub-events starting with each of the rest of the stimuli are all formed this way.

As a result, a node is formed for every possible fragment of the event, which may start from anywhere in the event and end anywhere after the starting point.

Abstraction

Abstraction, in this continuous-time event-based model, is viewed as the formation of sub-events that are shared by multiple events.

Sub-events that may possibly be contained by the events perceived in the future form nodes so that they can be referred to later. This actually has been partially demonstrated in the example shown in Figure 4, in which the nodes for only one type of the sub-events of the perceived event, namely the segment ones, are formed.

Actually, the nodes for all the non-segment sub-events are also formed similarly. For example, the second and the fourth stimulus of an event will form a node with the interval between them, based on which another node for the second, the fourth, and the fifth stimulus will also be formed.

Any sub-event of a perceived event is considered to be a possible pattern that the event may be following. When a number of events indeed follow a particular pattern, they share that sub-event by referring to the node representing it either directly as the antecedent or the consequence of them, or indirectly. Meanwhile, the node for the pattern itself, is probably sharing some more general patterns with other nodes. And a network of events of different levels of detail, that is, different levels of abstraction, is formed this way.

The generalisation of the model works in a way in which, in the beginning of the learning, the perceived item is analysed and all various aspects of it are represented individually. These aspects, which overlap with each other, actually include all possible more general and more abstract items, i.e. categories. For example, when *a particular car as observed this morning* is perceived, various aspects of it such as *a particular car*, *car*, *vehicle*, *wheel*, *a particular wheel* and a lot of things we do not have names for are also represented and stored at the same time. As more and more items are perceived by the agent, some aspects of the perception, e.g. *car*, are found to be more useful than others, e.g. *a particular wheel*, because they are shared by more items and thus are more reliable in making the predictions. These aspects gain more credit and have stronger influence. And they can be considered as the basic level categories defined by Rosch (Ros78).

Short-term Memory

The nodes that are formed for a single event are discussed in the previous sections. However, the intelligent agent is supposed to receive a continuous series of stimuli, without separators.

One option is to take all the stimuli from the start till present as a big event. But that will cost too much space and time. In the model, when a sensory input is stimulated, only a limited number of previous stimuli will be linked to it to form new nodes. The stimuli that occur too early are simply ignored, as if they have never occurred. The number of nodes that are formed is limited this way. Those stimuli that are used to form new nodes are considered to be in a *short-term memory*.

A limited size of short-term memory may seem to completely prevent long series from being learnt. But actually it does not. This is discussed in the next section.

Recognition

Recurring Event

When an event that has been previously perceived occurs, no new node is formed. Rather, the event is *recognised* by an existing node. This *recognition* process takes place when

1. The antecedent and consequence of an existing node is perceived and both of them are still being held in the short-term memory; and
2. The difference between the perceived interval and that of the existing node is within a tolerance range.

When either of the above two conditions is not met, even if the other one is, no recognition takes place and instead a new node is formed.

Familiarity

The capacity of the short-term memory is designed in relation to the number of nodes instead of the number of stimuli of the events being held.[2] When the recognition process happens, the existing node will take the place of its antecedent and consequence in the short-term memory. This makes more space in the short-term memory so that more information can be held in it. Note that just like more than one stimulus of a single sensory input can be held in the short-term memory, a single node can be recognised more than once in a short period of time and thus has more than one instance being held in the short-term memory.

Recognition can happen recursively. The nodes that refer to other nodes can be recognised after their antecedents and consequences are recognised.

The learning process can also take place upon the recognition. New nodes can be formed based on the recognised ones or combinations of the recognised nodes and raw stimuli.

Therefore, no matter how long a series is, it can still be learnt piece by piece, as long as it recurs enough times.

Overlapping Events

Assuming two events that have already been learnt independently recur in an overlapping manner, in which, one event starts before the other finishes. Both events will be recognised. And a new node will be formed with one of them as the antecedent and the other as the consequence. The interval, by definition, is the delay between their first stimuli.

Such nodes, with their antecedents overlapping with their consequences, are dealt with in the same way as other nodes. They are also used to make the predictions, as discussed in detail in the next section.

Prediction

Knowledge and its Use

A node is not only an event. It is also a piece of *knowledge*. It is a fraction of the understanding of what the environment is like. It is a prediction that if its antecedent happens, its consequence happens after the interval.

The possession of a collection of nodes is the possession of the knowledge of some aspects of the environment. To make use of the knowledge, an intelligent agent makes *predictions*.

[2] In the current model, the short-term memory is represented by a fixed number of spaces, each of which can store either a stimulus or a recognised node.

The sole purpose of this knowledge structure is to allow the model *to make predictions of what will happen based on what has already happened*. Whenever the antecedent of a node is perceived, it makes a prediction that its consequence will happen after the interval. A large number of predictions may be made simultaneously by multiple nodes from all over the network. These predictions can then be used to make decisions when the model is put into a decision making process.

Confidence

The prediction is actually made in the form of the *confidence*, which is an attribute of the node, ranging from 0 to 1. From one perspective, the confidence of a node can be viewed as the degree of certainty held by the node that its consequence is about to happen at the given time. From another perspective, it can also be viewed as the degree of certainty held by the node that the event represented by the node itself is actually happening at the given time.

The confidence of a node is determined by various factors, the most important of which is the time since its antecedent last happened.

We define the *expectation* of a node, which ranges from 0 to 1. It usually stays at 0. When the antecedent of the node is perceived and the exact interval has passed, it reaches 1. The time when the expectation reaches 1 is called the *expected time*. When the time is around the expected time and within a tolerance range, the expectation gets a value between 0 and 1. The closer the time is to the expected time, the higher the expectation is. Currently a cosine function is adopted. See Figure 5 for illustration.

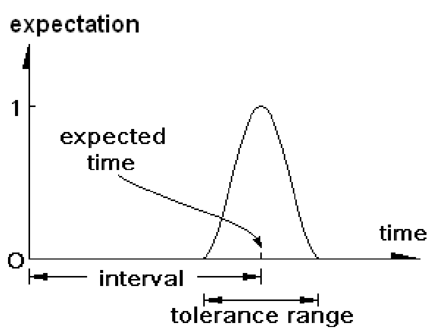

Figure 5: Expectation Changing over Time: Assuming the antecedent of the node occurs at time 0 and the consequence does not happen, the expectation of the node starts to increase when the time is within the tolerance range and reaches its maximum value of 1 at the expected time. After that, it gradually decreases to 0.

If the consequence of the node does happen when its expectation is greater than 0, the prediction is considered to be verified and the node stops making the prediction by changing its expectation to 0 immediately.[3]

[3]Note that the antecedent is assumed to have completely happened at this point. More complicated cases are discussed in subsequent sections.

In the simplest case, the confidence of a node is simply its expectation as in

$$c = e \tag{1}$$

where c is the confidence of the node and e is the expectation of it. Other factors that influence the confidence are discussed in subsequent sections.

Reliability

The predictions made by different nodes are not treated the same way.

A prediction made by a particular node at a specific time may turn out to be either right or wrong. On the one hand, some nodes may tend to always make right predictions. On the other hand, some nodes may tend to always make wrong predictions. The knowledge represented by the former better reflect what the environment is like than the latter.

Each node has its *reliability*, which may range from 0 to 1. The higher the reliability of a node is, the stronger influence the node has on the overall perspective. And the reliability of a node is actually defined as its degree of accuracy as follows.

$$r = \frac{p}{a} \tag{2}$$

where r is the reliability of the node, p denotes how many times the event represented by the node has been perceived, and a denotes how many times the event represented by its antecedent has been perceived. This is because every time the antecedent of the node is perceived, it makes a prediction, but only when the node itself is perceived afterwards, the prediction is considered to be accurate.[4]

The confidence of a node is always influenced by its reliability. Having taken into account the reliability, r, we can now define the confidence of a node, c, as

$$c = re \tag{3}$$

Interaction

The predictions are made by the nodes that are interconnected in the network. And they are not only determined by the nodes that make them through the general principle of prediction discussed previously, but also influenced by other nodes directly or indirectly in a number of ways.

The confidence of a given node can be influenced by a node that has a direct connection to it, which falls into one of the four types listed below.

1. **Antecedent** A node has a unique antecedent. Its influence on the confidence of the node is already discussed in the case where it is a sensory input. Actually, if the antecedent itself is another node, the way it influences the confidence of the node in question is similar, only that its own confidence plays a role in it. To be more specific, assuming the antecedent is the only source of the confidence of the node, its confidence c, then, can actually be represented as

$$c = c_a = c're \tag{4}$$

[4]This definition of the reliability takes a perspective of the overall statistical accuracy. Alternatively, a definition that favours the more recent experiences may be taken.

where c' is the confidence of the antecedent of the node. We use c_a to denote the part of the confidence of the node that is contributed by its antecedent.

2. **Consequence** A node has a unique consequence. When both the antecedent and consequence of the node gains a confidence value of 1 when its expectation e is still greater than 0, as the event is considered over, e is set to 0 and thus c_a changes to 0 as well.
3. **Antecedent parent** Any node that has the given node as its consequence is an *antecedent parent* of the given node. A node may have zero or more antecedent parent. Even if there is no evidence that the antecedent of the node is happening, the node may gain confidence through its antecedent parents. Actually, each antecedent parent passes the c_a part of its confidence to the node. In other words, the confidence that the node gains from each of its antecedent parent is simply the confidence that the antecedent parent gains from its own antecedent.
4. **Consequent parent** Any node that has the given node as its antecedent is a *consequent parent* of the given node. Like the antecedent parent, a node may have zero or more consequent parent. But unlike the antecedent parent, a consequent parent of the node passes the non-c_a part of its confidence, that is, the consequence it gains from its antecedent parents and consequent parents, to the node in question.

To combine the confidence that a node gains from both its antecedent and all the parent nodes, the confidence of the node, c, is defined as

$$c = 1 - (1 - c_a) \prod_{i \in A \cup C} (1 - c_i) \tag{5}$$

where A is the set of all its antecedent parents, C is the set of all its consequent parents, and c_i is the confidence that the node gains from the parent node i.[5]

A Demonstration of the Model

Currently we are still working on testing the model against various experimental data. Whereas in this paper we show a toy program that can demonstrate how the model learns on a real-time basis.

As shown in Figure 1, a grid of sensory inputs represented by squares is displayed by the program. Users are allowed to stimulate these sensory inputs by clicking on them or pressing the keys. Multiple sensory inputs can be stimulated simultaneously by pressing multiple keys at the same time.

The program learns both the spatial and temporal patterns of the stimuli and keeps making the predictions of which sensory inputs are about to be stimulated. The sensory inputs stimulated by the user are shown in red. And the sensory inputs predicted by the program are shown in blue. The more confident the program is about a prediction, the brighter the predicted sensory input will be displayed. The network size indicates how much knowledge has been learnt.

The program can demonstrate that repeated spatial and temporal patterns, even with random interference in either a spatial or a temporal sense, will be learnt. Or from another perspective, similarities between different processes will be traced. The more consistently a pattern is followed, the more confident the prediction about it will be. And longer and more complicated sequences can be learnt after shorter and simpler ones are learnt.

[5] The confidence is considered as the probability of some sort. Various sources of the confidence are also simply considered to be independent events. Conventional probability theory is used to combine them together.

Conclusion

A continuous-time prediction-oriented machine learning model based on a semi-hierarchical knowledge representation has been presented as an attempt to combine spatial and temporal perception. It allows the intelligent agent to acquire the knowledge in both static and dynamic environments; to recognise learnt spatial and temporal patterns and build new knowledge upon them; and to make the predictions in a distributed manner through the antecedent-consequence representation of the knowledge. A demonstration program is also presented to show how the model works.

References

Maroua Bouzid and Antoni Ligeza. Temporal causal abduction. *Constraints*, 5(3):303–319, 2000.

R. Carter. *Exploring Consciousness*. University of California Press, 2002.

C. R. Gallistel and J. Gibbon. Time, rate and conditioning. *Psychological Review*, 107:289–344, 2000.

J.A. Hampton. Similarity-based categorization: The development of prototype theory. *Psychologica Belgica*, 35:103–125, 1995.

J. K. Kruschke. Alcove: An exemplar-based connectionist model of category learning. *Psychological Review*, 99:22–44, 1992.

R. M. Nosofsky. Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General*, 115(1):39–57, 1986.

Ivan P. Pavlov. *Lectures on Conditioned Reflexes*. Liveright Publishing Corp., 1928.

Eleanor H. Rosch. Natural categories. *Cognitive Psychology*, 4(3):328–350, May 1973.

Eleanor Rosch. *Principles of Categorization*, pages 27–48. John Wiley & Sons Inc, 1978.

R. A. Rescorla and A. R. Wagner. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical Conditioning II*, pages 64–99, 1972.

B. F. Skinner. *The Behavior of Organisms: An Experimental Analysis*. Prentice Hall, New Jersey, 1938.

Eugene J. Surowitz. Importing space-time concepts into agi. In *Proceedings of the Second Conference on Artificial General Intelligence*, Advances in Intelligent Systems Research. AGI-09, 2009.