The path to true artificial intelligence

With the existing technology we are able to achieve incredible feats using artificial intelligence and machine learning. We have achieved unprecedented success in training machines to do tasks for us without explicitly training them. We are currently on the verge of fully autonomous machines. But, we are still reliant on multiple ML models to achieve this task without still having a system which is truly intelligent. A truly intelligent system is expected to be able to think, perceive and realize the world without being trained on every instance.

With this article we will be exploring the definition of system 2 machines, general artificial intelligence and the JEPA architecture which provided breakthroughs in our general understanding of machine learning which is about to revolutionize the industry by providing an architecture for building truly intelligent machines.

Introduction

This article discusses the prospects of machine learning where the system is capable of perceiving the world and learning common sense. This is the implication of the fundamental goal of artificial intelligence which is to create a system which is at least as intelligent as a human. This is the definition of system 2 machines.

In the book “Thinking fast and slow” [1] the fundamental differences between system 1 and system 2 are discussed. System 1 is fast, instinctive and emotional whereas system 2 is deliberate and logical. When these definitions are extended to machine learning, system 1 is reliant on training and chooses the most appropriate reaction based on previous experiences. This process is fast and spontaneous. System 2 machines, think, perceive and take logical actions. These actions are based on knowledge rather than experience.

Another crucial aspect is the difference between how humans learn as compared to how current ML learns. As stated by the author in this paper [2] Humans and most animals are capable of learning a huge amount of information in a short period of time. To give an example of this, a new driver who has never driven a car before is capable of learning how to drive in less than 20 hours of training. Whereas current autonomous driving systems have taken complex models and thousands of hours of training and yet are not as good at it.

The question to ask at this point is, what differentiates how humans learn with the way machines learn. Is it just the fact that humans have access to large amounts of data? Do humans have the extraordinary ability to process that data and learn from it efficiently in a short period of time? Or is it the fundamental process of learning?

The answer to this question lies in the difference between system 1 and system 2 machines. Humans are capable of learning a task and applying the learning to a different but similar task whereas machines are not just as efficient at it. By extending the above example further, a novice driver who has never driven on snow would still be capable of understanding the basic principles of driving on snow as he knows how to drive and by utilizing the knowledge of the world, the driver knows that snow is slippery. Based on these observations, the driver would take the necessary steps to adapt the driving style. So how do we train machines to think like humans?

This issue is addressed by the paper[2] where the author proposes an architecture that comprises multiple models capable of utilizing the information learnt by all of them. This is the implementation of a concept known as world models[3]. This approach believes that we can achieve true intelligence only through world models which are capable of understanding the world's various aspects just like how a human mind thinks. The human brain uses various sensory inputs and rather than just memorizing the scenario, it draws inferences and observations from those inputs that are then used to make decisions and predictions for other tasks.The paper also presents the JEPA(joint embedding predictive architecture) which can be used in a hierarchical manner to mimic the human brain and reproduce the learning and thinking process of humans.

The paper mainly focuses on 3 issues and solutions.

- How can machines understand the world and use it to learn, predict and act based on observations? The solution to this is using a multi component architecture, to generate a cognitive system where all modules are interconnected in terms of differentiability with respect to each other.
- How can machines learn to plan based on reasoning? The paper proposes using a hierarchical JEPA (H-JEPA) which learns a hierarchy of representations.
- How to train the model and to reduce the complications of gathering and labeling data? The author proposes using a non-contrastive self supervised model that learns and predicts simultaneously.

We will explore each of these problems and solutions in the next section.

Building a model for autonomous intelligence

To understand the architecture we first need to identify and understand the need for such an architecture. Discussing the reasons why the existing models do not satisfy the requirements would be redundant. Hence we identify the requirements in a more relevant way. We look at the requirements such that we frame it as a problem where the need to simply replicate the human brain. The reasons for that are as follows.

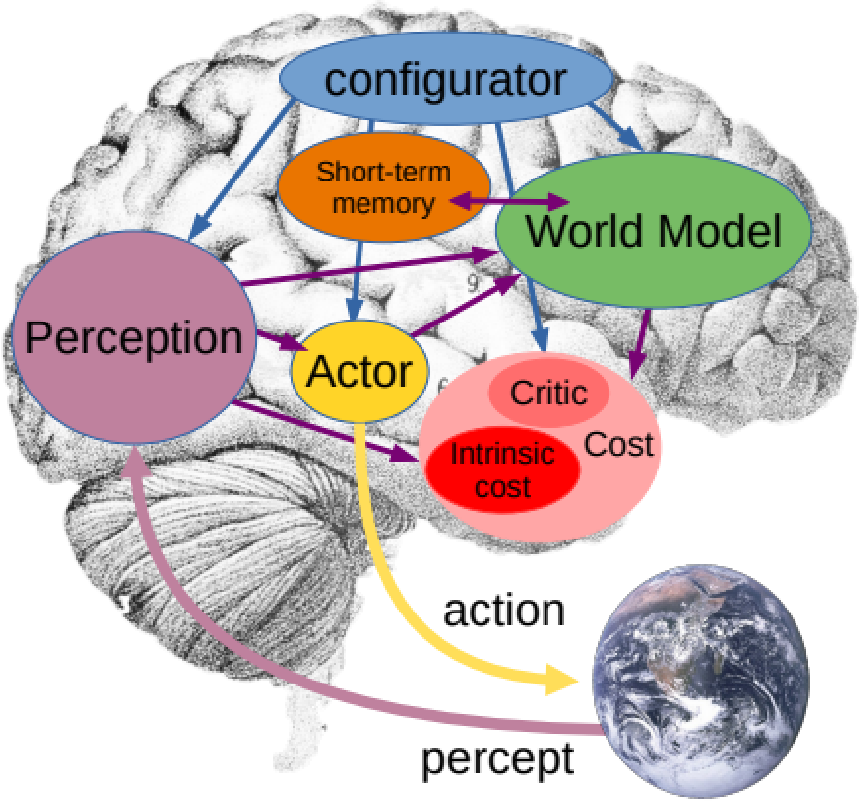
- Humans and machines learn differently.
- A hierarchical learning architecture is beneficial to learning the world.
- Common sense and inference is key than predicting an outcome through explicit example based training.

So far, we have tried to solve the problems through some of the methods mentioned below.

- Multitask models - we train one model to perform multiple tasks. By doing this we allow the model to learn the representations rather than the data itself so each task supports the learning of other tasks.
- Multimodal models - we build models to train on different kinds of data. This produces a rich representation of the data allowing for a better understanding of the task.
- Simulation based training - Used most commonly in autonomous systems where getting examples of every possible scenario is impractical, we teach the machine what to do by doing it.
- Meta learning - we need the model to learn progressively and continuously. So we teach the model how to learn.

All of the techniques mentioned above are state-of-the-art. However, individually none of these techniques comes close to replicating how humans learn. So what we need is a model where we use all of the above techniques in such a way that we create a single model capable of doing all of the above tasks, learn faster, continuously and understand the data through representations, be capable of using different types of data, and also learn by observation. Another important feature of such a model would be the ability to work in a hierarchical method. Research has already shown that humans learn in a hierarchical way and recent advances in autonomous systems have shown that stacking models allows us to learn from rich data and produce realistic representations of the world. Tesla has presented these kinds of models in their AI Open day and they are currently used in their FSD (full self driving) systems.

An architecture for autonomous intelligence



The above architecture presents a high level overview of the proposed architecture where we have various modules each contributing to the goal of achieving autonomous intelligence. The various models are as follows.

- **Configurator -** The role of the configurator module is executive control. It pre-configures the perception , the world module ,the cost and the actor for the current task. It collects the inputs from the other modules and transforms the parameters and their attention circuits to fit a particular task. This gives the ability to use a general model.
- **The perception module -** When understanding the world, not all information is relevant to the task at hand. The perception module, receives the perception, and it selects the features of the perception to estimate the state of the world with respect to the task at hand. This is the key to observability.
- **The world model -** This is the most complex module of the architecture. It contributes to 2 key goals. To piece together the missing information not obtained through perception, and to predict the future states of the world. In other words, the world model learns the natural dynamics and properties of the real world. The configurator pre-configures the world model to make it relevant to the current task. The world model

infers various possible states of the world which are represented in a latent space using latent variables to handle uncertainties. On a general view, the world model plays the role of a simulator.

- **The cost module -** This is an energy based model. The energy is a measure of the discomfort of the agent. To put it in terms of an example, for an autonomous vehicle, the discomfort could be the level of uncertainty in its action. For a humanoid, it could be the measure of empathy towards another entity. The goal here is to decrease the energy in the system. This works together with the agent module where the agent takes actions to lower the average energy. The energy is calculated by two submodules. The intrinsic cost module and the trainable critic module. The intrinsic cost module is nonmutable and is hardwired. It computes the cost based on certain pre-configured policies. Eg - proximity to heat, or standing in the way of traffic is wired to output a high energy level. The critic module is trainable and predicts the future intrinsic energies based on the current state and actions. The trainable critic module is configurable by the configurator to suit a certain task.
- **The short term memory -** It stores the experiences of past present and future states of the world. It is updated by the world model. This is similar to the replay buffer in a DQN(Deep Q Network) . The world model can query this short term memory to fulfill a query to itself. It is also used to train the critic model by using the past states.
- **The Actor Module -** Generates a list of actions which is used by the world model to predict the future states. For a current task, the goal is defined by a cost as configured by the configurator. The cost module computes the cost based on current, past and future states associated with the proposed action. The actor has access to the cost and gradients from the cost modules and may adjust its actions based on the cost. Finally, it may output a given sequence of actions based on a policy module that directly generates an action based on the predictions of the world module perception or may generate actions based on the inputs from the cost modules. Here, the former is similar to system 1 and the latter is similar to system 2. This whole circuit is called the perception action loop. More on this in the next section.

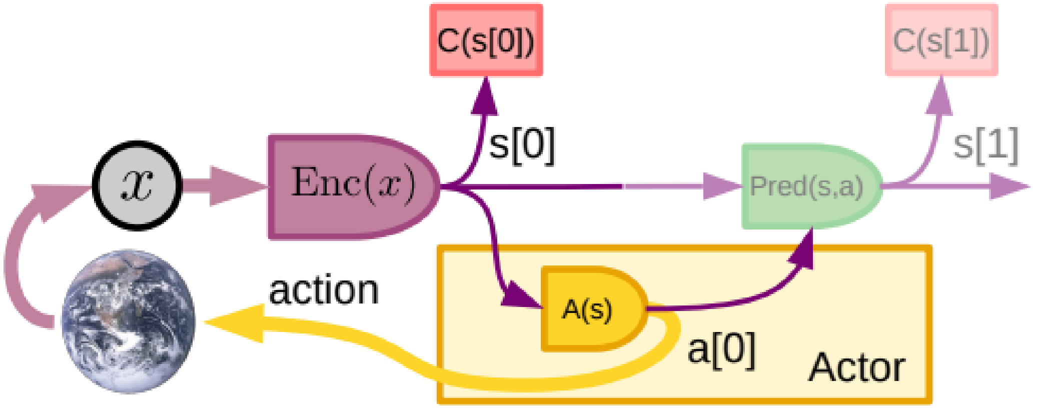
Acting based on the perception

We have established that acting based on observing is the key to making a system intelligent. Hence, the proposed architecture in the paper has two kinds of mechanisms to observe and act. It is representative of system 1 as mode 1 and system 2 as mode 2. We have glanced over how both of them differ in the previous sections.

Mode 1 is policy based, less expensive and faster. It takes in the current observation from the perception module and generates an action based on policies. It does not evaluate the consequences of the policy as there is no contribution from the cost module. Further, as the actions are not evaluated, the configurator configures the cost module to generate 0 gradient meaning the system is not trainable in this mode.

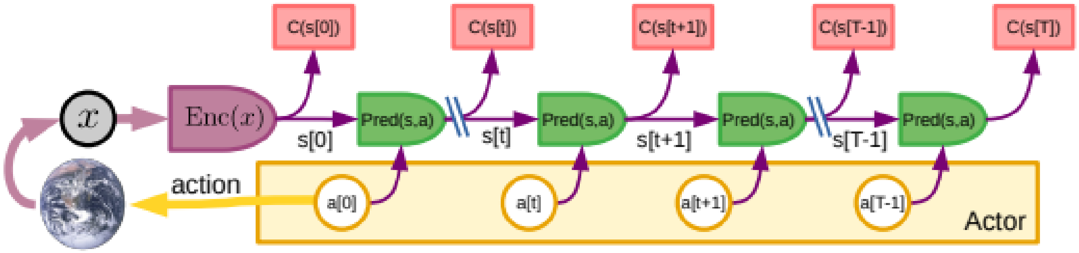
Mode 2 is more deliberate and involves planning. The system is also trainable in this mode. The input is generated from the perception module. The cost module calculates the immediate energy of the system. The actor proposes a set of actions based on the current state of the world. The world model simulates the real world using which the cost module again calculates the cost of the proposed actions. The actor adjusts the action until we find the cost converges. The actions are then executed. The information from the whole episode is stored in the short term memory buffer.

Although we are generating practically useful predictions in mode 2, it is expensive. The system has to go through a high performance phase engaging all the modes. It makes it slow. We need to be able to adjust our policies based on our experiences. Hence, we use the information stored in the buffer to retrain the policies when the system is operating in mode 2. Eventually, we end up with well adjusted policies that we will be able to produce the same quality output using mode 1 and mode 2 will only be used for learning new tasks.



Mode 1- perception action circuit. The prediction is based on policies

.



Mode 2 - The world model is engaged to plan and determine the consequences of the proposed action. Then the actions are adjusted to be the most compatible for the current task.

How does JEPA fit in this architecture??

One of the most complex and consequential components of the proposed system is the world model and it implicitly understood that to get high quality predictions about the world the world model should be accurately trained. As the world model is based on pattern completion, i.e predicting the unobserved events of the world, we are presented with 3 issues when training such a model.

- We need a diverse collection of training data. This reduces uncertainty and creates diverse representations of the world. It is important to note, quantity is not merely as important to quality and diversity in our case.
- The world is not completely predictable. This is more impactful than the first problem as it leads to uncertainty. A small fraction of uncertainty at this stage can lead to bad performance of the mode.
- The third issue is that of planning and abstraction. Humans break down complex tasks into smaller parts and these smaller goals are achieved extending to fulfilling the larger task. We need our architecture to be able to plan in a hierarchical manner.

To address these issues firstly, we use self-supervised learning where we learn the relationship between the input and the output rather than being able to predict the output. The general framework is based on EBMs (energy based models). The EBM outputs a low value when y is loosely obtainable from x . This helps us in concretely designing a mode 2 system where the dependencies between past states and future states are understood. In other words, we capture the representation of the world rather than predicting the world by making sure more information is learnt.

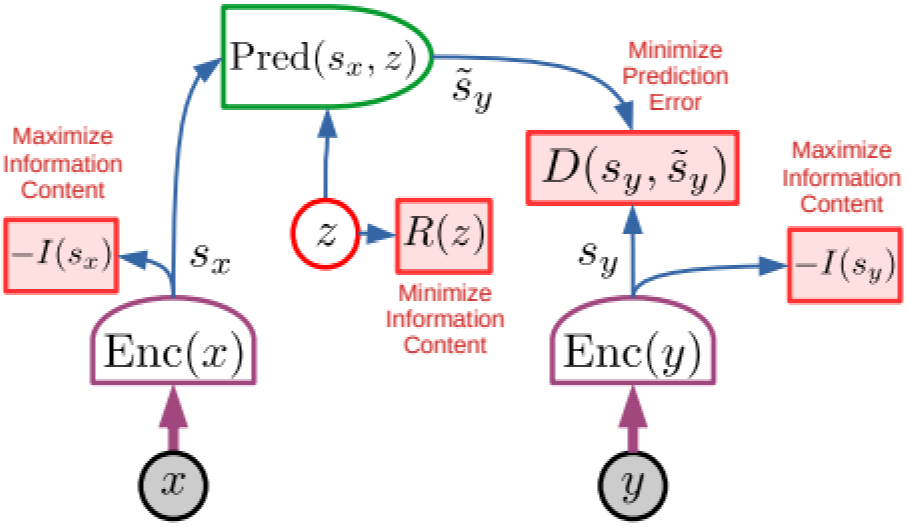
To handle uncertainty we use latent variables. A latent variable is an input value which is inferred rather than observed. We use the latent variable to parametrize the relationship between the input and the output. It is used to represent information about the output that cannot be extracted from the input. In our case, we combine the latent variable with the EBM to create an energy function that depends on the input(x) output(y) and the latent variable(z) which outputs a low energy when y can be represented with x and z.

When we build an architecture using the principles discussed above we arrive at JEPA(Joint Embedding Predictive Architecture). It is an architecture for world models where we cannot directly predict y using x and in that terms it is non generative.

We use 2 encoders one each for input and output and these encoders may not share the same architecture. We predict y from x using z and the energy is simply the error of the prediction. But, the prediction happens in a representation space which means that we do not need to predict every dimension of y from this. Traditionally, EBMs are trained using contrastive methods but, studies have shown that these methods fail when the number of dimensions of x and y are high. As we are representing the world, even encoding these values still produces high dimensional data. So we follow the following principles instead of contrastive methods.

- Maximize the information of the encoding (s_x) obtained from x
- Maximize the information of s_y obtained from y
- Make s_y easily predictable from s_x
- Minimize the information contained in the latent variable z

The first 2 principles prevent vanishing loss due to information collapse by producing rich representations. The 3rd principle makes sure that both representations are relevant to each other. The last principle also prevents information collapse.

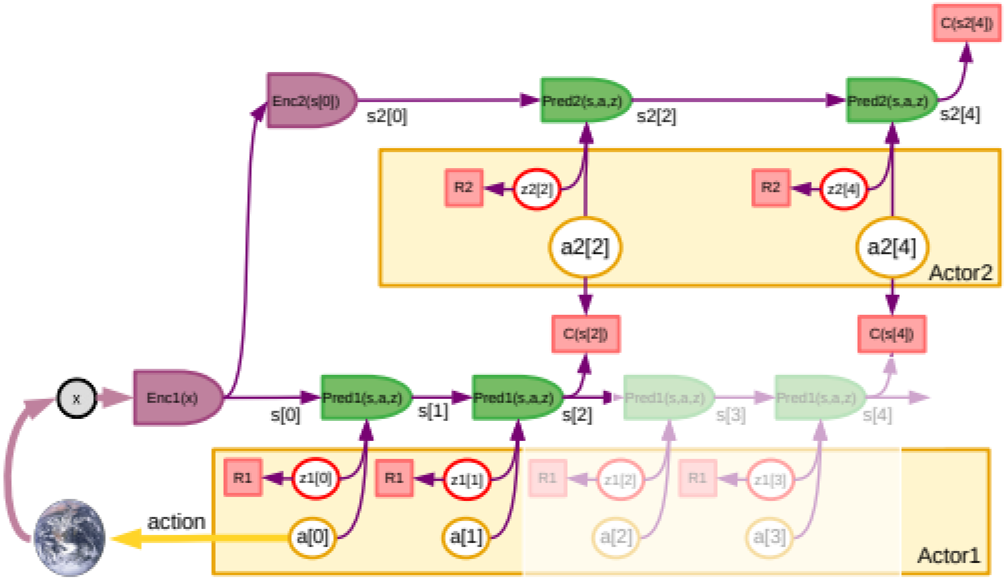


The architecture of a simple JEPA following the 4 principles of non contrastive methods.

This architecture can be used in a hierarchical architecture. The advantages of doing so is that we can use JEPAs individually to make predictions of multiple orders. The combined output of the architecture will be the decomposition of complex tasks where each smaller task will be handled by a JEPA. This is called H-JEPA(hierarchical JEPA). It has the ability to learn representations on various levels of abstractions. In the diagram below, the first JEPA learns lower level abstractions and performs short term predictions. The representations produced by the encoders are used by another JEPA to learn higher level abstractions and produce long term predictions.

This architecture again mimics system 1 and system 2 where the lower level JEPAs are used for immediate and spontaneous predictions and higher level JEPAs are used for planning and long term prediction. This allows us to make predictions on multiple time scales too.

Let us better understand this using an example. An autonomous car using this architecture would predict the trajectory and the controls of the car for a short term using lower level abstractions. However, as we cannot predict the long term trajectory of the vehicle as it is dependent on external factors, it creates multiple possible scenarios and has potential trajectories based on the situation it may encounter in the long term.



The above diagram shows hierarchical JEPA (H-JEPA)

Conclusion

To summarize the article, we have discussed the paper “A Path Towards Autonomous Machine Intelligence” where the goal is to build a model that perceives, observes and makes predictions on the world using common sense. This is a major milestone in machine learning as we are finally making headways to achieve artificial general intelligence. This is just the beginning of a new era of machine learning. The proposed architecture is a union of reinforcement learning techniques, deep learning, multimodal systems all combined to form one all powerful model capable of mimicking a human. However, with our current understanding of the technology and existing resources it is not practically possible to implement a system like this. But, I am glad to say we are not far away from creating one.

References

- [1]. Kahneman, Daniel. *Thinking, fast and slow*. Macmillan, 2011.,
- [2]. LeCun, Yann. "A path towards autonomous machine intelligence." *preprint posted on openreview* (2022).
- [3]. Ha, David, and Jürgen Schmidhuber. "World models." *arXiv preprint arXiv:1803.10122* (2018)