AGI: Large-Scale Neural Network Modelling

Poondru Prithvinath Reddy

December 19, 2019

AGI : LARGE-SCALE NEURAL NETWORK MODELLING

Poondru Prithvinath Reddy

ABSTRACT

One of the interesting applications is in creating computers at the AGI (Artificial General Intelligence) level equal to human intelligence. Designing AGI is basically creating the human brain on a computer. General intelligence has tons of components and required testing on many functions from image recognition, to the ability to write an essay, to solving Inverse Kinematic problems, etc. In this paper, we present a AGI – a large scale neural network model to achieve General Intelligence involving different components. The AGI model contains three subsystems : (1) EEG based system where Xiang Zhang et all, proposed a novel deep neural network based learning framework that affords perceptive insights into the relationship between the EEG data and brain activities and designed a joint convolutional recurrent neural network that simultaneously learns robust high-level feature presentations through low-dimensional dense embeddings from raw EEG signals. The proposed approach has been to use results of this study as it is and use simulated conditions as true input for our study; (2) Image system that contains an encoder to convert the input into abstract representations, and a deep image reconstruction which optimizes the output of the decoded images so that it more closely resembles the actual or true images, in combination with a multi-layered convolutional neural network (CNN) to simulate the same processes that occur when a human brain perceives images; (3) a LSTM that combines inputs in the forms of both EEG and Image, and predict text symbols associated with images and next images accordingly. In this work, the

proposed AGI model illustrates the ability to incrementally learn different functions and form a machine programming loop that enables interactions between EEG signals and Image system, and possibly possess human-like general intelligence.

INTRODUCTION

One of the most interesting applications is in creating computers at the AGI(Artificial General Intelligence; equal to humans) level of intelligence.

Right now, we have ANI (artificial narrow intelligence); AI that is good at specific tasks. Example A self-driving car won't be good at predicting stock, but it's intelligent at driving. In the future, we're striving for AGI which can drive cars and predict stocks plus do everything in between.

Designing AGI is basically creating the human brain, on a computer. In order to reach AGI, It would be ideal if AGI designs itself.

It is possible to achieve AI designing AI, and perhaps Genetic Algorithms could offer solution to this.

Presently, what we're lacking is architecture for our powerful computers to run intelligently. So in theory, we can use our super powerful computers, and have them use the genetic algorithm to find the best structure for their code and they would evolve their code by running the codes on various tasks to test intelligence.

The above function will be very lengthy and complex. General intelligence has tons of components, and it is necessary to test the code on many functions from image recognition, to the ability to write an essay, to solving Inverse kinematic problems, etc. This will take many years to develop.

METHODOLOGY

In an attempt to model human-level General Intelligence patterns in machines, We created a large-scale artificial neural network inspired by the human vision, audition, dynamics, memory and attention processes that take place as people are performing a given task including the maintenance and manipulation of information.

We proposed a AGI model with an aim to form a human-like general intelligence programming process in a machine.

The Artificial General Intelligence model has three key components: an EEG system, an image system and an artificial neural network implemented by LSTM. An EEG system consists of a unified deep learning framework that leverages recurrent convolutional neural network to capture spatial dependencies of raw EEG signals based on features extracted by convolutional operations and temporal correlations through RNN architecture. Also, an Autoencoder layer is fused to cope with the possible incomplete and corrupted EEG signals to enhance the robustness of EEG classification. The results of their study, Xiang Zhang et all, has been used as it is and used simulated conditions as true input for our study.

The second sub-system, an image system that contains an encoder to convert the input into abstract representations, and a multi-layered CNN to classify image scenarios from real level representations. The final component of the AGI Model mimics the human brain by a LSTM, combining inputs of both image and EEG representations to predict text symbols associated with images and next images accordingly.

We evaluated the AGI Model in a sequence of experiments and found that it successfully acquired general intelligence tasks in a cumulative way. The technique also formed the 'machine programming loop," showing an interaction between EEG signals and images. In the future, the AGI model could aid the development of more advanced AGI, which is capable of human-level general intelligence strategies on a machine.

ARCHITECTURE

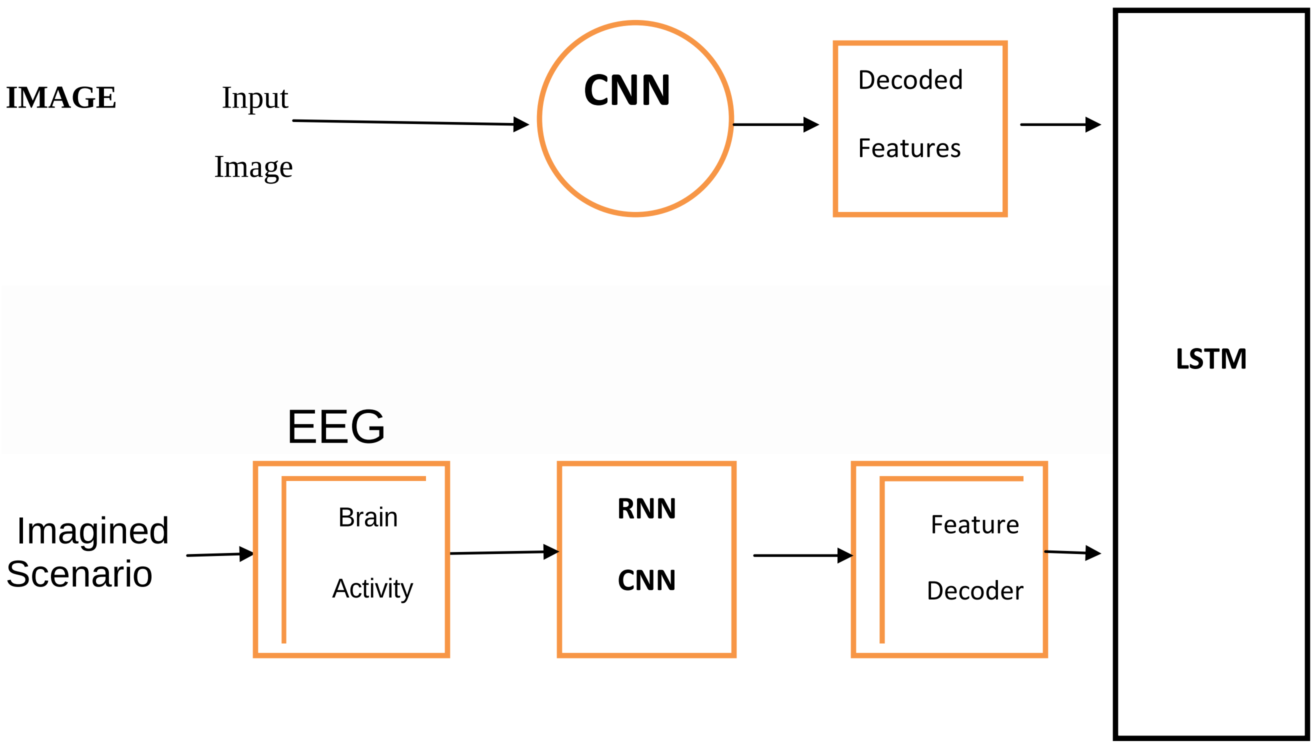

Figure 1 : AGI Model

IMAGE SUBSYSTEM

We have already heard of image or facial recognition or self—driving cars. These are real-life implementations of Convolutional Neural Networks (CNNs). We implement these deep, feed-forward artificial neural networks by overcoming overfitting with the regularization technique called “dropout”.

We have used the MNIST dataset for traing and testing the image processing.The **MNIST database** (Modified National Institute of Standards and Technology database) is a large database of handwritten digits that is commonly used for training various image processing systems. The MNIST database contains 60,000 training images and 10,000 testing images. To load the data, we first need to download the data from the link and then structure the data in a particular folder format to be able to work with it.

From the above, we can see that the training data has a shape of 60000 x 784: there are 60,000 training samples each of 784-dimensional vector. Similarly, the test data has a shape of 10000 x 784, since there are 10,000 testing samples.

The 784 dimensional vector is nothing but a 28 x 28 dimensional matrix. That's why we will be reshaping each training and testing sample from a 784 dimensional vector to a 28 x 28 x 1 dimensional matrix in order to feed the samples in to the CNN model.

As a first step, we convert each 28 x 28 image of the train and test set into a matrix of size 28 x 28 x 1 which is then fed into the network.

The Deep Artificial Neural Network

We used three convolutional layers:

- The first layer will have 32-3 x 3 filters,
- The second layer will have 64-3 x 3 filters and
- The third layer will have 128-3 x 3 filters.

In addition, there are three max-pooling layers each of size 2 x 2.

We used a RELU as our activation function which simply takes the output of max_pool and applies RELU.

Flattening layer:

The Output of a convolutional layer is a multi-dimensional Tensor. We want to convert this into a one-dimensional tensor. This is done in the Flattening layer. We simply used the reshape operation to create a single dimensional tensor.

Fully connected layer:

Now, let's define and create a fully connected layer. Just like any other layer, we declare weights and biases as random normal distributions. In fully connected layer, we take all the inputs, do the standard z=wx+b operation on it. The Fully Connected Layer has 128 Neurons.

We added Dropout into the network to overcome the problem of overfitting to some extent and also to improve the training and validation accuracy. This way, turning off some neurons will not allow the network to memorize the training data since not all the neurons will be active at the same time and the inactive neurons will not be able to learn anything.

The Test results show good accuracy between training and validation data

CONVERTING EEG SIGNALS TO TEXT

An electroencephalography (EEG) based Brain Computer Interface (BCI) enables people to communicate with the outside world by interpreting the EEG signals of their brains to interact with the world. In a research paper, Xiang Zhang et all (https://arxiv.org/pdf/1709.08820.pdf) proposed a novel deep neural network based learning framework that affords perceptive insights into the relationship between the MI-EEG(Motor Imagery EEG) data and brain activities. They designed a joint convolutional recurrent neural network that simultaneously learns robust high-level feature presentations through low-dimensional dense embeddings from raw MI-EEG signals. They also employ an Autoencoder layer to eliminate various artifacts such as background activities. The proposed approach has been evaluated extensively on a large scale public MI-EEG dataset and a limited but easy-to-deploy dataset collected in their lab. The results show that the adopted approach outperforms a series of baselines and the competitive state-of-the art methods, yielding a classification accuracy of 95.53%. The applicability of their proposed approach is further demonstrated with a practical BCI system for typing.

The main offerings of this paper are highlighted as follows:

- Designed a unified deep learning framework that leverages recurrent convolutional neural network to capture spatial dependencies of raw EEG signals based on features extracted by convolutional operations and temporal correlations through RNN architecture, respectively. Moreover, an Autoencoder layer is fused to cope with the possible incomplete and corrupted EEG signals to enhance the robustness of EEG classification.

- Evaluated extensively the model using a public dataset and also a limited but easy-to-deploy dataset that was collected using an off-the-shelf EEG device. The experiment results illustrate that the proposed model achieves high level of accuracy over both the public dataset (95.53%) and the local dataset (94.27%). This demonstrates the consistent applicability of the proposed model.

- Also presented an operational prototype of a brain typing system based on the proposed model, which demonstrates the efficacy and practicality of adopted approach.

The proposed model consists of a design - an RNN model consisting of three components: one input layer, 5 hidden layers, and one output layer. There are two layers of Long Short-Term Memory (LSTM) cells among the hidden layers. While RNN is good in exploring the temporal (inter-sample) relevance, it is unable to appropriately decode spatial feature (intra-sample) representations. To exploit the spatial connections between different features in each specific EEG signal, a CNN structure is designed. The CNN structure is comprised of three categories of components: the convolutional layer, the pooling layer, and the fully connected layer. The convolutional layer contains a set of filters to convolve with the EEG data and then through the feature pooling and non-linear transformation to extract the geographical features. CNN is well-suited to extract the spatial relevance of the 2-D input data efficiently.

Next, a feature adaptation method is designed to map the stacked features to a correlative new feature space which can fuse the temporal and spatial features together and highlight the useful information. To do so, an Autoencoder layer is introduced to further interpret EEG signals, which is an unsupervised approach to learning effective features. The Autoencoder is trained to learn a compressed and distributed representations for the stacked EEG feature X'. The input of Autoencoder is the stacked temporal and spatial feature X'. Assume h, $\mathring{X}'$ denote the hidden layer and output layer data, respectively.

The data transformation procedure is described as the following:

$$h = \mathrm{Wen}\ X' + \mathrm{ben}$$

$$\mathring{X}' = \mathrm{Wde}\ h + \mathrm{bde}$$

where Wen, Wde, ben, bde denote the weights and biases in the encoder and decoder.

LSTM SUBSYSTEM

The LSTM subsystem contains a LSTM and a fully connected layer. It receives inputs from both EEG and image subsystems in a concatenated form of $\mathbf{c}(t) = [\mathbf{T}(t), \mathbf{I}(t)]$ at time t, and gives a prediction output $\mathbf{a'}(t) = [\mathbf{T'}(t), \mathbf{I'}(t)]$, which is expected to be identical to $\mathbf{a}(t + 1) = [\mathbf{T}(t + 1), \mathbf{I}(t + 1)]$ at time t+1. This has been achieved with a next image prediction (NIP) . So given an input image, the LSTM can predict the corresponding image description. The strategy of learning by predicting its own next element is essentially an unsupervised learning.

The Training is based on the next image prediction (NIP). The LSTM-FC is trained by the NIP principle, where the goal of the LSTM-FC is to output the representation vectors (including both text and image) of the next image / element. At time T, the LSTM of AGI Model generated the guided digit instance, which required the understanding of the previous text description and observed images.

The LSTM subsystem was trained separately after vision and EEG components had completed their functionalities. We have trained the network to accumulatively learn different images, and the related text results. Finally, it is demonstrated how the network forms a thinking loop with text descriptions and observed images.

The LSTM layer serves as working memory, that takes the concatenated input [**T**,**I**] from both EEG and image subsystems, and output the predicted next image representation that could be fed back into both subsystems to form a guided loop.

RESULTS

The objective of this study is to classify images along with image descriptions given by participants in text form on the test dataset. The technique involved training neural networks to associate patterns of brain activity with human thought about images in text form. Both the image and its description are concatenated to predict the outcome of the Model.

In this paper, we have not implemented converting EED signals to Text but taken the results of the study by Xiang Zhang et all (https://arxiv.org/pdf/1709.08820.pdf

) as the basis. However, the results of the study are simulated into different image descriptions in Text form to test our architecture of AGI Model.

After 200 steps training, AGI Model could not only reconstruct the input image but also predict the element / image with associated text symbols and correct image description just after the image classification. AGI has the capacity to correctly predict the next image and the associated description with correct text symbols at the proper time point. After training of 200 steps, AGI Model could classify various images with correct text description (accuracy = 16%). Note that, the classification process is not performed by large dataset, but by small number of training steps or iterations which is resulting in less accuracy.

CONCLUSION

In this paper, we have described AGI – a large scale neural network model to achieve human-like General Intelligence involving different functions. We presented a hybrid deep learning model to decode the raw EEG signals for the aim of converting the user"s thoughts to texts. The model employs the RNN and CNN to learn the temporal and spatial dependency features from the input EEG raw data and then stack them together. Our proposed approach adopts a method called " deep image reconstruction " which optimizes the output of the decoded images so that it more closely resembles the actual or true images, in combination with a multi-layered CNN. We evaluate our approach on simulated dataset for EEG and MNIST dataset for Image subsystem. The results are encouraging and form the basis for creating human-level General Intelligence on a machine.

REFERENCES

1. Xiang Zhang, Lina Yao, Quan Z. Sheng, Salil S. Kanhere, Tao Gu, Dalin Zhang "Converting Your Thoughts to Texts: Enabling Brain Typing via Deep Feature Learning of EEG Signals" , https://arxiv.org/pdf/1709.08820.pdf