International Journal of Engineering & Technology, 7 (1.1) (2018) 696-698

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Implementation of conventional communication system in deep learning

Satyanarayana. P [1], Charishma Devi. V [2] *, Sowjanya. P [3], Satish Babu. N [2], Syam Kumar. M. N. V. S [2]

[1] Professor, Department of Electronics and Communication Engineering, KLEF
[2] Student, Department of Electronics and Communication Engineering, KLEF
[3] Research Scholar, Department of Electronics and Communication Engineering, KLEF
**Corresponding author E-mail: charishma.valluri@gmail.com*

Abstract

Machine learning (ML) has been broadly connected to the upper layers of communication systems for different purposes, for example, arrangement of cognitive radio and communication network. Nevertheless, its application to the physical layer is hindered by complex channel conditions and constrained learning capacity of regular ML algorithms. Deep learning (DL) has been as of late connected for some fields, for example, computer vision and normal dialect preparing, given its expressive limit and advantageous enhancement ability. This paper describes about a novel use of DL for the physical layer. By deciphering a communication system as an auto encoder, we build up an essential better approach to consider communication system outline as a conclusion to-end reproduction undertaking that tries to together enhance transmitter and receiver in a solitary procedure. This DL based technique demonstrates promising execution change than traditional communication system.

Keywords: *Auto Encoder; Deep Learning; Machine Learning; Neural Networks.*

1. Introduction

Computers can learn from past experiences and that is machine learning do. The past experience of computers is data. Let's see an example we consider 2 different houses based on square feet and obviously have different prices, we can consider the best house using machine learning based on graph between sq. Feet houses on one axis and price on another axis. We use machine learning because we have huge number of computers and huge amount of data that is beyond human comprehension and we have lots of computers and we take their use to work with data and make better decisions and computers are cheap [1]. Machine learning is used for data sciences. It is use of algorithms to create an idea of data to machines. This algorithm produces optimization of data and gives the best results.

All we do in machine learning is a black box where the logic is not open to people. Some examples of machine learning are self-driving cars, facial recognition, stock trading systems, simple data analysis, learning credit card fraud detection. There are lot of terms that we use in machine learning most of them are similar but they have a different context in machine learning [2]. We have datasets and we compare our data based on the data sets which contains columns and rows and are called features and

instance respectively. If the data has same values that values won't change in the dataset and this dataset is used to train the machine which is used to predict the output of present given input. There are different types of features like numeric, interval, ordinal, categorial. Machine learning is divided into two different parts called supervised and unsupervised. Unsupervised is the one in which machine learned how to work on its own and supervised is the one in which machine it is under no one's supervise. The popular algorithm used to make clustering is the k-means algorithm and K is the no of clusters that we end up at the end and problem is that we end up we need to give the value of k beforehand [3].

Machine learning is closely related to data mining rather than artificial intelligence. For machine learning we need to have a huge data set which is fetched by the classifier which determines the output from dataset. Machine learning is dependent quite a lot on math but not always on math but all the time it is about designing neural network but it is the base. The basic language we use is the python programming language. Python is one of the ways to work on machine learning we can also use apple machine like mac book and ios11 have M L kit and we can give data to that and it can predict output. Implementation of machine learning can be done in java script also and has some future in coming days right now we cannot say anything surely.

2. Deep learning and neural networks

Neural network is a piece of math. There are many neural networks recently which are good in recognizing images and also for recognizing the handwritten words. Basic pre-requisite for understanding neural network is that we need perceptron. Neural networks are inspired by brain and contains neurons which in our case contains numbers. For example consider a neural network with a bunch of neurons corresponding to 28/28 pixels of input image of number 9. Total 784 neurons which contains the value that defines the grey scale value of the corresponding pixel. All these neurons make the first layer of network. Last layer contains 10 neurons that contains the numbers as output. In between it contains many hidden layers [4].

Deep learning is also machine learning which contains more number of layers than that of machine learning neural network. Deep learning methods are more accurate than humans and GPU make training of network much faster and finally huge date became accessible to deep learning data. One type of neural network for deep learning is convolutional neural network which mainly works on image recognition. Deep usually refers to number of layers in neural network. Usually machine learning contains two to three layers while deep neural network contains as many as hundreds of layers.

The core of deep learning is the artificial neural networks and its core is artificial neuron and each neuron is connected to other neuron on other layers. Applications of deep learning are many. It is used in natural language processing and it has many examples like sentiment analysis which discusses about reviews of a product. We can't do it manually because we have thousands of reviews of thousands of products and deep learning has been proved to outperform all other algorithms.

We can also use that in part of speech recognition, question answering where we are given a question and we get the answers, language translation, word analogies, object detection in which the deep learning algorithm detects the image and all the things in that image, automatic caption generation to an image and also what is happening in the image with so much accuracy, automatic tagging in face book in which it recognizes who the person in face book and automatically tags, automatic hand written text, medicine where in images generated by MRI's can actually analyse the image and output is the result where he has tumour or not like that, using deep learning to help blind people to make them walk through the path so that he can understand what is present in front of him helping him to avoid hurdles and to have a safe journey on his own and also help them read a book by text reading, recommender systems in which based on our past searches some sites provide what we might be interested at that time, self-learning cars in which it has many sensors and process the data using deep learning and reinforcement learning and based on that data the car moves forward and efficiently uses the space around them.

Deep learning got the name from the number of layers in the artificial neural network as deep learning has more number of hidden layers. Below figure is example network which contains 5 layers. First layer is the input layer through which we give input to the network and the next three layers are the hidden layers where the processing of data takes place. The number of hidden layers vary from network to network and are can be in the ranges of three to ten. The last layer is the output layer where we get the output of the network. To process the data using deep neural network we need high computing processors.

Each layer in deep neural network has activations function to work on the data. Some activation functions are ReLu, Sigmoid etc... and each neuron in the network has some weight and they get adjusted during training period so that when it is deployed it gives the most probable result. These weights are adjusted based on the data are on algorithm called back propagation which is implemented based on gradient descent which is an optimization technique. It improves the accuracy of the output of our models prediction. In neural network we need to arrange the layers so that they presume the output. First each neuron is initialized with certain weights. These weights are needed to be varies to get the accurate value and is done by minimizing the errors and we do that process by gradient descent. We need the weights of the neurons optimal and is achieved by this.

3. Difference between deep learning and machine learning:

Machine learning estimates output based on the predefined features while deep learning can handle things on its own. Deep learning requires needs to compute all features and so requires huge amount of data when compared to that of machine learning.

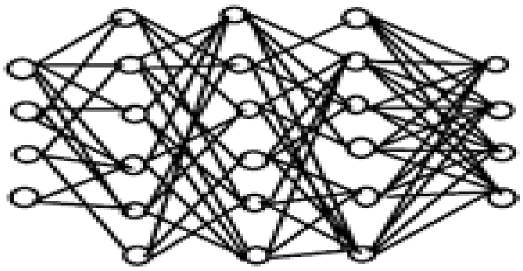

Fig. 1: Deep Neural Network example.

To enter such large datasets and run program effectively deep learning needs GPU and very high-end processors while machine learning just needs CPU [5]. Deep learning is also known as an end to end system as it predicts output in just one step. Training time of deep learning is very high because of high range of mathematical computations when compared to machine learning. Consider 2 hidden layers each with 16 neurons. And also activation in one layer determine the activation in other layers and also the key of the network has an information processing mechanism comes down to exactly how these activations in one layer bring about activations in other layers. And then we train the network to recognize recognise images.

4. Auto-encoder

An auto-encoder is an interesting variant in which it both encodes and decodes the data with notably two changes. First, the number of input neurons and output neurons of the network are same because of which the output size of image is same an input and most importantly the image is also same as the input. Generally, why would anyone want to design a model which gives as output as the input is that not a waste of time. Now we consider the second part where we might find a congested area in the network where the number of neurons needed to be present are lesser than needed. So the network has to compensate for the shortage and need to work as efficiently as possible.

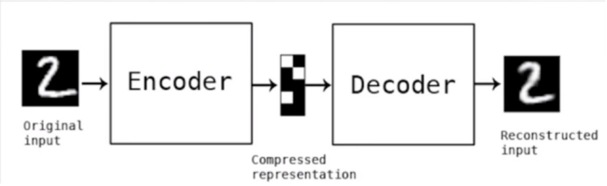

Fig. 2: Auto-Encoder Working Example.

Auto encoders are capable of giving the output even with the less neuron. so, this might be used to reduce the noise in the given picture which states that auto encoders are used in de-nosing the image. An auto encoder tries to learn the identity function in an unsupervised manner to bring the output which is like input but by unsupervised what we meant was that only the data is available without any prior training which mean it learned on its own. As an example consider an 10 by 10 pixel image as the input which means we have 100 input units but we have only 50 hidden units therefore we are making the system learn the compressed format of the input given the vector of hidden unit activations. To make the original 100-unit output, if the inputs are IID random, learning the structures will be difficult. However in real all the data we take in is nit random and in most of the cases the input we take in is correlated. This algorithm would be smart enough to discover the correlations in the input data [6]. In fact, a simple auto encoder can end up learning a low representation very similar to PCA's. because of the depth auto encoder, us much more powerful than PCAs.

De-noising auto encoders are same as the regular auto encoder. But in this case we randomly corrupt the input data by presuming some of the input data as zeros so as to allow more robustness. Input vector is randomly corrupted where some inputs are set to zero. The auto encoder models the data and corrupts it say it to be y and the output vector reconstructs the data z from y. Now the output z is compared with the original input vector data. As we know nothing is perfect in the start we do get few errors in the start and this error is back propagated to the network to optimize and get the correct output. Auto encoders can be stacked. For each additional layer that is going to get added we are increasing the modelling capacity. However random initialization of model weights might get us into trouble in future so for that we do training in layer wise. It is a type of neural network mostly used for dimensionality reduction in computer vision. A novel idea is to replace the traditional communication with the auto encoder will leave a lots and lots of challenges. The three parts of communication system namely transmitter, receiver and channel will be considered as encoder, noisy layer and decoder respectively. Number of messages will be given to the auto encoder system as a whole and the original message is retrieved. We use libraries which include Keras, TensorFlow, numpy, matplotlib.In this paper, we considered input messages represented by ‘M’. The bits corresponding to M is $K = \log_2 M$. The rate of the channel is R = K/n_channel_use. n_channel_use is the term used for representing the usage number of channels. We use n_channel_use as the number of bits corresponding to M. Therefore, our rate is R=1. At first, we generate 10,000 random integer values which vary between 0 and M-1. Each value is one hot encoded with a vector of length M. the whole sample data is converted to numpy array. Now, we will construct a neural network. Our network contains 6 layers. Input layer, two dense layers, Gaussian noise layer, and two dense layers at the decoder.

The dimensions of the layers are as follows.

- Input: M nodes.
- Dense1: M nodes.
- Dense2: n_channel use nodes.
- Gaussian noise layer: n_channel use nodes.
- Dense3: M nodes.
- Dense4: M nodes.

The model is trained with the generated samples. Now, the encoder part and decoder part are separated and tested with newly generated samples. The output is tested for bit error rate (BER). Fig.3 differentiates BER of a communication system employing Quadrature Phase Shift Keying (QPSK) modulation against BER achieved by the trained auto encoder (n=2, k=2), we also provide the BER of 8PSK against trained auto encoder (n=3, k=3) in Fig.4. This result proves that the auto encoder has learned without any prior knowledge of an encoder and decoder function that together achieves the better performance than the traditional communication system.

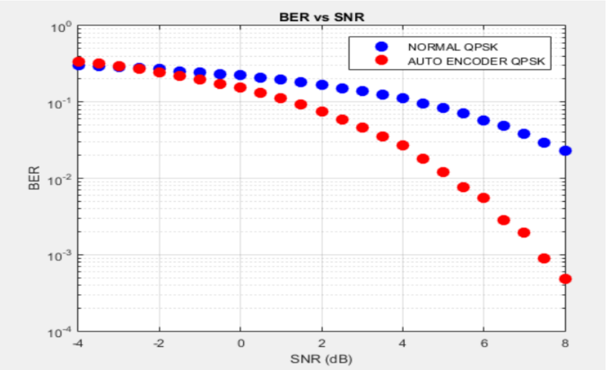

Fig. 3: BER Vs SNR Graph of Conventional QPSK Output and Auto Encoder (N=2, K=2) Output.

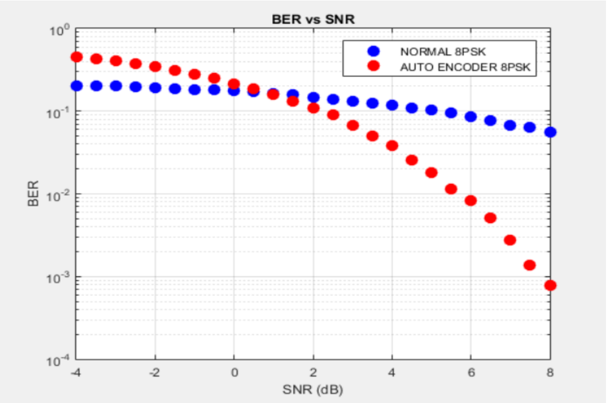

Fig. 4: BER vs. SNR Graph of Conventional 8PSK Output and Auto Encoder (N=3, K=3) Output.

5. Conclusion

This paper discusses about a promising new utilization of DL to the physical layer. It presents a new perception about communication system as a conclusion to-end remarking enhancement utilizing auto encoders to mutually learn transmitter and receiver usage. Comparing with traditional communication system this model reveals better BER performance.

References

[1] C. E. Shannon, “A mathematical theory of communication,” Bell Syst. Tech. Journal, vol. 27, pp. 379–423, 623–656, 1948.

[2] T. J. O’Shea and J. Hoydis, “An introduction to machine learning communications systems,” arXiv preprint arXiv: 1702.00832, 2017.

[3] N. S. Muhammad and J. Speidel, “Joint optimization of signal constellation bit labelling for bit-interleaved coded modulation with iterative decoding,” IEEE Commun. Lett. , vol. 9, no. 9, pp. 775–777, 2005.

[4] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv: 1412.6980, 2014.

[5] M. Abadi et al., “TensorFlow: Large-scale machine learning on heterogeneous distributed systems,” arXiv preprint arXiv: 1603.04467, 2016. [Online]. Available: http://tensorflow.org/.

[6] F. Chollet, “Keras,” 2015. [Online]. Available: https://github.com/fchollet/kera.