International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 09 | Sep 2019 www.irjet.net p-ISSN: 2395-0072

Recognition of Handwritten Characters based on Deep Learning with Tensorflow

JEMIMAH K

Research Scholar, School of Computer Science and Engineering, Bharathidasan University, Trichy, India.

Abstract - *Artificial Neural Network (ANN) and Deep Learning has brought a massive twist in the field of Machine Leaning, in recent times. The advancements in Deep Learning is mainly due to the new state-of-art innovations in the field of Computer Vision (CV). Convolutional Neural Network (CNN) is at the center of spectacular advances in Deep Learning that mixes Artificial Neural Network (ANN) and recent Deep Learning strategies. It is well known that Convolutional Neural Networks (CNN) have been the source of many major breakthroughs in the field of Deep Learning where it powers major advances in of Computer Vision (CV), where self-driving cars, robotics, drones, security, medical diagnoses, and treatments for the visually impaired are some of its applications. Convolutional Neural Networks (CNN) are widely used in pattern recognition, speech recognition, face recognition, sentence classification, text categorization, document analysis, scene and handwritten digit recognition. In this paper, a CNN model is built to recognize the handwritten digits with various hidden layers and epochs. Using Tensorflow, which is Google's open source machine learning and deep learning framework, the model is conveniently and flexibly built. The experiment is done using Modified National Institute of Standards and Technology (MNIST) dataset.*

Key Words: Recognition of Handwritten Characters, Artificial Neural Network (ANN), Deep Learning, Convolutional Neural Network (CNN), MNIST dataset and Hidden Layers.

1. INTRODUCTION:

In the recent years, Deep Learning has become somewhat a buzzword in the tech community. Generally, Deep Learning is a model which takes an input X and uses it to predict an output Y. Deep Learning algorithm uses Neural Network to find the association between a set of inputs and outputs. Neural Networks is one of the most popular algorithm at present which outperform other algorithms in accuracy and speed. Recent advancements in the field of Deep Learning has made computer vision application leap forward: from unlocking mobile phone with our face to safer self-driving cars.

Deep Learning's fundamental example is Convolutional Neural Network (CNN), which is a sophisticated model that pushes the evolution of Artificial Intelligence (AI) by offering systems that simulate different types of biological human brain activity. Convolutional Neural Network (CNN) is the most widely used Machine Learning algorithm for image classification and it is the architecture behind computer vision applications.

Convolutional Neural Network (CNN) is a go-to model on every image related problems. The main advantage of Convolutional Neural Network (CNN) compared to its predecessors is that it automatically detects the important features from the inputs without any human supervision. It is successfully applied to recommender systems, natural language processing and more. Convolutional Neural Network (CNN) is a very powerful and an efficient model which performs automatic feature extraction process to achieve superhuman accuracy.

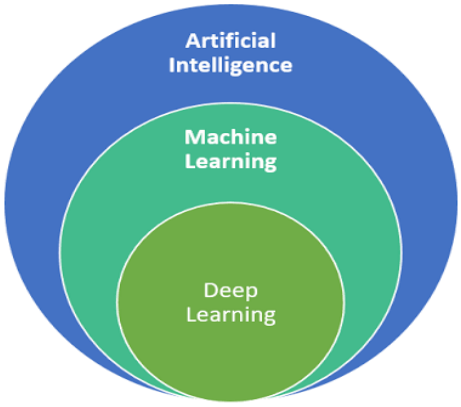

Fig 1: *Relationship between AI, Machine Learning and Deep Learning.*

Tensorflow is an open-source framework for Machine Learning. It is a software library for numerical computation. It is a product by Google Brain Team in Google's Machine Intelligence research organization for conducting research in the field of Machine Learning and Deep Neural Network. Tensorflow, which is a product of Google has proven to be a scalable, clean, flexible and efficient library. Tensorflow operations are arranged in a graph known as Computational Graph. A cloud service that was launched by Google is used to run Tensorflow Graphs. The Computational Graph can be broken into chunks and can run on any CPUs and GPUs parallelly. It is known as parallel computation. Tensorflow provides several optimization nodes to search for parameters that minimizes the cost function.

The proposed model gives an accuracy of about 98% with a very low error rate of 0.2% which is a good rate and the predictions are done. The existing systems have a separate preprocessing step but in a Convolutional Neural Network (CNN) model for Recognition of Handwritten Characters based on Deep Learning don't have a separate preprocessing step because during preprocessing some features might get affected and lost.

Convolutional Neural Network (CNN) for Handwritten Character recognition is not like other model where the features are selected and extracted. Here, the model automatically draws the features from the input images which is called feature map. Then the spatial dimensions are reduced that gives the model a faster computing power for training the data. Finally, the hidden layers gives out some value which is changed to a range of values using activation function to exactly classify the classes.

2. RELATED WORKS:

A comparative study on the various algorithms for digits classification with their respective accuracies [1]. The feature selection method used for handwritten digits classification consist of various algorithms (i.e., linear, non-linear, ensemble and deep learning). The experimental results proved that 60% of the training time was reduced by using the subset of the complete set of features [2].

An Android Application was developed for recognizing handwritten characters using Neural Network technique. Random datasets were used for conducting the experiments. It uses structural features and k-nearest neighbor classifier to classify the characters [3].

The recognition and implementation of handwritten characters in Chinese language uses a Deep Learning model where the dataset is divided into four sub-sets with different classification iterations. The dataset has 260 different people's handwritten characters [4].

A Multiple-Cell Size (MCS) approach was proposed to utilize the Histogram of Oriented Gradient (HOG) feature with Support Vector Machine (SVM) for handwritten digits classification [5]. A combined approach of feature selection and machine learning technique was used to find the handwritten characters where preprocessing was done for the input images. Feature extraction, feature selection and classification are the techniques used in the paper [6].

Handwritten recognition is very complicated because of miscellaneous human handwriting, different size and shape of letters, angle. Using feature extraction technique a sliding window is shifted over the image from left to right which identifies the important characters of the image [7]. The Deep Neural Network based Convolutional Neural Network model is analyzed using MNIST dataset which achieves an accuracy rate of 99% at sigmoid activation function with cross-entropy error function [8].

3. DATASET:

Recognition of Handwritten Characters uses MNIST dataset, known as Modified National Institute of Standards and Technology database which is a large database of handwritten digits. These handwritten digits are in image format which is fed into the system as inputs. The inputs are used as training and testing data in the field of machine learning.

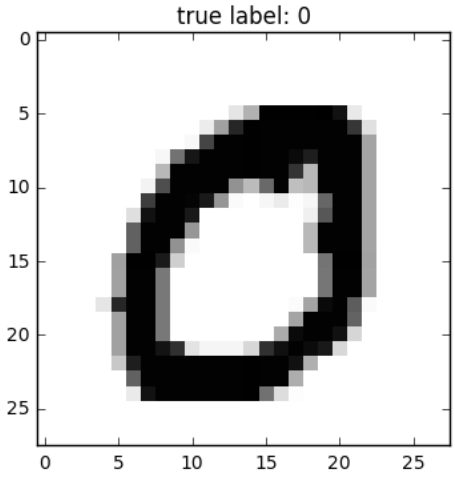

Fig 2: *MNIST dataset image 28x28 pixel*

The MNIST database contains 70,000 images which has been separated into training and testing dataset where 60,000 images are used for training models and 10,000 images are used for testing the model. The MNIST dataset contains 256 different people's handwritten digits from different parts of the world.

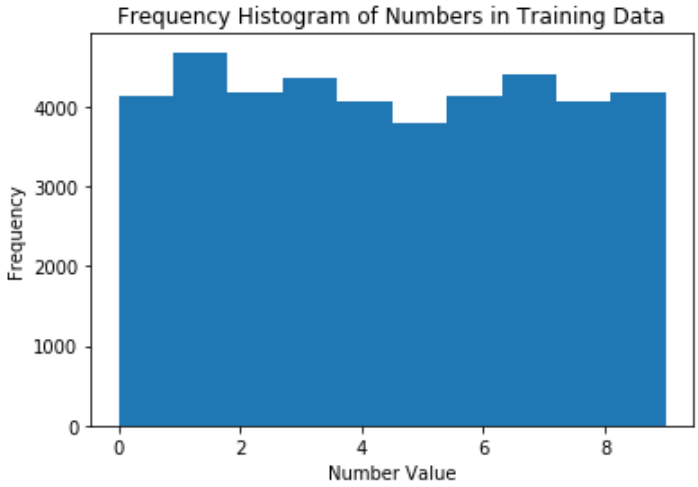

Fig 3: *Histogram of Numbers in Training Data with Frequency.*

4. PROPOSED METHODOLOGY:

In Deep Learning the word 'deep' refers to the number of hidden layers in the Neural Network. Recognition of Handwritten Characters based on Deep Learning technique i.e., Convolutional Neural Network (CNN) uses seven hidden layers. It consist of the three main phases- Convolution, Pooling and Fully-Connected Layer.

4.1 Convolution:

Convolutional layer is the main building block of the Convolutional Neural Network (CNN) model. Convolutional layer does the Filtering process to the input images which creates a feature map. The image is taken from the MNIST dataset and is sent to the first layer that is the Convolutional Layer. In Deep Learning model there is no separate preprocessing step for the input images from the dataset. The image is filtered using a convolutional filter and then is sent to the next layer for pooling.

4.2 Pooling:

Pooling layer reduces the spatial dimension of the image that enables to reduce the number of parameters which both shortens the training time and combats overfitting. There are two main pooling layers known as max-pooling and min pooling. The max-pooling takes the maximum values from the selected region and the min-pooling takes the minimum values from the selected region. Some of the main advantages of reducing the spatial dimensions are:

- The computational performance increases when the spatial information decreases.
- When the spatial information decreases simultaneously the parameters that is used to train the model gets decreased, thus it reduces the chance of overfitting.

4.3 Fully-Connected Layer:

The output layer in a Convolutional Neural Network (CNN) is the Fully-Connected layer where the inputs from the other layer is flattened and is sent to transform the output into number of classes in the Neural Network. The activation function used in this model is Rectified Linear Unit (ReLu). The activation function squashes the values into a range of value. Softmax is the last layer that predicts one of the 10 classes.

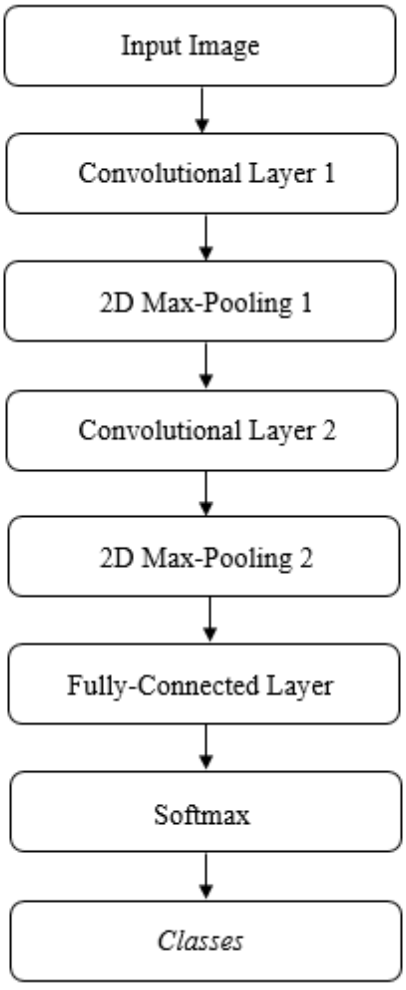

Fig 4: *The flow of CNN Classifier.*

5. Training and Testing:

Training and testing process is done and the accuracies of training and testing is displayed below with a reduced error rate of 0.03%. Out of 70,000 datasets, 60,000 datasets are used for training the model and 10,000 datasets are used for testing. For optimization, Adam optimizer is used instead of the classical stochastic gradient descent procedure.

```
After 0 training epoch(s), loss on training batch is 2.8360793590545654, accuracy on test batch is 0.8899999856948853
After 1 training epoch(s), loss on training batch is 2.1476352214813232, accuracy on test batch is 0.8600000143051147
After 2 training epoch(s), loss on training batch is 1.7951462268829346, accuracy on test batch is 0.9200000166893005
After 3 training epoch(s), loss on training batch is 1.529175043106079, accuracy on test batch is 0.949999988079071
After 4 training epoch(s), loss on training batch is 1.4293618202209473, accuracy on test batch is 0.9300000071525574
After 5 training epoch(s), loss on training batch is 1.394352912902832, accuracy on test batch is 0.9700000286102295
After 6 training epoch(s), loss on training batch is 1.2092708349227905, accuracy on test batch is 0.9800000190734863
After 7 training epoch(s), loss on training batch is 1.118780493736267, accuracy on test batch is 0.9700000286102295
After 8 training epoch(s), loss on training batch is 1.032992959022522, accuracy on test batch is 0.949999988079071
After 9 training epoch(s), loss on training batch is 0.9551498293876648, accuracy on test batch is 0.949999988079071
After 10 training epoch(s), loss on training batch is 0.8923606276512146, accuracy on test batch is 0.9599999785423279
After 11 training epoch(s), loss on training batch is 0.8201137781143188, accuracy on test batch is 0.949999988079071
After 12 training epoch(s), loss on training batch is 0.7496619820594788, accuracy on test batch is 0.949999988079071
After 13 training epoch(s), loss on training batch is 0.6898198127746582, accuracy on test batch is 0.9399999976158142
After 14 training epoch(s), loss on training batch is 0.6447305679321289, accuracy on test batch is 0.9700000286102295
After 15 training epoch(s), loss on training batch is 0.6015048623085022, accuracy on test batch is 0.949999988079071
After 16 training epoch(s), loss on training batch is 0.563193678855896, accuracy on test batch is 0.949999988079071
After 17 training epoch(s), loss on training batch is 0.5285437107086182, accuracy on test batch is 0.949999988079071
After 18 training epoch(s), loss on training batch is 0.477291077375412, accuracy on test batch is 0.9399999976158142
After 19 training epoch(s), loss on training batch is 0.4460993707180023, accuracy on test batch is 0.9599999785423279
After 20 training epoch(s), loss on training batch is 0.4262148141860962, accuracy on test batch is 0.9599999785423279
After 21 training epoch(s), loss on training batch is 0.5982959270477295, accuracy on test batch is 0.9399999976158142
After 22 training epoch(s), loss on training batch is 0.37646162509918213, accuracy on test batch is 0.9800000190734863
After 23 training epoch(s), loss on training batch is 0.34219101071357727, accuracy on test batch is 0.9700000286102295
After 24 training epoch(s), loss on training batch is 0.3099372386932373, accuracy on test batch is 0.9599999785423279
```

Fig 5: *Training accuracy and Optimization.*

6. Results and Discussion:

Convolutional Neural Network (CNN) is a very basic and fundamental deep learning technique. The accuracies of the previous algorithms are less as compared to the Convlutional Neural Network (CNN) model. The training accuracy reaches to a level of 98% and loss rate of 0.03%. The model gives a good prediction rate. The model has certain issues too, that leads to misclassification. One fact for such misclassication is that the images dataset are not centered.

7. Conclusion:

Recognition of Handwritten Characters based on Deep Learning with Tensorflow gives the most accurate classification and prediction values which can be taken for further research. The training time taken by the Convolutional Neural Network (CNN) model is very less as compared with any other model. The error rate or the rate of misclassified characters are also less compared with the previous models. Convolutional Neural Network (CNN) is not a lot different from other machine learning models but it tries to find patterns in the dataset. In future, further improvement can be made in the hidden layers to avoid misclassification and more data can be used for training.

References:

[1] Aanchal Jain and Brij Kishore Sharma, "Analysis of Activation Functions for Convolutional Neural Network based MNIST Handwritten Character Recognition", International Journal of Advanced Studies of Scientific Research (IJASSR), Vol 3 Issue 9, Jan 2019.

[2] Areej Alsaafin and Ashraf Elnagar, "A Minimal Subset of Features Using Feature Selection for Handwritten Digit Recognition", Journal of Intelligent Learning Systems and Applications (JILSA), Vol 9, 2017.

[3] D. A. Deepak Gowda, S. Harusha, M. Hruday, N. Likitha and B.G. Girish, "Implementation of Handwritten Character Recognition using Neural Network", International Journal for Innovative Research in Science and Technology (IJIRST), Vol 5 Issue 12, May 2019.

[4] Edwin Wira Putra and Iping Supriana Suwardi, "Structural Offline Handwritten Character Recogntion using Approximate Subgraph Matching and Levenshtein Distance", Elsevier, Volume 59, 2015, Pages 340-349.

[5] Fengzhi Dai, Zhongyong Ye and Xia Jin, "The Recognition and Implementation of Handwritten Character based on Deep Learning", Journal of Robotics, Networking and Artificial Life (JRNAL), Vol 6 Issue 1, June 2019.

[6] Hamayun A. Khan, "MCS HOG Features and SVM Based Handwritten Digit Recognition System", Journal of Intelligent Learning Systems and Applications (JILSA), Vol 9, 2017.

[7] Manoj Sonkusare and Narendra Sahu, "A Survey on Handwritten Character Recognition (HCR) Techniques for English Alphabets", Advances in Vision Computing: An International Journal (AVC), Vol.3 Issue No.1, March 2016.

[8] Meer Zohra and D.Rajeswara Rao. "A Comprehensive Data Analysis on Handwritten Digit Recognition using Machine Learning Approach", International Journal of Innovative Technology and Exploring Engineering (IJITEE), Vol 8 Issue 6, April 2019.

[9] Saptadeepa Kalita, Diwakar Gautam, Ashok Kumar Sahooand Rajiv Kumar, "A Combined Approach of Feature Selection and Machine Learning Technique for Handwritten Character Recognition", International Journal of Advanced Studies of Scientific Research (IJASSR), Vol 4 Issue 4, June 2019.

[10] Srivastava, Shobhit&Kalani, Sanjana& Hani, Umme& Chakraborty, Sayak. (2017). Recognition of Handwritten Digits using Machine Learning Techniques. International Journal of Engineering Research and. V6. 10.17577/IJERTV6IS050456.

[11] Yann LeCun, Leon Bottou, YoshuaBengio, and Patrick Haffner. "Gradient-based learning applied to document recognition". Proceedings of the IEEE, 86(11):2278–2324, 1998.