Handwriting Recognition using Transfer Learning

Kirandeep Singh
Computer Science and Engineering
Dr.B.R. Ambedkar National Institute of Technology
Jalandhar, India
kirandeeps.cs.20@nitj.ac.in

Harmandeep Singh
Computer Science and Engineering
Dr.B.R. Ambedkar National Institute of Technology
Jalandhar, India
harmandeeps.cs.20@nitj.ac.in

Gunwant Wankhede
Computer Science and Engineering
Dr.B.R. Ambedkar National Institute of Technology
Jalandhar, India
gunwantw.cs.20@nitj.ac.in

***Abstract*—This paper describes enhancement of hand drawn shapes recognition model to learn to recognize handwritten characters by using Transfer Learning process. We will first build a model to learn and predict hand drawn shapes and follow up to make a new model to learn the characters. This model can further be transferred to make Optical Character Recognition (OCR). Learning to predict shapes is an easy task and is not computationally expensive and transferring its knowledge to learn and predict handwritten characters will also become cheaper as there is no requirement to learn the feature extraction (which will be there as Convolutional Neural Network (CNN) will be used in models).**

Keywords*—*Handwriting recognition, Shape recognition, Alphanumeric character recognition, DL (Deep Learning), NN (Neural Network), CNN (Convolutional Neural Network), Transfer learning, TF(TensorFlow), Keras.

I. Introduction

Transfer learning plays an important role in training a model to learn some task in situations where we have a limited number of training examples or we want to train the model for a specific task in less time. It is also helpful in reducing the computational cost as previously learnt knowledge for one process can be again used for another process instead of training the whole model again from scratch even if previous knowledge holds good and is accurate enough for use. Transfer learning is being used by many giants in Deep learning and Machine Learning like DeepMind and OpenAI.

Handwriting recognition is a powerful tool which have gained interest of many researchers since a long time as it is the most needed task to transfer handwritten or printed documents to digital versions without much effort. It has gained a lot of popularity and many AI models are out there in the market to try. Handwriting or Text recognition have its application in a lot of fields – like in tourism to read signs, symbols, and text and convert them to digital text and translate them to their own regional language. It can also be used to make digital versions of ancient scripts as those documents are very long and not easily accessible to many who want to learn or make use of it. This makes it easier to even produce a translated version of these documents.

Transfer learning and Handwriting recognition are made easier with the help of TensorFlow API. Tensorflow is mainly made to develop and train Neural Networks and provide a lot of flexibility to the structure. Tensorflow and Keras come free of cost and are pre-installed in Google Colab. Google colab along with Tensorflow provides a good framework for Machine Learning and Deep Learning. Easy use of CNN(Convolutional Neural Network) is also provided by Tensorflow and Keras. CNN is a neural network which have at least one Convolutional Layer. The main purpose of the Convolutional layer is to learn and extract features (i.e. it can recognise as many features and patterns as possible) from the given image or a tensor. A tensor in a TensorFlow program is a primary data structure. Each tensor is multidimensional (i.e a tensor can have many dimensions and the number could be very large) data structure. A tensor most commonly consists of scalars, vectors, or matrices. The elements of a Tensor can hold integer, floating-point, or string values as per the requirement.

II. Motivation

Humans are capable of solving and recognizing any kind of problem but for machines to work like humans, a variety of tactics or procedures should be used. Despite all of the progress made in this field, there is still a major research gap that has to be filled.

Our main goal is to apply machine learning approaches using transfer learning to discover the principles that will be employed in automatic handwritten alphanumeric character recognition for document images. The area of study here is the recognition of handwritten alphanumeric characters that have been drawn in a distorted manner. Because there are so many different ways to write every alphanumeric character, the main goal here is to recognise handwritten alphanumeric characters.

III. References of Previous works

- Ahlawat, Savita, Amit Choudhary, Anand Nayyar, Saurabh Singh, and Byungun Yoon. 2020. "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)" Sensors 20, no. 12: 3344. https://doi.org/10.3390/s20123344

This research paper explains and tests various Neural Network Configuration by varying number of layers, Kernel size and various other hyper parameters.This article shows various configurations and accuracy of final network on testing set along with total time elapsed to train the neural network. It also describes the accuracy score for various approaches (like Convolutional neural network and deep learning) on different datasets (MNIST, The Chars74K, Urdu Nasta'liq handwritten dataset (UNHD)) with different handwritten data features like using Pixel based approach or geometrical based approach or both.

- N. Aneja and S. Aneja, "Transfer Learning using CNN for Handwritten Devanagari Character Recognition," 2019 1st International Conference on

Advances in Information Technology (ICAIT), 2019, pp. 293-296, doi: 10.1109/ICAIT47043.2019.8987286.

The above research article focuses on pre-trained models for recognising handwritten Devanagari alphabets using Deep Convolutional Neural Networks (DCNN) and transfer learning . As a fixed feature extractor, AlexNet, DenseNet, Vgg, and Inception ConvNet are used in this study. AlexNet, DenseNet 121, DenseNet 201, Vgg 11, Vgg 16, Vgg 19, and Inception V3 each have 15 epochs. Where Inception V3 outperforms AlexNet in terms of accuracy, by obtaining 99 percent accuracy with an average epoch time of 16.3 minutes, whereas AlexNet outperforms Inception V3 in terms of speed, with 2.2 minutes per epoch and 98 percent accuracy. This helps us to compare various techniques and network types for feature extraction and handwriting recognition.

- Adeline Granet, Emmanuel Morin, Harold Mouchère, Solen Quiniou, Christian Viard-Gaudin. Transfer Learning for Handwriting Recognition on Historical Documents. 7th International Conference on Pattern Recognition Applications and Methods (ICPRAM), Jan 2018, Madeira, Portugal. ⟨hal-01681126⟩

This research journal talks about the process of handwriting recognition on historical documents with the help of transfer learning.In this experiment, we illustrate how a CNN-BLSTM-CTC neural network performs when trained on a combination of datasets such as RIMES, Georges Washington, and Los Esposalles for the job of transcribing handwritten titles of Italian Comedy plays.

- J. C. Aradillas Jaramillo, J. J. Murillo-Fuentes and P. M. Olmos, "Boosting Handwriting Text Recognition in Small Databases with Transfer Learning," 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2018, pp. 429-434, doi: 10.1109/ICFHR-2018.2018.00081.

The above research paper deals with offline handwritten text recognition problems with the main motive of training the model on reduced or small training datasets. This paper mainly focuses on the use of Convolutional neural networks along with long short-term memory recurrent units (LSTM) and is trained on IAM dataset.

IV. Proposed Solution

Transfer learning with the problem of handwriting recognition is done by us in order to achieve the minimum time limit to train a model to a high accuracy. Here we first created a dataset of few shapes that have enough features and drawing strokes to train all the possible features and drawing strokes. These shapes include basic shapes like circle (○), triangle(△), square(□) and diamond (◇) and some derived shapes like 4 x 4 grid and infinity (∞) symbol. These shapes are chosen so that each and every kernel must be trained in a good way. All the possible features like all joints 'L', 'T' and '+' can be trained using a 4 x 4 grid. Few examples include as the presence of 'L' joint as in lower left corner of characters 'L' and 'D' 'L' joint is also present in upper right as in 'J' and many other characters have similar joints, We can also observe few 'T' joints as present in characters like 'T' and is present twice once above and once below as in character 'I' and on its presence on left and right in character 'H' and also plus '+' joints are present in the 4 x 4 grid which is present in character with a cut in them as present in 't'. So, the kernel will be trained by all these possible right angular joints and in all possible directions. infinity symbol is is used to train the kernel for small loops and arches which occupy approximately half of the drawn characters like a loop present in the lower part of the character as in '6', and an arc in the lower half in character as in '5', a lower loop and an upper loop as found in '8' and many other features like that. These smaller loops were not easily detectable by the kernel filters trained on a circle which can also be said as "a loop that occupies the entire drawn space of the input character".

And the previously discussed dataset is trained on a Convolutional Neural Network (or CNN). This dataset is smaller in size. This property makes it easier to get trained. Then we can transfer the knowledge learned to learn the digital characters or numeric character dataset as provided by MNIST to fulfill our given task. For this we will pop or remove the last classification layer from our model which provides classification for six distinct shapes as present in the dataset then we will add a new layer of size of 10 so that we can classify the 10 numeric digits as present in MNIST dataset and we will train the weights and biases of last layer and prevent re-training of Convolutional Neural Network (or CNN) by freezing layer as we have trained our model to predict the features ot stroke shapes that can be present in the input images. This maens our second model to be trained will be faster than as compared to the version where we do not use transfer learning for the same task. As we are training the weights and biases of one layer only that is the last layer which will classify by getting the features from previous layers.

V. Experimental Setup and Results

Various technologies and libraries were used in this project. Python was the primary programming language used throughout the project. Tensorflow is an open source software library for implementing machine learning and artificial intelligence. In Tensorflow keras is a module which helps in developing and evaluating deep learning models. The sequential module is a part of keras, it allows you to create models layer-by-layer, majorly used for deep learning algorithms such as Artificial Neural Networks (ANN) and Convolutional Neural Networks (CNN).
It is a linear stack of methods that groups a linear stack of layers into a tf.keras.Model. According to its name, its main task is to arrange the layers of the Keras in sequential order. In this model, the data flows from one layer to another layer. The flow of data is continued until the data reaches the final layer. Two data sets were required for transfer learning, one of which was provided by MNIST and the other was created

from hand drawn shapes.For importing, cleaning and preparing the datasets we have used the pandas python library. Numpy is a package used for scientific calculation and is also used in various parts of our project..Data was using train test split, train test split is a model validation procedure that allows you to simulate how a model would perform on new/unseen data .Google collab was the IDE used for coding and documenting this project. The Result of the project is that we now have a Machine learning modal that can be used for text recognition from images using transfer learning mechanism.

VI. Comparative Analysis

Because of the wide range of writing styles and letter sizes, handwritten character identification is an important issue. This project takes into account various handwriting styles from various people. A higher quality image will undoubtedly take significantly longer to calculate than a lower resolution image. Shape distortion is a typical phenomenon in practical image learning systems since various people's handwriting has varying character shapes.

Many researchers have performed research in this field using various concepts such as Deep Learning and Neural network configurations which includes CNN(Convolutional Neural Network), Fully connected Neural network, Deep neural network (DNN), deep belief network (DBN). They implemented these concepts on different datasets (MNIST, The Chars74K, Urdu Nasta'liq handwritten dataset (UNHD)). The trials were carried out using a random and conventional collection of handwritten shapes and digits . The results demonstrate that DNN is the most accurate algorithm among the three NN techniques, with the highest accuracy rate. It was found that with the use of Simple neural network, we obtained approximately 98 percent accuracy on the given dataset as per TensorFlow guide. https://www.tensorflow.org/tutorials/quickstart/beginner

The research done on Handwriting recognition using CNN(Convolutional Neural Network) achieved better accuracy. The prescribed study is unique in that it investigates all parameters of CNN architecture to provide the greatest recognition accuracy among peer researchers for MNIST digit recognition. The recognition accuracy provided in this study using a fine-tuned pure CNN model outperforms that reported by peer researchers using an ensemble architecture. The achieved accuracy was near 99 percent for MNIST dataset. In this research, six scenarios were evaluated for CNN architecture with three layers and five cases for CNN architecture with four levels. The amount of feature maps, stride sizes, padding, dilation, and received receptive fields varied between situations. The research used MNIST dataset and categories the input photos into training and test images. The pre-processing strategy was used on both the training and test datasets and data was normalized so that it falls between 0 and 1. Data preprocessing is critical in any recognition procedure. Scaling, noise reduction, centering, slanting, and skew estimation were employed to convert the input pictures into a format appropriate for segmentation. Many algorithms perform better once data has been standardized and whitened. To determine the precise settings for data pre-processing, one must experiment with several techniques. The MNIST dataset pictures were size-normalized into a fixed image of size 28x28 in the current study. Using this labeled data, a CNN model was trained and recognition accuracy was calculated.

On the contrary, in our research done on handwriting recognition with the transfer learning, we achieved an accuracy of 82 percent. Although the accuracy is a little less, it is faster in training. We created our own dataset which includes shapes with different drawing strokes. These shapes include Circle, triangle, square, diamond and some derived shapes like 4 x 4 grid and infinity symbol. These shapes cover all the strokes and this dataset is trained using CNN(Convolutional Neural Network).This trained dataset helps us in handwriting recognition.

VII. Conclusions And Future Work

Our first version of the model that was trained on shapes dataset was faster to be trained and when transferred to MNIST dataset was also faster to be trained as there were only the weights and biases of one layer to be trained. This Procedure also gave a satisfactory accuracy of 82% on MNSIT dataset.

Our first model which was trained on shapes is also capable of transferring its knowledge to EMNIST (or extended MNIST) which contains handwritten characters also or any other problem that contains similar shapes and features.

VIII. References

TensorFlow. "Save and load Keras models." *TensorFlow*, 10 January 2022, https://www.tensorflow.org/guide/keras/save_and_serialize. Accessed 5 April 2022.TensorFlow. "The Sequential model." *TensorFlow*, 10 January 2022, https://www.tensorflow.org/guide/keras/sequential_model. Accessed 5 April 2022.

TensorFlow. "TensorFlow 2 quickstart for beginners." *TensorFlow*, 26 January 2022, https://www.tensorflow.org/tutorials/quickstart/beginner. Accessed 5 April 2022.

TensorFlow. "Transfer learning and fine-tuning." *TensorFlow*, 10 January 2022, http://tensorflow.org/guide/keras/transfer_learning. Accessed 5 April 2022.

Ahlawat, Savita, Amit Choudhary, Anand Nayyar, Saurabh Singh, and Byungun Yoon. 2020. "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)" Sensors 20, no. 12: 3344.

Ahlawat, Savita, Amit Choudhary, Anand Nayyar, Saurabh Singh, and Byungun Yoon. 2020. "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)" Sensors 20, no. 12: 3344. https://doi.org/10.3390/s20123344

N. Aneja and S. Aneja, "Transfer Learning using CNN for Handwritten Devanagari Character Recognition," 2019 1st International Conference on Advances in Information Technology (ICAIT), 2019, pp. 293-296, doi: 10.1109/ICAIT47043.2019.8987286.

Adeline Granet, Emmanuel Morin, Harold Mouchère, Solen Quiniou, Christian Viard-Gaudin. Transfer Learning for Handwriting Recognition on Historical Documents. 7th International Conference on Pattern Recognition Applications and Methods (ICPRAM), Jan 2018, Madeira, Portugal. ⟨hal-01681126⟩

J. C. Aradillas Jaramillo, J. J. Murillo-Fuentes and P. M. Olmos, "Boosting Handwriting Text Recognition in Small Databases with Transfer Learning," 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), 2018, pp. 429-434, doi: 10.1109/ICFHR-2018.2018.00081.