**COMP 551: Mini Project 3 – Modified MNIST**

**Group 25: Erion Hysa, Nicolas Barreyro Lopez and Shubhika Ranjan**

13 December 2020

**Abstract: The task assigned for this project was to build a deep learning model for image analysis and classification of the Modified MNIST dataset. The task is to assign class for each image based on the recognition of handwritten digits. The image contains 1 to 5 digits ranging from 0 to 9. If the nu7mber of digits in an image is less than 5, then the remaining labels are associated with a special class called “no digit” annotated by label 10. This implies that we have 11 total classes, and each image is associated with 5 such classes. The convolutional neural networks (CNN) will automatically capture the relevant features required for recognition. The proposed algorithm reached an accuracy of 98.3% on the test set and ranked the 54th on Kaggle’s Modified MNIST competition.**

**INTRODUCTION:**

In this project, the objective is to identify the 5 classes associated with each image from a modified version of the MNIST dataset. In recent decade, deep learning techniques show an excellence on machine learning tasks and on different data types e.g. text, speech, image, and video. Deep neural networks automatically capture the feature information of the input data. In deep learning, the art is instead, architecture engineering. In the present study, we take advantage of Convolutional Neural Network (CNN) to perform the classification task

**DATASET AND SETUP**

The provided dataset for the project is a variation on the famous MNIST dataset. The MNIST dataset (Modified National Institute of Standards and Technology) is a large database of handwritten numbers used to train neural networks. The dataset at hand for this project is a collection of training\_images (56000 images), training\_labels(corresponding labels) and test\_images. Each image consist of handwritten digits, ranging from 1 to 5 digits per image. The label for each image contains 5 classes, 0 to 9 for each corresponding digit and 10 for “no-digit” class.

**PROPOSED APPROACH**

We tried the bounding box approach, where the aim is to use Computer vision techniques to create a bounding box around each digit in the image and extract the digits. The bounding box was generated by performing image thresholding and finding contours. The digits were then extracted and unskewed using interpolation techniques. The digits were cropped as separate images. The entire processing was carried out using OpenCV. We used a training model having high accuracy on MNIST dataset to predict the digits in our processed images. This was followed by manually determining the number of digits in each image, and if it is less than 5 we assigned “no-digit” class to the remaining digits in an image. CNN was used to train on this image data rather than the original MNIST datasets.

Another method explored was to directly feed in raw images into the CNN models. We presumed that this approach might be useful for this task since theoretically, CNN automatically captures the relevant features from the input data.

**RESULTS**

Several methods with different scenarios have been examined in this project until acceptable results were achieved. The most important results are presented in the following subsections.

As a common strategy in image classification, we examined the model with an appropriate classifier with an acceptable performance for the original MNIST dataset. CNN trained on original MNIST with accuracy of over 99% was used for predicting our unprocessed dataset. Prediction results were poor and could achieve only around 70% accuracy on Modified MNIST training set. Next, CNN was used to train stacked images of cropped digits from bounding box approach on original dataset. We achieved a maximum accuracy of 98.3% with this method hinting that a simple approach might work better.

In order to determine the best hyper-parameters for data augmentation, grid search was used. The parameters are also set smoothly in order to prevent loss of information (e.g. cutoff of digits). Learning from the various trials led to the accuracy rate of 98.3%.

**DISCUSSION AND CONCLUSION**

When working with data in the format of pictures, each pixel is highly dependent on the neighboring pixels. The most important conclusion of this project is the strength of CNN in handling image data. Despite modifying the original hand written image data, the CNN showed an excellent performance in learning complicated properties of the data. Due to natural feature capturing characteristic inherent in deep neural networks, it is instead needed to wisefully design an appropriate architecture and set the hyperparameters. The CNN enables different hidden units in the hidden layers to feed information to each other, i.e. now the network also knows when pixels are close together, or are sitting together in different ways that the network can learn. As a future study, we hope to use this trained model for such applications. In the future, the computation time can be reduced by fitting the dataset using a smaller architecture, while ensuring the same accuracy. Similarly, exploration of methods to speed up convergence can be explored in detail.

**Statement of Contributions**

The collaborative work of this team involved everyone contributing to the project. Shubhika and Nicolas worked on data preprocessing and analysis, various feature extraction and classifier implementation. Erion along with Nicolas worked on data visualisation and hyper parameter tuning. Shubhika and Erion collaboratively worked on the report.