1st Competition - Report

Manuela Bergau s4543645, Stergios Morakis s1047752 Shima Yousefi s1039415

March 2020

1 Introduction

Machine Learning techniques get more and more attention these days. The online platform Kaggle.com provides a platform to get hands on experience with different machine learning techniques using non trivial data in so called Kaggle competitions. For the course 'Machine Learning in Practice' we are taking part in two different Kaggle competitions.

The first competitions is called "Bengali AI - Handwritten Grapheme Classification". The competition's goal is to predict handwritten grapheme components, the smallest units of a writing system for any language. The Banglali graphemes are combinations of the following parts:

- Grapheme roots which can be vowels, consonants or conjuncts
- Vowel Diacritics
- Consonant Diacritics

We were given a set of images containing handwritten Bengali graphemes. The goal is to separately classify the three constituent elements mentioned above in each of the images. To solve this image classification task our team explored numerous topics of Machine Learning, such as the pros and cons of different architectures in Convolutional Neural Networks and various Augmentation techniques. In this report we are reflecting our research and experiment results.

2 Our approach

As all members of this team are fairly new to working with Convolutional Neural Networks, we first started by some background reading and by looking into already published notebooks. It came to our attention that the architectures that were used by the majority of Kaggle competitors taking part in the competition were either the Residual Network (ResNet), which was introduced in 2015, or the Densely Connected Neural Network (DenseNet), whose paper got published in 2017. As we are all new to the field, we decided that we would use a basic model instead of a complicated one to make sure that we could fully understand the model. As a next step, we found a simple Convolutional Neural Network with three output layers, which was used as a starting point for our code. [1]

To implement our model we made use of Kaggle notebooks. The platform grants each user 30h of GPU time per week. With three team members that proved to be enough for our needs. Nevertheless, using a free service has its downsides. When running a draft session the notebook frequently got disconnected during computation. Also computation power varies during the day, probably depending on the workload of the servers.

2.1 Preprocessing

The images in the dataset vary in size and the graphemes are not always centred. Therefore, we established a base size of 64x64 pixels for all images. Before resizing them, we applied a method provided by the cv2 package to turn the pixels whose value exceeded the threshold of 30 into 255. This step supports finding the contours. The contours in the binary images are based on the energy within each image. This way we are emphasising the handwritten grapheme components. These steps ensured that the data would be turned into valuable information for our network to process, while all the necessary information would be kept.

2.2 Augmentation

Augmentation has been one of the most interesting topics in the Kaggle Discussion. As we know, having a large enough dataset is crucial for the performance of a Convolutional Neural Network. However, we can improve the performance of the network by changing the data we already have slightly. The Kaggle Discussion introduced us to various augmentation techniques and packages. The ones we experimented with on this project were shift, scale, rotate, crop, gridmask, cutmix and mixup, most of which were provided by the python package Albumentation. The score of our model was improved with the addition of the following augmentation techniques:

- CoarseDropout
- ShiftScaleRotate
- GridDistortion

3 Best working model

In our best scoring model, we used a simple CNN with three output layers. We extended the data with augmentation. Therefore, we needed to adapt the data handling.

[1] The notebook can be found here.

In the original notebook an image generator class is used to handle the data. With this class only limited augmentation is possible. With the Keras Sequence class we are able to hand a list of albumentation augmentations to be applied to the data.

4 Results

In the first notebooks that we handed in we changed model parameters, such as padding and dropout rate, to investigate their effect. The retrieved scores varied between 0.7681 and 0.8347. As the original notebook reported a score of 0.95 we did not test further parameter changes, but turned to augmentation testing, batch size adjustment and total number of epochs.

The first public score for the notebook, which had no augmentation applied, was 0.9389. We attempted improving that score by using different augmentation techniques and eventually the score of 0.9470 was reached. This is an improvement of approximately 0.01. After finely tuning the augmentation probability parameters, we reached our best public score of 0.9498. (Final Private Score = 0.8885)

5 Author Contributions

Overall, all members of this team worked well independently and within a team setting, with minor supervision. A non-exhaustive list of the main contributions for each individual is described as follows:

- *Manuela Bergau:*
  Set up the main model and coded the Keras sequence class to simplify data augmentation. Tested various types of augmentation techniques and adjusted their hyper-parameters for a vast improvement in the model's final score.

- *Stergios Morakis:*
  Experimented with a variety of code snippets as well as hyper-parameter optimisation and merged appropriate parts of them to fit the model's needs based on past research on this field.

- *Shima Yousefi:*
  Planned, developed and tested the preprocessing steps of the algorithm. In addition, explored various notebooks and discussion threads supporting the team with undiscovered types of augmentation techniques, hyper-parameter optimisation and more.

6 Evaluation of the Process

In summary, working with Convolutional Neural Networks has been a challenging but exciting experience that introduced us to different variants of architecture types and augmentation techniques. The discussion forum in Kaggle as well as the reproducibility of many papers and blogs gave us a respectable place in the competition that proudly reflecting our efforts.

7 Appendix

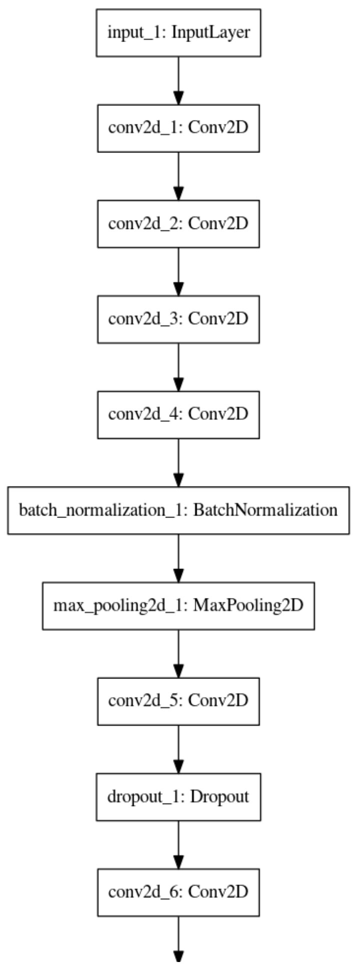

Figure 1: The structure of our model part 1

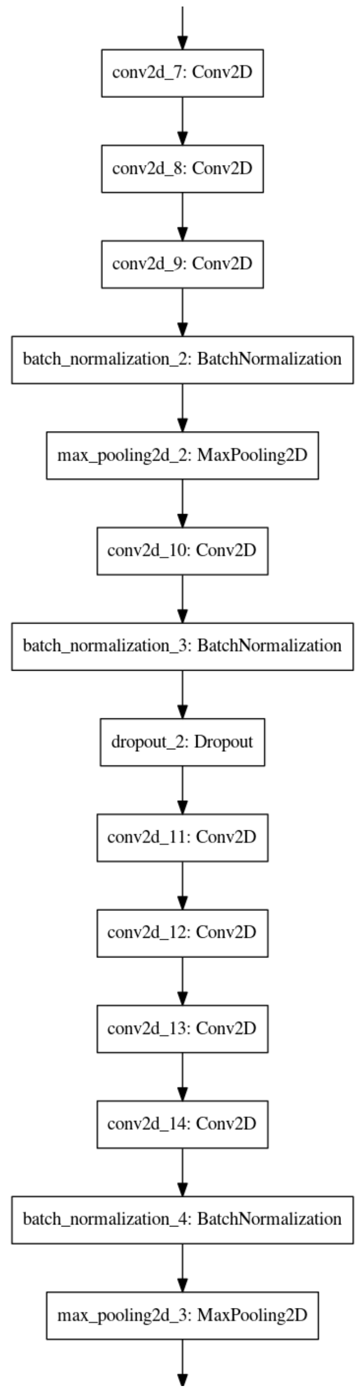

Figure 2: The structure of our model continued part 2

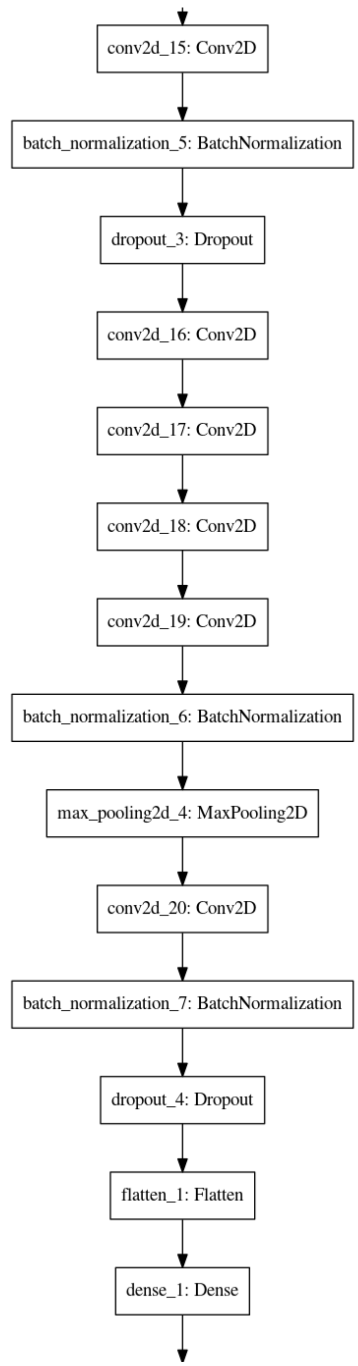

Figure 3: The structure of our model continued part 3

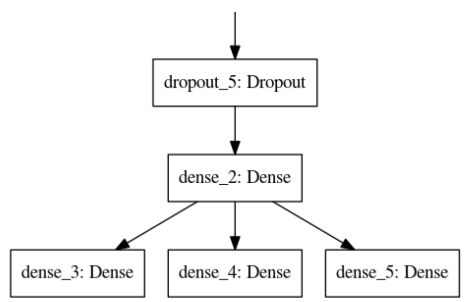

Figure 4: The structure of our model continued part 4