DEEP LEARNING SKIN LESION CLASSIFICATION

GROUP 35: BARTO RADMAN (2062836), DANIEL KOLTAI (2123303), MARIO VELLA (2085712), MARTA ANGELO (2126580)

TILBURG UNIVERSITY

STUDENT NUMBER

Barto Radman: 2062836

COMMITTEE

Dr. Gorkem Saygili

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

October 6, 2023

ACKNOWLEDGMENT

i want to thank my mom

Abstract

Computer vision is one of the most researched areas in deep learning. Milestone papers have proposed the use of CNNs for classification tasks performed on image data. This paper summarizes our attempt at skin lesion classification data. It goes through the necessary libraries, train/test splitting, normalization, and comparison of three CNN models of various complexity. Each of the design choices and the performance are then reported along with the accompanying visualization plots.

1 PREPROCESSING AND EXPLORATORY DATA ANALYSIS

First the necessary libraries for processing and visualising data, along with libraries for building the models, were imported. These include: pandas, numpy, matplotlib, seaborn, tensorflow and keras. The provided image data (10015 skin lesion images along with their respected labels) were first loaded for train/test splitting and normalization. The data was split into 60 percent training, 20 percent validation and 20 percent test data. The data was then standardized to float values between 0 and 1 by dividing by 255 (pixel values). Fifteen samples were then randomly selected and for each sample the class label was displayed (Figure 1). Last exploratory data analysis step was to create a plot to visualize the class (in)balance. The plot displayed large disparity in the sample, namely 67 percent of data belonged to the "Menalocytic nevi" class and 1 percent of data belonging to "Dermatofibroma" class. The other classes were also largely underrepresented (Figure 2) providing a first obstacle for generalizable classification.

2 BASELINE MODEL

After the data was preprocessed and visualized, a baseline CNN algorithm is implemented. The network consisted of two consecutive convolutional layers (size 64 and 32 with 3x3 filter size). Convoluitional layers used ReLU activation function followed by a max pooling layer of 2x2 size. ReLU activation is a standard non-linear function used in many different neural networks due to the ease of training and good performance. The function outputs the input directly if its is a positive value. In the case of a negative value input the function outputs a 0 (Brownlee, 2020). After the convolution layers, two dense layers of 32 units with ReLU activation were utilized. The output layer had 7 units (one for each class) and used the softmax activation function. Softmax is an ideal activation function for a multi-classification task as it outputs probabilities to each output neuron (Khan, 2023). The model was compiled with Adam optimizer, categorical cross entropy for loss and accuracy as an evaluation metric. Categorical cross entropy was chosen due to the multi-class nature of the task. The model was then fitted to the previously described data, with 10 training epochs and the batch size of 32. (Figure 3) Displays the training and validation accuracies (and losses). From the graph we can observe that the model is overfitting around the fourth epoch of training, indicating that the model is too complex and not generalizable. (Figure 4) Displays the ROC curve and the AUC score. The AUC score measures the ability of the model to distinguish between the classes. For the baseline model the AUC scores were all above 0.81, indicating that the model can detect true classes more often. Furthermore, the confusion matrix is shown. Unsurprisingly, due to the heavy under representation of the classes, the model predicts the majority class most often. Last, other performance measures should be discussed. While the accuracy on the validation and testing data is adequate (0.75), the precision, recall and f1 scores leave much to be desired for all of the minority classes. The algorithm does not learn the minority classes, likely due to extremely low representation. The baseline model informs on and confirms one of the largest weaknesses, the big disparity in data. The further examined models attempt to correct these weaknesses. The rest of the metrics can be seen in the source code of the baseline model.

3 ENHANCED MODEL

The baseline model has made it clear that a more balanced dataset is necessary in order to make the model generalize to all of the classes. Thus, before enhancing the baseline model it was necessary to augment the minority classes. The majority class was excluded from the data and each minority class was augmented by rotating, shifting flipping, adding noise through brightness and nearest pixels. We chose to augment the data instead of just copying minority classes or artificially sampling. It was hoped that augmented data would provide the model with more quality data (with new information) along with noise which could allow the model to generalize better. New class distribution can be seen in (Figure 5). The bar graph still displays unbalance in the data, however the model will receive many more training examples and will generalize better then in the baseline model. Other fine tuning decisions were made mostly with trial and error. Due to the long training times (up to an hour) only the changes to hyper parameters which were thought to influence the model the most were made. First the baseline model was trained once with the new augmented data. The model proved slightly better performance and generalizability. In the end, the enhanced model added dropout layers after each convolution layer to prevent overfitting along with making the model faster to train. L2 regularizer with lambda = 0.001 was included for each hidden layer in order to keep the weights of the model small and further prevent overfitting. Lastly, the convolution layers used LeakyReLU activation (alpha = 0.3) function to address possible negative weights. The model also used 'He' initialization which is suited for activation functions which are not symmetric around 0 (ReLU). The initializer sets weights around M = 0 and SD = sqrt(2/n). Number of epochs and additional layers were not included in the model due to the computational limitations. Once these hyperparameters were chosen, the model was evaluated on the same metrics as the baseline model. Training and Validation loss and accuracy displayed that the model was not overfitting, conveying that the model's complexity captures the data information more adequately than the baselines model (Figure 6). ROC AUC score is shown in (Figure 7). AUC scores vary high between the classes. The majority classes to which the model has been trained the most are predicted correctly above 0.81 percent of time showing much better generalizability. However, the three minority classes have around 0.6 correct predictions, making the model still limited. Confusion matrix has also displayed improvement in the enhanced model, with more classes getting predicted (correctly) overall. Other necessary metrices can be found in the source code. Precision, recall and f1 scores have all improved from the baseline model, proving that the model is better suited for the classification of the skin lesions. These improvements in the model performance can be attributed to better data. The goodness of data representation has great impact on machine learning algorithms, especially deep learning (Karthi et al., 2021). Thus our model has provided greater information to the model through image augmentation. Rest of the parameters were included to both increase the generalizability and decrease complexity of the model, while regularizing weights and allowing negative weights through LeakyReLU. A better approach to hyperparameter tuning would be a limited grid search, which could also take into account and test the different amount of convolution layers, number of epochs along with the tuning of hyperparameters that were made in this paper. However, due to limited training time and computational limits, the hyperparameters had to be chosen based on educated guesses and inspections of the performance after fitting. We believe that with more resources, the network can be fine tuned to perform even better on the dataset. The enhanced model remains as a valuable tool for evaluating skin lesions, especially of the majority class. Another possibility would be to use the current state of the art vision transformer networks which employ attention units. While these networks are usually used for natural language processing, they could be refined and used along side CNNs (Shen, 2020).

4 TRANSFER LEARNING

The last model to be contrasted with the baseline and enhanced model is the model built using transfer learning. Transfer learning is a term describing the technique of taking a pre-trained model (pre-trained weights) in which the dense layers are frozen. New dense layer is then inserted and the model is evaluated on our baseline data (without the augmentations). Transfer learning is often applied when the dataset is small (not allowing the new model to learn suitable image representations) or unbalanced, such as the case in this paper. The model chosen for the task of skin lesion classification is ResNet50, a deep learning model trained on ImageNet data with 48 convolution layers and two pooling layers (He et al., 2016). Once the last ResNet50 layers were removed, we flattened the model and added a dense layer (size 64) with relu activation function and an ourput layer (size 7) to classify the outputs. The model was then compiled using same hyperparameters as the baseline model and trained for 10 epochs. The transfer learning model displayed simillar performance to our enhanced model, despite the limited data that was provided (0.68 training and validation accuracy). ROC AUC scores conveyed good predictive capabilities for the majority classes and struggled only with one class (class 6). However, the confusion matrix showed that the model had a similar limitation as the baselines model, namely it was overly predicting the over represented majority class. Other metrics can be seen in more detail in the source code. It remains interesting to consider feeding more balanced, augmented data to the transfer model. It could be predicted that in that case the model could outperform the enhanced model on all of the metrics. For now it outperforms the baseline model due to more fine tuned weights however falls short due to the unbalanced data.

5 CONCLUSION

The paper discussed three potential CNN models, the baseline model, enhcnace model and transfer learning (ResNet50). The greatest obstacle for the models seems to be the unbalanced data. Baseline model displayed overfitting along with heavy bias towards the over represented majority class. The enhanced model improved on the architecture while including a more balanced data set leading to less overfitting and better generlaization power. Lastly, the transfer learning model showed clear improvement through pre-trained weights and could be further imrpoved with more high quality data. The paper presented possible future implications along with possible changes in the architecture of the enhanced model.

REFERENCES

Brownlee, J. (2020, August 20). A gentle introduction to the rectified linear unit (ReLU). MachineLearningMastery.com. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/

He, K., Zhang, X., Ren, S., amp; Sun, J. (2016). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr.2016.90

Karthi, S., Kalaiyarasi, M., Latha, P., Parthiban, M., amp; Anbumani, P. (2021). Emerging applications of Deep Learning. Integrating Deep Learning Algorithms to Overcome Challenges in Big Data Analytics, 57–72. https://doi.org/10.1201/9781003038450-4

Khan, M. A. I. (2023, April 7). Introduction to Softmax classifier in pytorch. MachineLearningMastery.com. https://machinelearningmastery.com/introduction-to-softmax-classifier-in-pytorch/

Shen, D. (2020). Enhance Image Classification Performance via Unsupervised Pre-Trained Transformers Language Models. https://doi.org/10.21203/rs.3.rs-93060/v1

6 APPENDIX: FIGURES

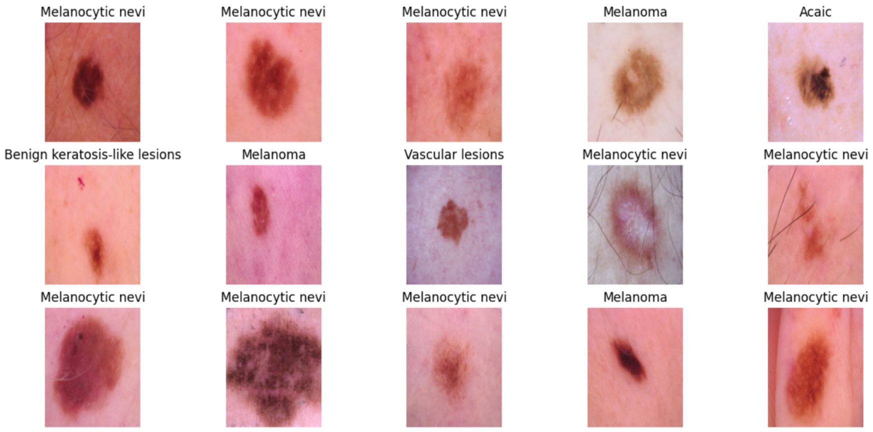


Figure 1: Sample display

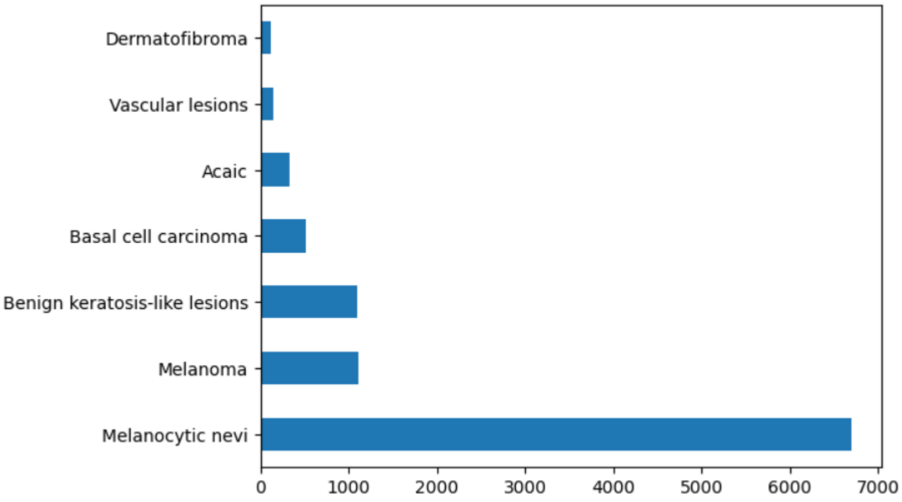
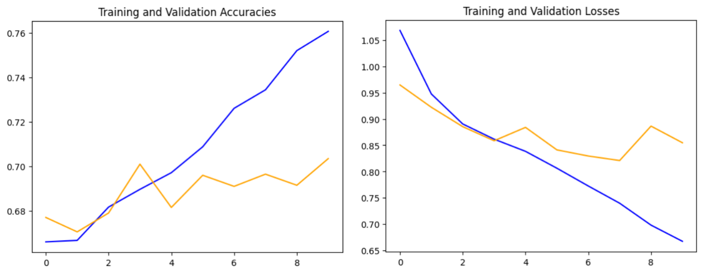
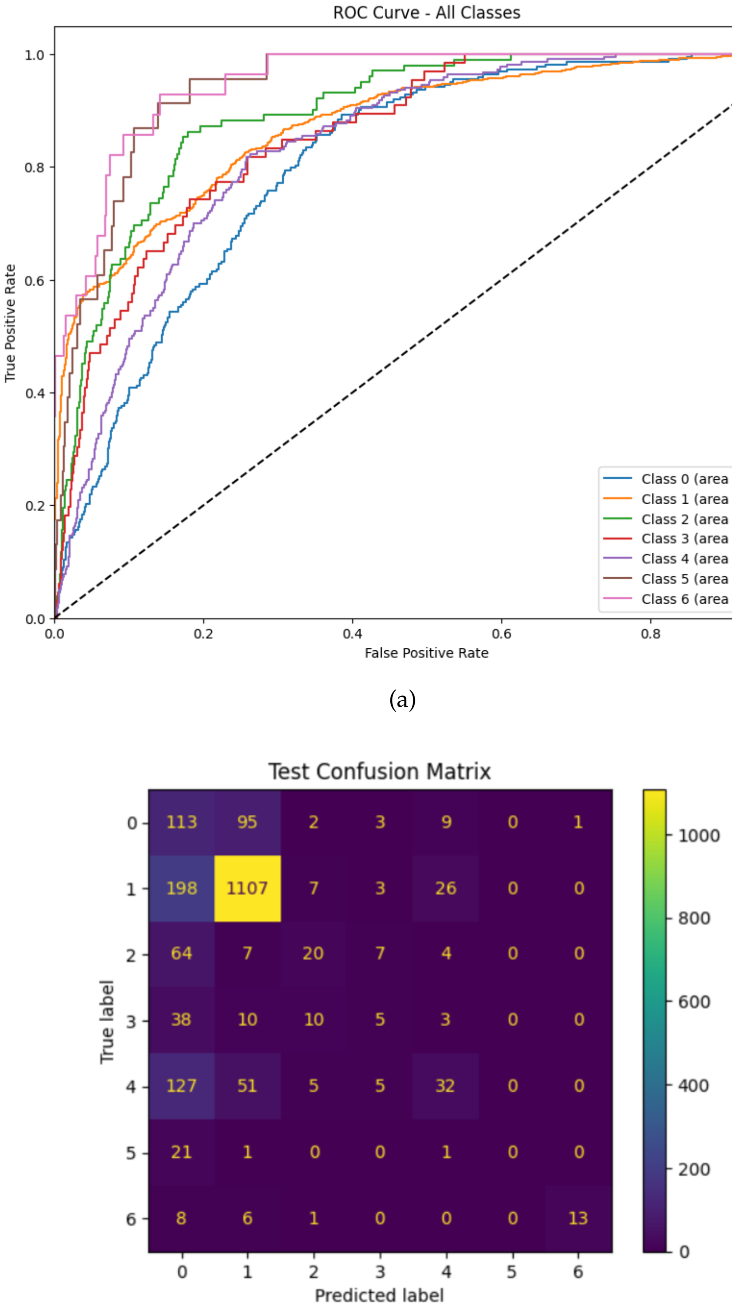


Figure 2: Sample display



(a) Baseline accuracy (b) Baseline loss

Figure 3



(a)

(b)

Figure 4: ROC AUC and Confusion Matrix - Baseline Model

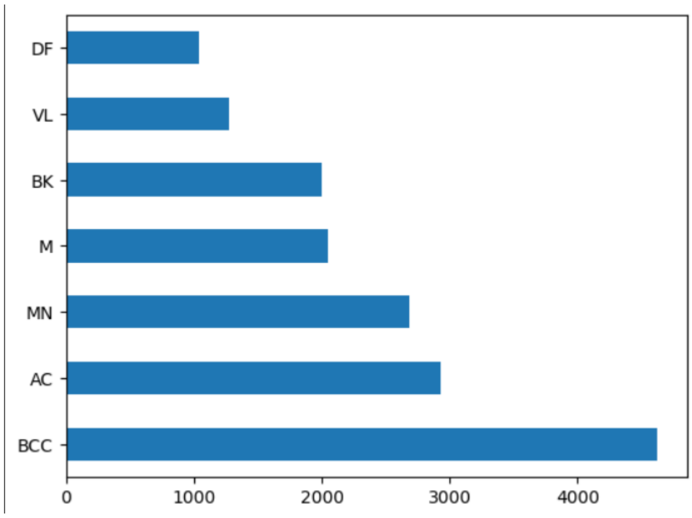
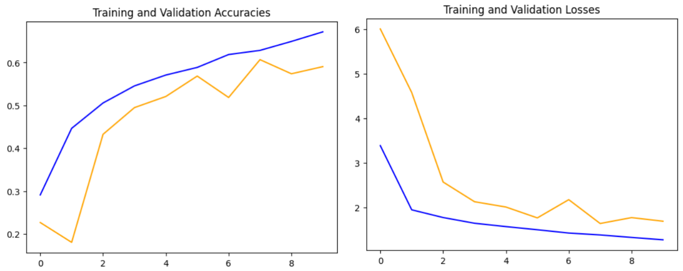


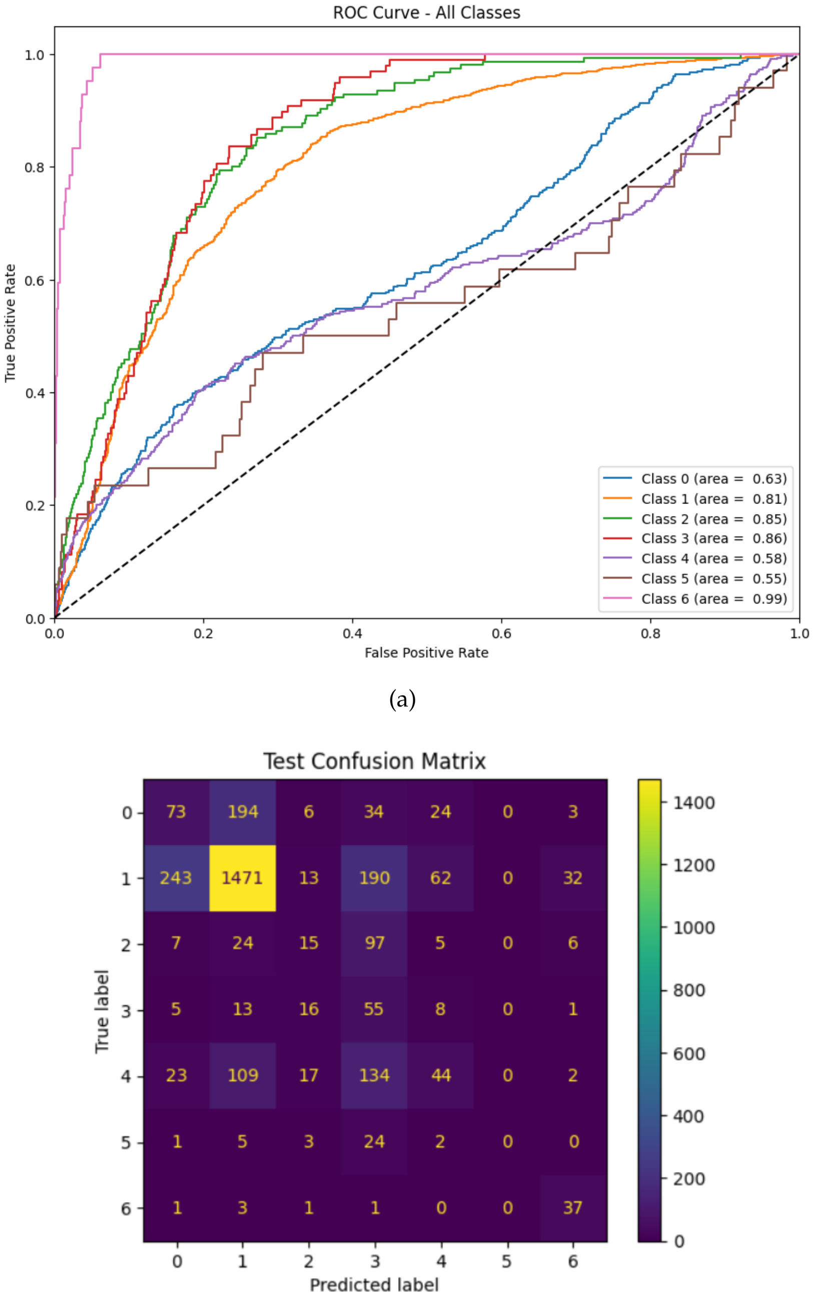
Figure 5: Samples with augmented images



(a) Enhanced accuracy

(b) Enhanced loss

Figure 6



(a)

(b)

Figure 7: ROC AUC and Confusion Matrix - Enghanced Model