Skin Cancer Classifier using Dermatoscopic Images of Pigmented Lesions - Progress Report -

Brief Project Description

Skin cancer is the 19th most common form of cancer for humans [1], and the most prevalent in the United States where it is estimated that 1 in every 5 Americans will suffer from skin cancer at some point in their life [2]. As such, the prevention and early detection of skin cancer is crucial. Detecting skin cancer involves carefully analyzing one's moles, skin lesions and overall skin tone. Dermatologists are highly trained in this skill, but the diagnosis process can be lengthy and costly for a patient due to the need for an expert supervision. Our project goal is to create an ML model that is able to classify an image of a skin lesion as being malignant or benign. The use of supervised machine learning in this field would be appropriate since a classifier trained with labeled images of different skin cancer lesions would greatly aid in the early detection. Our final design would help considerably people with non-medical knowledge to easily understand the possibility of them having skin cancer by simply providing an image of the suspected lesion before going through the costly and potentially lengthy process of being diagnosed by a dermatologist.

Individual Contributions and Responsibilities

1. *Working Together*

Our team is closely working together using Zoom Video Calls when important team decisions are needed and for the crucial parts of the project (i.e. data cleaning, model training and testing). Tasks are split evenly and carried out individually but final revision will always be done together before each submission. We are also using a Messenger chat group to communicate individual progress and any question or doubt we have to the rest of the team. At the end of each meeting we always choose a date for the next meeting in order to keep internal deadlines clear and commitment high; on average we have been meeting once a week after classes. To make sure individual work runs smoothly and no one overwrites each other's text and/or code we use Google Docs and Google Colab so that we can easily add comments to different sections and keep track of who did what and last edits.

2. *Accomplishments and Next Tasks*

Until now, the team has completed the data loading and splitting, the development of a baseline model of several deep CNN models such as AlexNet and Resnet. These models have been individually trained to achieve the highest possible degree of accuracy. We are currently working on testing how these models perform over the test dataset to select the one with the highest final test accuracy as our best model. So far the team has been on schedule and we can predict with a fairly high degree of confidence that we will be able to complete the project in time and have some extra time for finishing touches before the final presentation.

Table 1: Project management planning

Team Member	Task	Internal Deadline	Completion
Valentina	Draft for Ethical Considerations and Risk Register	02/09/2021	Done
Ines	Draft for Introduction and Data processing	02/09/2021	Done
Meha	Draft for Background and Related Work	02/09/2021	Done
Samarth	Joined late due to partners dropping out. Minor draft editing.	02/11/2021	Done
Team	Final Revision of Project Proposal	02/12/2021	Done
Ines, Meha, Valentina	Data Loading	02/26/2021	Done
Ines	Develop baseline model	03/05/2021	Done
Meha	Develop an train Deep CNN - AlexNet	03/19/2021	Done
Valentina	Develop an train Deep CNN - Resnet 18	03/19/2021	Done
Samarth	Develop and train Deep CNN - Resnet 50 & 101	03/19/2021	Done
Ines, Meha, Valentina, Samarth	Complete Progress Report	03/22/2021	Done
Team	Test Deep CNN & Compare with Baseline	03/26/2021	In progress
Team	Complete Final Report and Presentation Slides	04/03/2021	-
Team	Recording of Final presentation	04/04/2021	-
Team	Final presentation Q&A	04/06/2021-04/08/2021	-

Notable Contribution

The first step towards the creation of our skin cancer classifier has been the data loading and splitting. This process has been initially problematic because the database we originally chose in the project proposal was too big to be used on Google Colab. We therefore selected a new dataset from Kaggle [3] from the same source as the original one smaller in size. We therefore proceeded in splitting the 3,297 images into training, validation and testing datasets with a proportion of 70:20:10.

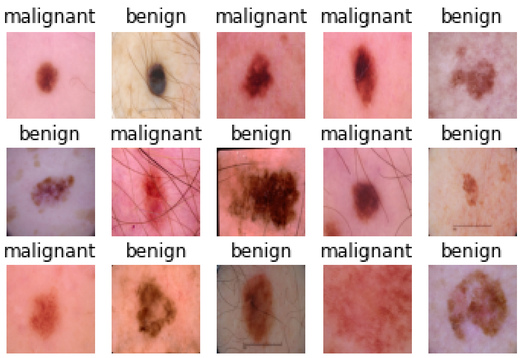
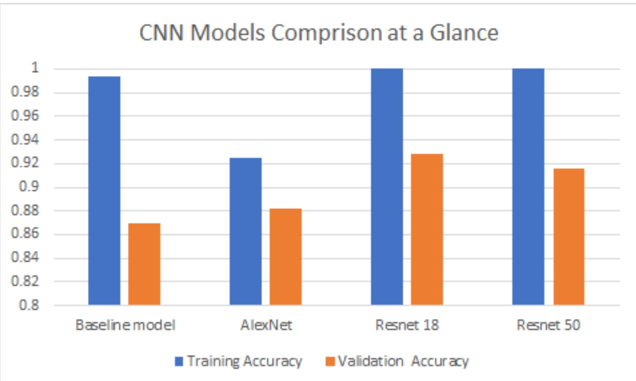


Figure 1: Sample of labelled data

My main contribution however has been on the development and training of a Resnet 18 model. It makes sense to use this model for our project goal because its complex structure allows us to achieve high accuracy in image recognition and classification: it is made of 5 convolutional layers, followed by an average pool and one fully connected layer, using softmax activation function. For training I have proceeded as follows:

1. I have taken the Resnet 18 model from the pytorch library and without any pre-training I have trained it with batch_size=128, epochs=50 and learning_rate=0.001. I was able to achieve a validation accuracy of 85%. [Figure 2]
2. Then, I have run a pre-trained Resnet 18 model with the same hyperparameters in order to confirm that it was actually able to achieve higher accuracy. Indeed, I got validation accuracy of 88%. [Figure 3]
3. I have therefore decided to proceed with tuning the hyperparameters on the pre-trained model. From the graphs produced so far I could see that 50 epochs were unnecessary so I decided to reduce them to 30 and decrease the learning rate to 0.0001. Validation accuracy was improved to 92.0%. [Figure 4]
4. With this last set of hyperparameters however, it was very clear that the model was overfitting the training data; by trying many different other combinations to avoid that I was only able to slightly reduce the overfitting by halving the number of epochs and decreasing the learning rate to 0.00003. Validation accuracy is now lowered to 89.7%. [Figure 5]

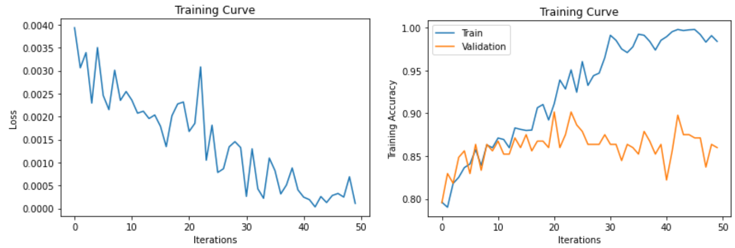


Overfit is definitely the biggest problem I encountered and still partially remains. Comparing my results with other models currently being developed by my team member [Figure 6] Resnet 18 remains for now a good option to keep into consideration as primary model but further comparisons with the testing dataset needs to be performed before making the final decision.

Figure 6: Comparison of different models

Figure 2: first implementation of Resnet 18

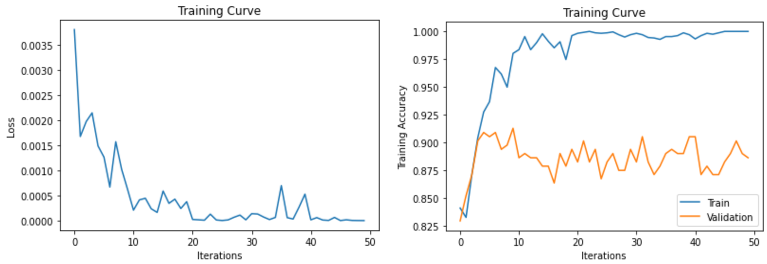
batch_size=128, num_epochs=50,learning_rate=0.001




```
Final Training Accuracy: 0.9839865149599663
Final Validation Accuracy: 0.8598484848484849
```

Figure 3: second implementation of Resnet 18 (with Pretrained Model)

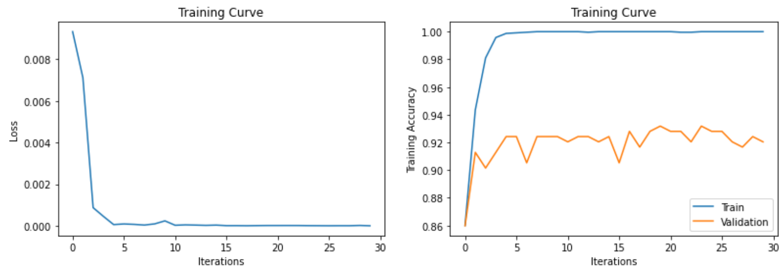
batch_size=128, num_epochs=50,learning_rate=0.001




```
Final Training Accuracy: 1.0
Final Validation Accuracy: 0.8863636363636364
```

Figure 4: third implementation of Resnet 18 (with pretrained model)

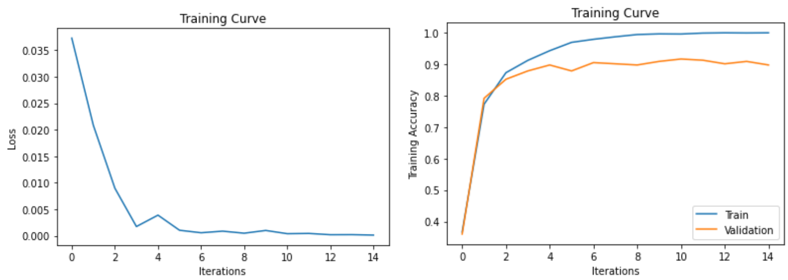
batch_size=128, num_epochs=30,learning_rate=0.0001




```
Final Training Accuracy: 1.0
Final Validation Accuracy: 0.9204545454545454
```

Figure 4: fourth implementation of Resnet 18 (with pretrained model)

batch_size=128, num_epochs=15, learning_rate=0.00003




```
Final Training Accuracy: 1.0
Final Validation Accuracy: 0.8977272727272727
```

Link to Google Colab Notebook

https://drive.google.com/file/d/1OiL1nujlLVtbSYQeQf724qv1HzVIX8GW/view?usp=sharing

References

[1] "Skin cancer statistics", World Cancer Research Fund International. 2018 [Online]. Available: https://www.wcrf.org/dietandcancer/cancer-trends/skin-cancer-statistics

[2] "Skin Cancer Facts & Statistics", The Skin Cancer Foundation, The Skin Cancer Foundation. 2017 [Online] Available: https://www.skincancer.org/skin-cancer-information/skin-cancer-facts

[3] C. Fanconi, “Skin Cancer: Malignant vs. Benign,” Kaggle, 19-Jun-2019. [Online]. Available: https://www.kaggle.com/fanconic/skin-cancer-malignant-vs-benign