Classification of melanomas using a convolutional neural network

Renaud Dekeyser, *Student, EPB*

Abstract—Melanomas is the deadliest skin cancer and early diagnosis is a key for an effective treatment. Dermoscopic images can be used for a first a diagnosis. Therefore, a deep learning model was developed for the classification of skin lesions as part of a challenge from SIIM-ISIC. First of all, the data were preprocessed to make a balanced dataset and to make data augmentation. Then the model used, a convolutionnal neural newtork, is discussed. Finally, the model was trained over 25 epochs and obtained an accuracy of more than 0.73. The results obtained were compared to the ones obtained using transfer learning with a pretrained base.

Index Terms—Melanoma classification, deep learning, transfer learning, SIIM-ISIC 2020 challenge.

1 Introduction

Skin cancer is one of the most common cancer in the world according to the WHO [1]. In the different types of skin cancers, melanoma has the higher mortality rate [2] and is the mains cause of death due to skin cancers, even if it represents only 1-3% of the skin cancers [3]. An early diagnosis, like for all cancers, is essential for an effective treatment but the diagnosis with the eyes are limited. That's why, computer aided diagnosis are developed [3].

Currently, one of the ways to diagnose melanoma is to use dermoscopy [4]. The aim of this project is to develop a deep learning model that is trained on dermoscopic images in order to be able to identify malignant melanomas among skin lesions. Thus, this is a binary classification problem where skin lesions have to be classify as malignant or benign. Therefore, this project was divided into three mains part : the data prepossessing, the model and a comparison with pre-trained models from Google and Oxford. This project is based on the data given by the SIIM-ISIC Melanoma Classification Challenge 2020. A link to it can be found in the appendix A and the code was done using the TensorFlow API [5] in Python.

2 The data preprocessing

The dataset contains 33,126 images of skin lesions but in it, there are only 584 images of malignant melanomas. This make this an imbalanced dataset. This is a problem that has to be handled. The data also provide information about the patient like the gender, the position of lesions and the age. These data could be used to improve the quality of the model but weren't used here. This project was more focused on the computer vision problem. Finally, its important to mention that images were of varying shapes. That's why, those pictures were reshaped to 160x160 in order to have more standardized data and because the model need to have an unique input shape. The shape was chosen due to memory limitation. However tests on different sizes were conducted and didn't result in significant changes in the model accuracy. Moreover, smaller sizes were used in the literature and obtained decent results [6].

2.1 Imbalanced dataset

This dataset is imbalanced because the malignant cases represents less than 2% of all cases. The problem is if the model is trained on this dataset, only the majority class is going to be predicted. Indeed, this is going to lead to a 98% global accuracy which is an overall good result. However the model can't be used because it's overfitting and if it was tested on a balanced testing dataset the accuracy would seriously drop.

The problem of classes imbalance can be handled in different ways. The majority class could be downsampled [7], the minority could be oversampled [8] [6] or class weights could be set [5]. The downsampling of the majority class was chosen for several reasons. Firstly, because the data are too much imbalanced thus class weights couldn't be set. Secondly, the downsampling decreases the number of images and therefore the training time is also decreasing. This makes the empirical phase of defining the model, much faster. Thirdly, the dataset obtained using downsampling was big enough according to some examples of dataset found in the literature [2] (935 images for training and 233 for validation). However only using downsampling for data is not the best solutions. A hybrid solution, using class weight and over/downsampling, are more commonly used [9]. This may be one improvement for this project.

2.2 Data Augmentation

In most of the dataset, the data augmentation has to be used due to the limitation of samples per class and because the performance of models often improves with more data provided [10]. There are different strategies to increase the variety of data but here the focus is going to be set on techniques for images. Data augmentation on images often consists in random transformation on image like rotation,

- *M. Dekeyser was with the Laboratory of Image Synthesis and Analysis, École polytechnique de Bruxelles, Brussels.*
E-mail: Renaud.Dekeyser@ulb.be

Manuscript sent may 17, 2022

cropping, contrast shift, etc. In this project, four transformations were used :
- contrast shift,
- flip along vertical axis,
- flip along horizontal axis,
- random rotation.

The data are going to pass trough this four randoms transformations during every epochs before being used. This data augmentation prevents the model to overfit. It was the more significant improvement during the empirical phase because it allows the model to be trained longer with less overfitting.

3 Architecture of the model

The model is built around two main blocks : a convolutional base and a dense head in order to form a convolutional neural network (CNN). A CNN is a deep learning algorithm which have for general purpose to analyse image [11]. Therefore the descriptions of the different blocks that compose the model will be explain separately. The architecture of the model is mainly inspired by tutorial from Tensorflow [5] and inspired by these books [12] [10].

3.1 Convolutional Base

The aim of the convolutional base is to learn to extract features from the pictures and these features will be analysed by the dense head. The table 1 presents a summary of this block.

TABLE 1
Convolutional base

Conv2D(64,3x3, activation='relu')
BatchNormalization()
MaxPooling2D()
Conv2D(128,3x3, activation='relu')
BatchNormalization()
MaxPooling2D()
Conv2D(256,3x3, activation='relu')
BatchNormalization()
MaxPooling2D()

The conv2D() layers are used in order to convolute image with 3x3 kernels. Kernels are sometimes called features filters. For example, the Sobel operator is a kernel used to make edge detection. Thus, conv2D() layers extract features and form a feature map, also called the activation map. Then the MaxPooling2D() layers downsample this map using the local maximum value. This operation is done in order to extract more general features from the pictures. Moreover, over relatively small distances, the operation of max-pooling is translation invariant. This is important because all the images doesn't have the same perspective. This operation also strengthens the model against noises contains in the data. Furthermore, the activation function chosen is the rectifier function as it's generally done [10].

Finally, the last layers used here are batchNormalization(). The aim of the layers is to keep the data normalise between the hidden layers of a deep-learning model. The normalisation is done across every mini-batch. This has for effect to prevent the model from overfitting, increase the learning rates [11].

3.2 Dense Head

As mentioned before the goal of this head is to use the feature extracted in order to classify the images. The table 2 presents a summary of this block.

TABLE 2
Dense Head

Flatten(),
layers.Dense(128, activation='relu')
BatchNormalization()
Dropout(0.2)
Dense(64, activation='relu')
BatchNormalization()
Dropout(0.2)
Dense(32, activation='relu')
BatchNormalization()
Dropout(0.2)
Dense(1,activation='sigmoid')

So, the flatten layer is used to make the connection between the head and the base. The dense layers contains the neurons of model (respectively 128, 64 and 32 neurons). The last dense layers contains only one neuron because it uses a sigmoid as activation function. However, a softmax function could be used too. Indeed, if there were more than two classes in the problem then this would have been mandatory. The activation with sigmoid means that the model will give a label, a float, between 0 and 1 to every pictures. In order to make the classification, a threshold of 0.5 will be set. The pictures with a label above 0.5 will be part of 1 class and the others will be part of the second class.

Furthermore, in this block dropout layers were added. This layers randomly dropout the output of some layers [11]. This techniques is used to fight against statistical noises and prevent the model to overfit [10].

Finally the model was compiled using Adam for optimizer. It's a stochastic gradient descent method that is commonly used. The loss function is cross-entropy and more specifically the binary cross-entropy. And the only metrics followed were the loss and the binary accuracy. The second one was used because the classes have the same frequency and because it's a binary classification problem. Sensitivity and specificity could also have been followed.

4 Training

The first training was done over 100 epochs. The result can be seen on figure 1.

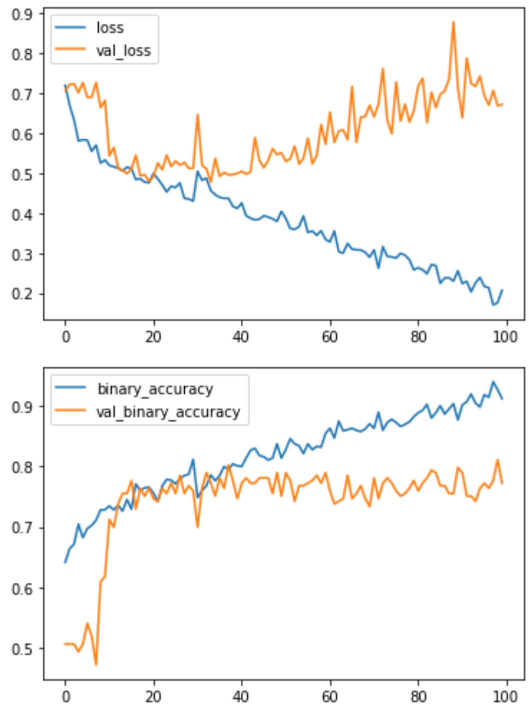

Fig. 1. Training over 100 epochs

After 25 epochs on the first graph, some overfitting begin to appear despite the fact that dropout and batch normalization layers were added. Therefore model that is going to be used will be trained on only 25 epochs. The result on only 25 epochs can be seen on figure 2.

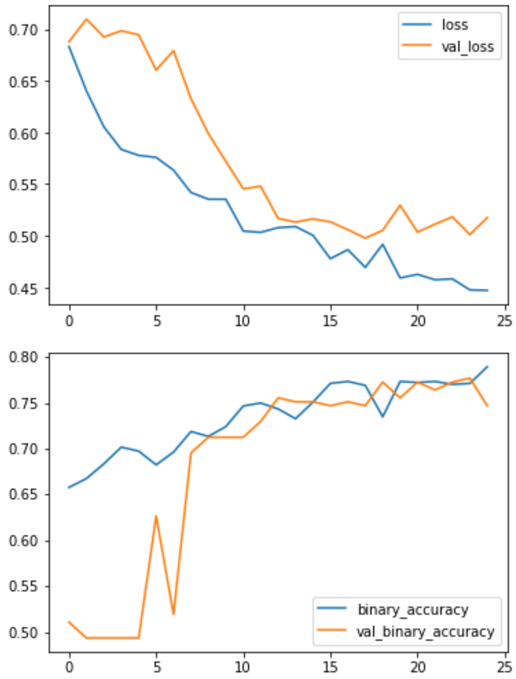

Fig. 2. Training over 25 epochs

5 Results

For analysing the results, i.e. the predictions made by the model using the validation dataset, a confusion matrix was plotted. The label 0 correspond to the benign skin lesions and 1 to the malignant.

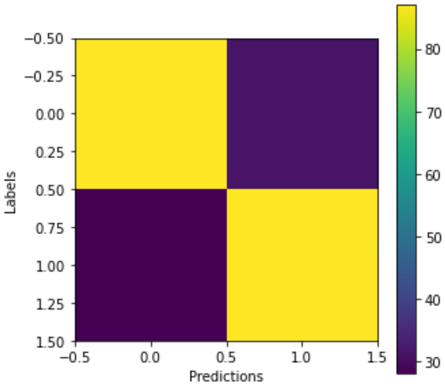

Fig. 3. Confusion matrix of the model

The confusion matrix highlights the fact that the model don't predict more one class than the other one which is a good thing due to the balanced dataset. However, it also highlights the small accuracy of 0.7468 with about 60 false predictions over 233 images. This accuracy is confirmed by the graph obtain during the training phase. It's quite low according to the literature [2] where the accuracy obtained reach over 0.95 with comparable dataset and even smaller. It shows that there still a lot room of improvement for this project. However, improvement were hard to gain because the lack of reproducibility during the empirical development of the model. Indeed, training with the same model and with the number of epochs doesn't provide always the same results, the accuracy was oscillating between 0.73 to 0.77 during testing on the kept model. This could be explained by the fact that the benign images are chosen and then split into test and validation randomly every time the code is launched. Therefore, data aren't always exactly the same. So, if this is true, it highlights the importance of data on the model.

5.1 Comparison with other models using transfer learning

To explore a bit more the results, the model defined above is compare with two well known model for classification: MobileNet from Google and VGG16 from Oxford. The models were pretrained and the base weren't retrained. Only the head was retrained on 25 epochs in order to be able to make predictions and no fine-tuning was done. Thus, there were much less weight to train. Also, the same dataset were used with every model. The confusion matrices can be seen on figures 4 and figure 5.

It's not mention on the confusion matrices but VGG16 and MobileNat, both have an accuracy of 0.73 after 25 epochs. What emerges from the confusion matrices is that both are predicting one class more than the other but the class differ for the two models. Thus for VGG16, on one hand there are more true benign cases but on the other hand there are more false benign cases and vice-versa for MobileNet. That's the major different with the model define above. However, it turns out that the accuracy is very similar

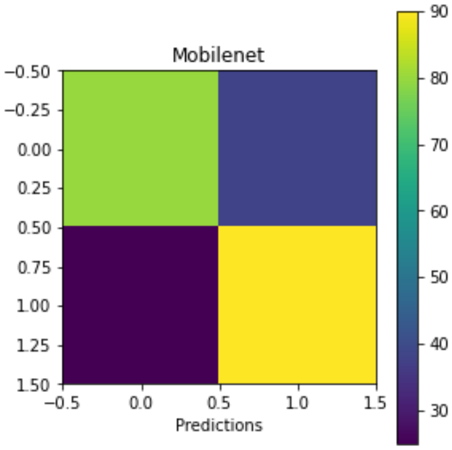

Fig. 4. Confusion matrix of the MobileNet model

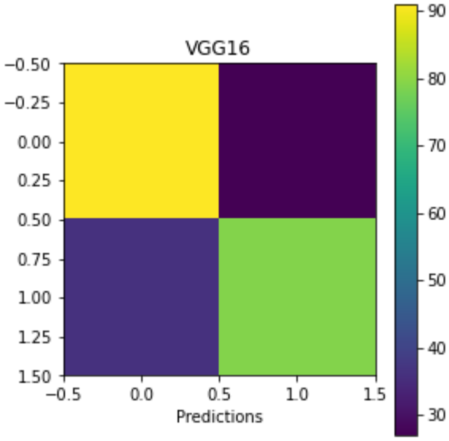

Fig. 5. Confusion matrix of the VGG16 model

between all the model even if the majority of the VGG16 and MobileNet weren't retrained on the data. So, maybe this is due to the data prepossessing. Indeed the data are one most of important parameters for the development of a good deep learning algorithm.

6 Future Improvements

First, an early stopping method could be implemented to avoid some arbitrariness in choosing the number of epochs. These methods are used to stop the training when a metrics stops improving. Often, the chosen metric is loss, as it allows to stop learning before an overfitting occurs.

In addition, the hardware impacts the result of the training. A study has shown, using Google Collab, that the training was faster, gets better and more consistent results when using a GPU rather than a CPU [13].

Furthermore, an another study [14] shown that additional objects decrease the recognition performance. In the dataset of the challenge, some images were very hairy. Thus, a method to remove hair could be added to the prepossessing step. The same study also shown that Gaussian blur and contrast shift could have some positive impact on the model quality. However, this impact varies greatly from one model to another but for sure, there is still room for improving the prepossessing steps of the model.

Finally, the large majority of the dataset was composed of light skins. This is explained by the fact that melanomas affect more people with light skins [2]. Due to this lack of variance in the skins tones, this model can't be used in every conditions. It's important to mention it because it can't impact the results obtained here. Indeed, the test and validation subdataset come from the same dataset so this error can't be expressed mathematically with these data. Therefore, a reinforced dataset could be an improvement.

7 Conclusion

As conclusion, a deep learning model was developed for a binary classification problem of skin lesions images but there are still room for improvement. The problem of the lack of reproducibility during the testing has to be treated in order to improve the development of the model. Furthermore, the prepossessing of data could also be improved. For example, only around 3% of the available pictures were used due to the method chosen for balancing the dataset. Moreover, other data about the patient were provided and could be used to have a better precision. Furthermore, other possible improvement were identified. However, the model still achieved an accuracy 0.7468 and make balanced predictions after only 25 epochs. Finally, these results were compared to the ones obtained using transfer learning techniques with the models VGG16 and MobileNet with pretrained weights. The results are comparable, and neither the built model or the pretrained model outperformed clearly the other.

Appendix A
Links

Here some links that could be usefull.

- The link to the SIIM-ISIC Melanoma Classification Challenge 2020:

https://www.kaggle.com/competitions/siim-isic-melanoma-classification

- The link to the repository of the code:

https://github.com/redekeys/PROJ-H419-Melanoma-Classification

References

[1] WHO, "Radiation: Ultraviolet (UV) radiation and skin cancer."

[2] A. Naeem, M. S. Farooq, A. Khelifi, and A. Abid, "Malignant melanoma classification using deep learning: Datasets, performance measurements, challenges and opportunities," *IEEE Access*, vol. 8, pp. 110575–110597, 2020.

[3] C.-I. Kim, S.-M. Hwang, E.-B. Park, C.-H. Won, and J.-H. Lee, "Computer-aided diagnosis algorithm for classification of malignant melanoma using deep neural networks," *Sensors (Basel, Switzerland)*, vol. 21, 08 2021.

[4] G. A. Holmes, J. M. Vassantachart, B. A. Limone, M. Zumwalt, J. Hirokane, and S. E. Jacob, "Using Dermoscopy to Identify Melanoma and Improve Diagnostic Discrimination," *Federal Practitioner*, vol. 35, pp. S39–S45, May 2018.

[5] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015. Software available from tensorflow.org.

[6] R. Kaur, H. GholamHosseini, R. Sinha, and M. Lindén, “Melanoma Classification Using a Novel Deep Convolutional Neural Network with Dermoscopic Images,” *Sensors*, vol. 22, p. 1134, Jan. 2022. Number: 3 Publisher: Multidisciplinary Digital Publishing Institute.

[7] M. A. Kassem, K. M. Hosny, and M. M. Fouad, “Skin Lesions Classification Into Eight Classes for ISIC 2019 Using Deep Convolutional Neural Network and Transfer Learning,” *IEEE Access*, vol. 8, pp. 114822–114832, 2020. Conference Name: IEEE Access.

[8] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018.

[9] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, “Training deep neural networks on imbalanced data sets,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, (Vancouver, BC, Canada), pp. 4368–4374, IEEE, July 2016.

[10] H. El-Amir and M. Hamdy, *Deep Learning Pipeline: Building a Deep Learning Model with TensorFlow*. Berkeley, CA: Apress, 2020.

[11] V. Verdhan, *Computer Vision Using Deep Learning: Neural Network Architectures with Python and Keras*. Berkeley, CA: Apress L. P, 2021.

[12] P. Sarang, *Artificial Neural Networks with TensorFlow 2: ANN Architecture Machine Learning Projects*. Berkeley, CA: Apress, 2021.

[13] M. N. Qureshi and M. S. Umar, “Performance Evaluation of Novel Convolution Neural Network Architecture for Melanoma Skin Cancer Diagnosis on Different Hardware Processing Units,” *Journal of Physics: Conference Series*, vol. 1950, p. 012039, Aug. 2021. Publisher: IOP Publishing.

[14] B. S. Akkoca Gazioğlu and M. E. Kamaşak, “Effects of objects and image quality on melanoma classification using deep neural networks,” *Biomedical Signal Processing and Control*, vol. 67, p. 102530, May 2021.