Project_plag_report

by Koushik V

Submission date: 30-Apr-2020 01:05PM (UTC+0700)
Submission ID: 1311894235
File name: project_report_for_plagiarism.PDF (1.18M)
Word count: 4755
Character count: 24302

Abstract

Skin cancer is the most common form of malignancy seen in humans, it is primarily diagnosed by visually looking at affected areas. Diagnosis begins with initial screening and then by dermoscopic analysis, a biopsy and then by histopathological examination. Automatic classification using machine learning and deep learning techniques is a challenging task due to the fine grained difference in the appearance of skin lesions((A skin lesion is part of the pores and skin that has an abnormal boom or appearance in comparison to the pores and skin around it). There are different types of skin cancer and treatment for each type is different. So identification of type of skin cancer is important. Automated process helps to make diagnosis easy. In this paper 7 different types of skin cancer are predicted. They are, **Melanocytic nevi , Melanoma, Basal cell carcinoma, Actinic keratoses, Vascular lesions, Dermatofibroma, Benign keratosis-like lesions**. The dataset used for the project is collected from kaggle.com. This dataset contains more than 10000 images of 7 different types of cancer. The final test accuracy obtained by this model on this dataset is 93.01%.

Contents

Chapter 1

Introduction

1.1 Introduction

Deep learning is a machine learning branch that fashions high-stage abstractions in facts using many processing layers. In this paper one such model CNN is discussed and used. It was initially designed to recognise cursive numbers and is later proved to be useful in object detection. These models are proved to be powerful classification tools. CNN models inclusive of GoogleNet, VGG and ResNet have showed better performances in image classification and recognition.

Despite the advancements in technology, however the inefficiency within the clinical dataset has restricted the utility of deep learning in biological data. Melanoma is a most usual skin cancer that showed huge mortality rate. It is estimated that nearly 9,730 deaths have occured due to melanoma in 2017. Basal cell carcinoma commonly referred to as as BCC is the most usual skin cancer, however is commonly not fatal.

So it is very important for both health care services to diagnose the type of cancer and to develop an efficient method to discriminate different types of skin cancer will be useful for initial screening. In the study, I have used a novel CNN algorithm (Snapshot Ensemble with Resnet50) in an effort to improve a classification tool using biological images of 7 different known skin lesions - Dermatofibroma, Melanocytic nevi, Basal cell carcinoma, Benign keratosis, Actinic keratoses, Vascular lesions, Melanoma,.

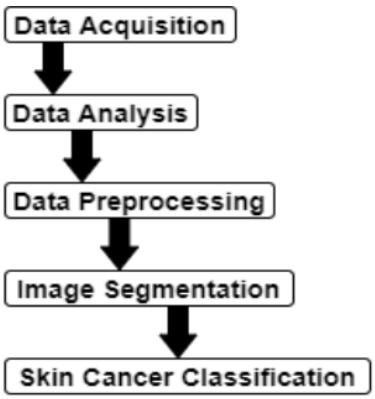


Fig.1: Flow used in Skin Cancer Diagnosis

In this study we used CNN based novel architecture using Snapshot Ensemble upon resnet50 architecture to classify the skin lesions. with the application of normal CNN architecture I got

an accuracy of 75.43% (+/- 1%) with the validation set and an accuracy of 76.14%(+/- 1%) for test set. When **Snapshot Ensemble**(explained in 4.2) was used the final accuracy was risen to **93.01%** for test set. This improvement is achieved by implementing all the functions in the flow diagram given in Fig.1.

1.2 Related Discussion

Huge efforts should be made to improve a image classification methods for more precise prediction of lesions. In one of the old studies, machine-guided diagnostic methods depending on feature extaction method imaged a considerable diagnostic capability with some types of skin cancer, which includes melanoma. However, an AI algorithm would not create precise diagnoses over a large varieties class of skin cancers. Not very long back, deep CNN architectures became popular in object classification using feature learning especially with image data. Extensive reasearch from ILSVR(ImageNet Large Scale Visual Recognition Challenge) has depicted that object classification abilities of CNNs can exceed over that of human diagnosis abilities.

Many dermatologic studies showed the uses of machine or deep learning. For example, Liao et ai. used a CNN based model to classify top level 23 categories such as, viral infections, bullous dise etc, with 23000 images. It showed an accuracy of 73.1% and 91.0% respectively for rates at which a model gives output of the correct label with top-1 and top-5 predictions for a given image. They have used a binary classification CNN based model whivh gave an AUC of 0.96 for carcinoma diagnosis using the above mentioned Edinburgh dataset with 707 cases and gave AUC of 0.96 for melanoma diagnosis which has 225 cases.

Chapter 2

Exploratory Data Analysis(EDA)

Here we discuss about the different features of the dataset, their distributions and the count of that types present in the dataset. This is helpful to analyse the nature of our data and helps us in the data processing step. First we will see the number of instances of data present for every possible values of every feature of data i.e. feature wise study of the data. Before jumping into the analysis part let us look at from where the data is collected.

2.1 Data Collection

Data used in this project contains images of several skin lesion types diagnosed using medical methods. This meta data and images used in this project are obtained from kaggle.com. dataset link: https://www.kaggle.com/kmader/skin-cancer-mnist-ham10000

Original data sources[2][3] as mentioned in kaggle are
https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T
https://arxiv.org/abs/1902.03368

2.2 Feature wise study

1) cell_type

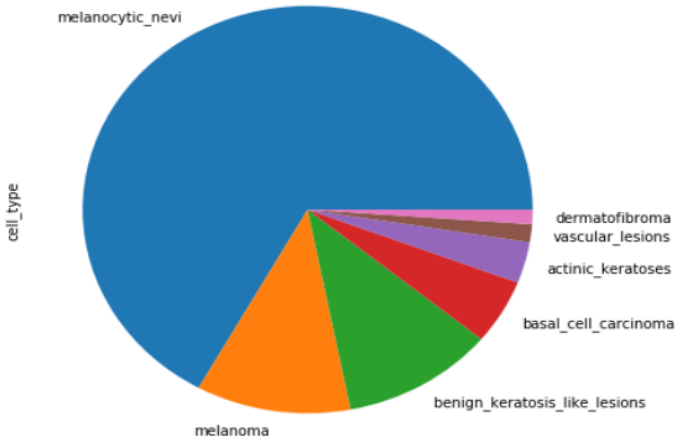


Fig 2.1

2) Technical Validation field (ground truth) which is dx_type

This field contains four catogeries,

1. Histopathology(Histo): It is the study of diseased tissue where the study is done by biopsy which is done by very specialized doctors known as pathologists.

2. Confocal: this is most commonly known as confocal laser scanning microscopy (CLSM) is one of the optical imaging technique which is used for improving optical resolution, contrast of a micrograph by using a spatial pinhole to block out-of-focus light in image formation.

3. Follow-up: this is a special type of technique which is used only for nevi. If diagnosis of nevi does not depict any betterment within 3 visits that is 1.5 year, then that type of nevi is declared as malignant and those images taken comes under this category.

4.Consensus: This type of groundtruth is confirmed of malignancy without histopathology or any follow-up. They take the expert consensus rating of two different methods given by HK and PT. Images are taken for educational purposes.

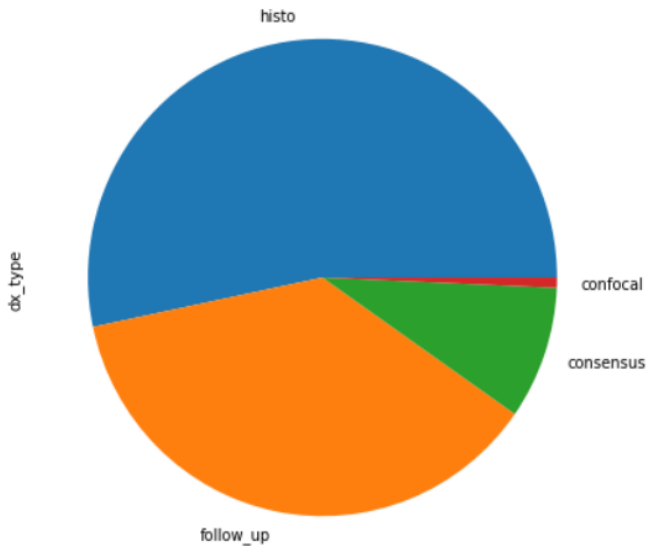


Fig 2.2

This pie chart shows that dataset contains more than half portion of data from Histopathology.

3) Localization

This field describes about the physical area from which the samples are collected from individuals.

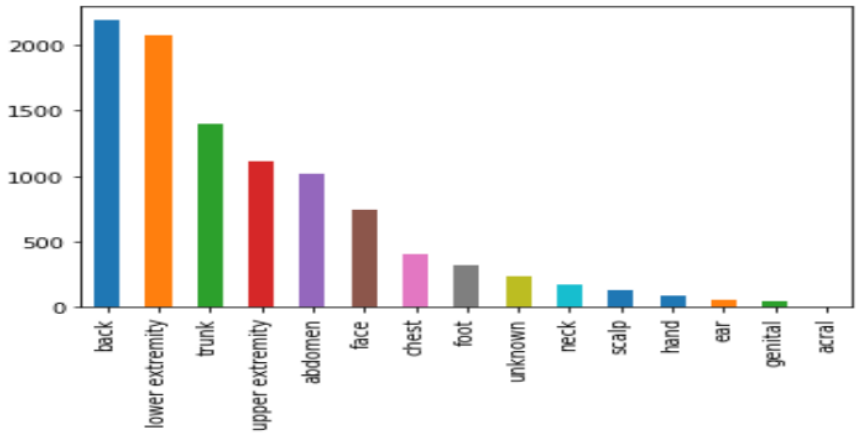


Fig 2.3

This bar graph shows back and lower extremity are two highly affected areas of skin cancer.

4) Age

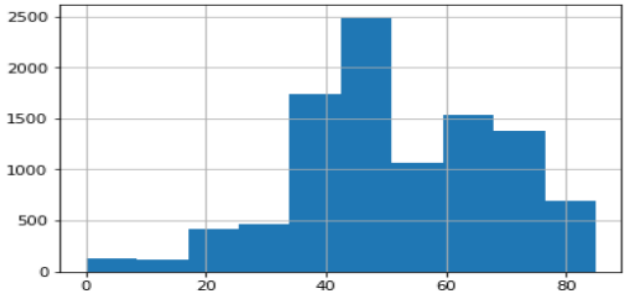


Fig 2.4

It is clear that most effected people by skin cancer are between 35 and 50 years of age.

5) Sex

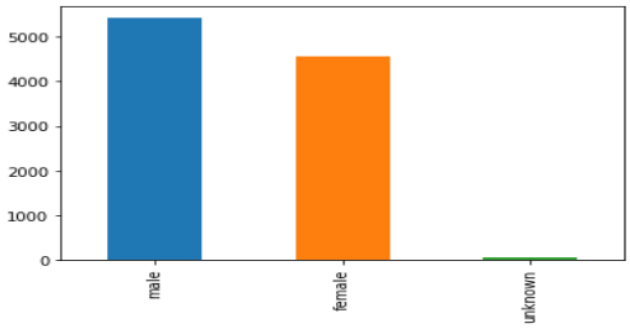


Fig 2.5

Not much to differentiate here with this bar graph.

2.3 Comparison with lesion type

1) Age wise distribution of skin cancer types

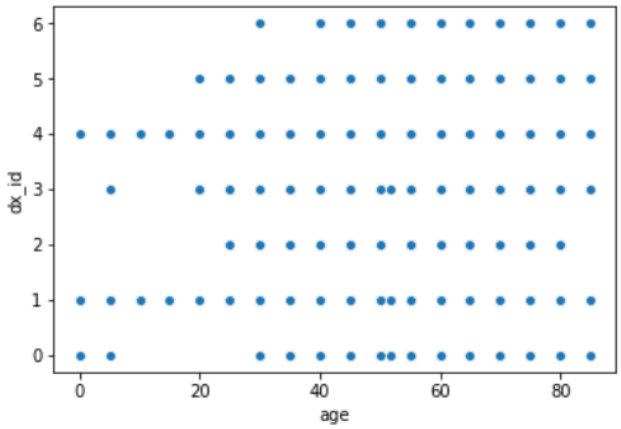


Fig 2.6

This scatter plot reveals that skin cancer types 2, 3, 5 and 6 which are Melanocytic nevi, Dermatofibroma, Basal cell carcinoma and Vascular lesions are not seen much below the age of 20 years.

2) Sex wise distribution of skin cancer type

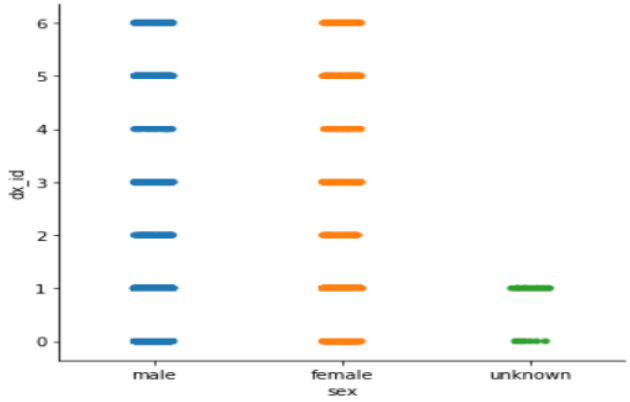


Fig 2.7

This cat-plot shows that there are no skin cancer types which are specific to a particular sex, that any of 7 skin cancer types can be seen in both males and females.

This concludes the data analysis part. we have analyzed the data clearly and now with this knowledge, data cleaning and preprocessing should be applied efficiently for a optimal training of dataset with our model.

Chapter 3

Data Processing

The next stage before training the data is data processing. Various steps applied here are,

1) Data Cleaning

2) Resizing Images

3) Data Normalization

4) Data Balancing

5) Data Augmentation

Each of these steps are discussed here.

3.1 Data Cleaning

Data cleaning is a important step in machine learning. Data cleaning plays an important role in building a proper model. Proper data cleaning can make or break the project. There is a popular belief that "Better data beats fancier algorithms".

Different steps in data cleaning are,

1) Unwanted observations removal

This means deleting duplicate, irrelevant and redundant values from dataset. Duplicate data arise mostly during data collection. Irrelevant observations are those observations that does not fit the certain problem that we are trying to solve**.** Our data does not contain any unwanted or duplicate, so we ignore this step.

2) Handling missing data

This type of data poses tricky problems in machine learning. This data cannot be ignored or removed from dataset because this data may contain important features specific to the corresponding class other than that of missing feature.

In this step we find all the null values in the data and replace them with mean values of that field. In our data only '**age**' field has null data as shown in and they are replaced by the mean values.

3.2 Resizing Images

original images in the dataset are all of same size but the size is very large. The images are of dimension (450 x 600 x 3) as seen in Fig 3.1. This will have a huge computation time while training and TensorFlow cannot handle this.

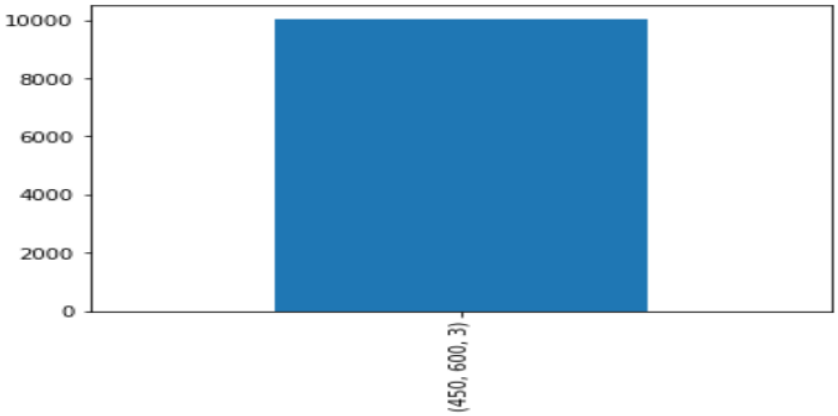


Fig 3.1

Let us resize the images by keeping the same size ratio so that no information is lost. I have resized the images to dimension (75 x 100 x 3). Now the shape of all images and their count is shown in Fig 3.2.

```
(75, 100, 3)    10015
Name: image, dtype: int64
```

Fig 3.2

3.3 Data Normalization

Original images are represented in colour code format with 3 values of Red, Blue and Green for each pixel and each value ranging from 0 to 255. So the normalization is applied as follows,

Normalized Image = (Original Image - mean(all original images)) / standard_deviation(all original images)

After normalization, each value of colour code format changed to a range of -2 to 2 which is preferred by neural networks.

3.4 Data Balancing

we have seen from data analysis that the data is highly **unbalanced**. Number of instances of each lesion type present in original data are shown in Fig 3.3.

```
melanocytic_nevi                  6705
melanoma                          1113
benign_keratosis_like_lesions     1099
basal_cell_carcinoma               514
actinic_keratoses                  327
vascular_lesions                   142
dermatofibroma                     115
Name: cell_type, dtype: int64
```

Fig 3.3: Unbalanced Data Category Count

This would make the model more or less biased towards certain classes. To solve this issue, I artificially added images to fewer categories to make them as equal as those of largest class. To **resample** these, we randomly chose and copy the images of fewer class. This will create duplicate images. To address this issue we have to perform **data augmentation** which will be discussed below. After balancing the number of instances of each categories are shown in Fig 3.4.

```
melanocytic_nevi                  6705
dermatofibroma                    5750
vascular_lesions                  5680
melanoma                          5565
benign_keratosis_like_lesions     5495
basal_cell_carcinoma              5140
actinic_keratoses                 4905
Name: cell_type, dtype: int64
```

Fig 3.4: Balanced Data Category Count

3.5 Data Augmentation

Deep neural networks perform better with large amount of data. Aim of this step is to create images that depict the features of its class in every possible angle. This makes sure that at whatever angle the image may be taken, our trained model can predict it with more precision.

Different techniques used for this are,

1) Randomly rotate the images in the range from 0 to 180 degrees

2) Randomly zoom images

3) Randomly flip images horizontally

4) Randomly flip images vertically

5) Randomly shift the images horizontally

6) Randomly shift the images vertically

Before starting the training, the dataset is divided into 80% training data and 20% test data. And this training data is again divided into 80% training data and 20% validation data. The data augmentation is only performed on the finally obtained training data, Because only this data is used for model training.

Chapter 4

Image Classification using CNN

4.1 Why CNN for skin cancer image classification?

Convolution Neural Networks are widely used for image classification. They tend to produce better results than other classification algorithms because of their efficiency to handle large data. CNN uses a unique way of image classification by applying different filters to find the specific features in images. For example, let us see some sample images from different lesion types from out data set as shown in Fig 4.1.

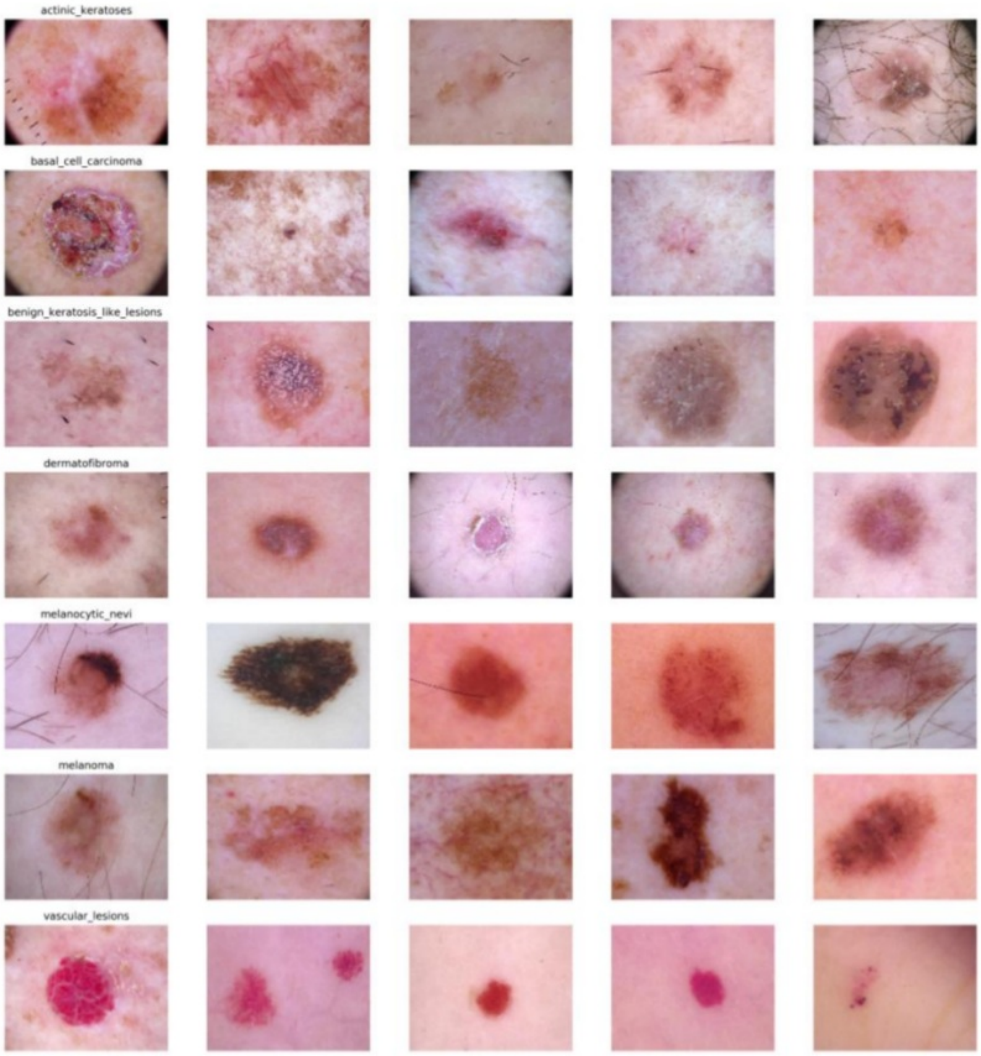


Fig 4.1: Category wise sample images

In these images we can observe that important features in most of the images are circles and disk like shapes in these images. So, to differentiate between these lesion types our model should be able to focus on those circles and disk like shapes in the images. This ability to grasp the deeper features hidden in the images makes CNN very useful and puts it above other classification models. A typical CNN architecture is shown in Fig 4.2 (https://miro.medium.com/max/1400/0*KTak4HI5Lp3w8D6l).

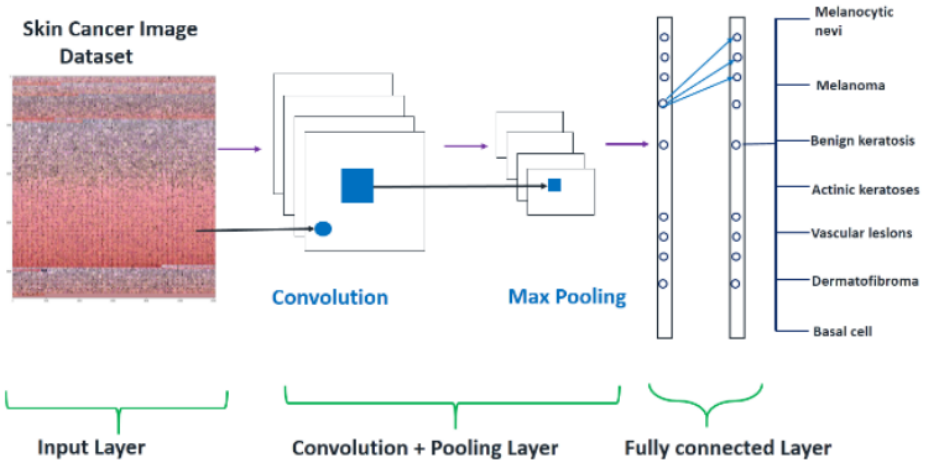


Fig 4.2

This ability to grasp the deeper features hidden can be observed by using relevant filters in the convolution layer. After applying the filters the output of **convolution layer** is given as input to **pooling layer** where the raw output is converted to predetermined features of the corresponding filters. In this way a raw image is converted to a list of features that we desired when applying filters. Then using these list of features we use normal neural network architecture to classify into one of the seven lesion types using **Fully Connected Layer**.

4.2 Different Concepts Used

This section deals with some concepts used over a basic CNN model to increase the prediction accuracy of the model.

1) Transfer Learning

The idea behind the transfer learning is to use a CNN model which is pre trained on ImageNet data as lower layers of our model so that it can capture some generic features and

we fine tune the higher layers to our specific domain and redefine the last layer to output 7 values corresponding to different lesion types. After experimenting with different pre trained architectures for 5 epochs, I found that Resnet50 is giving better results. So I adopted that architecture as it gave the best validation accuracy upon training on our data. I also used Adam optimizer with weight decay to reduce overfitting.

2) Cyclic Learning Rate Scheduling

To improve the results and to make the model converge at global minimum instead of local minimum we have to periodically increase the learning instead of determining the optimal learning rate exponentially, I have used a Cyclic Learning Rate Scheduling. This varies the learning rate cyclically, which helps the model to escape several global minimum. It also eliminates the necessity to find an optimal maximum learning rate manually. The Cyclic learning rate used is given in Fig 4.3.

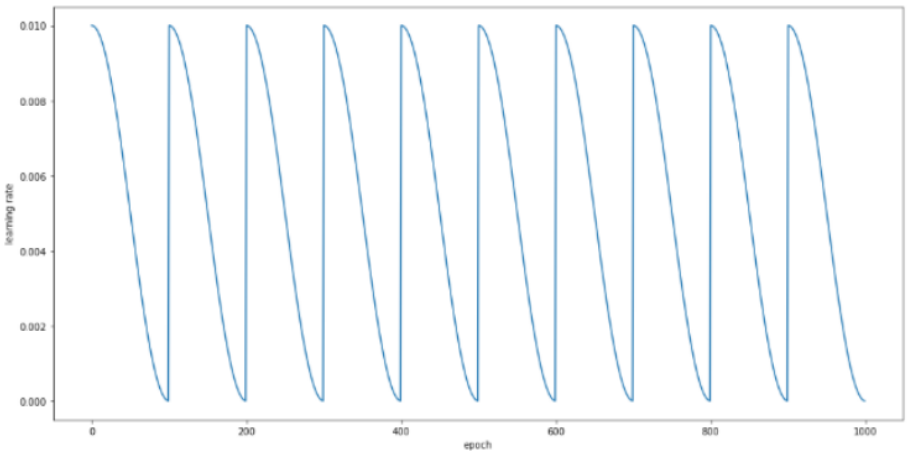


Fig 4.3: Cyclic Learning Rate

3) Snapshot Ensemble

Normal ensemble models are very powerful in improving the model performance. But, they also are computationally very expensive to separately train the model using different algorithms used in ensemble learning. So I have adopted a different ensemble technique known as **snapshot ensemble** with cyclic LR scheduling.

The main idea is to save the model parameters periodically during training. when a model converges to local minima during a cycle these parameters are saved and the learning rate is then increased to apply another model. In this way it allows us to gather an ensemble of models in a single training cycle.

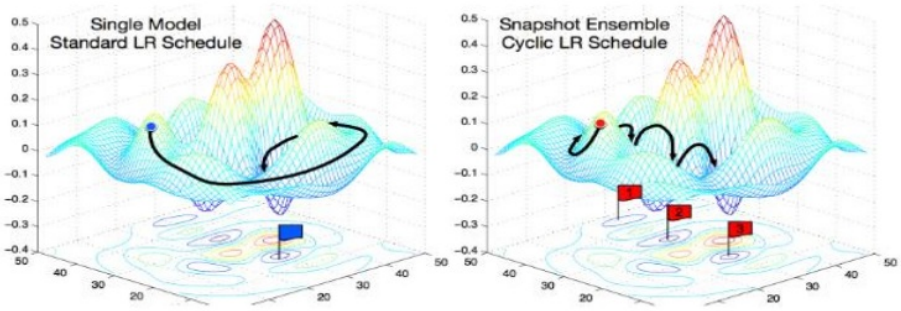


Fig 4.4

Fig 4.4(https://arxiv.org/abs/1704.00109) shows the differences between normal model and an snapshot ensemble model. In the later, we gradually save the snapshots at each local minimum and so we reach global minimum very fast. But in normal model we have to travel for a long time to reach global minimum. So this ensemble model helps us to reach global minimum in less epochs.

This can be implemented using the formula,

$$\alpha(t) = \frac{\alpha_0}{2}\left(\cos\left(\frac{\pi \mathbf{mod}(t-1, \lceil T/M \rceil)}{\lceil T/M \rceil}\right) + 1\right)$$

This formula gives the learning rate similar to that of [Fig 4.3]. So we can implement this model by simply applying this formula to get the desired learning rate.

4.3 Model Building

The model is built in python using Keras Sequential API. In this API, we have to attach one layer to the model at a time.

First the Resnet50 architecture is added to the model. This is a pre trained architecture to capture generic features.

Then a dropout layer is added, dropout is a new regularization method, here a portion of nodes are ignored randomly for each training sample. this makes our model to learn features in distributed way. Dropout also improves the generalization and thereby reduces overfitting.

Layer (type)	Output Shape	Param #
resnet50 (Model)	(None, 2048)	23587712
dropout_1 (Dropout)	(None, 2048)	0
dense_1 (Dense)	(None, 128)	262272
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 7)	903

Total params: 23,850,887
Trainable params: 9,194,503
Non-trainable params: 14,656,384

Fig 4.5

Then a dense layer with 128 nodes is added, this is a part of fully connected layer where different features from resnet are converted to give a output from 128 nodes. This is just an artificial neural network classifier.

Then again a dropout layer is added which is followed by the output dense layer with 7 nodes corresponding to 7 different lesion types, the net output gives the probability of each class.

My CNN architecture is,

Input => Resnet50 => Dropout => Dense => Dropout => Dense => Output

Complete details are shown in [Fig 4.5]. In both the dense layers we used 'relu' activation functio. It adds non linearity to the model.

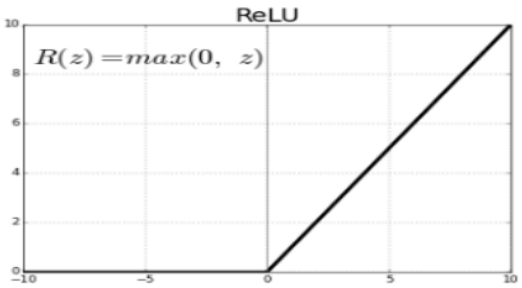


Fig 4.6

ReLU stands for rectified linear unit. It is the most commonly used activation function.

4.4 Training the Model

After building the model, it is compiled using Adam optimizer. when this model is trained with exponential learning rate reducer, test accuracy was risen to nearly 83% from 77% of normal CNN without resnet(Some other model). Then after using the Cyclic LR Scheduling and the **Snapshot Ensemble** techniques for training, test accuracy was risen to 92.71%.

This accuracy is obtained by trying different combinations of epochs and number of ensemble models. I have observed that during normal training procedure there is not a significant increase in the accuracy and even loss remained constant or even start increasing after **30 epochs**.

So to reduce the computation time, I tried different number of models and number of epochs combination so that, **(no. of models)** x **(no. of epochs per model**) = **30**, such that total epochs for entire model remains 30. Finally among them the combination,**(No. of models = 3, and No. of epochs per model =10)** seems to be the optimal solution. **Max learning rate** is set to **0.001.** Learning rate decreases and then increases periodically to max learning rate. **Batch size** is set to **10**. Then model fitting is done with these parameters set on the training data, and validation data which is segmented earlier is used for validating the performance during training process. Once the training is done, Test data is used to evaluate the final model performance.

Chapter 5

Results

The proposed CNN model is used to classify the skin cancer type of the dataset that comprises the skin lesion images of humans. Images below shows the results of classification using test data and validation data.

The model gave the test accuracy of 92.71% and validation accuracy of 91.2% as shown in Fig 5.1.

```
2003/2003 [==============================] - 2s 822us/step
1603/1603 [==============================] - 1s 776us/step
Validation: accuracy = 0.912040  ;  loss_v = 0.522199
Test: accuracy = 0.927109  ;  loss = 0.467480
```

Fig 5.1

Fig 5.2 shows the changes in accuracy and loss of training data and validation data during training process as the epochs progresses.

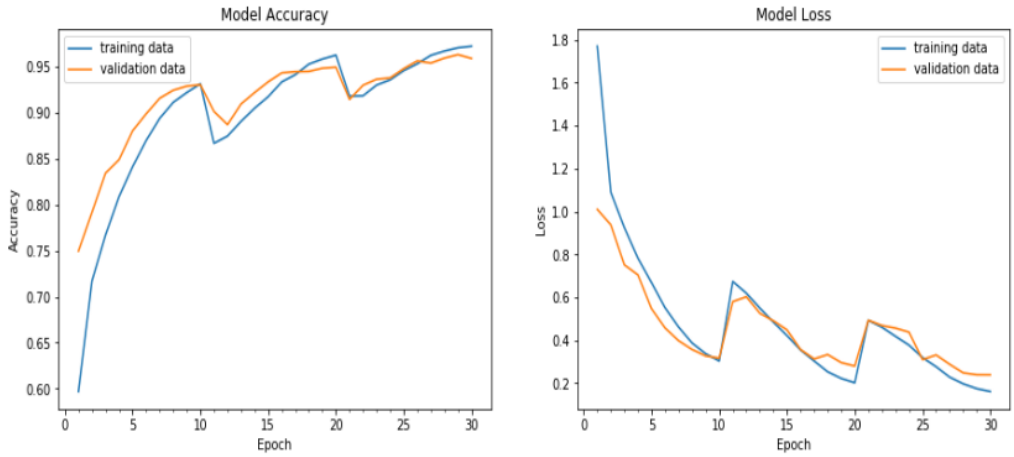


Fig 5.2

Fig 5.3 shows the confusion matrix of validation data.

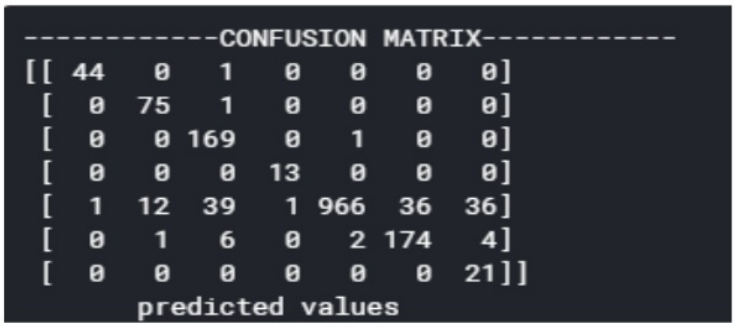


Fig 5.3

Fig 5.4 shows the confusion matrix of test data.

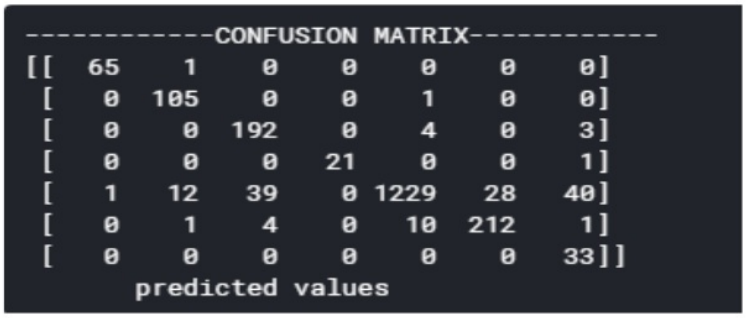


Fig 5.4

Fig 5.5 shows the Incorrectly predicted fraction of each lesion type of validation data .

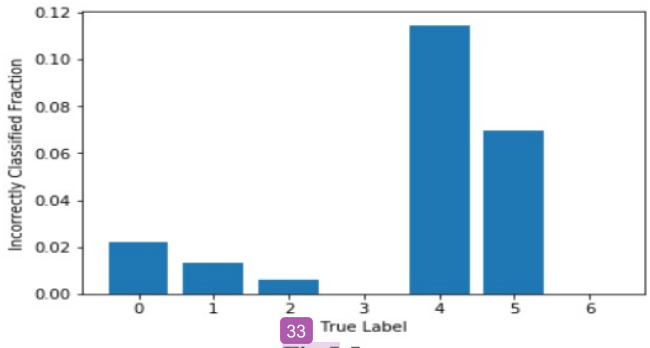


Fig 5.5

Fig 5.6 shows the Incorrectly predicted fraction of each lesion type of test data .

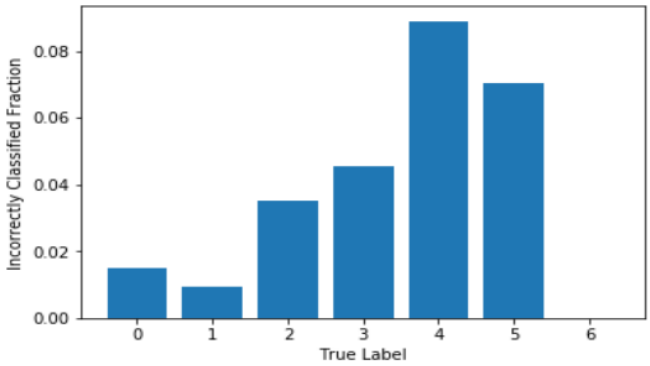
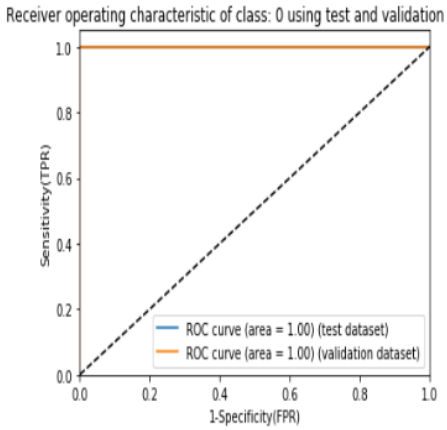


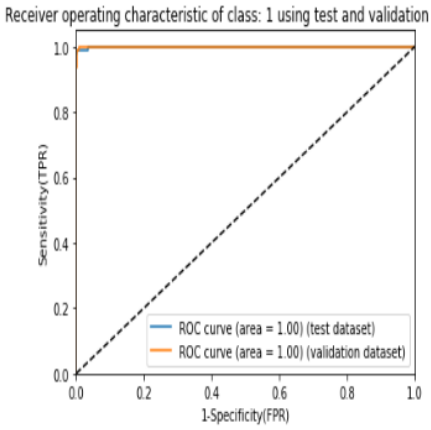
Fig 5.6

The following images shows the ROC(Receiver Operating Characteristic) curves of both test and validation data for each class.

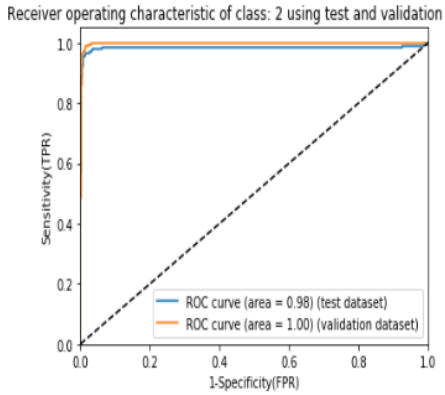
A



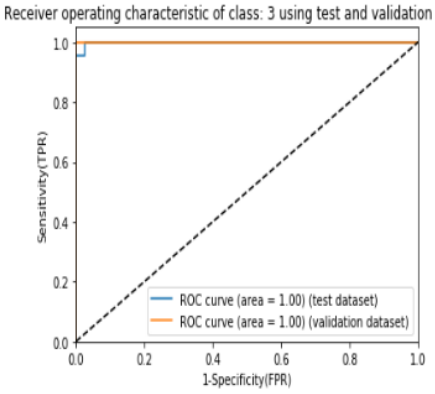
B



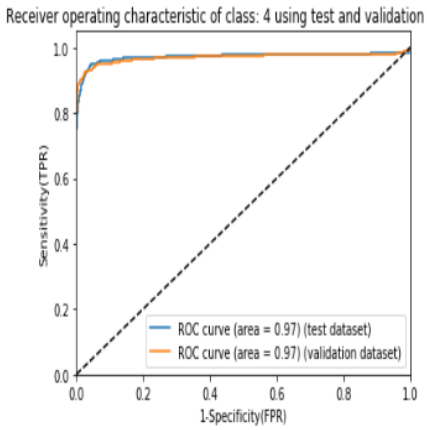
C



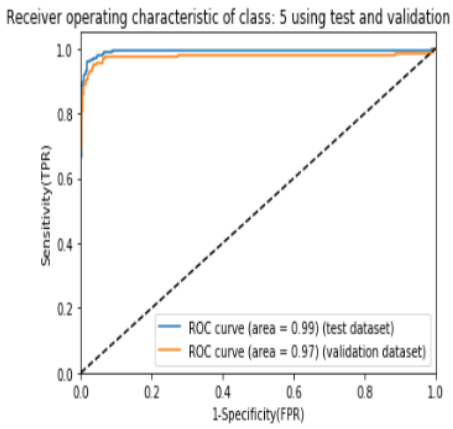
D



E



F



G

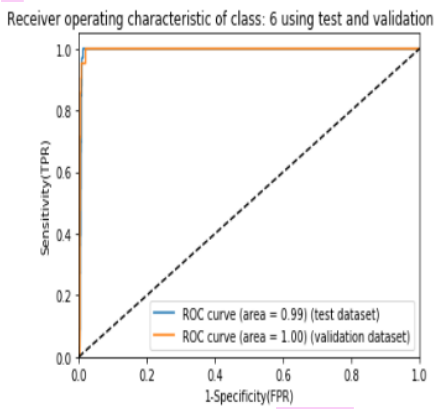


Fig 5.7: ROC curves of each lesion types

Ensemble Accuracy Calculation

Snapshot Ensemble model used above have 3 models, when each snapshot saved after the model execution is evaluated using test data we get three different probabilities from three models. Using these three probabilities an **ensemble probability** can be calculated by taking the **average** of all three probabilities and the resultant class with highest probability is added to predicted data. Thus we can get an ensemble model with **ensemble accuracy**. That accuracy along with model accuracies are shown in Fig 5.8.

```
model 1: accuracy = 0.8882
model 2: accuracy = 0.9051
model 3: accuracy = 0.9271
ensemble: accuracy = 0.9186
```

Fig 5.8

Improving Ensemble Accuracy

In the above ensemble accuracy calculation we have taken the average of probability of all 3 models. Instead of average we can take **weighted average** to produce better results. But the problem is, how to chose the weights for different models. This can be done by initialising the weights randomly and then check for improvement, if there is any improvement then add new weights to **best weights**. If there is no improvement then increment the no improvement counter. We continue this process until the no improvement counter reaches certain limit and this limit is known as **patience**. Thus the final accuracy and final improved weights can be calculated using Algorithm 1.

By applying this method to our test data the final **ensemble accuracy is 93.01%**. The improvement process and the resultant best weights are shown in [Fig 5.9].

```
improvement: 0.9216
improvement: 0.9226
improvement: 0.9236
improvement: 0.9261
improvement: 0.9266
improvement: 0.9286
improvement: 0.9291
improvement: 0.9296
improvement: 0.9301
best weights are [0.04802128 0.14556977 0.80640896]
```

Fig 5.9

Algorithm 1: Improving Ensemble accuracy

```
best_score = ensemble_score
best_weights = None
no_improvements = 0
while no_improvements < 5000: #patience
    select new_weights randomly
    Normalize new_weights
    new_score = evaluate(preds, new_weights)

    if new_score is greater than best_score:
        no_improvements = 0
        best_score = new_score
        best_weights = new_weights
        print(best_score)
    else:
        no_improvements += 1

print(best_weights)
```

Fig 5.10 shows the precision, recall, f1-score and support values for test and validation data using best weights obtained. Predictions are made using best weights and they are compared with true values to calculate these metrics.

Validation data Report

	precision	recall	f1-score	support
bkl	1.00	0.98	0.99	45
nv	0.99	1.00	0.99	76
df	0.82	0.99	0.90	170
mel	0.87	1.00	0.93	13
vasc	0.99	0.93	0.96	1091
bcc	0.86	0.92	0.89	187
akiec	0.47	1.00	0.64	21
micro avg	0.94	0.94	0.94	1603
macro avg	0.86	0.97	0.90	1603
weighted avg	0.95	0.94	0.94	1603
samples avg	0.94	0.94	0.94	1603

Test data Report

	precision	recall	f1-score	support
bkl	1.00	1.00	1.00	66
nv	0.96	0.98	0.97	106
df	0.85	0.97	0.91	199
mel	0.88	0.95	0.91	22
vasc	0.99	0.94	0.96	1349
bcc	0.90	0.94	0.92	228
akiec	0.53	1.00	0.69	33
micro avg	0.95	0.95	0.95	2003
macro avg	0.87	0.97	0.91	2003
weighted avg	0.95	0.95	0.95	2003
samples avg	0.95	0.95	0.95	2003

Fig 5.10

These are the various results obtained on the skin cancer image data using this CNN model.

6

Chapter 6

Future work and Conclusion

6.1 Future work and other ideas

This section contains the future work to be done and some ideas that are either not implemented or implemented improperly.

Proper Data Collection

In many of the image classification projects correct results depends mostly on dataset. Our dataset contains 10,015 images, but most of those images are of only one class Melanocytic Nevi. It is important to collect a balanced data for better predictions.

Also our dataset has only seven different lesion types whereas in practical there are many more different types of skin cancer. One of the future work is to collect a better dataset with more balanced data with more lesion types.

Also our data only contains cancer images of patients. This model can only tell different types of cancer, but cannot tell whether the lesion itself is malignant or not. Along with the seven classes if we can get images of non malignant lesions and add them to our data and then train the model, we can also detect cancer and also tell the type of cancer.

Other ensemble methods

We also tried some other ensemble methods where model is trained separately for each algorithm. One such example is to try linear regression as one model, Decision tree as another algorithm, CNN as another algorithm and calculate the ensemble accuracy as mentioned in previous chapters.

The models We have tried are computationally much expensive with not much significant increase in accuracy. One future work is to try and find better ensemble models which produces better accuracy.

Generative adversarial networks

Data augmentation and class balancing plays an important role in model performance as seen previously. Besides classic image processing pre defined generative models can be used to improve performance. Some such models are, BAGAN for balancing and DAGAN for augmentation, etc,.

Different Activation Functions in CNN

In our CNN architecture we have used **ReLU** activation function. This is most commonly used activation function but we can try some novel activation functions available such as **MISH** activation function. It is a self regularised non-monotonic neural network activation function given as,

$$f(x) = x * \tanh(\ln(1 + e^x))$$

There are some other similar activation functions like **SWISH**. We can also use a customized activation function.

6.2 Conclusion

This work involved a novel method involving Snapshot Ensemble with Resnet50 architecture for skin lesion type identification based on image data. This introduces a new method to identify lesion type before medical diagnosis and helps the medical staff to diagnose the type predicted by the model first. This reduces a lot of time and effort. This model uses a Convolution Neural Network(CNN) within a TensorFlow framework. For the specific cancer types the accuracy ranges from 88% to 100%. and the overall test accuracy of the model is 93.01%.

Compared to traditional medical diagnosis this method is better in two ways. First, This method will not be influenced by diagnostic medical instruments, that is error in the instruments can occur during medical diagnosis and this will not have any effect in this method if the images are of good quality. With more and more cancer cases our model becomes more and more powerful unlike medical instruments which has wear and tear. Second, this method does not depend on stage of cancer and the cancer can be detected even in an early stage. But some medical diagnosis can detect some types of skin cancer only when it reaches advanced stage and this does not occur with this model. If among seven types, we add ane more class which is non malignant canser type and then train the model, our model can even detect whether it is malignant or not and also tells the type of cancer.

This model may have some limitations like the prediction probability cannot be 100%. But it provides a good basis for furthur diagnosis and will be very helpful.

References

[1] Han, S. S., Kim, M. S., Lim, W., Park, G. H., Park, I., & Chang, S. E. (2018). Classification of the clinical images for benign and malignant cutaneous tumors using a deep learning algorithm. *Journal of Investigative Dermatology*, *138*(7), 1529-1538.

[2] Codella, N., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S., Gutman, D., ... & Kittler, H. (2019). Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*.

[3] Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, *5*, 180161.

[4]Liao, H., & Luo, J. (2018). A deep multi-task learning approach to skin lesion classification. *arXiv preprint arXiv:1812.03527*.

[5]Codella, N. C., Nguyen, Q. B., Pankanti, S., Gutman, D. A., Helba, B., Halpern, A. C., & Smith, J. R. (2017). Deep learning ensembles for melanoma recognition in dermoscopy images. *IBM Journal of Research and Development*, *61*(4/5), 5-1.

[6] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, *542*(7639), 115-118.

[7] Arevalo, J., Cruz-Roa, A., Arias, V., Romero, E., & González, F. A. (2015). An unsupervised feature learning framework for basal cell carcinoma image analysis. Artificial intelligence in medicine, 64(2), 131-145.

[8] Giotis, I., Molders, N., Land, S., Biehl, M., Jonkman, M. F., & Petkov, N. (2015). MED-NODE: a computer-assisted melanoma diagnosis system using non-dermoscopic images. *Expert systems with applications*, *42*(19), 6578-6585.

[9] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 1026-1034).

[10] Ioffe, S., & Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv e-prints.

[11] Agarwal, A., Negahban, S., & Wainwright, M. J. (2012). A simple way to prevent neural networks from overfitting. *Ann. Stat.*, *40*(2), 1171-1197..

Project_plag_report

ORIGINALITY REPORT

11%	4%	6%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Seung Seog Han, Myoung Shin Kim, Woohyung Lim, Gyeong Hun Park, Ilwoo Park, Sung Eun Chang. "Classification of the Clinical Images for Benign and Malignant Cutaneous Tumors Using a Deep Learning Algorithm", Journal of Investigative Dermatology, 2018 Publication	1%
2	oasiswebdevelopment.com Internet Source	1%
3	Sounil Biswas, Hongfei Wang, R. D. (Shawn) Blanton. "Reducing test cost of integrated, heterogeneous systems using pass-fail test data analysis", ACM Transactions on Design Automation of Electronic Systems, 2014 Publication	1%
4	complexity.cecs.ucf.edu Internet Source	<1%
5	"Deep Learning to Detect Skin Cancer using Google Colab", International Journal of Engineering and Advanced Technology, 2019	<1%

	Publication	
6	ir.uiowa.edu Internet Source	<1%
7	Submitted to University of Lincoln Student Paper	<1%
8	Submitted to Deakin University Student Paper	<1%
9	Lee Yi Bin, Goh Yeh Huann, Lum Kin Yun. "Study of Convolutional Neural Network in Recognizing Static American Sign Language", 2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 2019 Publication	<1%
10	"OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis", Springer Science and Business Media LLC, 2018 Publication	<1%
11	Submitted to Feng Chia University Student Paper	<1%
12	Submitted to University of Sydney Student Paper	<1%
13	issuu.com Internet Source	<1%

14	discovery.ucl.ac.uk Internet Source	<1%
15	Submitted to Sri Lanka Institute of Information Technology Student Paper	<1%
16	Zidrina Pabarskaite, Aistis Raudys. "A process of knowledge discovery from web log data: Systematization and critical review", Journal of Intelligent Information Systems, 2006 Publication	<1%
17	Submitted to National College of Ireland Student Paper	<1%
18	"Image Analysis and Recognition", Springer Science and Business Media LLC, 2017 Publication	<1%
19	www.ijert.org Internet Source	<1%
20	Submitted to University of Limerick Student Paper	<1%
21	krishikosh.egranth.ac.in Internet Source	<1%
22	Submitted to Queen Mary and Westfield College Student Paper	<1%
23	image-net.org Internet Source	<1%

24	Submitted to Bilkent University Student Paper	<1%
25	Qi Kang, Ying Fu, Hua Huang. "Deep Color Image Demosaicking with Feature Pyramid Channel Attention", 2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), 2019 Publication	<1%
26	www.nature.com Internet Source	<1%
27	Submitted to University of Melbourne Student Paper	<1%
28	George S. Liu, Michael H. Zhu, Jinkyung Kim, Patrick Raphael, Brian E. Applegate, John S. Oghalai. "ELHnet: a convolutional neural network for classifying cochlear endolymphatic hydrops imaged with optical coherence tomography", Biomedical Optics Express, 2017 Publication	<1%
29	www.mdpi.com Internet Source	<1%
30	arxiv.org Internet Source	<1%
31	Submitted to Freedom High School Student Paper	<1%

32	Submitted to The Hong Kong Polytechnic University Student Paper	<1%
33	Submitted to University of Auckland Student Paper	<1%
34	Submitted to Universiti Teknologi MARA Student Paper	<1%
35	"Proceedings of the Global AI Congress 2019", Springer Science and Business Media LLC, 2020 Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		