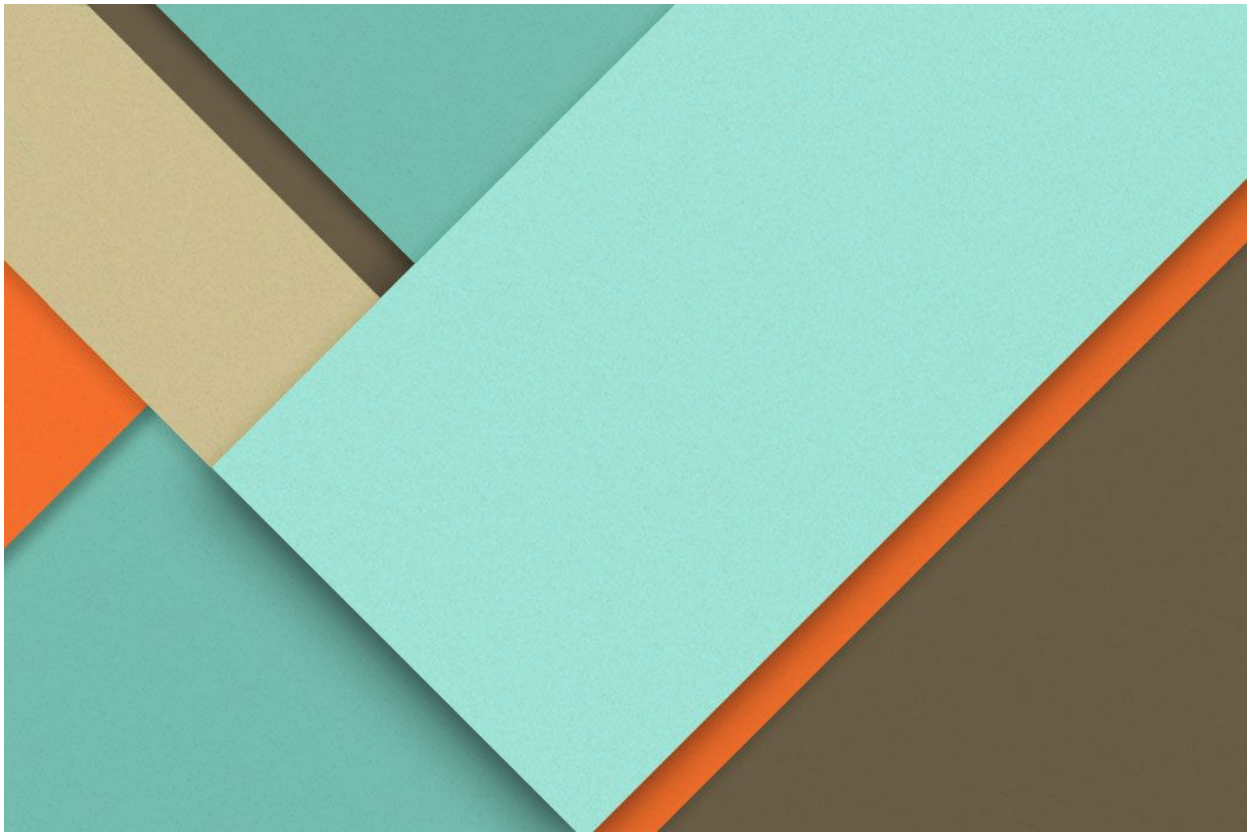


MRNet

Team :

Mohamed Essam Mohamed ID : 3879

Mostafa Yousry ID : 4505

Overview

The MRNet dataset consists of 1,370 knee MRI exams performed at Stanford University Medical Center. The dataset contains 1,104 (80.6%) abnormal exams, with 319 (23.3%) ACL tears and 508 (37.1%) meniscal tears; labels were obtained through manual extraction from clinical reports.

Goals

1- detect the patient suffering from (abnormal OR ACL OR meniscal tears)from different angle (Axial, Coronal, sagittal) and print the accuracy of this detect.

2- Build a deep CNN model to perform the classification.

3- Use Transfer Learning and ensemble.

4- Plot accuracy, F-score, and loss of both training and validation sets per epoch.

Steps :

1- Loading dataset using **glob.glob()** and ***np.load()*** the sort the data set using **Sorted()**.

2- Extract each exam stack slice (Contain s slices) to 3 because most of model CNN accept only 3 channel .

3- Loading the labels of training data for each knee.

4-we use **InceptionV3**. model with shape 265*265*3.

5- get the output of the **InceptionV3** model "feature extraction" and add Dense layers so that the model is designed to classify our data.

6- freeze the layers and unfreeze the rest.

7- the loss function is binary_crossentropy as our labels are binary***(0|1)***

Adam optimizer use small learning rate(lr) to ***0.0001.***

8 - Using call back method to manage our model to calculate f_score.

9- *train the model with 2214 training data & 46 validation data.*

save the model that has the min validation loss.

10 - plot the training loss & accuracy, validation loss & accuracy & F-score.

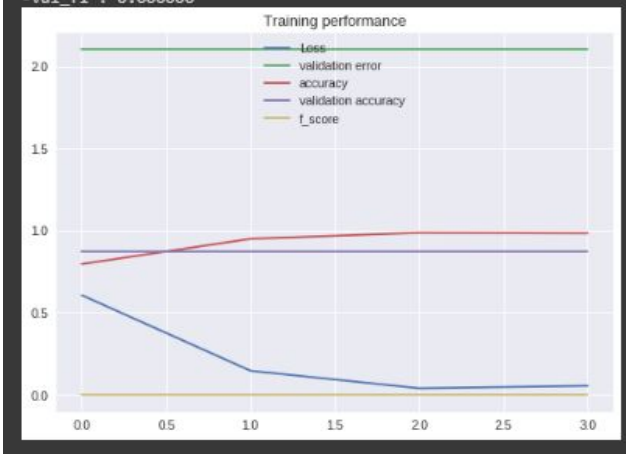
11- Evaluate model.

12- *predict if the knee has an acl tear by using the data of the 3 knee angel (axial, coronal & sagittal) and the 3 models that accept one of these data and predict if the knee has an* ***abnormal tear or not*** *or* ***has acl or not*** *or* ***has meniscus*** *tear or not by doing a majority voting between the 3 models*

Screenshot :

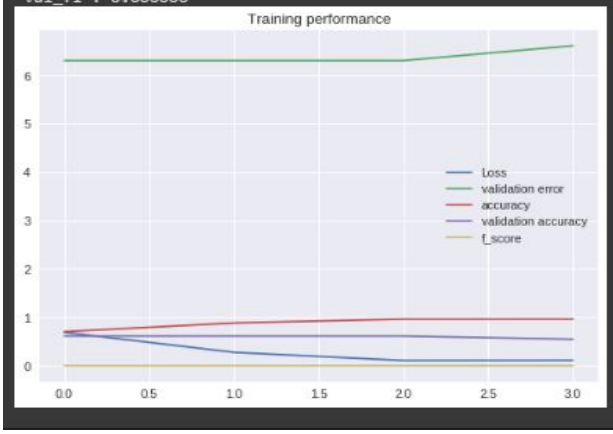
```
axial_acl_model = fit_model(training_set_axial, training_label_acl, "axial_acl_model")

Train on 2214 samples, validate on 46 samples
Epoch 1/30
2214/2214 [==============================] - 20s 9ms/step - loss: 0.6054 - acc: 0.7958 - val_loss: 2.1024 - val_acc: 0.8696
/usr/local/lib/python3.6/dist-packages/sklearn/metrics/classification.py:1437: UndefinedMetricWarning: F-score is ill-defined
  'precision', 'predicted', average, warn_for)
-val_f1 : 0.000000
Epoch 2/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.1449 - acc: 0.9494 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
Epoch 3/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.0384 - acc: 0.9864 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
Epoch 4/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.0533 - acc: 0.9833 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
```




```
axial_meniscus_model = fit_model(training_set_axial, training_label_meniscus, "axial_meniscus_model")

Train on 2214 samples, validate on 46 samples
Epoch 1/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.6851 - acc: 0.7005 - val_loss: 6.3071 - val_acc: 0.6087
/usr/local/lib/python3.6/dist-packages/sklearn/metrics/classification.py:1437: UndefinedMetricWarning: F-score is ill-defined
  'precision', 'predicted', average, warn_for)
-val_f1 : 0.000000
Epoch 2/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.2720 - acc: 0.8799 - val_loss: 6.3071 - val_acc: 0.6087
-val_f1 : 0.000000
Epoch 3/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.1035 - acc: 0.9603 - val_loss: 6.3071 - val_acc: 0.6087
-val_f1 : 0.000000
Epoch 4/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.1067 - acc: 0.9612 - val_loss: 6.6150 - val_acc: 0.5435
-val_f1 : 0.000000
```



```
coronal_abnormal_model = fit_model(training_set_coronal, training_label_abnormal, "coronal_abnormal_model")
```

```
Train on 2214 samples, validate on 46 samples
Epoch 1/30
2214/2214 [==============================] - 20s 9ms/step - loss: 0.5443 - acc: 0.7918 - val_loss: 2.0795 - val_acc: 0.8696
-val_f1 : 0.930233
Epoch 2/30
2214/2214 [==============================] - 20s 9ms/step - loss: 0.1849 - acc: 0.9286 - val_loss: 2.3734 - val_acc: 0.8478
-val_f1 : 0.917647
Epoch 3/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.0794 - acc: 0.9729 - val_loss: 2.1145 - val_acc: 0.8478
-val_f1 : 0.917647
Epoch 4/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.0689 - acc: 0.9738 - val_loss: 10.0714 - val_acc: 0.3043
-val_f1 : 0.448276
```


```
coronal_acl_model = fit_model(training_set_coronal, training_label_acl, "coronal_acl_model")
```

```
Train on 2214 samples, validate on 46 samples
Epoch 1/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.6430 - acc: 0.7827 - val_loss: 2.1024 - val_acc: 0.8696
/usr/local/lib/python3.6/dist-packages/sklearn/metrics/classification.py:1437: UndefinedMetricWarning: F-score is ill-defined a
  'precision', 'predicted', average, warn_for)
-val_f1 : 0.000000
Epoch 2/30
2214/2214 [==============================] - 21s 9ms/step - loss: 0.3604 - acc: 0.8379 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
Epoch 3/30
2214/2214 [==============================] - 21s 10ms/step - loss: 0.1910 - acc: 0.9264 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
Epoch 4/30
2214/2214 [==============================] - 21s 10ms/step - loss: 0.0989 - acc: 0.9666 - val_loss: 2.1024 - val_acc: 0.8696
-val_f1 : 0.000000
```

Our Accuracy to predict :

1- abnormal : `79.16666666666666`

2- ACL : `55.00000000000001`

3- meniscus *:* `56.666666666666664`