UDACITY

‹ Return to Classroom

Pneumonia Detection from Chest X-Rays

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Dear Udacious Learner,
Congratulations you made it!!! 🏆
This is a great submission and I know it's been hours of great brainwork which has of course yielded a great outcome. Keep up with the great work and remain dacious

Extra Resources

- You may want to expand your knowledge on this project by reading this article on Efficient Pneumonia Detection in Chest Xray Images Using Deep Transfer Learning and also this article on Pneumonia Detection Using an Improved Algorithm Based on Faster R-CNN

Exploratory Data Analysis

- Students create distributions of diseases and comorbidities in their dataset
- Students create distributions of basic demographics of the patients who make up their datasets (such as age, gender, patient position,etc.)
- Students can use the above distributions to draw conclusions about how they will need to set up their model training

their model training

Really outstanding job with Exploratory analysis of data. The plots representing the age and gender distribution are really meaningful and self-explanatory. Great work

- Students use python's imshow to visualize medical images during EDA
- Students create distributions of intensity values of the pixel-level data within images and compare them both within and across diagnoses
- Students use both of these methods of inspecting images to draw meaningful conclusions about what their model will train on

- Good job analyzing the images at pixel by pixel level. Required plots are meaningful and results are well displayed. For future projects it could be interesting to extract some conclusions at this stage of the EDA.

Model Building & Training

- Students create a set of training data and a set of validation data that each have the appropriate proportions of positive and negative cases for their intended use (training and validation)

The dataset is correctly balanced and the training, validation, and test dataset are well Split.

```
train_data, val_data = skl.train_test_split(vargs,
                                test_size = 0.2,
                                stratify = vargs['pneumonia_class'])


#Condition 1 - To have EQUAL amount of positive and negative cases of Pneumonia in Training¶
p_inds = train_data[train_data['pneumonia_class']=="1"].index.tolist()
np_inds = train_data[train_data['pneumonia_class']=="0"].index.tolist()

np_sample = sample(np_inds,len(p_inds))
train_data = train_data.loc[p_inds + np_sample]

#Condition 2 - To have positive cases of Pneumonia in the Test Set as same as in all_df

val_data = val_data.sample(int(len(train_data) / 100) * 20)

percent = int(round((all_df[all_df['pneumonia_class'] == "1"].shape[0] * 100) / all_df[all_df['pneumonia_class']
print("Percent: {0}".format(percent))
p_inds = val_data[val_data['pneumonia_class']=="1"].index.tolist()
np_inds = val_data[val_data['pneumonia_class']=="0"].index.tolist()

# The following code pulls a random sample of pneumonia data that's 1% as non-pneumonia sample.
p_sample = sample(p_inds,int(round(len(np_inds) / 100)))
val_data = val_data.loc[p_sample + np_inds]
```

- Student implements a class such as ImageDataGenerator from Keras to augment their training data only

- Student should not augment testing/validation data
- Student uses types of augmentation that are appropriate for medical imaging. There are no required types of augmentation
- Students should normalize the imaging data so the model weights do not go to infinity.

- Good job in with the augmentation here, for future project I will suggest you to implement other normalizations in order to check how they improve the algorithm Performance.

- Student monitors the training progress of their model using log loss
- Student changes training parameters to avoid overfitting and compares performances of different training paradigms.
- Student trains enough epochs until the loss is "stable"
- After training, student uses precision, recall, and F1 score to actually evaluate the utility of their model.
- Find a threshold to classify if an image is pneumonia or not.
- Students should show precision-recall curve and a curve of F1-score vs. threshold

Nice work monitoring the training process. A correct metric choice has been performed and the training results look correct.

```
Epoch 1/30
130/130 [==============================] - 56s 428ms/step - loss: 0.7020 - binary_accuracy: 0.5362 - val_loss:
0.6869 - val_binary_accuracy: 0.5400
Epoch 2/30
130/130 [==============================] - 53s 405ms/step - loss: 0.6868 - binary_accuracy: 0.5599 - val_loss:
0.6943 - val_binary_accuracy: 0.5300
Epoch 3/30
130/130 [==============================] - 53s 410ms/step - loss: 0.6746 - binary_accuracy: 0.5870 - val_loss:
0.6256 - val_binary_accuracy: 0.6800
Epoch 4/30
130/130 [==============================] - 53s 407ms/step - loss: 0.6673 - binary_accuracy: 0.5942 - val_loss:
0.6947 - val_binary_accuracy: 0.5100
Epoch 5/30
130/130 [==============================] - 52s 403ms/step - loss: 0.6592 - binary_accuracy: 0.6159 - val_loss:
0.6401 - val_binary_accuracy: 0.6600
Epoch 6/30
130/130 [==============================] - 53s 406ms/step - loss: 0.6547 - binary_accuracy: 0.6261 - val_loss:
0.6296 - val_binary_accuracy: 0.6300
Epoch 7/30
```

- Student can check DICOM header for image position, image type and body part on ALL .dcm files to check validity for their model using the pydicom python package.
- Student can read imaging data in from a .dcm file, preprocess the image and feed it into their model using the pydicom python package.

DICOM header parameters are correctly checked and taken into account at inference level.
Well done👍

FDA Description and Validation Plan

- **Student should provide an intended use statement**
- **Student should point to data from their EDA to describe who their algorithm is indicated for and what the clinical setting is in which their algorithm would be used**
- **Student should describe limitations of their algorithm and how false positives or false negatives might affect a patient**

- Awesome work done here!
- You have indeed understood the aim of the project which made it easier for you to state the limitations and clinical impact of the performance👍
- You did an excellent job! The proper age ranges for use, as determined by the EDA studies, are described in the indications of use🎉
- The limitations of the performance, as well as its clinical significance, have been carefully considered.

- **Students provide a flowchart or architecture diagram of their model**
- **Students should describe the DICOM checks they use before sending an image through their algorithm**
- **Students should describe the preprocessing steps they use.**
- **Students should describe the architecture of the classifier**
- **Students should describe augmentation and its parameters used**
- **Students describe the parameters used for training**
- **Students should show the behavior of training and validating loss**
- **Students should describe the performance statistics and threshold used in final validation**

- Well done including the normalization applied in the preprocessing step.
- The flowchart, DICOM checks, types of augmentation utilized, fine-tuning the model, train and validation loss graphs, explaining how the threshold was determined, and the Precision-Recall curve were all done quite well.

- **Students should provide information for the training set**
- **Students should provide information for the validating set**
- **Students should describe how the ground truth of the NIH dataset is created, the benefit and limitations.**

The training and test sets are correctly characterized and described. Meaningful parameters such as balance and heterogeneity have been highlighted.

- Students should describe the ideal dataset that they would receive from a clinical partner for their FDA Validation Dataset
- Students should describe how they would ideally create ground truth for this FDA Validation Dataset
- Students should describe the performance metric and the metic value that they would hold their algorithm to, supported by literature

The ideal dataset for validation is correctly characterized and requirements for its integrity are set. Really good job on the overall FDA😃

Suggestions

Please have a look at this link to get an idea of how to prepare medical image data for a machine learning algorithm

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

START