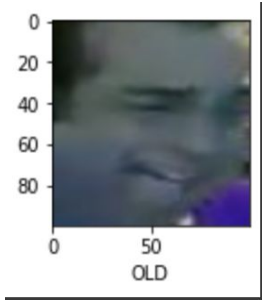
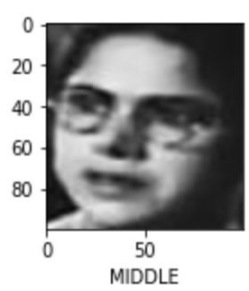
Facial Age Classification and Sentiment Analysis using Deep learning

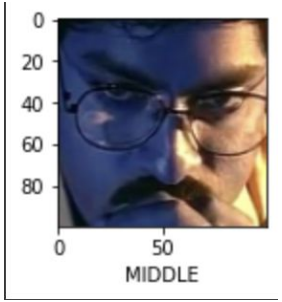
By- Group Number 5
Cheril Yogi (002790646)
Chinmay Dharwad (002771031)
Shubham Goyal (002747584)

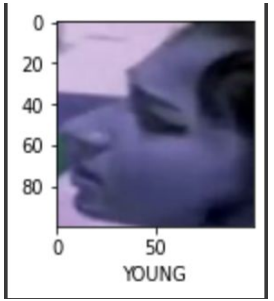
Facial age classification w/ Deep learning

- In this project we tried Deep learning and transfer learning for facial age classification of ~20K different images segmented in 3 age ranges
- The data had some level of imbalance in the three classes with ‘Middle’ age being the most populous
- We leveraged CNN with 2 dense layers, and Resnet50 (pretrained) models in this analysis



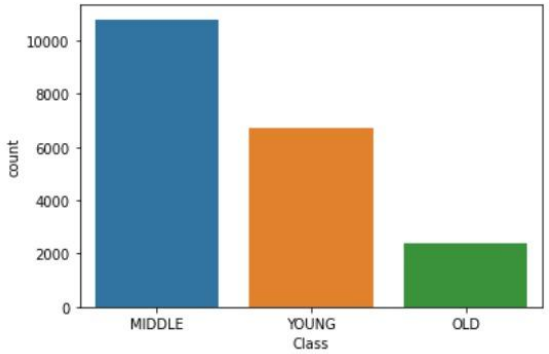






Pre-processing and Modeling

- The data required basic preprocessing like resizing, scaling and structuring into arrays to qualify for training
- We dropped the sizes of images to 150 and 100 respectively, as the process was computationally extensive
- We used stratified splitting to overcome class imbalance to a certain extent



Results and Model Comparisons

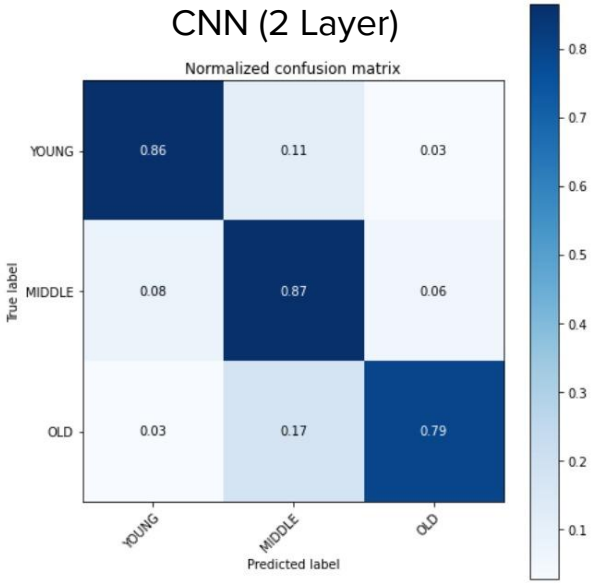
1) With CNN model we were able to achieve a significantly higher accuracy on the data with a difference of ~30%.

Models	**Train Accuracy**	**Test Accuracy**
CNN Model (2 dense Layers)	86.98%	85.62%
ResNet50 (Pretrained)	57.16%	53%

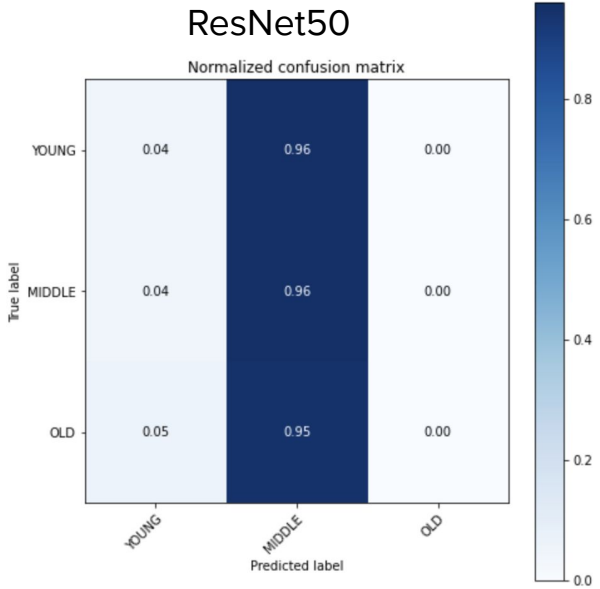
Results and Model Comparisons

2) One major factor that we observed in down-performance of Resnet50 was the class imbalance due to which the correct predictions for classes like ‘Old’ and ‘Young’ were minimal.

CNN (2 Layer)



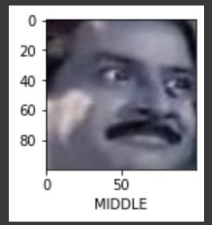
ResNet50



Results and Model Comparisons

- Reviewing some pictures that were labeled by ResNet50

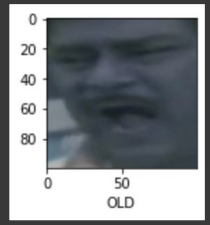
```
plot_sample(X_test, y_true,11)
```




```
[96] class_names[y_pred_classes[11]]

'MIDDLE'
```

```
plot_sample(X_test, y_true,1031)
```




```
[90] class_names[y_pred_classes[1031]]

'MIDDLE'
```

- Further study, we can try to curb the class imbalance with data augmentation and subsampling of majority classes

Sentiment Analysis : Data Scraping

- For the Naive Bayes Classifier we have used the review's data of Apple Airpods (2nd Gen.) from Amazon.
- For HTML document's parsing , beautifulsoup package is used along with requests_html library to make parsing simple and intuitive.
- There was high imbalance in the reviews with 87% positive response and 13% negative response.
- A variable named star_filter is used to achieve balanced datasheet, which led us to scrape one star rating at at time , and so we chose equal number of pages for each rating and mapped out "rating", "title" and "body" of the reviews using Pagination. And then tagged the reviews as 0 : negative, 1 : positive.

Sentiment Analysis : Naive Bayes Classifier

- While preprocessing the data, it was found that the “body” of the reviews were unusually large text while the “title” column is more precise and used better descriptive words.
- Bag of words CountVectorizer is used to convert the words present in the corpus into vector form , on the basis of the frequency (count) of each word that occur in the entire text.
- In this analysis we used Multinomial Naive Bayes Algorithm. The classifier algorithm guesses the tag of a text using Bayes theorem and calculates each tag’s likelihood for the given sample and outputs the tag with the greatest chance.

Sentiment Analysis : Conclusion

- The confusion matrix:

Original_Cateogry	0	1
final_category		
0	280	24
1	6	590

- The accuracy of the model turned out to be 96.6%.

```
print("accuracy is",correct/predict_df.shape[0])
```

```
accuracy is 0.9666666666666667
```

Sentiment Analysis: Conclusion with Imbalanced Datasheet

- The confusion matrix:

Original_Cateogry	0	1
final_category		
0	831	60
1	50	566

- The new accuracy turned out to be 92.7%

```
print("accuracy is",correct/predict_df.shape[0])
```

```
accuracy is 0.927007299270073
```

Thank You !!