Flower Classification with TPUs

Use TPUs to classify 104 types of flowers

Presented by:
Apurva Modi
01122493

The Challenge

In this competition, we’re challenged to build a machine learning model that identifies the type of flowers in a dataset of images (for simplicity, they sticked to 104 types).

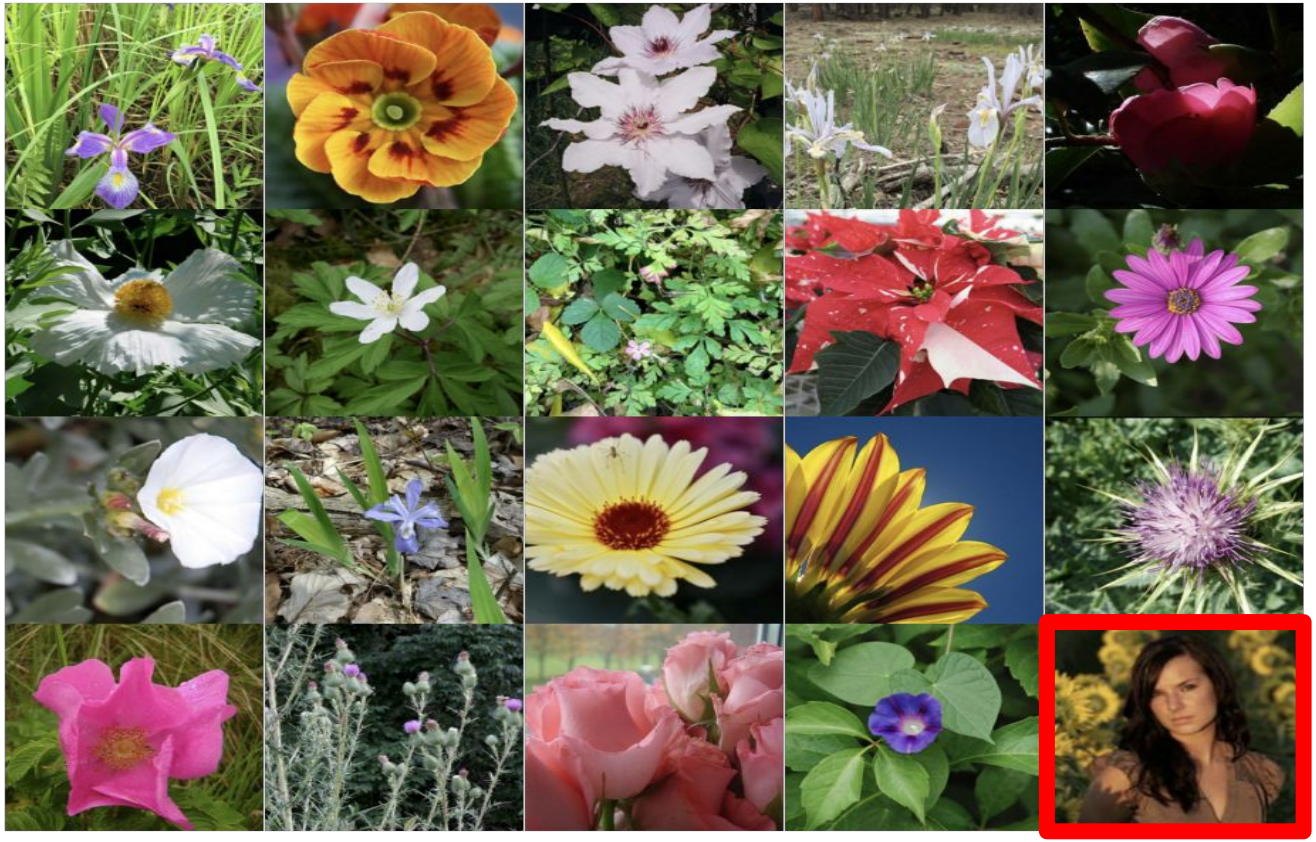
What was provided?

A starter notebook with TPU starter code and how to use model(Vgg16 , DenseNet201 and Xception) for training, validation and testing.

Training : 12753
Validation: 3712
Testing: 7382

Using 512 X 512

Dataset and anomalies



Their Approach - Vgg16

Training : 12753
Validation: 3712
Testing: 7382

512 X 512

Data Augmentation: random flips left right

Used transfer learning on keras sequential model with imagenet weights.

With step size of 100 for 80 Epochs
F1- score of 0.246

Initial Approach - Vgg16

Training : 12753
Validation: 3712
Testing: 7382

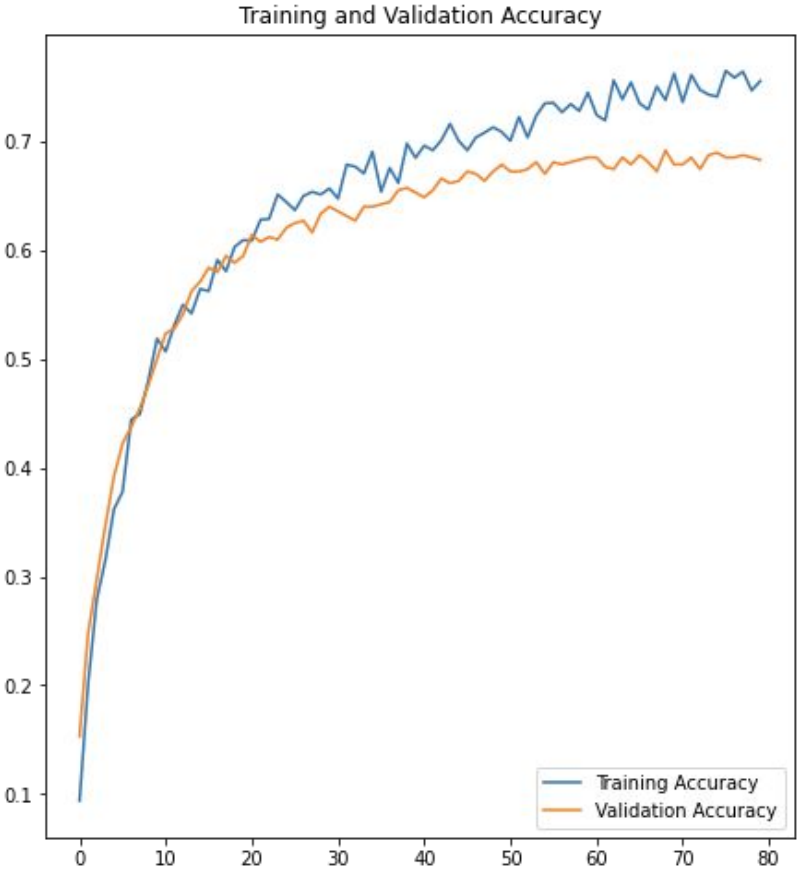
192 X 192

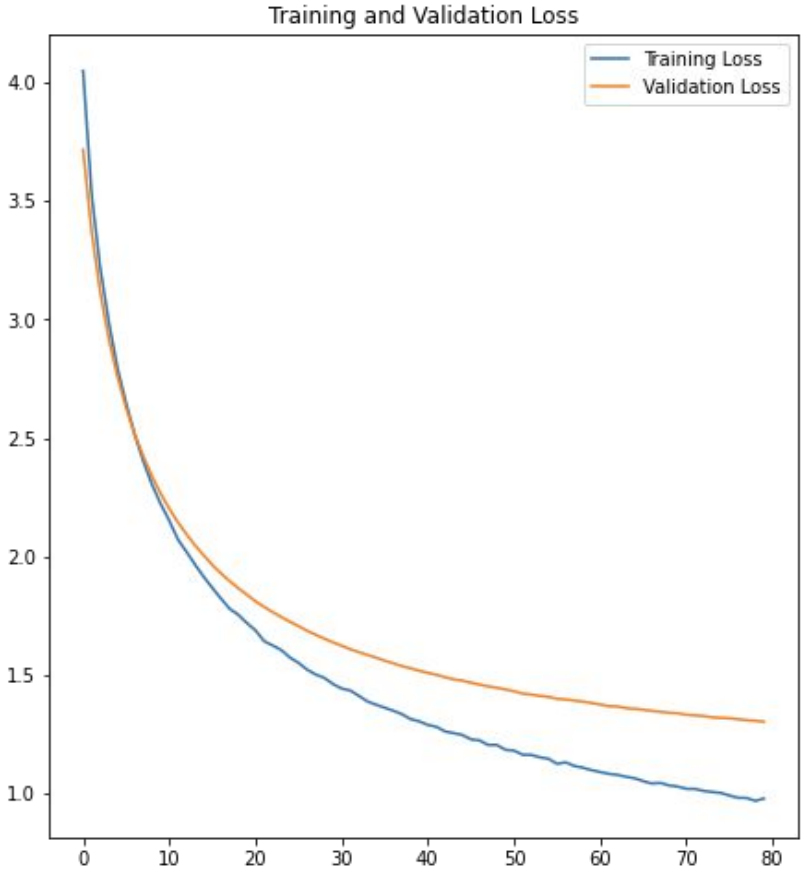
Data Augmentation: random flips left right up down and contrast.

Used transfer learning on keras sequential model with imagenet weights.

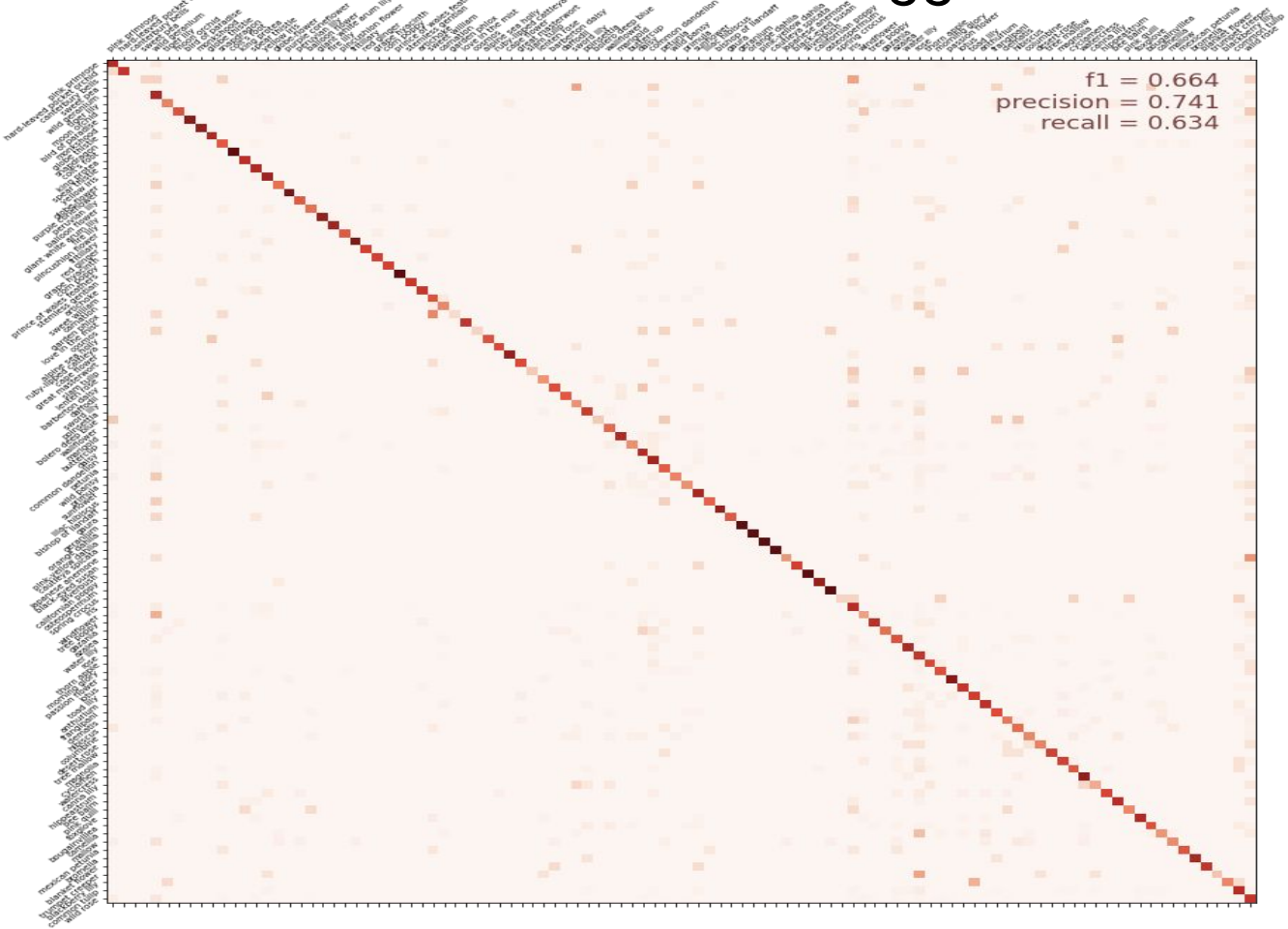
With step size of 100 for 80 Epochs
F1- score of 0.666
Kaggle rank - 667
Kaggle score 0.65409

Vgg16 with imagenet as weights

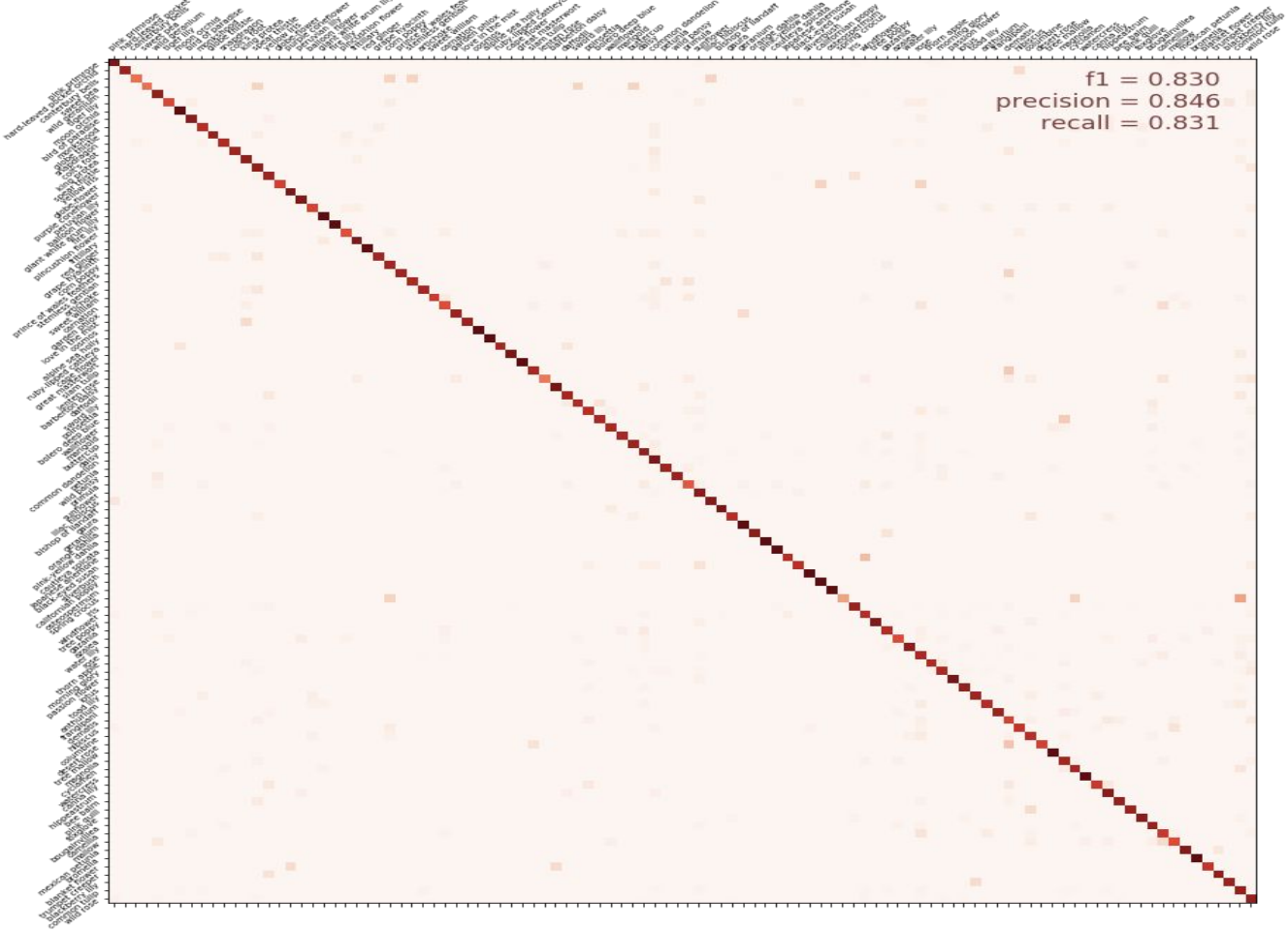




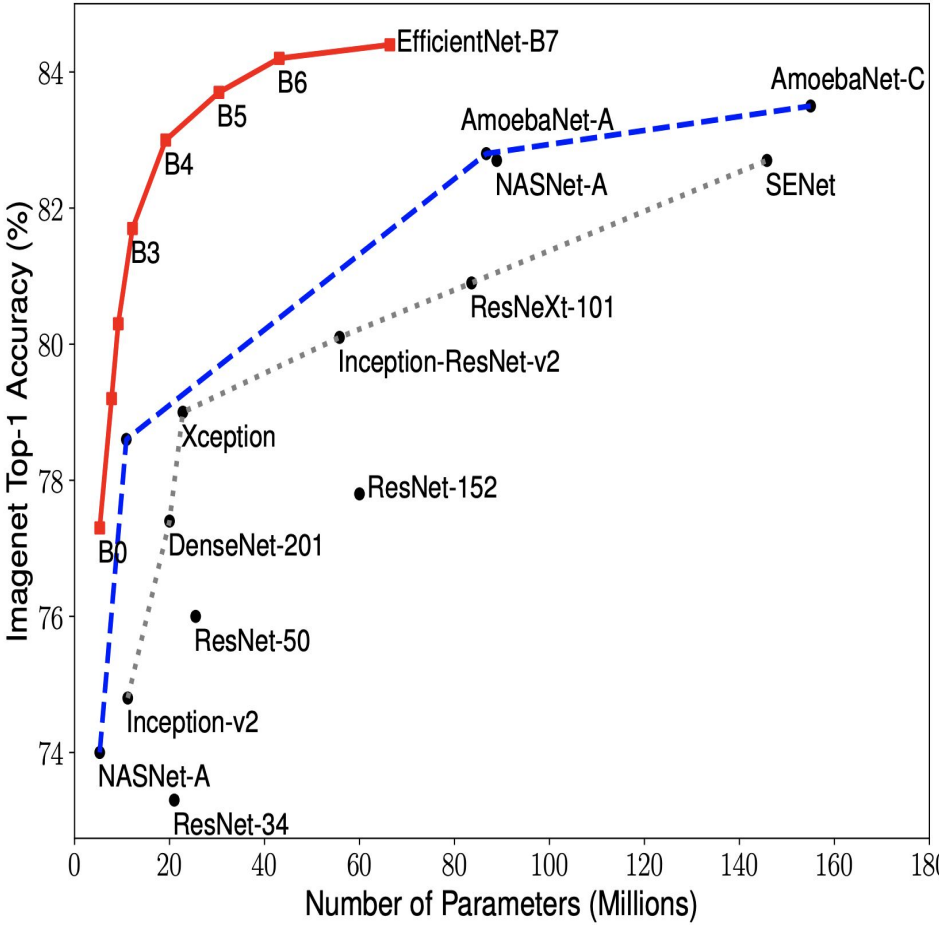
Confusion Matrix for Vgg16

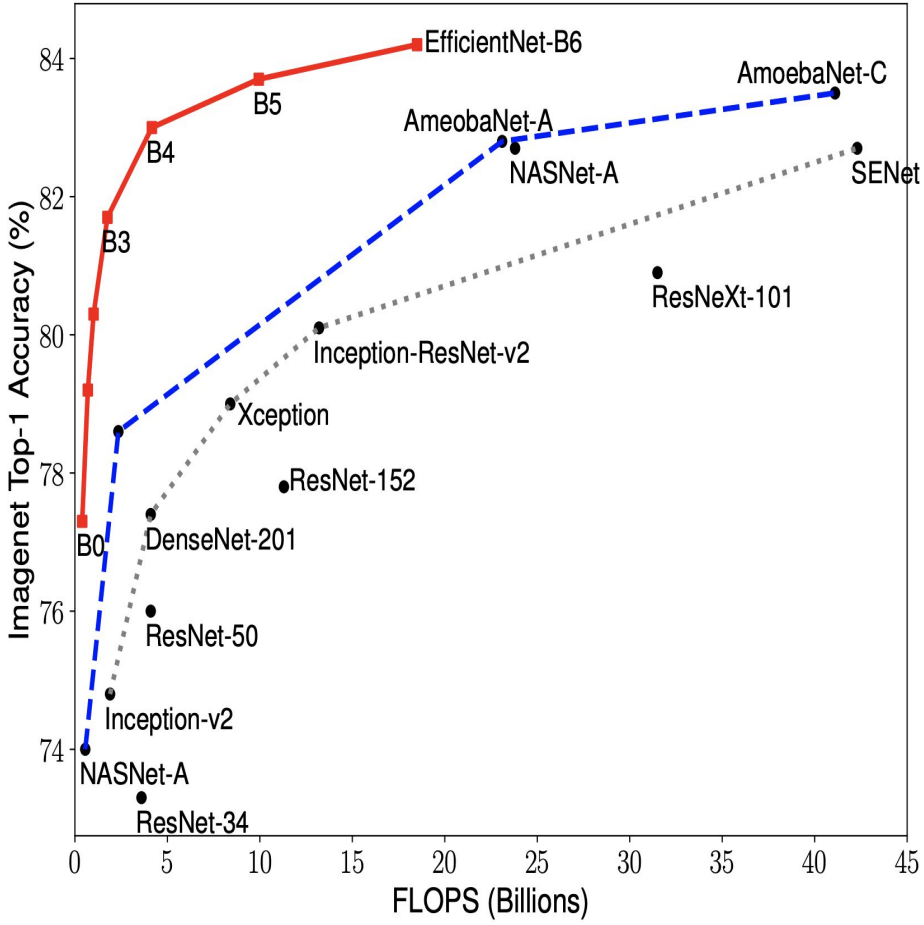


Confusion Matrix for Improved Vgg16



State of the art Model





State of the art model - EfficientNet (noisy-student)

Training : 12753
Validation: 3712
Testing: 7382

Dataset : 512 x 512

Data Augmentation: random flips left right

Used transfer learning on keras sequential model with imagenet weights.

Training Accuracy - 0.955
With step size of 100 for 21 Epochs

Ensemble: EfficientNet (noisy-student) + DenseNet201

Training : 16465
Testing: 7382

Dataset : 512 x 512

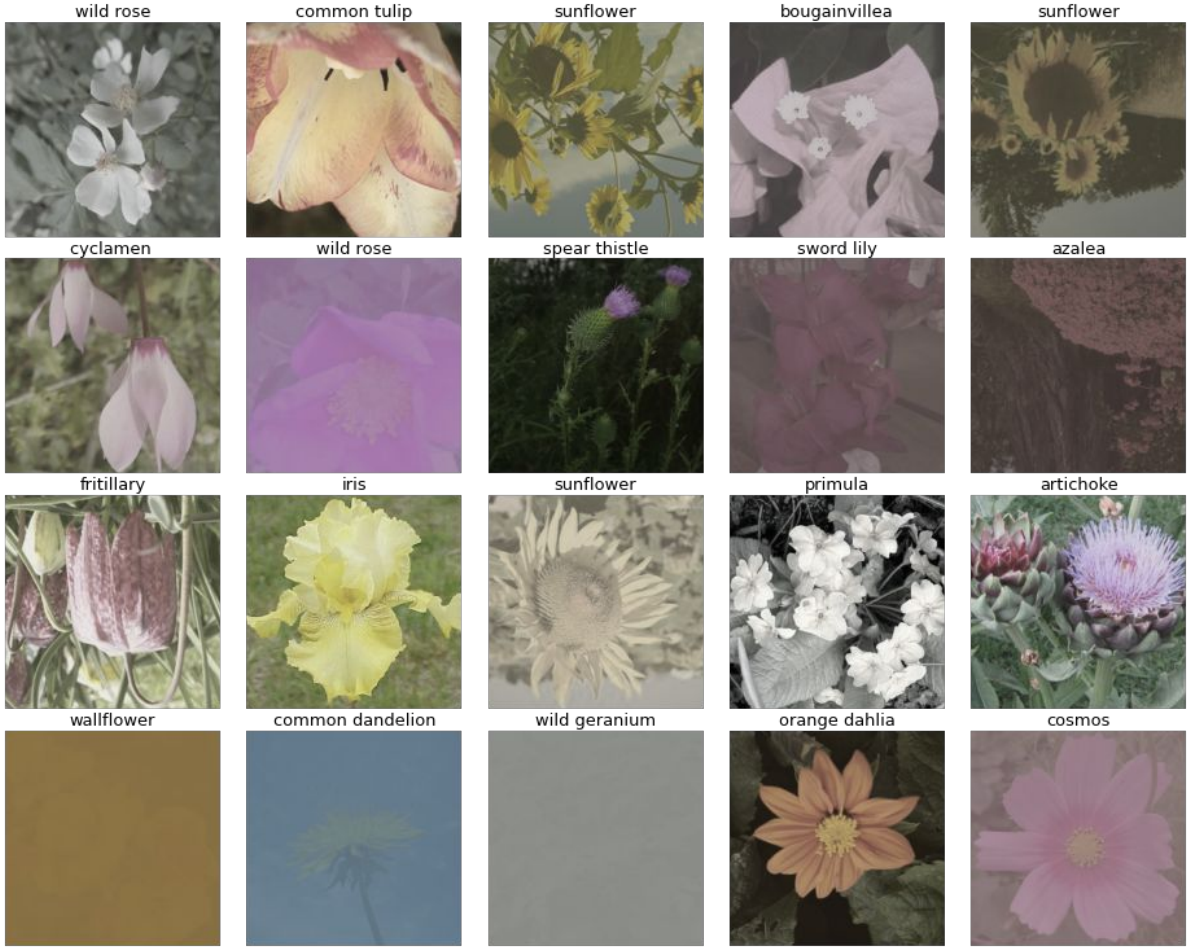
Data Augmentation: random flips left right

Used transfer learning on keras sequential model with imagenet weights for DenseNet201 and EfficientNetB7

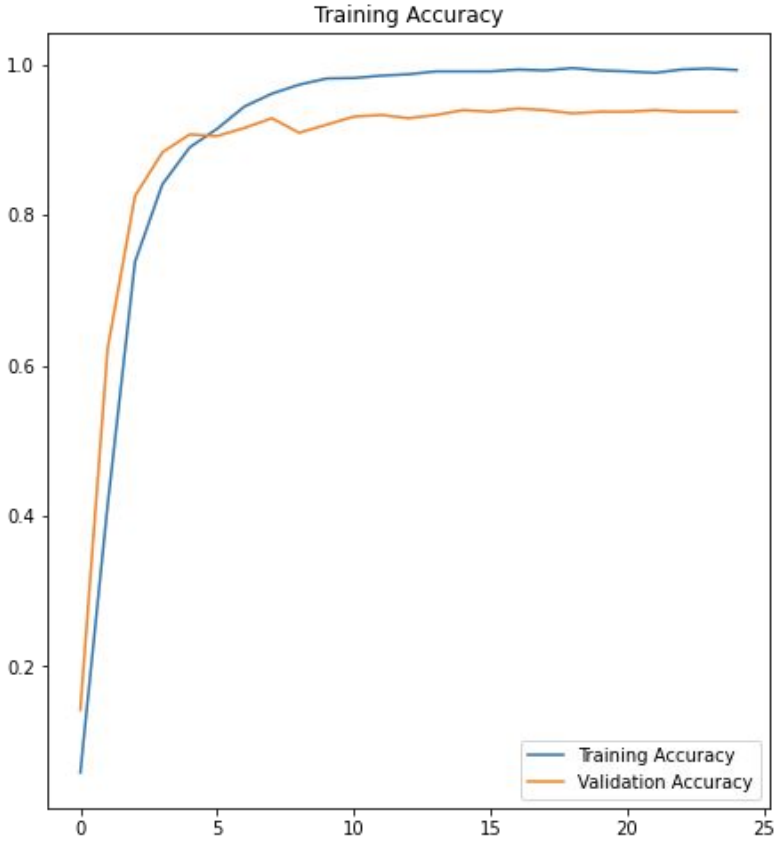
With step size of 100 for 25 Epochs

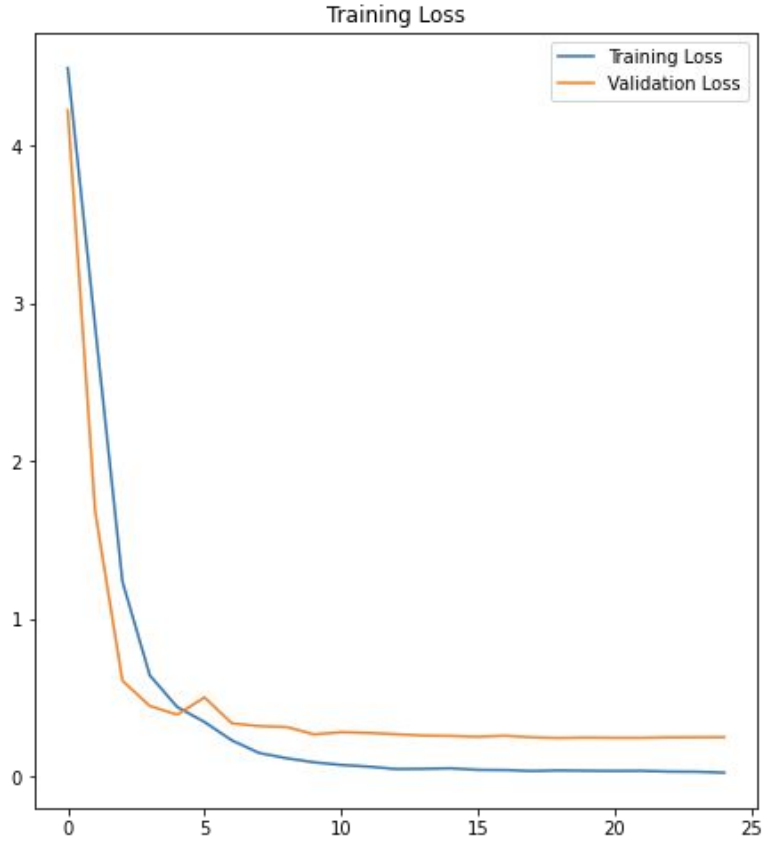
Training Accuracy - 0.9981
Training Loss - 0.0253

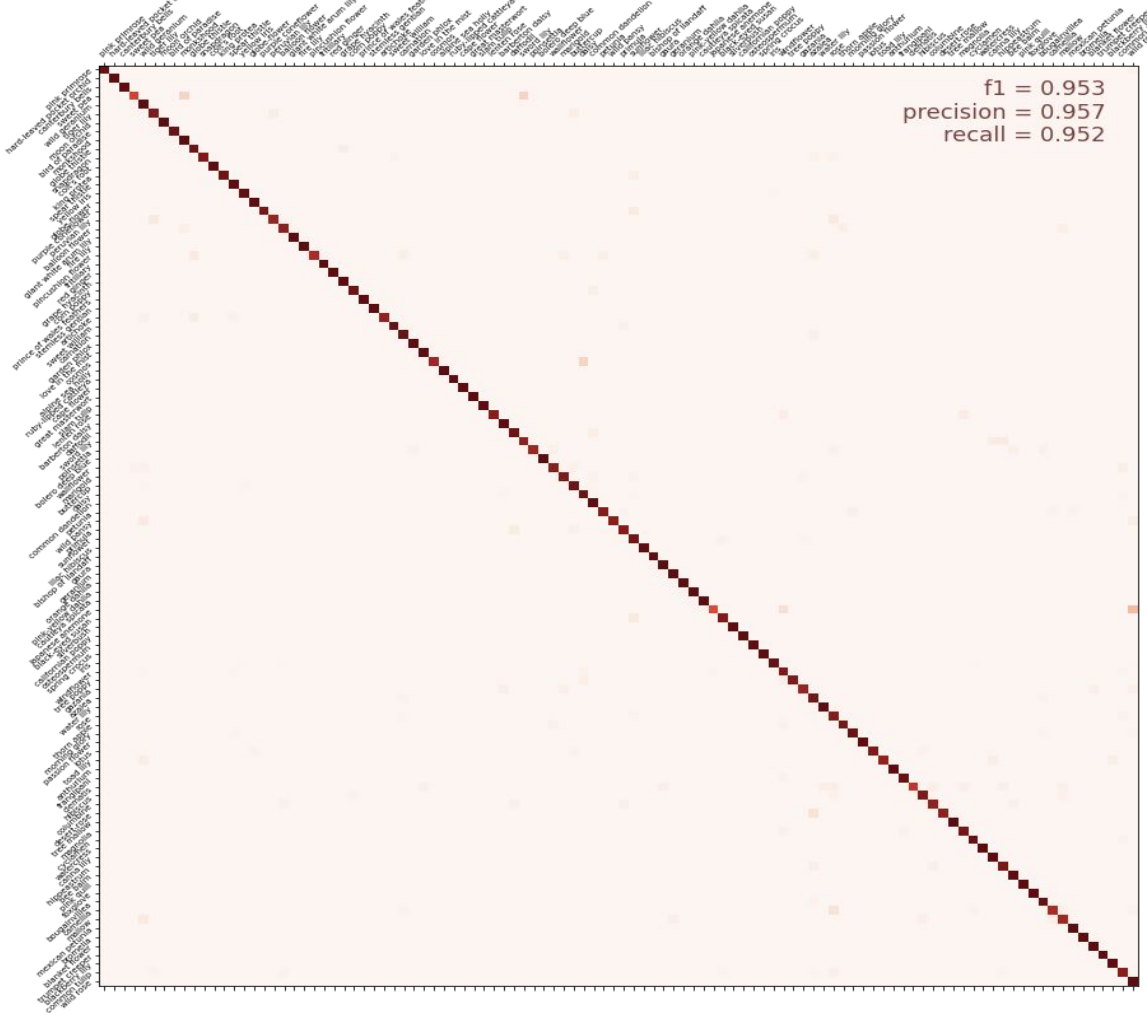
Data Augmentation: random flips left right up down contrast and saturation.



Ensemble: with custom weights






f1 = 0.953
precision = 0.957
recall = 0.952

Result Analysis.

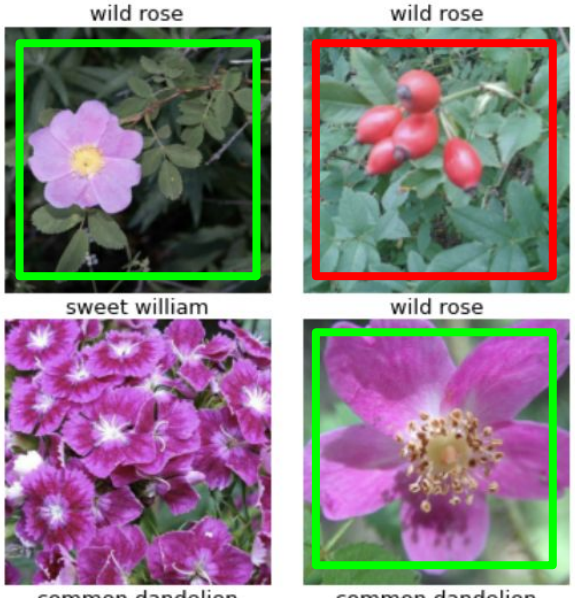
1. Image size of the training data matters if you are using a transfer learning as pre trained model are trained on similar size data.

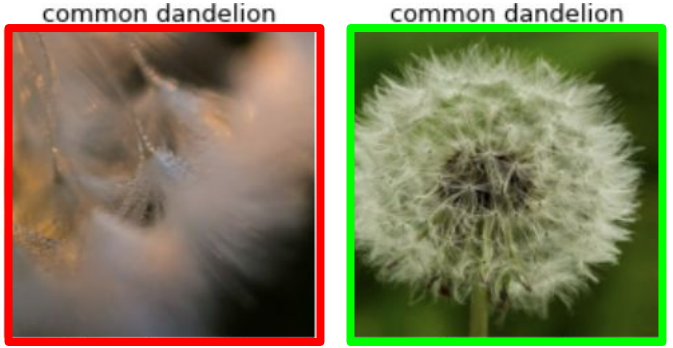
2. There were some underfit classes for the dataset which were resulting in mis prediction.

3. Finally, there were 11 classes which were mispredicting.

Future Works

1. Perform image segmentation to filter out flowers and to remove the background for some of the flower classes, which are mostly misclassified.
2. Calculate the best weight for the ensemble model.





Kaggle Submission

351	Apurva Modi		0.96011	7	~10s

Your Best Entry ↑

You advanced 73 places on the leaderboard!

Your submission scored 0.96011, which is an improvement of your previous score of 0.95569. Great job!

Tweet this!

403	Apurva Modi		0.95569	3	~10s

Your Best Entry ↑

You advanced 20 places on the leaderboard!

Your submission scored 0.95569, which is an improvement of your previous score of 0.95446. Great job!

Tweet this!

423	Apurva Modi		0.95446	2	now

Your Best Entry ↑

667	Apurva Modi		0.65409	1	1d

Your First Entry ↑