01

FINAL PROJECT

Ethan Baksh, Vaibhav Bhatla, Nicholas Friar, Divya Kartik, Sameer Narendran

INTRODUCTION

- Our Project was the “Car Classification” task where we had to use transfer learning to solve our problem
- This involved classifying 196 classes of cars from 16,185 different images
- We have tested multiple models ranging from Efficient Net B4 to InceptionV3.
- These models were taken from TensorFlow Keras Pre-defined Models, and one model was created on our own
- We used these models as a base and then added dense layers for classification, and applied regularization techniques

NETWORK ARCHITECTURE

Base Model: ResNet152V2

Global Average Pooling 2D layer (instead of Flatten)

Dense layer (with 2048 outputs and ReLu activation

Dense layer (with 2048 outputs and ReLu Activation)

Batch Normalization layer

Dropout layer (rate = 0.6)

Dense layer (with 196 outputs and Softmax activation)

```
resnet = ResNet152V2(weights="imagenet", include_top=False, input_shape = (224, 224, 3))
x = (resnet.output)
x = GlobalAveragePooling2D()(x)
x = Dense(2048, activation='relu')(x)
x = Dense(2048, activation='relu')(x)
x = BatchNormalization()(x)
x = Dropout(0.6)(x)
predictions = Dense(196, activation='softmax')(x)

model = Model(inputs=resnet.input, outputs=predictions)

for layer in model.layers[:30]:
   layer.trainable=False
for layer in model.layers[30:]:
   layer.trainable=True
```

MODEL TESTING

Model	Training Accuracy	Validation Accuracy
VGG16	0.08	0.06
VGG19	0.0204	0.0173
InceptionV3	0.7641	0.7643
ResNet50V2	0.8804	0.7726
MobileNetV2	0.8645	0.7476
ResNet152V2	0.9198	0.8350
EfficientNet b7	0.1125	0.0903

we tested the performance of a number of different models and decided that **ResNet152V2** was the best one

HYPERPARAMETERS

DATA TRANSFORMATION

```
rescale=1./255,
width_shift_range=0.1,
height_shift_range=0.1,
horizontal_flip=True,
vertical_flip=True,
zoom_range=0.2
```

EPOCHS

```
epochs: 50
batch_size: 32
```

LEARNING RATE

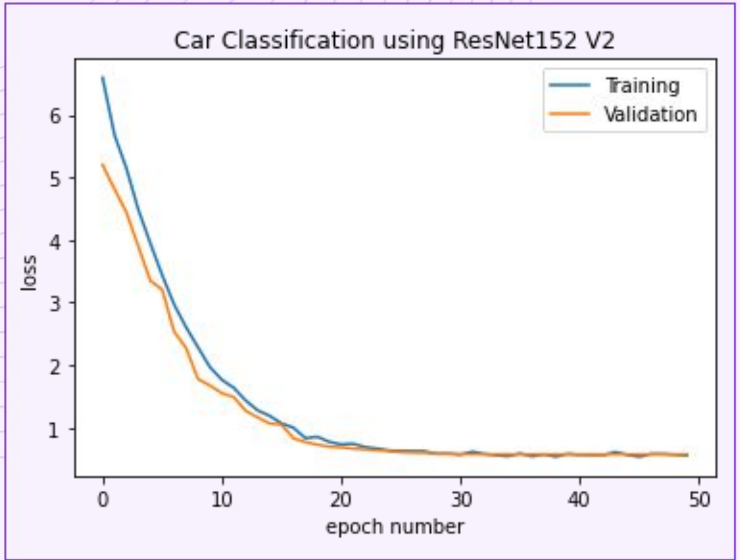
```
lr = 0.0001

ReduceLROnPlateau
('val_accuracy',
factor = 0.1,
patience = 1)
```

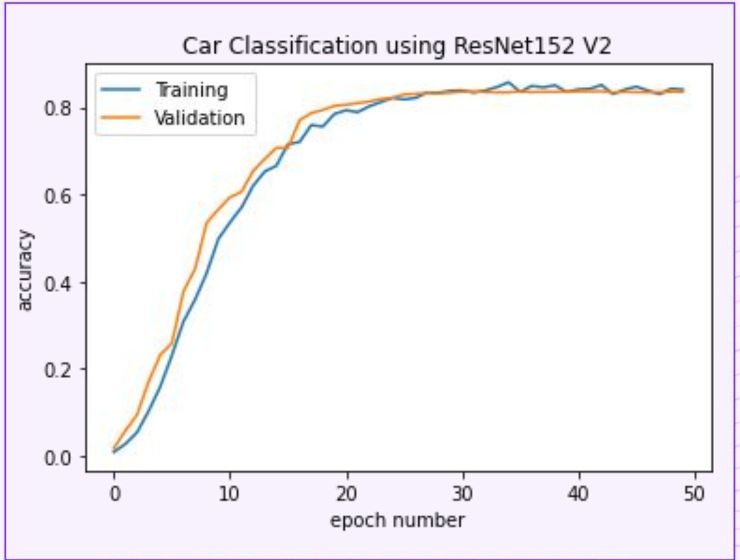
OPTIMISER

```
optimizer =
Adam(lr=0.0001)
```

MODEL PERFORMANCE



LOSS



ACCURACY

Training Accuracy: 91.98%
Validation Accuracy: 83.50%

CLASS ACCURACY

Class	Accuracy
1: Acura RL Sedan 2012	0.9286
2: Acura TL Sedan 2012	0.7143
3: Acura TL Type-S 2008	0.8667
4: Acura TSX Sedan 2012	0.7500
5: Acura Integra Type R 2001	0.5000
6: Acura ZDX Hatchback 2012	0.8824
7: Aston Martin V8 Vantage Convertible 2012	0.8462
8: Aston Martin V8 Vantage Coupe 2012	0.6500
9: Aston Martin Virage Convertible 2012	0.7059
10: Aston Martin Virage Coupe 2012	0.6667

CHALLENGES

- The dataset had trouble downloading onto Google Colab and made the computer very slow
- For the first few hours the accuracy was stuck at 0.3 to 0.5 percent (VGG19)
- After another hour, the accuracy went up to 2 to 3 percent (model created from scratch)
- The problem was resolved when the accuracy started climbing to 25 and 50 percent accuracy with the usage of different models.
- We had to test an assortment of models, which was troubling, because our computers could not handle the immense workload, therefore, making the process tedious and time-consuming

SOURCES

- 3D Object Representations for Fine-Grained Categorization. Jonathan Krause, Michael Stark, Jia Deng, Li Fei-Fei4th IEEE Workshop on 3D Representation and Recognition, at ICCV 2013 (3dRR-13). Sydney, Australia. Dec. 8, 2013.[
- Identity Mappings in Deep Residual Networks] (https://arxiv.org/abs/1603.05027) (CVPR 2016)
- Brownlee, Jason. “Transfer Learning in Keras with Computer Vision Models.” Machine Learning Mastery, 2 Sept. 2019, machinelearningmastery.com/how-to-use- transfer-learning-when-developing-convolutional-neural-network-models/.