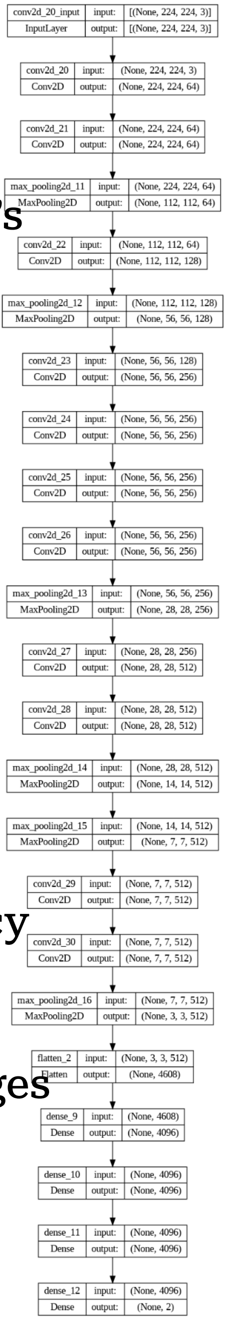
REPORT

- **DATA COLLECTION :**
  - Collected Data From Kaggle Resume Data for Resume Image’s and For Non Resume Image’s from Google ,Applications etc
  - Total no .of imag*es in dataset are 138*
- **DATA Prepocesing :**
  - *For Data preprocessing keras.preprocessing.image Image generator is used*
- **Classification of Resume Classification**
  - Binary Classification
  - CLASS : 2
  - OPTIMIZER : ADAM
  - LOSS FUNCTION : binary_crossentropy
  - METRIC : ACCURACY
- **MODEL SELECTION:**
  - First I have Made Custom Model then I archived 57.8 accuracy
  - Later On I have Used VGG16 , MobileNet then I archived 68.2 accuracy
  - By Tuning VGG16 I have Archived 81.62 Accuracy ,The Changes made
  - By Changing learning Rate , LossFunction ,Batch_Size .. ,We have Reached Optimium Accuracy

MODEL


conv2d_20_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
conv2d_20 | input: | (None, 224, 224, 3)
Conv2D | output: | (None, 224, 224, 64)
conv2d_21 | input: | (None, 224, 224, 64)
Conv2D | output: | (None, 224, 224, 64)
max_pooling2d_11 | input: | (None, 224, 224, 64)
MaxPooling2D | output: | (None, 112, 112, 64)
conv2d_22 | input: | (None, 112, 112, 64)
Conv2D | output: | (None, 112, 112, 128)
max_pooling2d_12 | input: | (None, 112, 112, 128)
MaxPooling2D | output: | (None, 56, 56, 128)
conv2d_23 | input: | (None, 56, 56, 128)
Conv2D | output: | (None, 56, 56, 256)
conv2d_24 | input: | (None, 56, 56, 256)
Conv2D | output: | (None, 56, 56, 256)
conv2d_25 | input: | (None, 56, 56, 256)
Conv2D | output: | (None, 56, 56, 256)
conv2d_26 | input: | (None, 56, 56, 256)
Conv2D | output: | (None, 56, 56, 256)
max_pooling2d_13 | input: | (None, 56, 56, 256)
MaxPooling2D | output: | (None, 28, 28, 256)
conv2d_27 | input: | (None, 28, 28, 256)
Conv2D | output: | (None, 28, 28, 512)
conv2d_28 | input: | (None, 28, 28, 512)
Conv2D | output: | (None, 28, 28, 512)
max_pooling2d_14 | input: | (None, 28, 28, 512)
MaxPooling2D | output: | (None, 14, 14, 512)
max_pooling2d_15 | input: | (None, 14, 14, 512)
MaxPooling2D | output: | (None, 7, 7, 512)
conv2d_29 | input: | (None, 7, 7, 512)
Conv2D | output: | (None, 7, 7, 512)
conv2d_30 | input: | (None, 7, 7, 512)
Conv2D | output: | (None, 7, 7, 512)
max_pooling2d_16 | input: | (None, 7, 7, 512)
MaxPooling2D | output: | (None, 3, 3, 512)
flatten_2 | input: | (None, 3, 3, 512)
Flatten | output: | (None, 4608)
dense_9 | input: | (None, 4608)
Dense | output: | (None, 4096)
dense_10 | input: | (None, 4096)
Dense | output: | (None, 4096)
dense_11 | input: | (None, 4096)
Dense | output: | (None, 4096)
dense_12 | input: | (None, 4096)
Dense | output: | (None, 2)

```
model = Sequential()
model.add(Conv2D(input_shape=(224,224,3),filters=64,kernel_size=(3,3),padding="same", activation="relu"))
model.add(Conv2D(filters=64,kernel_size=(3,3),padding="same", activation="relu"))
model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))
model.add(Conv2D(filters=128, kernel_size=(3,3), padding="same", activation="relu"))

model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu"))
model.add(Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu"))

model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))
model.add(Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu"))
model.add(Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu"))

model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))
model.add(Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu"))
model.add(Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu"))

model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))
model.add(Flatten())
model.add(Dense(units=4096,activation="relu"))
model.add(Dense(units=4096,activation="relu"))
model.add(Dense(units=4096,activation="relu"))

model.add(Dense(units=2, activation="softmax"))
```

Results :

Loss Function	Optimizer	Learning Rate	Batch Size	Epoch	Accuracy
Categorical Cross Entropy	Adam	0.00001	16	50	71.88
Categorical Cross Entropy	Adam	0.0001	8	50	62.5
Categorical Cross Entropy	Adam	0.000001	8	50	69.08
Categorical Cross Entropy	Adam	0.000001	16	50	70.65
BinaryCross Entropy	Adam	0.00001	16	50	59.88
BinaryCross Entropy	Adam	0.0001	8	50	81.25
BinaryCross Entropy	Adam	0.000001	8	50	77.78