AI PROJECT

INDIAN SIGN LANGUAGE DETECTOR

TEAM ID: PTID-AI-SEP22-1015

PROJECT ID: PRAICP-1000-IndiSignLang

INTRODUCTION & HOW DOES SIGN LANGUAGE WORK?

\* People who are hearing impaired communication with the help of gesture-energetic movements of the hand accompanied by a living facial expression

\* This language is also used at places where listening is not possible for ex... under water drivers, asutronuts….

**INDAIN SIGN LANGUAGE:**



**DATA SUMMARY:**

This dataset contain total 4972 jpg images in 24 classes A to y. as well as all classes data is imbalanced so we perform oversampling to balance the classes.

## TASK: MULTICLASS CLASSIFICATION

## WE DEVICE THIS PROJECT INTO MULTIPLE STEPS

* Make subset of training, validation and testing as well as perform oversampling to balance the all classes
* Prepare training, validation and testing set
* Get all classes labels
* Convert [Train, validation, Test] set into (X\_train,y\_train)(X\_valid,y\_valid),(X\_test,y\_test) and convert categorical lebel into binary metrix
* Visualise training images
* Used Resnet50, VGG16 & EfficientnetB3 pre-trained model
* Model Compilation
* Model Training
* Model Evaluation
* Model Saving
* Prediction on test data

LODING DATA / PREPARING DATA

* Make a subset of data into three parts train, test, and validation and balance the data of each class with the help of splifolder library.

DATA PROCESSING

[PREPARE TRAINING, TESTING, VALIDATION DATA]

* Found 3586 images belonging to 24 classes.
* Found 1390 images belonging to 24 classes.
* Found 1365 images belonging to 24 classes.

\* After perform oversampling 3586 samples for training, 1390 for validation and 1365 samples goes for testing

GET ALL CLASSES LABELS:

{'A': 0,'B': 1,'C': 2,'D': 3,'E': 4,'F': 5,'G': 6,'H': 7,'I': 8,

'K': 9,'L': 10,'M': 11,'N': 12,'O': 13,'P': 14,'Q': 15,'R': 16,

'S': 17,'T': 18,'U': 19,'V': 20,'W': 21,'X': 22,'Y': 23}

Split Training, Validation and Testing Data Into (X\_train,X\_test),

(X\_valid,y\_valid),(X\_test,y\_test) & Convert Labels Into Binary Matrix

Total 24 classes is present in this data we need to seprate the labels as well as encode the categorical integer into binary matrix

Visualise The Training Images:

****

**BUILD ARCHITECTURE OF RESNET50, VGG16, EFFICIENTNET-B3:**

RESNET50:

* In resnet50 model we train the existing weights to get a better

Result.

* Use dropout to supress overfitting problem as well as use batch

Normalization

* At last output layer use softmax activation function because the

Task is multiclass classification.

VGG16:

* In VGG16 we don’t train the existing weights
* Not use any dropout and batch normalization to check model

Performance

EFFICIENTNET-B3

* In this model also we train initial weights to get better result.
* Use Dropout and batch normalization to supress overfitting

Problem

* Add regularize at dense layer to reduce the losses

MODEL COMPILATION & TRAINING & EVALUATION:

RESNET50:

* Here we use categorical cross-entropy loss as well as Adamax

optimizer.

* Set learning rate and monitor validation loss and train Model on
* 300 epoch.
* validation loss is decreases after every epoch as well as validation accuracy is increases.

EVALUATION:

4/4 [==============================] - 1s 129ms/step - loss: 0.0067 - accuracy: 1.0000

Train loss & Accuracy: [0.00668429397046566, 1.0]

4/4 [==============================] - 1s 128ms/step - loss: 0.3905 - accuracy: 0.8828

Testing loss & Accuracy [0.3905099034309387, 0.8828125]

VGG16:

* Here we use categorical cross-entropy loss as well as Adamax

optimizer.

* Set learning rate and monitor validation loss and train Model on
* 300 epoch.
* VGG16 model perform well on training data but in

validation side accuracy is extreme less, as well as validation loss is also high

EVALUATION:

4/4 [==============================] - 1s 197ms/step - loss: 0.0771 - accuracy: 1.0000

VGG16 Training Loss & Accuracy [0.07713527977466583, 1.0]

4/4 [==============================] - 1s 198ms/step - loss: 1.7651 - accuracy: 0.4766

VGG16 Testing Loss & Accuracy [1.765055537223816, 0.4765625]

EFFICIENTNET-B3:

* Here we use categorical cross-entropy loss as well as Adamax

optimizer.

* Set learning rate and monitor validation loss and train Model on
* 300 epoch.
* VGG16 model perform well on training data but loss is very high or

validation side accuracy is around 75% , as well as validation loss is also high

EVALUATION:

4/4 [==============================] - 1s 149ms/step - loss: 3.4699 - accuracy: 1.0000

Training Loss & Accuracy [3.4699418544769287, 1.0]

4/4 [==============================] - 1s 147ms/step - loss: 4.0854 - accuracy: 0.9062

Testing Loss & Accuracy [4.085390090942383, 0.90625]

MODEL SAVING:

* Save model in training time with the help of model check point with

.h5 extension

PREDICTION & TESTING:

* Resnet50 model predict correctly on test image
* Not perform any testing on VGG16 because model not perform

Well.

* Efficientnet B3 mode do some wrong prediction because of loss is very high.

CONCLUSION:

* Resnet50 model perform well on training as well as testing data and both training and testing loss is less. as well as model predict correct result
* VGG16 model is not perform good on testing data
* EfficientnetB3 model training & Testing accuracy is good but training and testing loss is high, that's why model predict wrong output
* From above models we are select Resnet50 model

DEPLOY MODEL USING FLASK FRAMWORK

LIBRARY USED:

* Tensorflow
* Keras
* Matplotlib
* Glob
* Numpy
* Splifolder
* Resnet50
* VGG16
* Efficient-net B3

TOOL USED:

* Jupyter Notebook
* Pycharm