DRAWING MACHINE LEARNING CLASSIFIER

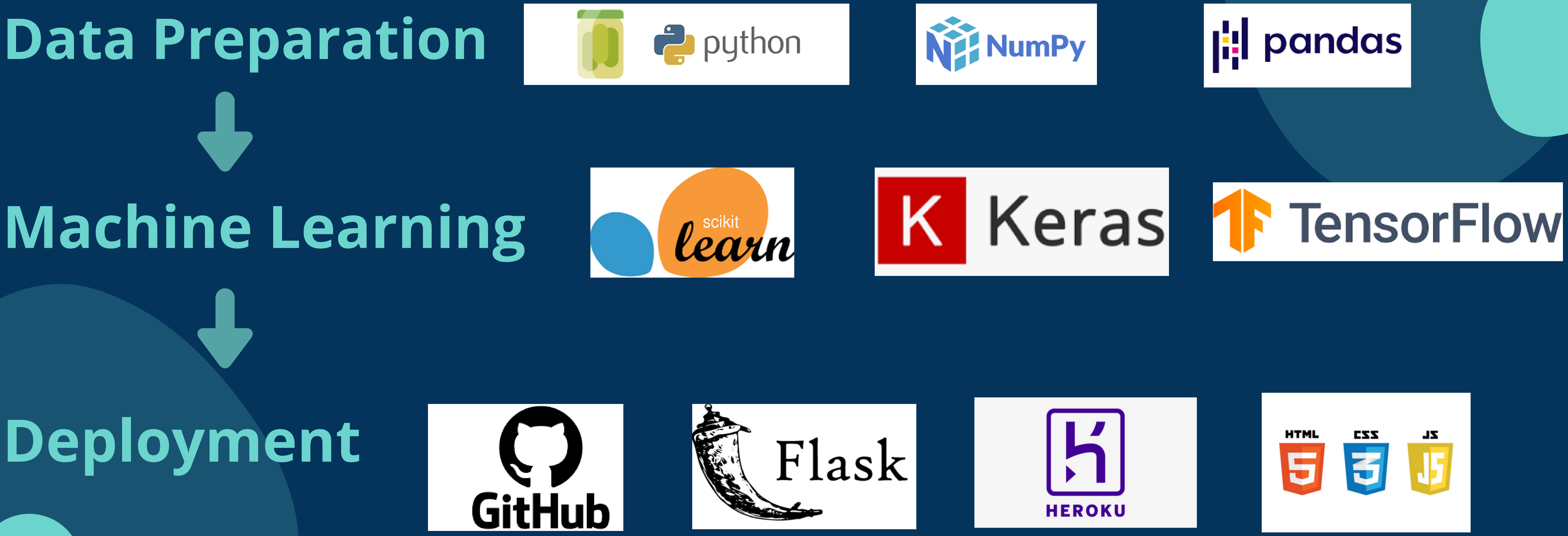
PRESENTED BY 4 MASKATEERS

ALANNA JAVIER, ANAND SHARAN, DONNA TIGRETT, & ELIZABETH KOEHLER

PROJECT OBJECTIVE:

To classify user drawings using a CNN machine learning classification model to predict which category of image best matches the drawing.

Project Workflow



Data Preparation

Machine Learning

Deployment

DATA PREPARATION STEPS

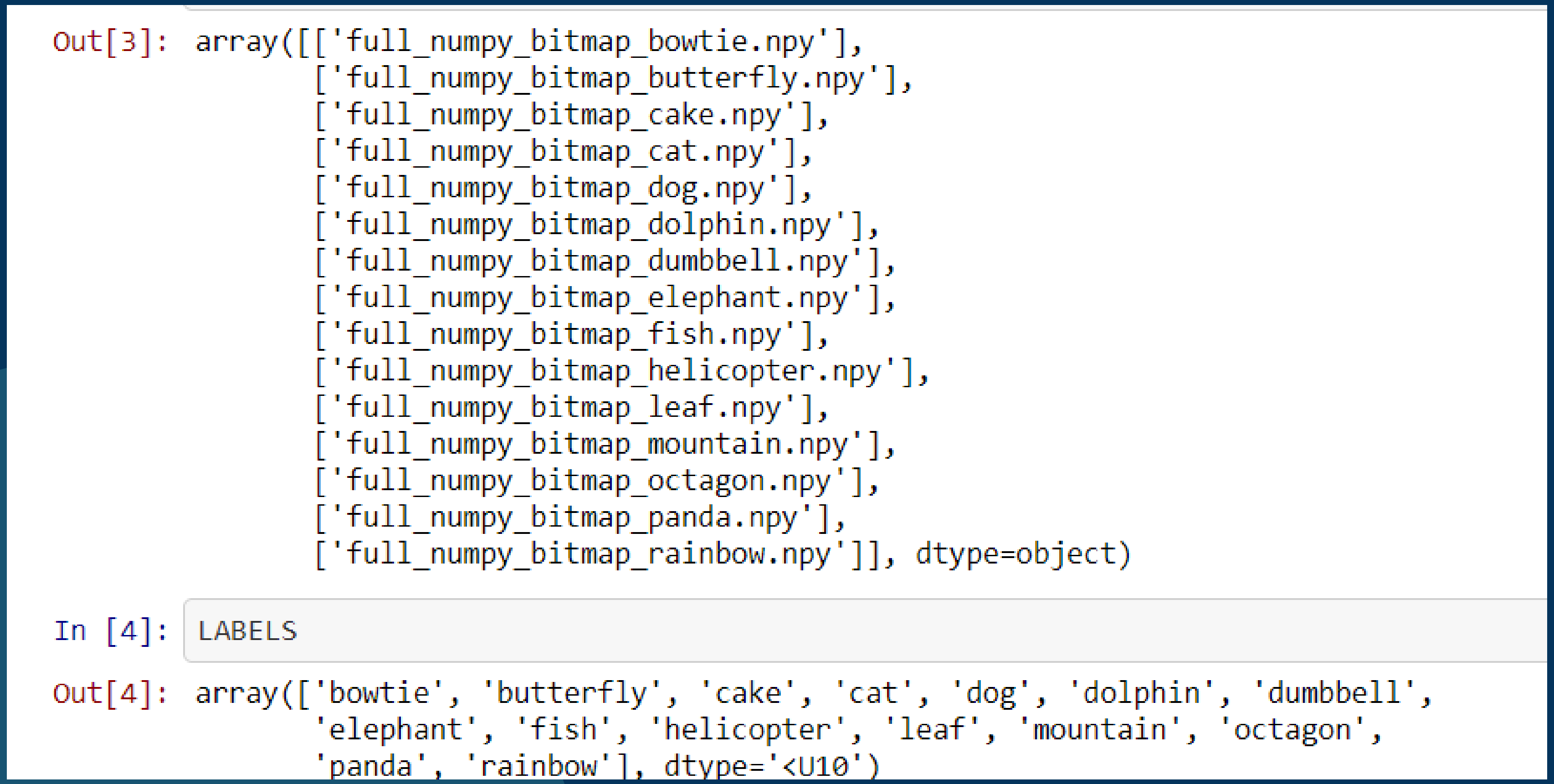
Data Source: Drawing classes in Numpy bitmap files

Quick, Draw! The Data

- Bowtie
- Butterfly
- Cake
- Cat
- Dog
- Dolphin
- Dumbbell
- Elephant
- Fish
- Helicopter
- Leaf
- Mountain
- Octagon
- Panda
- Rainbow

CREATE A PYTHON NUMPY ARRAY OF IMAGES

NUMPY Bitmap Files



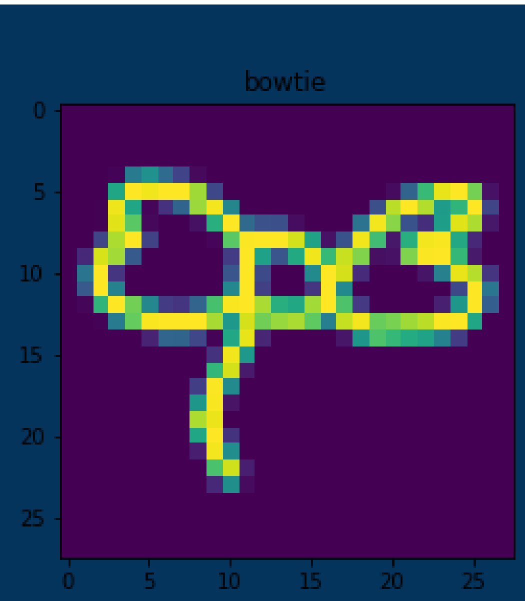
```
Out[3]: array([['full_numpy_bitmap_bowtie.npy'],
               ['full_numpy_bitmap_butterfly.npy'],
               ['full_numpy_bitmap_cake.npy'],
               ['full_numpy_bitmap_cat.npy'],
               ['full_numpy_bitmap_dog.npy'],
               ['full_numpy_bitmap_dolphin.npy'],
               ['full_numpy_bitmap_dumbbell.npy'],
               ['full_numpy_bitmap_elephant.npy'],
               ['full_numpy_bitmap_fish.npy'],
               ['full_numpy_bitmap_helicopter.npy'],
               ['full_numpy_bitmap_leaf.npy'],
               ['full_numpy_bitmap_mountain.npy'],
               ['full_numpy_bitmap_octagon.npy'],
               ['full_numpy_bitmap_panda.npy'],
               ['full_numpy_bitmap_rainbow.npy']], dtype=object)

In [4]: LABELS

Out[4]: array(['bowtie', 'butterfly', 'cake', 'cat', 'dog', 'dolphin', 'dumbbell',
               'elephant', 'fish', 'helicopter', 'leaf', 'mountain', 'octagon',
               'panda', 'rainbow'], dtype='<U10')
```

Source Image

Numpy Bitmap Image

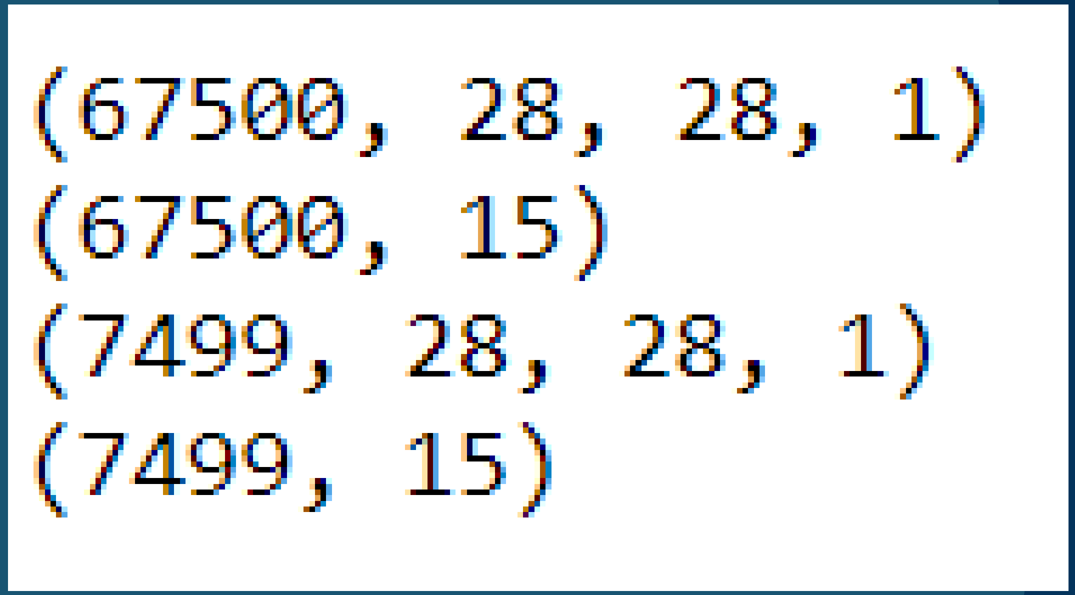


SPLIT THE IMAGES INTO TRAINING AND TESTING

90% of the images are used for training and 10% are used for testing

Dataset 75000 images
Train 67500 images
Test 7500 images

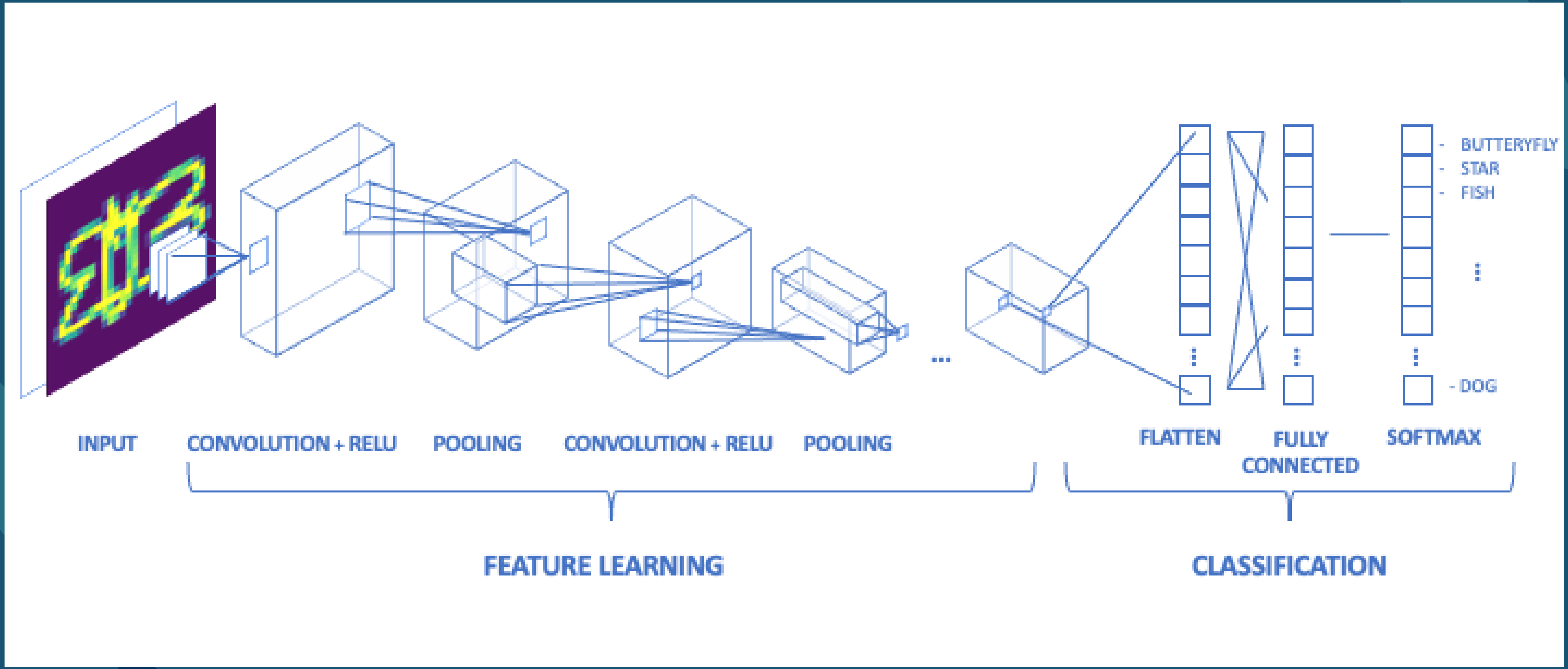
Train and Test image counts



```
(67500, 28, 28, 1)
(67500, 15)
(7499, 28, 28, 1)
(7499, 15)
```

MODEL CREATION AND TRAINING

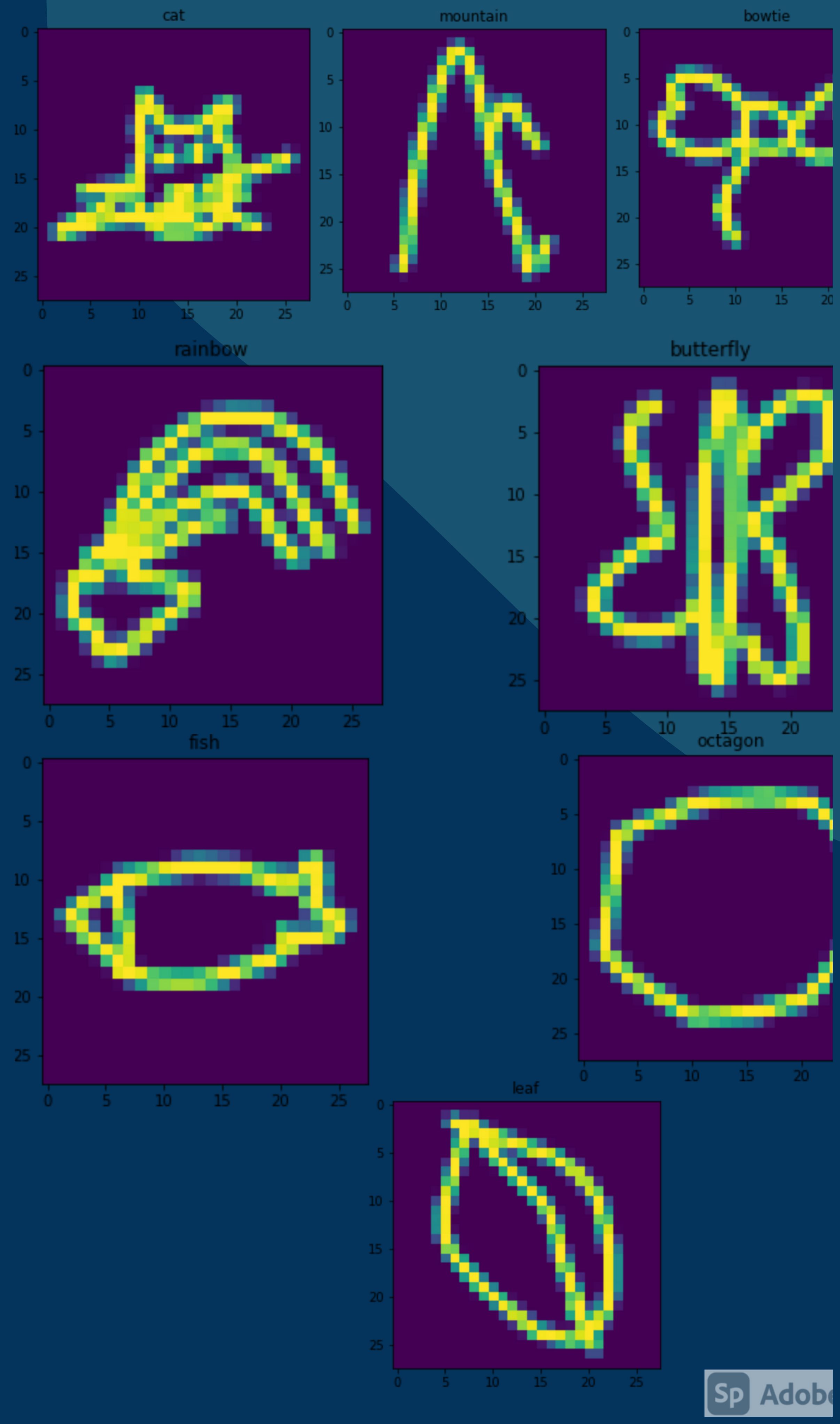
Convolutional Neural Network



MODEL IMAGES

Sample Training input images

1. Cat
2. Mountain
3. Bowtie
4. Rainbow
5. Butterfly
6. Fish
7. Octagon
8. Leaf



MODEL COMPILATION

Compile parameters

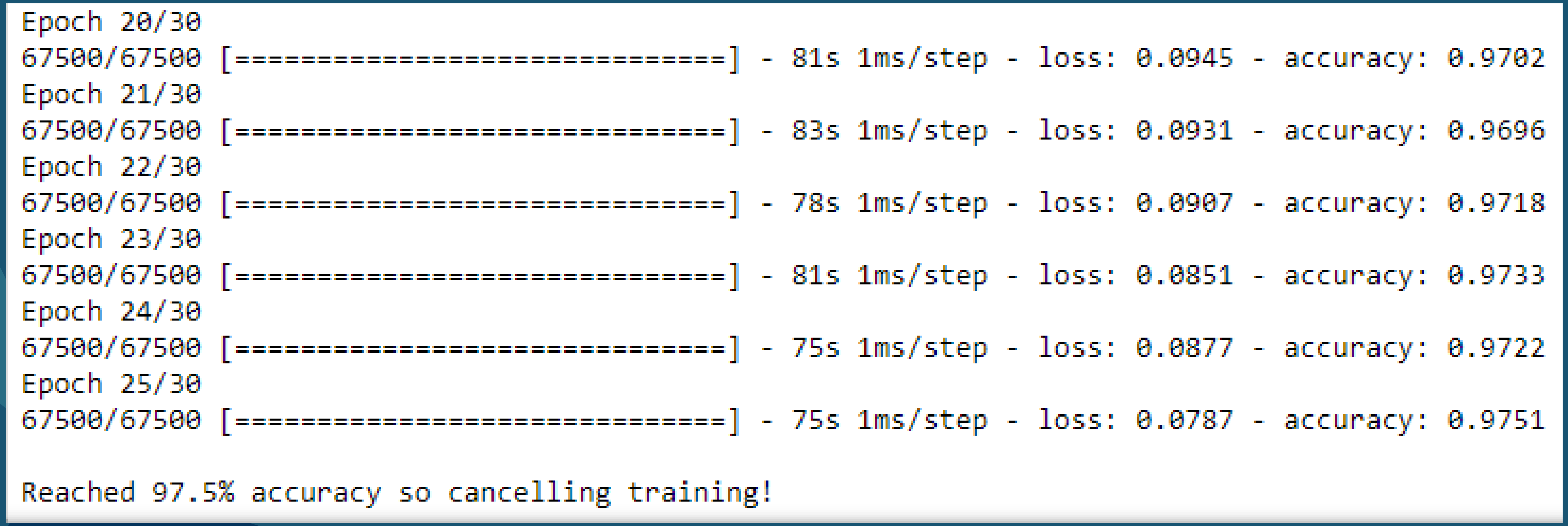
Loss function: Categorical Cross-entropy

Optimizer: Adam Optimizier

MODEL TRAINING

- Accuracy to over 97% is observed on the training data for 25 Epochs:
- The model training was stopped using callback when accuracy of over 97.5% was observed

Model fitting output



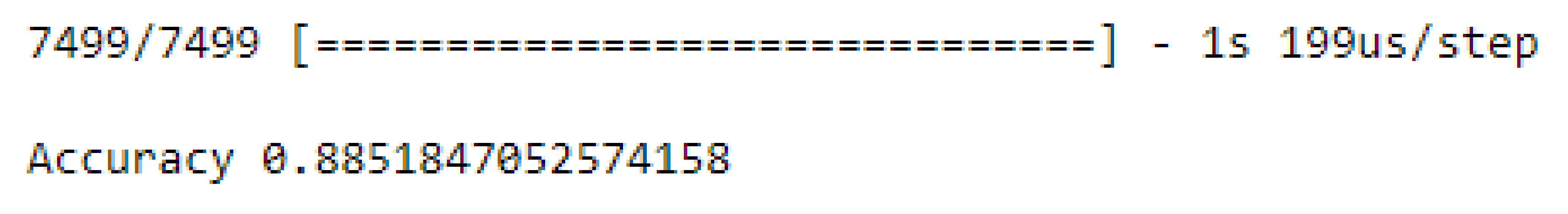
```
Epoch 20/30
67500/67500 [==============================] - 81s 1ms/step - loss: 0.0945 - accuracy: 0.9702
Epoch 21/30
67500/67500 [==============================] - 83s 1ms/step - loss: 0.0931 - accuracy: 0.9696
Epoch 22/30
67500/67500 [==============================] - 78s 1ms/step - loss: 0.0907 - accuracy: 0.9718
Epoch 23/30
67500/67500 [==============================] - 81s 1ms/step - loss: 0.0851 - accuracy: 0.9733
Epoch 24/30
67500/67500 [==============================] - 75s 1ms/step - loss: 0.0877 - accuracy: 0.9722
Epoch 25/30
67500/67500 [==============================] - 75s 1ms/step - loss: 0.0787 - accuracy: 0.9751

Reached 97.5% accuracy so cancelling training!
```

TEST MODEL

Accuracy to over 88% is observed on the test data of 7500 images

Model test output

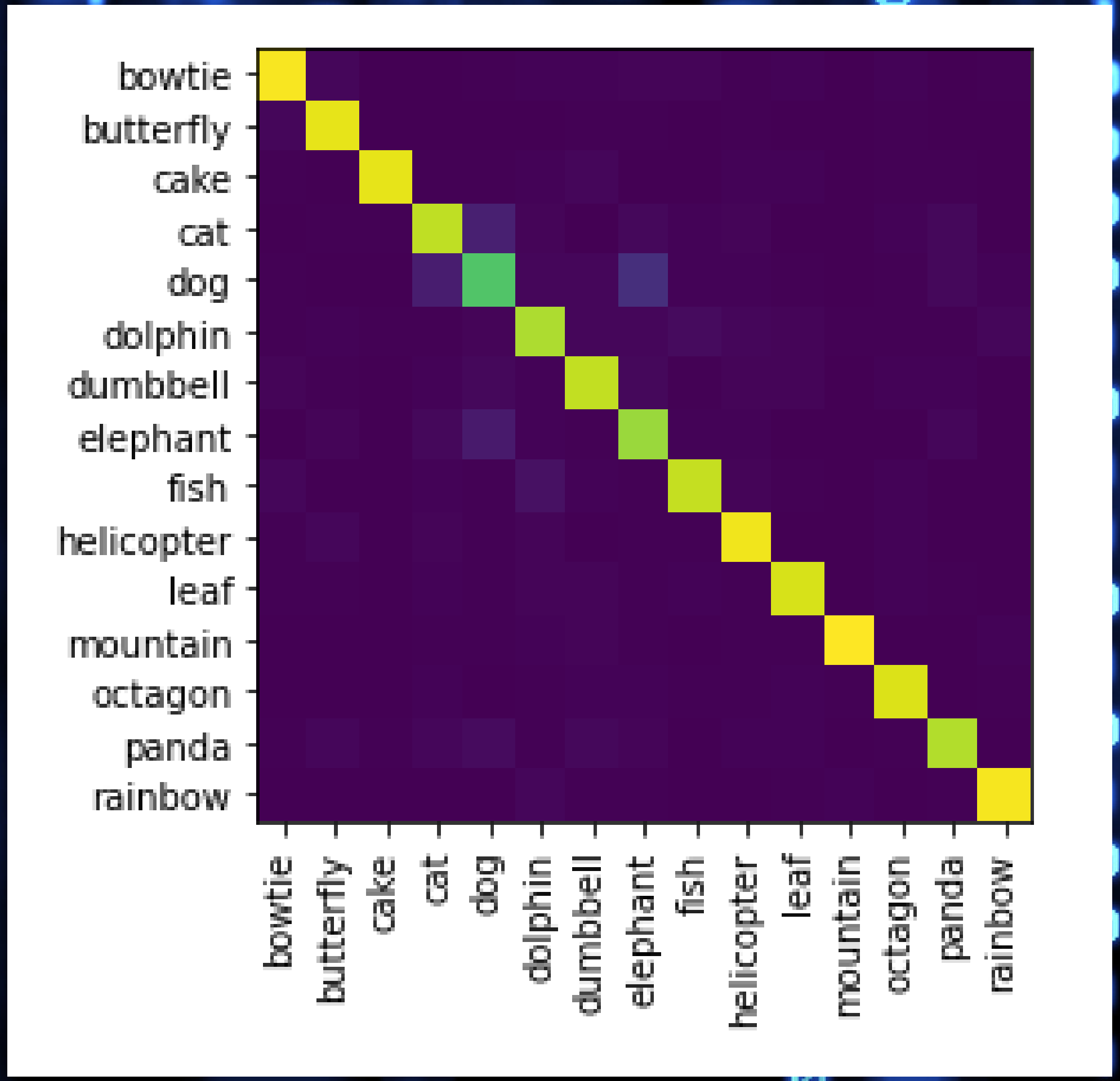


```
7499/7499 [==============================] - 1s 199us/step

Accuracy 0.8851847052574158
```

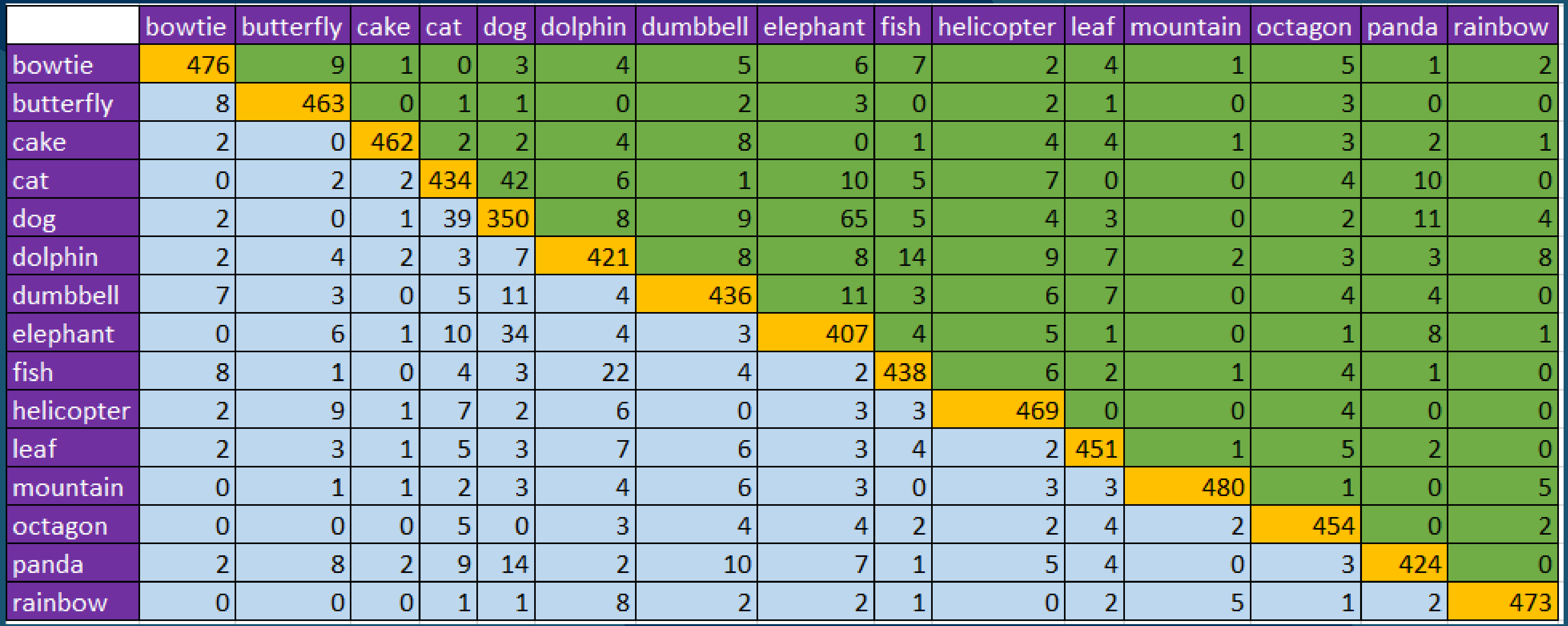
CNN MODEL

Confusion Matrix diagram



CNN MODEL

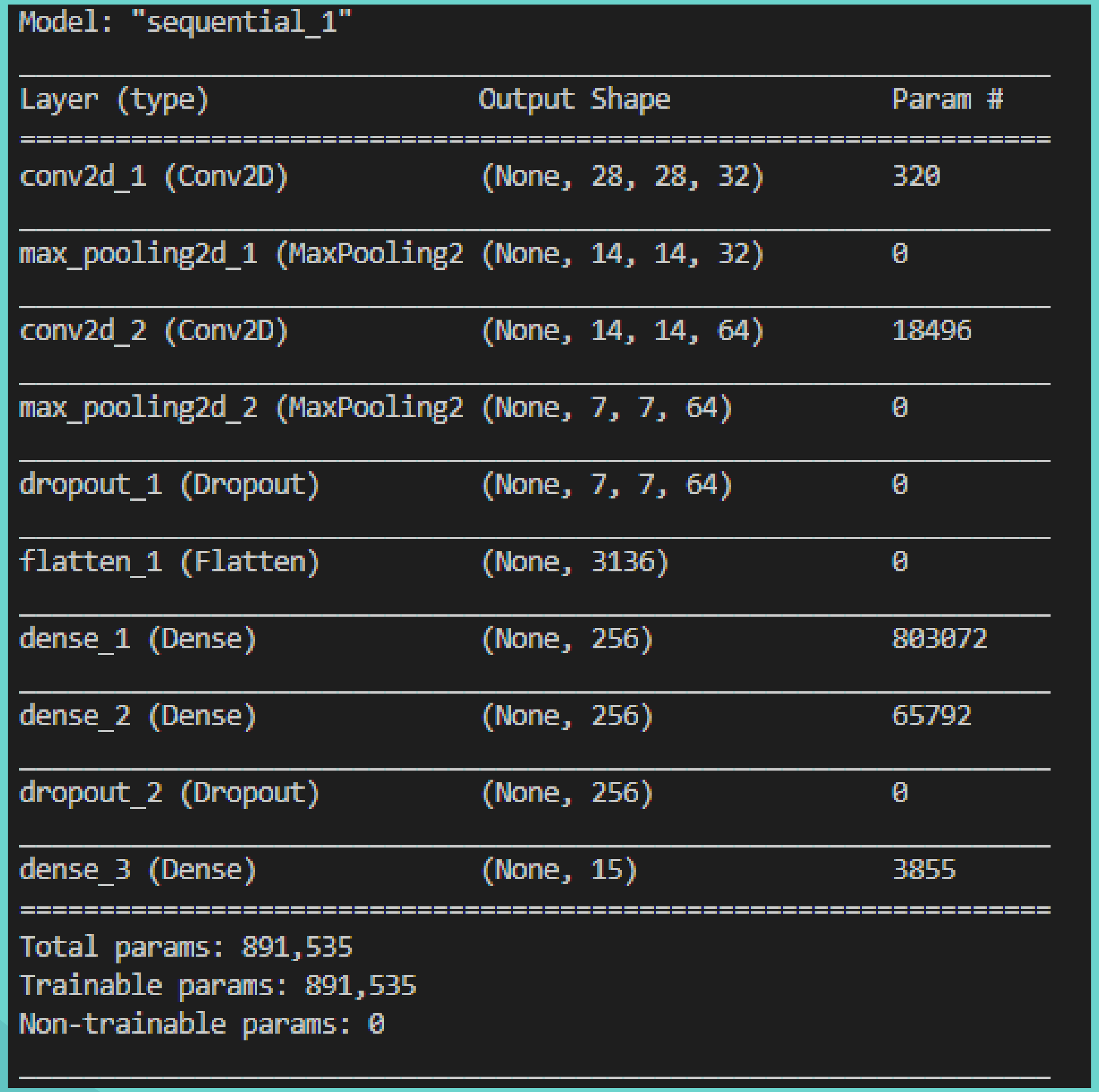
Confusion Matrix Values



	bowtie	butterfly	cake	cat	dog	dolphin	dumbbell	elephant	fish	helicopter	leaf	mountain	octagon	panda	rainbow
bowtie	476	9	1	0	3	4	5	6	7	2	4	1	5	1	2
butterfly	8	463	0	1	1	0	2	3	0	2	1	0	3	0	0
cake	2	0	462	2	2	4	8	0	1	4	4	1	3	2	1
cat	0	2	2	434	42	6	1	10	5	7	0	0	4	10	0
dog	2	0	1	39	350	8	9	65	5	4	3	0	2	11	4
dolphin	2	4	2	3	7	421	8	8	14	9	7	2	3	3	8
dumbbell	7	3	0	5	11	4	436	11	3	6	7	0	4	4	0
elephant	0	6	1	10	34	4	3	407	4	5	1	0	1	8	1
fish	8	1	0	4	3	22	4	2	438	6	2	1	4	1	0
helicopter	2	9	1	7	2	6	0	3	3	469	0	0	4	0	0
leaf	2	3	1	5	3	7	6	3	4	2	451	1	5	2	0
mountain	0	1	1	2	3	4	6	3	0	3	3	480	1	0	5
octagon	0	0	0	5	0	3	4	4	2	2	4	2	454	0	2
panda	2	8	2	9	14	2	10	7	1	5	4	0	3	424	0
rainbow	0	0	0	1	1	8	2	2	1	0	2	5	1	2	473

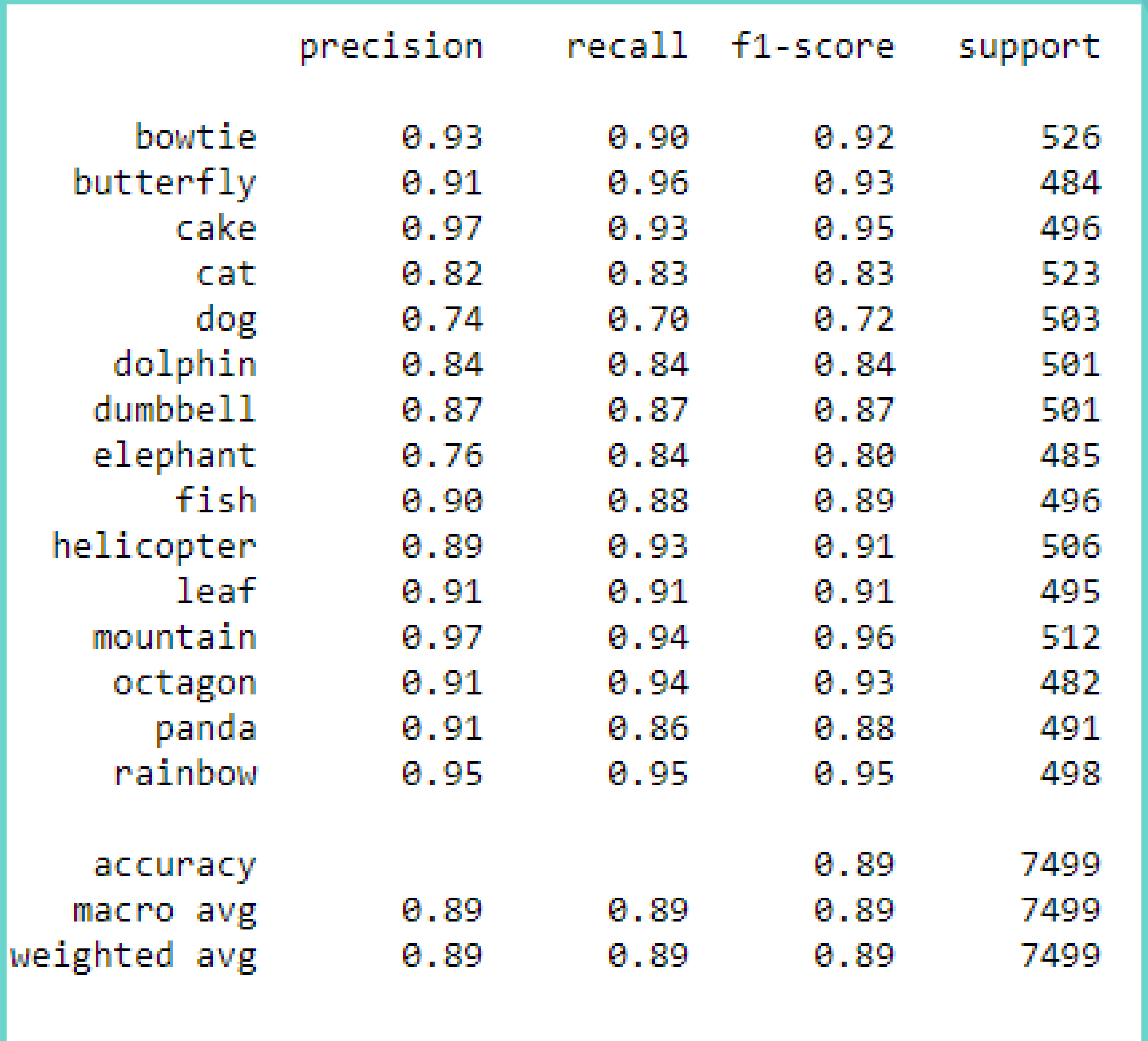
CNN MODEL LAYERS, PARAMETERS AND CLASSIFICATION REPORT

Model Summary



```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 28, 28, 32)        320
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 14, 14, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 7, 7, 64)          0
_________________________________________________________________
dropout_1 (Dropout)          (None, 7, 7, 64)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 3136)              0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               803072
_________________________________________________________________
dense_2 (Dense)              (None, 256)               65792
_________________________________________________________________
dropout_2 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_3 (Dense)              (None, 15)                3855
=================================================================
Total params: 891,535
Trainable params: 891,535
Non-trainable params: 0
_________________________________________________________________
```

Classification Report



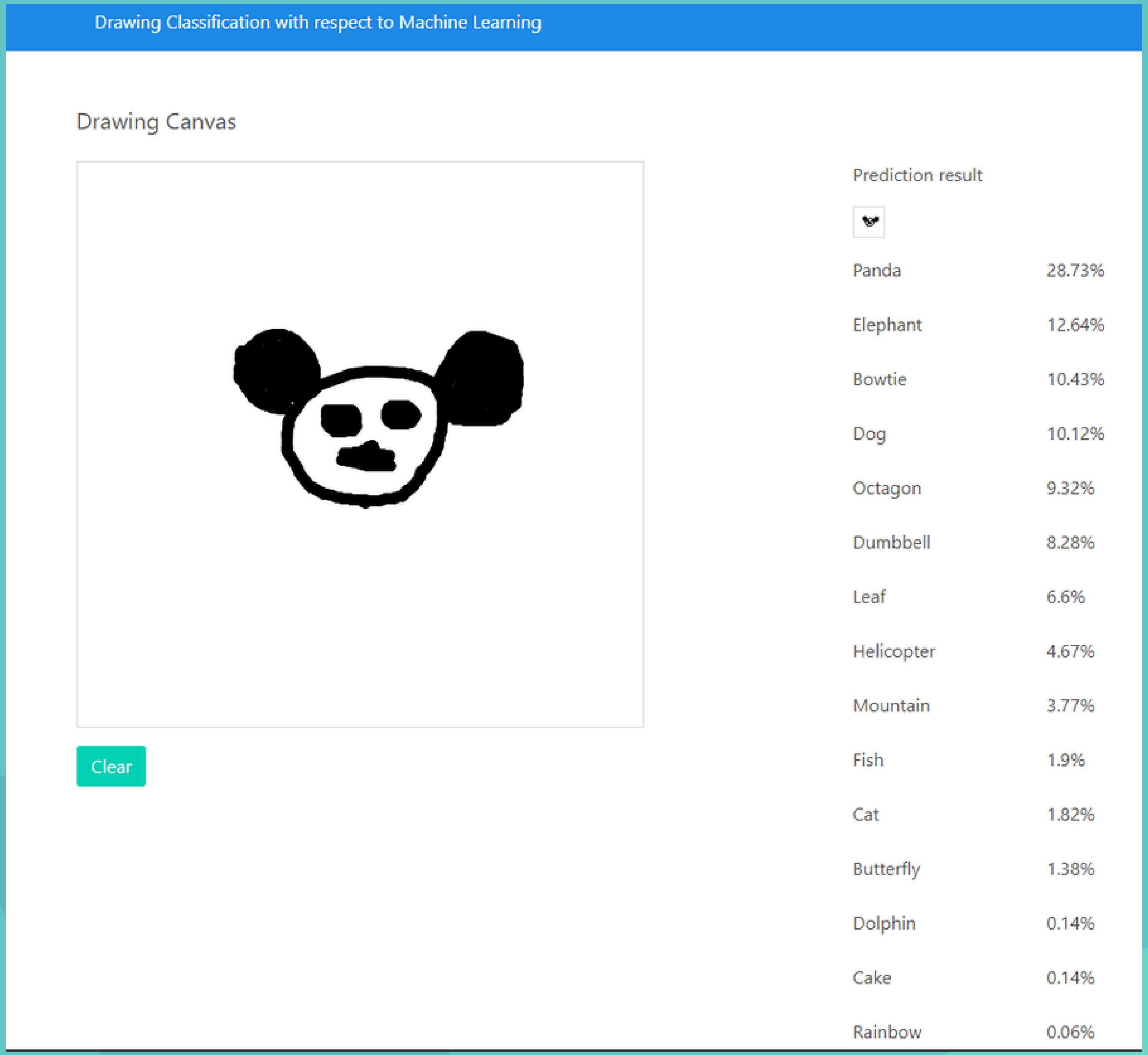
```
              precision    recall  f1-score   support

      bowtie       0.93      0.90      0.92       526
   butterfly       0.91      0.96      0.93       484
        cake       0.97      0.93      0.95       496
         cat       0.82      0.83      0.83       523
         dog       0.74      0.70      0.72       503
     dolphin       0.84      0.84      0.84       501
    dumbbell       0.87      0.87      0.87       501
    elephant       0.76      0.84      0.80       485
        fish       0.90      0.88      0.89       496
  helicopter       0.89      0.93      0.91       506
        leaf       0.91      0.91      0.91       495
    mountain       0.97      0.94      0.96       512
     octagon       0.91      0.94      0.93       482
       panda       0.91      0.86      0.88       491
     rainbow       0.95      0.95      0.95       498

    accuracy                           0.89      7499
   macro avg       0.89      0.89      0.89      7499
weighted avg       0.89      0.89      0.89      7499
```

HEROKU APP

https://drawing-ml.herokuapp.com



Drawing Classification with respect to Machine Learning

Drawing Canvas

Prediction result

Panda	28.73%
Elephant	12.64%
Bowtie	10.43%
Dog	10.12%
Octagon	9.32%
Dumbbell	8.28%
Leaf	6.6%
Helicopter	4.67%
Mountain	3.77%
Fish	1.9%
Cat	1.82%
Butterfly	1.38%
Dolphin	0.14%
Cake	0.14%
Rainbow	0.06%

THANK YOU,

Mudit, Landon, Ken, and to our classmates!