STAR-GALAXY CLASSIFICATION USING DEEP LEARNING

BY
AJAY DAS M
S3MCA
MAC23MCA-2008

Overview

1. INTRODUCTION
2. LITERATURE REVIEW 1
3. LITERATURE REVIEW 2
4. LITERATURE REVIEW 3
5. SUMMARY
6. PROJECT PROPOSAL
7. CONCLUSION

Introduction

- The challenge of accurately classifying astronomical objects as stars or galaxies has been a fundamental task in astrophysics for centuries.
- Traditional methods relied heavily on visual inspection and morphological analysis, which were labour-intensive and limited by human subjectivity and the capacity to process large data volumes.
- With the advent of modern sky surveys like the Sloan Digital Sky Survey (SDSS), the volume of astronomical data has grown exponentially, rendering manual classification impractical.

Introduction

- The proposed system uses Convolution Neural Network (CNN). The models will classify photometric data under two classes Star. An automated system with the help of deep learning methodology we can make these much easier.

- The models will classify photometric data under two classes Star and Galaxy. An automated system can be very helpful to offers significant benefits for star-galaxy classification, including reduced human error, increased scalability, and efficient handling of vast data quantities

LITERATURE REVIEW 1

Ganesh Ranganath Chandrasekar Iyer Krishna Chaithanya Vastare (2017). Deep Learning for Star-Galaxy Classification

TITLE	YEAR	DATASET	ARCHITECTURE ARCHITECTURE	ACCURACY
Deep Learning for Star-Galaxy Classification	2017	Dataset was taken from the Sloan Digital Sky Survey (SDSS). The dataset contains 30 million images.	Convolutional Neural Networks(CNN)	99.19

LITERATURE REVIEW 2

Kim EJ, Brunner RJ. Star-galaxy classification using deep convolutional neural networks. Monthly Notices of the Royal Astronomical Society. 2016 Oct 17:stw2672.

TITLE	YEAR	DATASET	ARCHITECTURE ARCHITECTURE	ACCURACY
Star-galaxy classification using deep convolutional neural networks	2016	photometric and spectroscopic data sets with different characteristics and compositions. data sets and the image pre-processing steps for retrieving cutout images	Convolutional Neural Networks (ConvNets)	99.48

LITERATURE REVIEW 3

Kennamer N, Kirkby D, Ihler A, Sanchez-Lopez FJ. ContextNet: Deep learning for star galaxy classification. In International conference on machine learning 2018 Jul 3 (pp. 2582-2590). PMLR.

TITLE	YEAR	DATASET	ARCHITECTURE ARCHITECTURE	ACCURACY
ContextNet: Deep Learning for Star Galaxy Classification	2018	The dataset used in the work consists of simulated images from the Large Synoptic Survey Telescope (LSST) observations, generated using the GalSim image simulation package.	**Local Network**: Convolutional Neural Networks (CNNs) **Global Network**: Recurrent Neural Networks (RNNs) **Prediction Network**: Fully Connected Neural Networks (FCNs)	ContextNet - 95%

LITERATURE REVIEW SUMMERY

REVIEW PAPER	ARCHITECTURE	ACCURACY
Deep Learning for Star-Galaxy Classification	Convolutional Neural Networks(CNN)	99.19
Star-galaxy classification using deep convolutional neural networks	Convolutional Neural Networks (ConvNets)	99.48
ContextNet: Deep Learning for Star Galaxy Classification	ContextNet	95

DATA SET

- The dataset is taken from the Kaggle repository, containing a collection of astronomical images captured using a 1.3-meter telescope located in Nainital, India.
- The dataset contains 3986 sample observations, with 942 Galaxies and 3044 Stars photometric data.
- Link : https://www.kaggle.com/datasets/divyansh22/dummy-astronomy-data/

PROJECT PROPOSAL

Objective

Develop a deep learning-based system for the accurate classification of astronomical objects as stars or galaxies, leveraging the benefits of reduced human error, increased scalability, and efficient handling of vast data quantities.

Methodology

Utilize Convolutional Neural Networks (CNNs) to build a binary classifier that can distinguish between stars and galaxies in photometric data. Implement data preprocessing, model training, and evaluation steps to optimize the classification performance.

DATASET EXPLORATION

SOURCE

- This data was created as a part of my project at Aryabhatta Research Institute of Observational Sciences (ARIES), Nainital, India
- The images were captured by the in-house 1.3m telescope of the observatory situated in Devasthal, Nainital, India

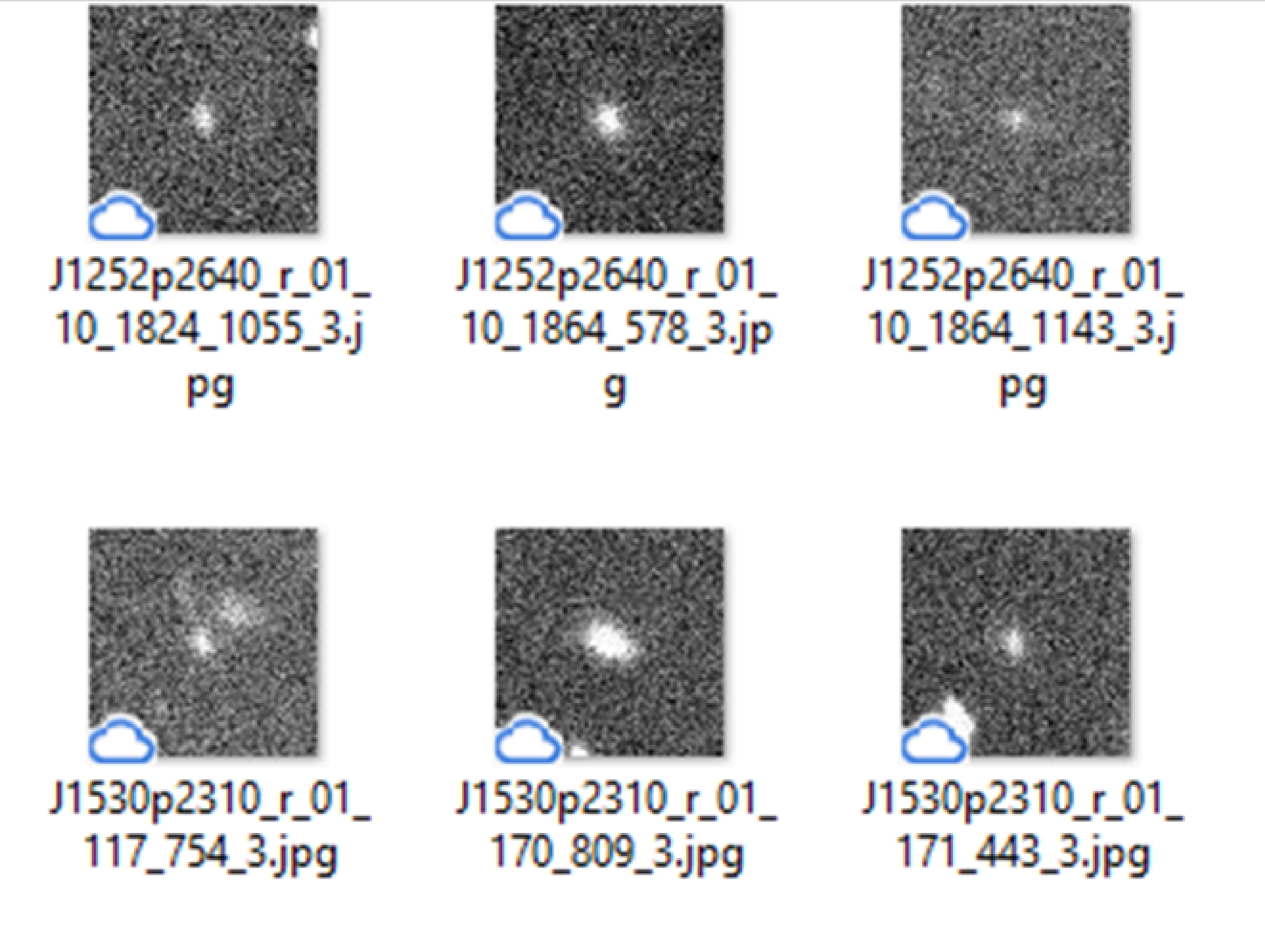
DATASET EXPLORATION

- All image in the dataset are reduced to 64 x 64 cutouts from the images to isolate the sources in a single image.

CLASS LABELS

The dataset is labeled with 10 distinct ayurvedic plant Some of them are:

- Star
- galaxy



Galaxy Data Example

Star Data Example

DATA PREPROCESSING

RESIZE IMAGE

Uniform Image Size: Since CNNs require fixed-size input images, all images in the dataset must be resized to a uniform size. A common choice for this type of classification task is 64x64 or 128x128 pixels, although the size can be adjusted based on the computational resources available and the complexity of the objects in the images.

Aspect Ratio Consideration: Ensure that resizing doesn't distort the images, particularly if the original images have different aspect ratios. In some cases, padding the images to maintain aspect ratios might be necessary.

NORMALIZE

Pixel Value Scaling: CNNs perform better when input data is normalized. Typically, image pixel values are scaled from their original range (0-255 for 8-bit images) to a range of 0-1 or -1 to 1. This is done by dividing the pixel values by 255. Normalization helps in speeding up the convergence during training by ensuring that the input features have a similar scale.

Data Augmentation

Data augmentation artificially increases the size of the training dataset by creating modified versions of images in the dataset. This helps the model generalize better and become more robust to variations.

Technique:

Rotation: Randomly rotate images within a certain range to simulate different orientations of celestial objects.

Flipping: Horizontally or vertically flip the images to introduce symmetry variations.

Zooming: Randomly zoom in on images to simulate different scales.

Brightness/Contrast Adjustments: Modify the brightness and contrast of the images to account for different lighting conditions in the data.

Translation: Shift images horizontally or vertically to simulate positional variance.

Splitting the Database

Training Set: 80% of the dataset is used for training. This is the subset of data the model will learn from.

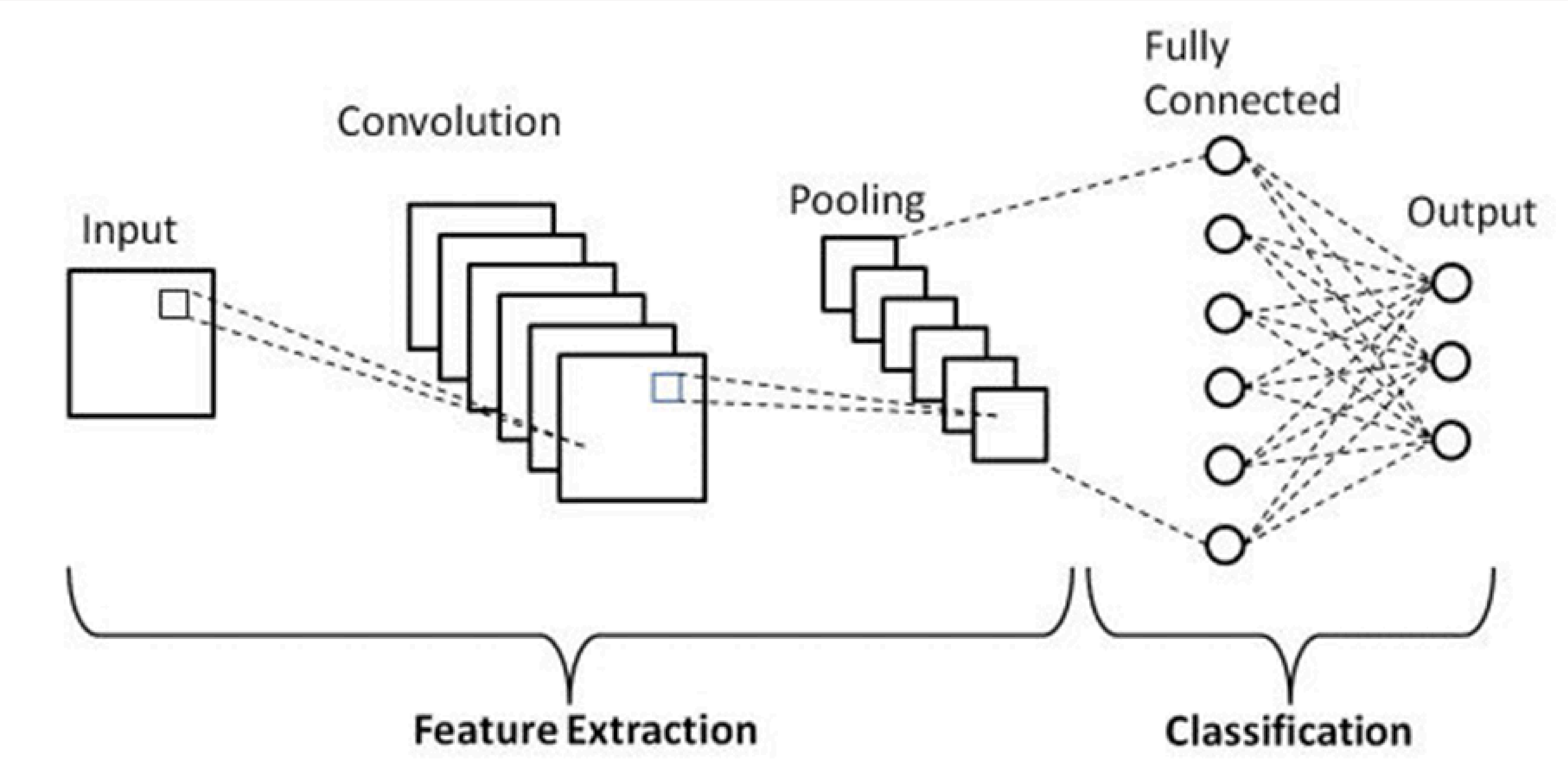
Test Set: The remaining 20% is reserved for testing the model after training to evaluate its performance on unseen data.

WORKING OF ALGORITHM

- Convolutional Neural Networks (CNNs) are a class of deep learning models commonly used for image and video recognition

- The main layers of CNN are:
  - Input Layer
  - Convolution Layer
  - Pooling Layer
  - Fully-Connected (dense)Layers
  - Output Layer

Architecture Diagram



1. Input Layer:

- Function: Accepts the raw input images, which in this case are 44x44 pixel images in five photometric bands (u, g, r, i, z).
- Role in the Project: Provides the initial data (images) to the network, which will be processed and analyzed to distinguish between stars and galaxies.

2. Convolutional Layers:

- Function: Applies convolution operations to the input data, using a set of filters (kernels) to extract features such as edges, textures, and shapes.
- Role in the Project: These layers detect various features at different levels of abstraction. Early layers might detect basic features like edges, while deeper layers detect more complex structures relevant to differentiating stars from galaxies.

4. Max-Pooling Layers:

- Function: Reduces the spatial dimensions (width and height) of the feature maps, retaining the most critical information while reducing the computational load and controlling overfitting.
- Role in the Project: By reducing the dimensionality, these layers help in abstracting the features detected by convolutional layers and make the network more computationally efficient.

5. Fully Connected (Dense) Layers:

- Function: Each neuron in these layers is connected to every neuron in the previous layer, which allows the network to combine the features extracted by the convolutional and pooling layers and make final predictions.
- Role in the Project: These layers are responsible for the final classification, combining all learned features to distinguish whether an object in the image is a star or a galaxy.

.

Output Layer:

- Function: Produces the final output, typically using a softmax function in classification tasks to produce probabilities for each class.
- Role in the Project: This layer outputs the probability of the image belonging to either the "star" or "galaxy" class, allowing for final decision-making in the classification task.

Functions And Packages

TensorFlow/Keras

- Sequential()
- Conv2D()
- MaxPooling2D()
- Flatten()
- Dense()
- Dropout()

NumPy

- array()
- reshape()
- mean()
- std()
- expand_dims()
- argmax()

Functions And Packages

Pandas:

- read_csv()
- DataFrame()
- DataFrame.head()
- DataFrame.describe()

Matplotlib/Seaborn:

pyplot.plot()

pyplot.imshow()

pyplot.show()

heatmap()

Functions And Packages

Scikit-learn:

sklearn.model_selection.train_test_split

sklearn.preprocessing.LabelEncoder

sklearn.metrics.confusion_matrix

sklearn.metrics.accuracy_score

sklearn.metrics.classification_report

sklearn.metrics.roc_curve

sklearn.metrics.auc

Functions And Packages

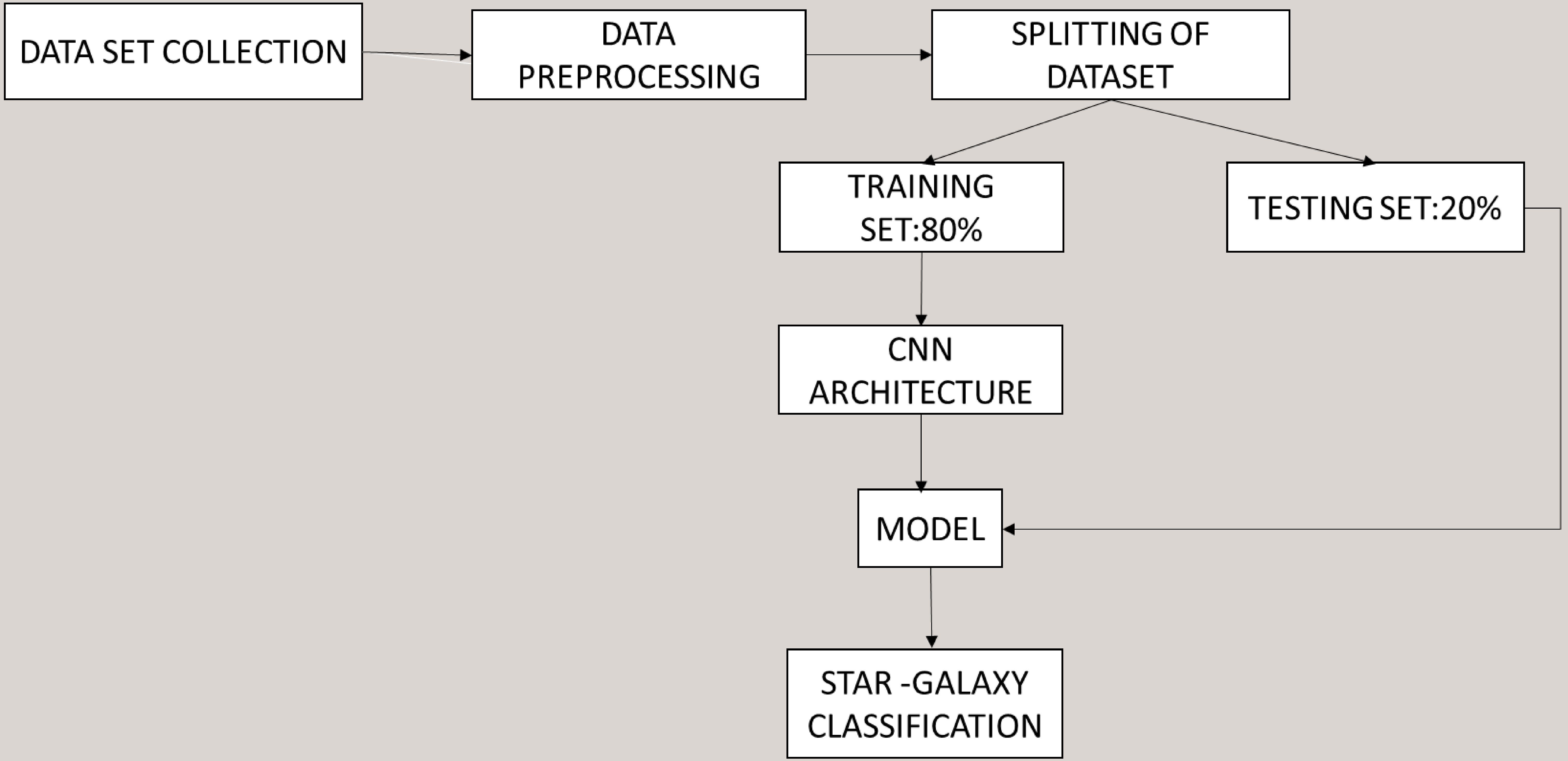
os:

- listdir()
- path.join()
- makedirs()

OpenCV:

-
- imread()
- cvtColor()
- resize()
- imshow()

PIPELINE DIAGRAM



Model Building

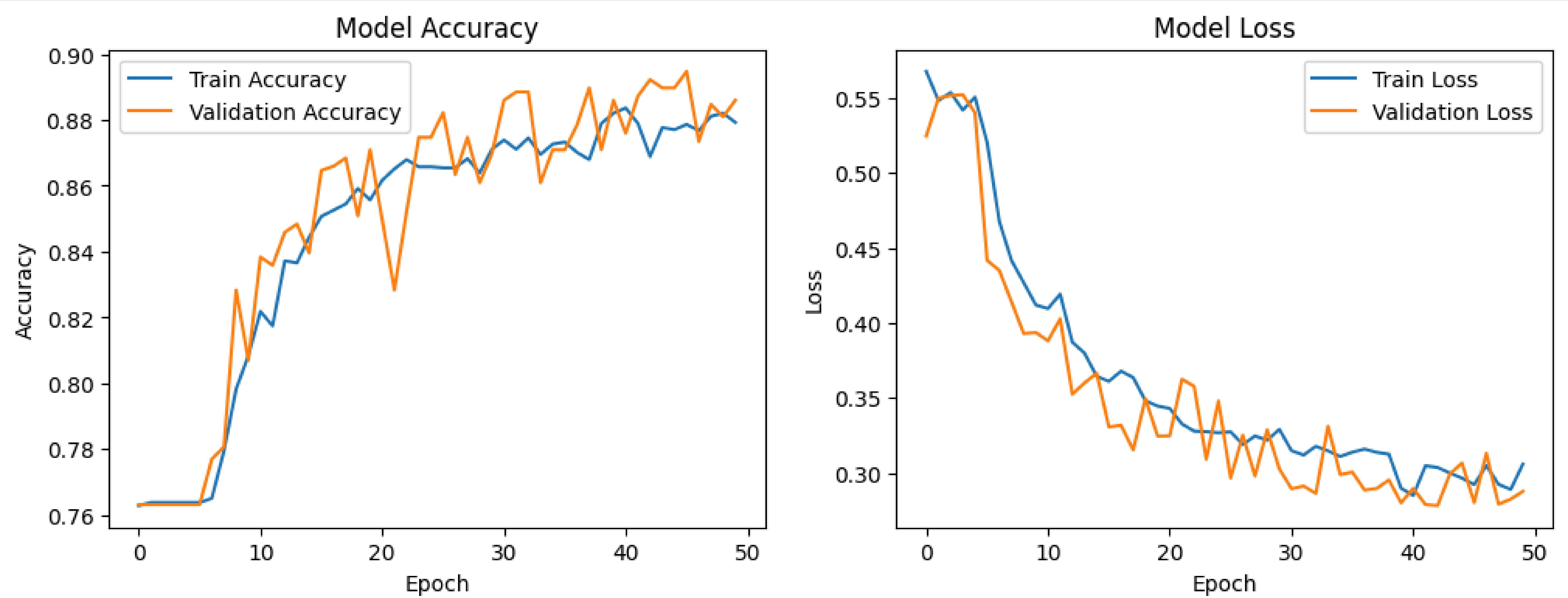
Add a little bit o The Convolution Neural Network is designed using the following layers:

1. Input Layer
2. Convolution Layer 1 (32 filters, 3x3 kernel, ReLU activation)
3. MaxPooling Layer 1 (2x2 pool size)
4. Convolution Layer 2 (64 filters, 3x3 kernel, ReLU activation)
5. MaxPooling Layer 2 (2x2 pool size)
6. Convolution Layer 3 (128 filters, 3x3 kernel, ReLU activation)
7. MaxPooling Layer 3 (2x2 pool size)
8. Flatten Layer
9. Dense Layer 1 (128 units, ReLU activation)
10. Dropout Layer 1 (50% rate)
11. Dense Layer 2 (64 units, ReLU activation)
12. Dropout Layer 2 (50% rate)
13. Output Layer (2 units, Softmax activation)

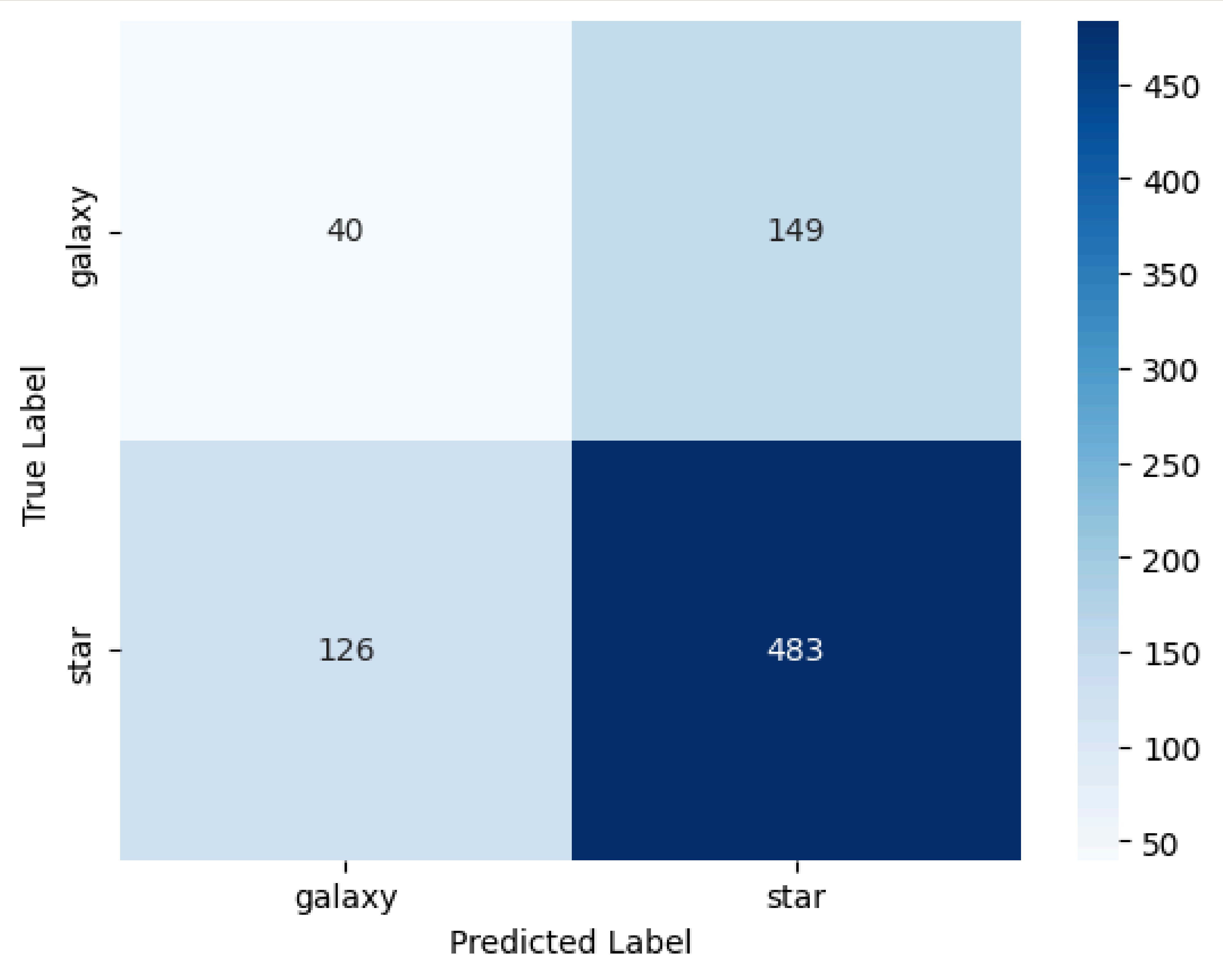
Evaluation Matrix

Metric	Galaxy	Star	Accuracy	Macro Average	Weighted Average
Precision	0.24	0.76		0.50	0.64
Recall	0.21	0.79		0.50	0.66
F1-Score	0.23	0.78	0.66	0.50	0.65
Support	189	609	798	798	798

Learning Curve



CONFUSION MATRIX



Thank You