Galaxy Classification Between AGN and Star-Forming Galaxies Utilizing A Convolutional Neural Network

Brandon Xu

Received October 17, 2023
Accepted December 14, 2023
Electronic access December 31, 2023

Galaxy images contain a wealth of information about the internal processes and histories of those galaxies. However, analysis of astronomical images has traditionally been done via hard coded algorithms, making it difficult to extract all of the information from direct images. Machine learning and convolutional neural networks are able to extract far more features from galaxy imaging than traditional algorithms, which can be used, among other goals, to construct cleaner classifications for galaxies. The unique advantage of CNNs in this context lies in their ability to automatically learn hierarchical representations of features from raw data. Galaxies exhibit diverse structures and formations, and CNNs excel in capturing these nuanced features. Patterns in images such as the spatial distribution of emission lines, the morphology of the galaxies, and the overall structure can contribute to the model's ability to classify galaxies. Using deep learning on the NASA SLOAN dataset, a publicly available large dataset of galaxy imaging, the use of machine learning algorithms to classify and separate Active Galactic Nuclei (AGN) galaxies and star-forming galaxies is explored. The NASA SLOAN dataset used in this study encompasses a substantial volume of galaxy images, providing a rich source for training and testing the CNN model. The dataset's composition includes coordinates of galaxies with positive emission line fluxes in H, NII, H, and OIII, signifying active star formation, allowing classification between AGN galaxies and star-forming galaxies. Such classifications have traditionally required expensive and time-consuming spectroscopy, so a neural-network based approach is highly incentivized.

Introduction

An important part of studying galaxies is to classify them into certain categories, allowing us to understand more about them. Accurate categorization enables researchers to shed light on the formation, development, and ultimate fate of galaxies. Precise classifications of galaxies offer a nuanced lens to examine the processes driving their evolution—ranging from the birth of stars to the active nuclei indicative of supermassive black hole accretion.

However, the process can be both expensive and time consuming. One way to classify galaxies is through spectroscopy. Unfortunately, despite traditional spectroscopic methods being extremely accurate, they entail the meticulous acquisition of spectral data from galaxies using large telescopes or specialized instruments. While this approach provides detailed insights into the chemical composition, redshift, and kinematics of galaxies, it comes with substantial challenges. The process demands significant observational time, making it infeasible for large-scale surveys. The requirement for access to expansive telescopes further limits the accessibility of spectroscopy, often sidelining smaller research initiatives.

Spectroscopy is a fundamental technique in astronomy used to study the properties of celestial objects, including galaxies. It involves the analysis of the light emitted or absorbed by these objects as a function of wavelength. By dispersing light into its component wavelengths, astronomers can examine the spectral features that provide valuable information about the object's composition, temperature, density, and motion[1].

However, spectroscopic observations typically require significant amounts of time. The more finely the light is spread out in a spectrograph, the longer it takes to receive good measurements. Obtaining a spectrum for each galaxy in a large sample can be time-consuming, especially when dealing with extensive surveys or datasets[2].

As a result, being able to classify galaxies using machine learning algorithms can prove to be a more efficient and cost-effective option. Even if machine learning algorithms cannot be completely accurate, an algorithm with a high percentage of success can be used to create samples of galaxy types with the ability to tune accuracy or sample purity.

This work attempted to leverage Convolutional Neural Networks (CNNs), a deep learning algorithm which can assign weights to important aspects of images. A CNN passes images through multiple layers, with each layer progressively extracting more complicated features from the image[3]. Based on the final convolution layer, the algorithm generates a set of confidence scores, indicating the likelihood of the image belonging to a specific "class" that it seeks to identify. Critically, many more (and more subtle) features can be extracted than traditional

algorithms have sought.

Specifically, a model was trained to classify between Active Galactic Nuclei (AGN) galaxies and star forming galaxies. AGN are galaxies with active nuclei, intense emission driven by the accretion of matter onto their central supermassive black holes. AGN may be star forming in their disk as well, and the tracers of AGN activity and star formation are similar, which can lead to incorrectly-inferred star formation rates. Emission line ratios are used to distinguish between the two types of galaxies.

This project aims to further explore the potential of machine learning in the field of galaxy classification. While previous studies have demonstrated high accuracy in using convolutional neural networks to classify images from the NASA SLOAN dataset, they relied on extensive manual sorting by volunteers. By specifically focusing on distinguishing between Active Galactic Nuclei (AGN) and star-forming galaxies, our research contributes additional evidence to the effectiveness of machine learning in galaxy classification. Moreover, the efficiency of data sorting is enhanced by leveraging mathematical formulas inherent in the Baldwin, Phillips, and Terlevich diagram[4].

The Baldwin, Phillips, and Terlevich (BPT) diagram is a 2D histogram which sorts between these two galaxy types. The rationale behind its efficacy lies in the distinct emission line characteristics exhibited by these two galaxy types. The BPT Diagram depends on four main sets of information: Hα flux, [N II] flux, Hβ flux, and [OIII] flux. Each refers to the amount of light or radiation emitted from different emission lines. The 2D histogram depends on two emission line ratios, as shown in Figure 1, and is plotted in log scale for more clarity. These are:

1. log([N II] flux/Hα flux)
2. log([OIII] flux/Hβ flux)

This project will use these emission line ratios to help with sorting through data.

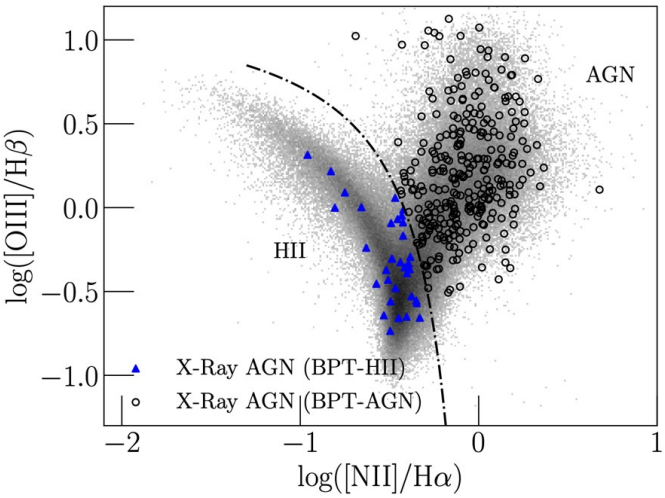

Fig. 1 A BPT Diagram is shown[5]. The line in the middle splits between Star-forming galaxies on the left, and AGN galaxies on the right.

Materials and Methods

Collecting Data

Large samples of galaxy images can be obtained from publicly available datasets, such as the NASA-Sloan Atlas[6], which takes data from the Sloan Digital Sky Survey (SDSS). Using the NASA-Sloan Atlas, the coordinates of a sample of galaxies with positive emission line fluxes in Ha, NII, Hb, and OIII — i.e. are obtained, which show signs of active star formation. The dataset needs to be sorted between AGN and Star-forming galaxies. The dataset is also filtered by sorting out sets that have emission line values at 0 or less, meaning that the galaxy is quiescent and is not forming stars. By plotting the dataset to recreate a BPT diagram, the galaxies from the NASA-Sloan Atlas can be sorted.

After creating the plot, Star-forming and AGN galaxies can be separated by recreating the line on the BPT diagram shown

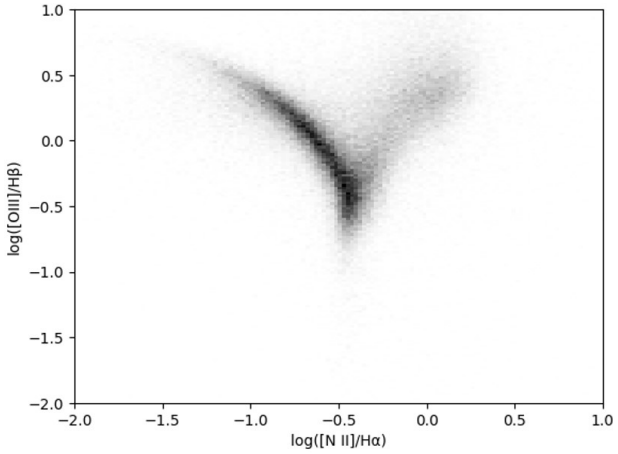

Fig. 2 The resulting plot of the NASA-Sloan Atlas data.

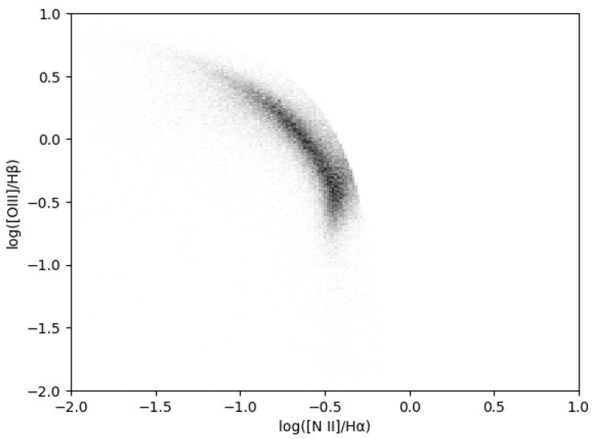

Fig. 3 The 2D histogram plot of star-forming galaxies in the NASA SLOAN dataset.

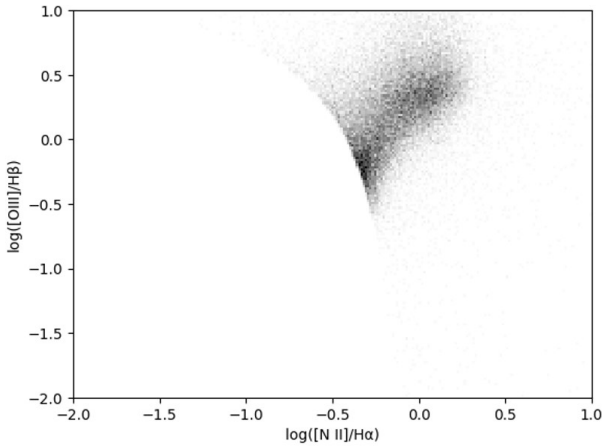

Fig. 4 The 2D histogram plot of AGN galaxies in the NASA SLOAN dataset.

in Figure 1. Although the exact equation of the line varies, it can be estimated with the equation[7]:

$$\log([\text{O III}]/\text{H}\beta) = \frac{0.61}{(\log([\text{N II}]/\text{H}\alpha) - 0.05)} + 1.3 \quad (1)$$

However, the equation appears to come back up to the right, creating an extra portion of points in the star-forming galaxies plot and cutting out a portion from the AGN galaxies plot. To fix this, a second condition was added. As shown in Figure 3, the extra points all have a value of $log([\text{N II}]/\text{H}\alpha)$ greater than -0.1. Consequently, the conditions for galaxies in the plot for star-forming galaxies are:

1. $\log([\text{O III}]/\text{H}\beta) < \frac{0.61}{(\log([\text{N II}]/\text{H}\alpha) - 0.05)} + 1.3$ (2)
2. $\log([\text{N II}]/\text{H}\alpha) < -0.1$ (3)

Other points not included within these conditions form the AGN galaxies plot. Using these conditions, two plots are created showing star-forming galaxies and AGN galaxies, in Figures 4 and 5, respectively.

After sorting the NASA SLOAN dataset into star-forming and AGN galaxies, the actual image files can be collected. Images were cut out from the DESI legacy imaging surveys sky viewer[8]. Given certain parameters, a link can be generated to cut out a jpg image of the galaxy. These parameters include:

1. Right ascension of measured object center (`ra`), a coordinate used to specify the east-west position of a celestial object
2. Declination of measured object center (`dec`), a coordinate used to specify the north-south position of a celestial object
3. Image size (`size`), set to 250 by 250 pixels for all images.
4. Image pixel scale (`pix_scale`), which was based on the radius of the galaxy measured in the NSA to accurately cut out the image, through the equation `pix_scale` = `radius` $\times 1.5 \times 2$/`size`.

Training the model

Once downloaded, the images can be proportioned for training and testing. In this case, 75% of images were used for training, 15% for validation, and 10% for testing. This results in 105980 images for training, 21195 images for validation, and 14130 images for testing. The chosen data splitting percentages are common choices used in machine learning development. The 75% of images used for the training set should be a sufficiently large set of data to allow the model to learn complex patterns and features within the data, and capture a diverse range of characteristics present in galaxy images. The 15% of images used for validation is crucial for hyperparameter tuning and preventing overfitting. A smaller percentage is allocated to the validation set to ensure that the model has ample data for training while maintaining an independent sample for performance validation. The remaining 10% of images in the test sample is reserved for evaluating the model's final performance. As the test data does not have an effect on the model itself, it avoids potential biases and can serve as a proxy for real-world scenarios.

The CNN model was trained on GPUs accessed through the Google Colaboratory service. Google Colab's GPUs help to accelerate the training process of CNNs. However, since our model required 30 hours of training, while Colab sessions are limited to 12 hours, model checkpoints were saved for every epoch. The CNN model consists of multiple layers. The input layer accepts images of size 128 by 128 pixels.

The three subsequent convolutional layers use 32 filters in the first convolutional layer, 64 in the second, and 128 in the third. The initial layers focus on capturing simple and local features, while subsequent layers build upon these to discern more complex patterns. The chosen numbers strike a balance between model complexity and computational efficiency, ensuring the network can capture both basic and intricate features present in galaxy images.

Each convolutional layer also uses a (3, 3) kernel size. It enables the model to capture spatial dependencies within a 3x3 region of the input, preserving local information while progressively learning hierarchical features. Larger kernel sizes might capture more global features but could increase computational complexity, potentially leading to overfitting. Smaller kernel sizes might overlook broader patterns. The (3, 3) size strikes a practical balance for extracting relevant features in the context of galaxy images.

Rectified Linear Unit (ReLU) activation functions were chosen after each convolutional layer. ReLU has been widely adopted in image classification tasks due to its simplicity and computational efficiency. ReLU introduces non-linearity to the model by outputting the input for positive values and zero for negative values. This choice is based on ReLU's effectiveness in mitigating the vanishing gradient problem, promoting faster convergence during training. Valid padding is utilized, resulting in feature maps of reduced spatial dimensions compared to the input image. This decrease is due to the constraints of only using complete filter windows during the convolution operation.

The first convolutional layer uses 32 filters, each of (3, 3) kernel size. After the convolutional layer, there is a max-pooling layer, which performs pooling with a (2,2) window of stride 2. The layer reduces the spatial dimensions of the feature maps generated by the previous convolutional layer, emphasizing distinctive features of the feature maps.

The second convolutional layer applies 64 filters of (3, 3) kernel size. This is followed by a second max-pooling layer. These layers perform the same process as the ones from before.

The third convolutional layer applies 128 filters of (3, 3) kernel size, followed by a third max-pooling layer. These layers perform the same process as the ones from before.

The flatten layer comes after the convolutional and max-pooling layers, and transitions the data from two-dimensional feature maps into a one-dimensional array. This prepares the information for further analysis in the fully connected layers.

The first fully connected, or dense layer has 128 units. The layer computes a weighted sum of inputs from the previous layer, and applies the ReLU activation function. The ReLU function keeps weighted sums the same if the results are positive, and outputs 0 if the results are negative.

The second fully connected layer performs the same function as the first, but has 64 units.

The output layer applies the Sigmoid activation function to output a single number from 0 to 1. 0 represents star forming galaxies, while 1 represents AGN galaxies.

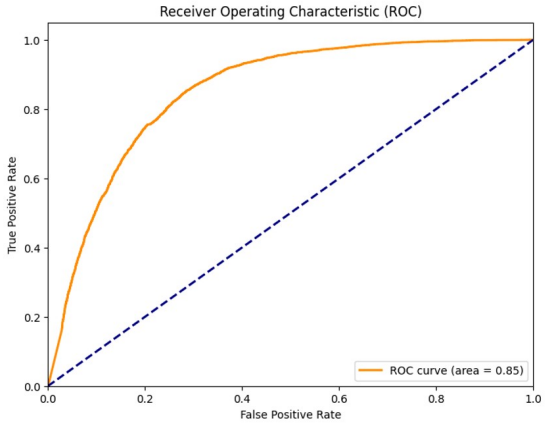

Fig. 5 The ROC curve for the trained model. It shows an AUC score of 0.85.

The model is compiled with an Adam optimizer and uses a Binary cross-entropy loss function. The model was trained for 10 epochs.

Results

A test sample of about 20 images found that the model had the highest accuracy at a prediction threshold of about 0.3, since using a threshold of 0.3 gave the test sample an accuracy of 100%. After accounting for the threshold, the model achieved an accuracy of about 78.1% on 14130 testing images. Other thresholds around the range from 0.15 to 0.4 produced similar accuracy results, although accuracies would decrease as the threshold moved away from 0.3. With no threshold, the accuracy is at 63.9%.

The Area Under the Curve (AUC) score is a measure of the overall performance of a binary classification model across various decision thresholds. The Receiver Operating Characteristic (ROC) curve is a graphical representation of the model's ability to discriminate between the positive and negative classes, and the AUC quantifies the area under this curve. An AUC score of 0.85 represents an 85% chance of ranking a randomly chosen positive instance (star-forming galaxies) higher than a randomly chosen negative instance (AGN galaxies).

The model had a precision of 0.6781. A precision of 67.81% implies that when the model predicts a galaxy as a star-forming galaxy, it is accurate about 67.81% of the time. The model had a recall of 0.7476. In other words, the model successfully identifies 74.76% of the star-forming galaxies. The model has a F1 score of 0.7111, which highlights the model's effectiveness in minimizing both false positives and false negatives. However,