INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

Deep Learning for Star-Galaxy Classification

K. Narayana Rao [1], K. Kishore Babu [2]
[1] professor, Assistant Professor [2]
Department of Computer Science Engineering
RISE Krishna Sai Prakasam Group of Institutions

Abstract Conventional star-galaxy classifiers are based on the reduced summaries provided by the star-galaxy catalogs. However, these classifiers need careful feature selection and involvement of domain experts at various stages of classification. Thus, the current mechanism is not extremely scalable. It is important to develop a scalable probabilistic classifier based on source information with minimal involvement of humans to overcome these shortcomings. In this project we tried to implement CNN based binary star-galaxy classifier proposed

1. Introduction

Over the past few years, advancements in technology has provided us extensive knowledge about the universe. New solar systems, planets, stars, etc are being discovered on a daily basis. Problem of classification in astronomy goes back as far as 18th century Messier. Morphological separation [2] [3] has been frequently used for star-galaxy classification. However, with the rate we are discovering new stars and galaxy systems makes it a very tedious task to use morphological separation. Also the current research in Dark Energy is coming up with a large Photometric survey called Dark Energy Survey (DES1). This survey is currently at a few petabytes of data. Manually processing this data is practically impossible even for experts in Astrophysics. Thus we need to explore different automated classification methods for star-galaxy classification.

2. Literature Survey

Machine learning (ML) methods solve classification problems in a more probabilistic manner. Thus we can solve the classification problem to the greatest accuracy. ML techniques have been a popular tool in various fields of Astrophysics. Especially, Neural Networks

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

(NNs) [1],[4], Support Vector Machines (SVM) [5], Random Forest (RF) ([2],[6],[7]), k-Nearest Neighbour (kNN) and NB just to name a few.

Recently, Edwardo Machado of CEFET/RJ., France, published a paper which encompassed and compared all the above mentioned algorithms for Star - Galaxy classification. Figure 1 shows the purity vs the magnitude for all the algorithms. It is evident that NN performs the best. Table 1 compares the algorithms based on the accuracy, Area under the ROC curve (AUC), Completeness galaxies and Purity galaxies [8]. Again from his results it is evident that NNs are the most promising. In this project we will focus on implementing Convolutional Neural Networks for Star-Galaxy Classification.

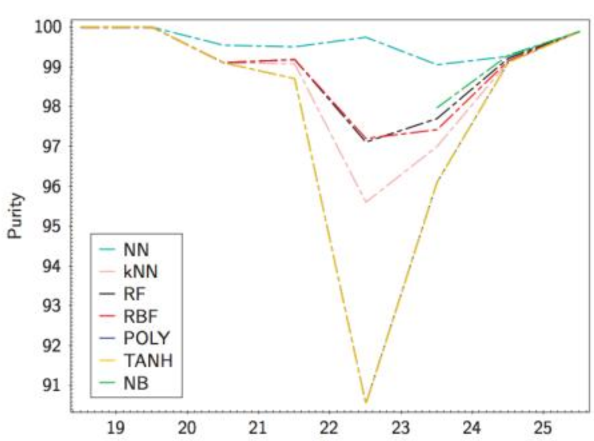

Figure 1: Magnitude vs Purity of various Star- Galaxy classification methods [8]

3. Data

We used the photometric and spectroscopic information from the Sloan Digital Sky Survey (SDSS) [9] DR 12 data set for training and testing the validity of the convolutional neural network model. The Sloan Digital Sky Survey [9] collects the data from 5 photometric bands namely u,g,r,i and z. The catalog covers over 300 million stars and galaxies.

Table 1: Results of Various Star-galaxy Classification methods [8]

Method	Accuracy	AUC	Completeness Galaxies	Purity galaxies
NN	99.19	0.984	99.84	99.34
RF	99.11	0.978	99.87	99.23
SVM_{rbf}	99.02	0.913	99.34	99.18
SVM_{poly}	98.51	0.961	99.95	98.56
SVM_{tanh}	98.51	0.961	99.95	98.56
kNN	98.89	0.945	99.87	99.02
NB	83.97	0.869	84.13	99.54

3.1. Processing Data

The data processing performed by us can be classified into the following stages:

• Catalog Fetch: We used the DR 12 context of SDSS's CASJobs server to select about 25000 entries which are either star or galaxies. For the sake of securing a clean data set following aspects were considered:

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688

1 The third class of celestial object in the survey is labelled ‘QSO’ and stands for Quasars. The Quasars are celestial bodies which can’t be classified into binary labels of interest. 2 Data points with photo metric observation errors were rejected. 3 The extinction parameters were included to apply corrections. 4 Data points with any warning were rejected.

• Montage Each entry in the catalog has 5 images associated with u, g,r,i and z. However, there are pixel overlaps across these images cause by the survey methodology. In this stage we used Montage [10] to align all the images with image of r band. The Montage’s ‘reproject’ algorithm [10] project all the images on a spherical surface and realigns the images with respect to the reference image.

• SExtractor The photometric images will not have the object of interest in the center. We used the SExtractor package [11] for extracting the pixels with information and center the object.

• Conversion to Luptitudes: The data points of magnitudes are in inverse hyperbolic sine magnitudes called luptitudes. Hence, we converted all the flux values to luptitudes.

• Extinction Correction Due to galactic dust photometric devices might induce errors into the images. Hence, we used the extinction parameters from the catalog to remove or neutralize these errors.

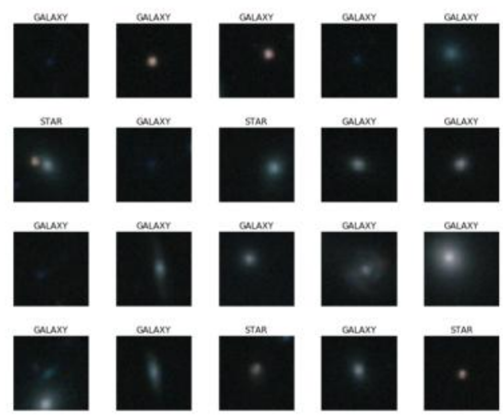

Figure 2: Input Data to CNN

4. Physical and Mathematical Framework - Deep Learning

Artificial neural networks have been studied for many years with a goal to achieve human-like performance for speech and image recognition [12]. For this application, we have utilized the concept of convolutional neural networks to perform star-galaxy classification. This section describes the mathematical model and the

ISSN: 2057-5688

architecture of the Deep Neural Network used to perform the classification task. The architecture of the CNN has a prominent role in its performance. It is proved by Edward Kim [4] that the use of the following architecture is the best for the star-galaxy classification. [1] compared CNNs with different Random Forrest algorithms (TPCs) and the authors results show that the following model works best for SDSS dataset. The network architecture is composed of eleven trainable layers. The first convolutional layer filters the 5 x 44 x 44 input image (i.e., 44 x 44 images in five bands u, g, r, i, z) with 32 square filters of size 5 x 5 x 5. For each CNN layer, we have employed Leaky ReLU as the activation function for the NN. The second layer filters the data with 32 filters, each of size 32 x 3 x 3 size. In the second layer, zero padding has been done on the border of the input. This was proposed by [1] to preserve the spatial resolution after convolution operation. Next we have a max-pooling layer which uses a filter of size 2x2 to reduce the dimension of the feature space. Then we have a stack of six additional convolutional layers, all with filters of size 3 x 3 and maxpooling layers with filters of size 2 x 2. Finally, we have three fully connected layers, where the first two have 2048 channels each (recommended by [1]) and the third performs binary classification using softmax function.

5. Implementation Details

6.1. Data Processing We used SQL query on CASJobs's DR 12 instance for getting the catalog. To process the data as described above we used Montage Wrapper (Astropy's v0.9.8)and SExtractor(v2.19.5) in Python 2.7.13 (Anaconda 4.4).

6.2. Neural Network The neural network was simulated used Nvidia's GeForce GTX 1050 Ti with 80% CNMeM and cuDNN 5.1. The scripting was done in Python 2.7.13 (Anaconda 4.4.0 64 Bit) using Theano (v 0.9) + Lasagne (v 0.2).

7. Results

7.1. Error Metric The system implemented is binary classifier and we used Area Under Receiver Operating Characteristic Curve (ROC). The ROC curve is constructed by plotting the True Positive Rate of classifier against its False Positive Rate. It is in the range 0-1 and higher the metric better the prediction accuracy. Fundamentally, this

ISSN: 2057-5688

metric helps us understand the classifier's ability when the threshold is changed. Hence, selected it.

7.2. Output Initially, we evaluated the model using 100 entries from the catalog and the found the average value of the error metric over 5 iterations to be equal to 0.94. Latter, we ran the simulation for 1500 entries and found the average value of error metric over 5 iterations to be 0.97. For these simulation, we used 80% of the data for training and 20% for testing.

8. Conclusion and Future work

8.1. Conclusion

In this project, we implemented the CNN based binary classifier suggested by [1] to classify the data from photometric catalogs as star or galaxy. Our main goal was to understand the working of CNN and its properties. As discussed in the literature survey, we are convinced that CNN based binary classification requires lesser involvement of experts in the subject and human error is reduced significantly. However, when we ran the data for about 1500 entries we got area under ROC as 0.97. This in agreement with author's reported area under ROC (0.99) for this model.

In the due course of project we also learnt a little bit about writing SQL query, using the Theano and Lasagne wrappers. Since, we new to python, it helped us appreciate the language and learn to use it. Further, we also would like to mention that each entry in the catalog corresponds to 62 MB of data in FITS format and we had access only to GPU 768 cuda cores and system with 50 GB of free space. Hence, due to limitations of hardware, we were not able to train more data. In future, we are planning to scale the operation and compare the performance of the model with other ML techniques in the literature studied. We would also like to study the effect of increase in neural network size on over fitting and our error metric (i.e.) Area Under ROC.

References

[1] Edward J Kim and Robert J Brunner. Star-galaxy classification using deep convolutional neural networks. Monthly Notices of the Royal Astronomical Society, page stw2672, 2016. 1, 3, 4

ISSN: 2057-5688

[2] EC Vasconcellos, RR De Carvalho, RR Gal, FL LaBarbera, HV Capelato, H Frago Campos Velho, M Trevisan, and RSR Ruiz. Decision tree classifiers for star/galaxy separation. The Astronomical Journal, 141(6):189, 2011. 1

[3] Marc Henrion, Daniel J Mortlock, David J Hand, and Axel Gandy. A bayesian approach to star–galaxy classification. Monthly Notices of the Royal Astronomical Society, 412(4):2286–2302, 2011. 1

[4] Edward J Kim, Robert J Brunner, and Matias Carrasco Kind. A hybrid ensemble learning approach to star–galaxy classification. Monthly Notices of the Royal Astronomical Society, 453(1):507–521, 2015. 1, 3 [5] Ross Fadely, David W Hogg, and Beth Willman. Stargalaxy classification in multi-band optical imaging. The Astrophysical Journal, 760(1):15, 2012. 1

[6] Nicholas Weir, Usama M Fayyad, and S Djorgovski. Automated star/galaxy classification for digitized possii. The Astronomical Journal, 109:2401, 1995. 1

[7] AA Suchkov, RJ Hanisch, and Bruce Margon. A census of object types and redshift estimates in the sdss photometric catalog from a trained decision tree classifier. The Astronomical Journal, 130(6):2439, 2005. 1

[8] Eduardo Machado, Marcello Serqueira, Eduardo Ogasawara, Ricardo Ogando, Marcio AG Maia, Luiz Nicolaci da Costa, Riccardo Campisano, Gustavo Paiva Guedes, and Eduardo Bezerra. Exploring machine learning methods for the star/galaxy separation problem. In Neural Networks (IJCNN), 2016 International Joint Conference on, pages 123–130. IEEE, 2016. 1, 2

[9] Sloan digital sku survey. http://www.sdss. org/. 1

[10] Montage. http://montage.ipac.caltech. edu/. 2

[11] Sextractor. https://www.astromatic.net/ software/sextractor. 2

[12] Richard Lippmann. An introduction to computing with neural nets. IEEE Assp magazine, 4(2):4–22, 1987. 2

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In

ISSN: 2057-5688

Advances in neural information processing systems, pages 1097–1105, 2012. 2, 4

[14] Dan C Cire¸san, Ueli Meier, Jonathan Masci, Luca M Gambardella, and Jürgen Schmidhuber. Highperformance neural networks for visual object classification. arXiv preprint arXiv:1102.0183, 2011. 2

[15] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In Proc. ICML, volume 30, 2013.