Convolutional Neural Networks for galaxy morphology prediction

P Manindra Kumar(192SP016), A Premkumar(192SP002), B Saikiran(192SP005), K Udaykumar(192SP011)
Department of ECE, National Institute of Technology Karnataka,
Surathkal, Mangalore, India - 575 025.

Abstract— **The main requirement to study the formation and evolution of galaxies is measuring the morphological parameters. The traditional method to carry out morphological analysis is by visual inspection which is time consuming. Only the trained experts can do the conventional way of analysis. But in order to have an automated system which can give the parameters directly using neural networks, we need a lot of images to train the network. So the surveys like Sloan Digital Survey (SDSS) made available a very large collection of images. To classify those images, Galaxy Zoo project introduced the crowdsourcing methodology. But still this method also didn't reach the required level of accuracy. Here we present a deep neural network model to classify the galaxies on the basis of their morphology. This greatly reduces the experts workload and gives the better accuracy.**

Keywords: Galaxy Zoo, crowdsourcing, augmentation

I. INTRODUCTION

In the world of space science, galaxies are the most fascinating elements to study. They exhibit a wide variety of shapes, sizes and colours. These are the parameters which indicate the age, formation conditions, evolution and interactions with other galaxies. Such studies require both the observation of large numbers of galaxies and accurate classification of morphologies. Large surveys such as the Sloan Digital Sky Survey (SDSS)[1] have resulted in the availability of image data for millions of celestial objects. However, manually inspecting all these images to annotate them with morphological information is impractical for either individual astronomers or small teams.

Many attempts were made to build automated classification systems for galaxy morphologies, but faced difficulties to reach the required levels of reliability and accuracy. The Galaxy Zoo project[2] was conceived to accelerate this task through the method of crowdsourcing. The original goal of the this project is to obtain the reliable morphological classification for approx 9,00,000 galaxies by online users to contribute classifications via a web platform. The project was much more successful than anticipated.

There are two recent developments in the Galaxy Zoo since its launch. First is the use of large strides in the fields of image classification and computer vision, primarily through the use of deep neural networks. Although the neural networks existed for several decades, they recently returned to the forefront of machine learning research. A significant increase in available computing power, along with new techniques such as rectified linear units and drop out regularization, made it possible to build more powerful neural network models.

Secondly, large sets of reliably annotated images of galaxies are now available as a consequence of the success of Galaxy Zoo. Such data can be used to train machine learning models thus increases the accuracy of their morphological classifications. Deep neural networks tend to scale very well as the amount of available training data increases. Neverthless it is also possible to train deep neural networks using techniques such as regularization, parameter sharing, model averaging and data augmentation.

In this project, we bring out a convolutional neural network model for galaxy morphology classification that is specifically in view of the images of the galaxies. We use both the translational and rotational symmetry in the images, and autonomously learns several levels of increasingly abstract representations of the images that are suitable for classification.

The rest of the paper is detailed as follows: we introduce the Galaxy Zoo project in section 2. We give an idea of related work in section 3. The methodology of the project is explained in the section 4. All the results and plots are given in the section 5. The paper ends with the conclusions and future scope of the project.

II. GALAXY ZOO

Galaxy Zoo is an online crowdsourcing project where users are asked to describe the morphology of galaxies based on colour images. In our project we use the image data from the galaxy zoo project. People across the web platform are asked various questions such as 'Is the galaxy looks spiral?' and 'Is the galaxy is rounded?'. The subsequent question is decided by the answer to the previous question. The classification scheme has 11 questions and 37 answers which are depicted in the table 1.

Because of the structure of the decision tree, each individual participant answered only a subset of the questions for each classification. When many participants have classified the same image, their answers are aggregated into a set of weighted vote fractions for the entire decision tree. These vote fractions are used to estimate confidence levels for each answer, and are indicative of the difficulty users experienced in classifying the image. More than half a million people have contributed classifications to Galaxy Zoo, with each image independently classified by 40 to 50 people.

[1] http://www.sdss.org/

[2] http://www.galaxyzoo.org/

	question		answers	next
Q1	Is the galaxy simply smooth and rounded, with no sign of a disk?	A1.1	smooth	Q7
		A1.2	features or disk	Q2
		A1.3	star or artifact	end
Q2	Could this be a disk viewed edge-on?	A2.1	yes	Q9
		A2.2	no	Q3
Q3	Is there a sign of a bar feature through the centre of the galaxy?	A3.1	yes	Q4
		A3.2	no	Q4
Q4	Is there any sign of a spiral arm pattern?	A4.1	yes	Q10
		A4.2	no	Q5
Q5	How prominent is the central bulge, compared with the rest of the galaxy?	A5.1	no bulge	Q6
		A5.2	just noticeable	Q6
		A5.3	obvious	Q6
		A5.4	dominant	Q6
Q6	Is there anything odd?	A6.1	yes	Q8
		A6.2	no	end
Q7	How rounded is it?	A7.1	completely round	Q6
		A7.2	in between	Q6
		A7.3	cigar-shaped	Q6
Q8	Is the odd feature a ring, or is the galaxy disturbed or irregular?	A8.1	ring	end
		A8.2	lens or arc	end
		A8.3	disturbed	end
		A8.4	irregular	end
		A8.5	other	end
		A8.6	merger	end
		A8.7	dust lane	end
Q9	Does the galaxy have a bulge at its centre? If so, what shape?	A9.1	rounded	Q6
		A9.2	boxy	Q6
		A9.3	no bulge	Q6
Q10	How tightly wound do the spiral arms appear?	A10.1	tight	Q11
		A10.2	medium	Q11
		A10.3	loose	Q11
Q11	How many spiral arms are there?	A11.1	1	Q5
		A11.2	2	Q5
		A11.3	3	Q5
		A11.4	4	Q5
		A11.5	more than four	Q5
		A11.6	can't tell	Q5

Table 1. All questions that can be asked about an image, with the corresponding answers that participants can choose from. Question 1 is the only one that can be asked of every image. The final column indicates the next question to be asked when a particular answer is given.

III. RELATED WORK

For more than two decades Machine Learning techniques and artificial neural networks have made a popular tool in astronomy research. Neural networks were initially applied for star-galaxy discrimination [1] and classification of galaxy spectra.

In the field of astronomy, the neural networks find widespread applications in the classification of galaxies. Most work in this domain proceeds by pre-processing the photometric data and then extracting a limited set of hand-crafted features that are known to be discriminative such as ellipticity, concentration, surface brightness, radii and log-likelihood values measured from various types of radial profiles.

Earlier work in this domain typically relied on much smaller datasets and used networks with very few trainable parameters (between 101 and 103). Modern network architectures are capable of handling at least 107 parameters, allowing for more precise fits and a larger morphological classification space. More recent work has profited from the availability of larger training sets using data from surveys such as the SDSS [3].

Another recent trend is the use of general purpose image features, instead of features that are specific to galaxies: the WND-CHARM feature set[5], originally designed for biological image analysis, has been applied to galaxy morphology classification in combination with nearest neighbour classifiers.

IV. METHODOLOGY

In this section, we discuss about our project and problems faced during the project. We first explain about the experimental procedure and overfitting problem which was the main drive behind our setup. This project consists of five steps as shown in the figure 1: data pre-processing, augmentation, viewpoint extraction, a convolutional neural network and model averaging.

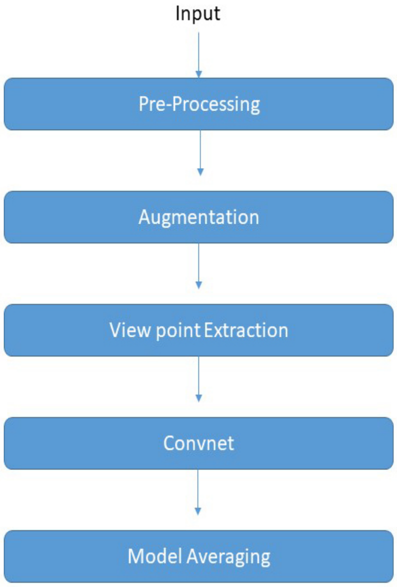
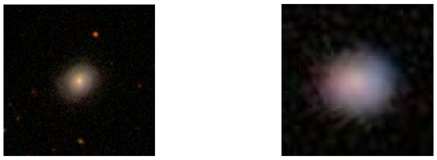


Fig. 1: Schematic Overview of the processing pipeline

A. Project Setup

We acquired the data set from the kaggle[3] that consists of 61,578 images for training and evaluation along with associated answer probabilities. The answers are gathered by the technique of crowdsourcing. We are not sure about the correctness of the answers. So we analyze the images by using the probabilties. We consider only the images that have the probability of correct answer greater than 0.5. Filter the images by using the respective probabilities. From the data set after filtering, the total images available for training are 25,188 and 6,298 for validation. The dimensions of an image is 424 x 424 pixels. A sample image from data set is shown in the figure 2a.

[3] https://www.kaggle.com/c/galaxy-zoo-the-galaxy-challenge/data



(a) Image with dimensions 424 x 424 pixels (b) Processed image with dimensions 69 x 69 pixels

Fig. 2: Sample images

B. Avoiding Overfitting

Modern neural networks typically have a large number of learnable parameters, several million in the case of our model. This is in stark contrast with the limited size of the training set, which had only 2.5 x 10^4 images. As a result, there is a high risk of overfitting: a network will tend to memorize the training examples because it has enough capacity to do so, and will not generalize well to new data. We used several strategies to avoid overfitting:

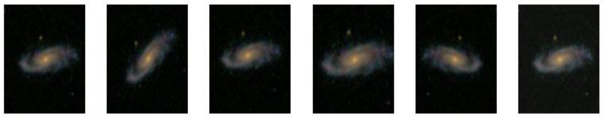
- **data augmentation:** extending the training set by randomly perturbing images in a way that leaves their associated answer probabilities unchanged;
- **regularization:** penalizing model complexity through use of dropout[2];
- **parameter sharing:** reducing the number of model parameters by exploiting translational and roarional symmetry in the input images;
- **model averaging:** averaging the predicitons of several models

C. Image Preprocessing

The filtered images were first cropped and rescaled to reduce the dimensionality of the input. It was useful to crop the images because the object of interest is in the middle of the image with a large amount of sky background(see figure 2a), and typically fits within a square with a side of approximately half the image height. We then rescaled the images to speed up training, with little to no effect on predictive performance. Images were cropped from 424 x 424 pixels to 207 x 207 pixels, and then downscaled 3 times to 69 x 69 pixels, shown in the figure 2b.

D. Data Augmentation

Due to the limited size of the training set, performing data augmentation to artificially increase the number of training examples is instrumental. Each training example was randomly perturbed in five ways, The flow which is followed is shown in the figure 3.



(a) none (b) rotation (c) translation (d) scaling (e) flipping (f) brightness

Fig. 3: Steps in Data Augmentation

- **rotation:** random rotation with an angle sampled uniformly between 0^0 and 360^0, to exploit rotational symmetry in the images.
- **translation:** random shift sampled uniformly between -4 and 4 pixels (relative to the original image size of 424 by 424 pixels) in the x and y direction. The size of the shift is limited to ensure that the object of interest remains in the capture of the image.
- **scaling:** random scaling with a scale factor sampled log-uniformly between 1.3^{-1} and 1.3.
- **flipping:** the image is flipped with a probability of 0.5.
- **brightness adjustment:** the colour of the image is adjusted as described in [2] by with two differences: the first eigen vector has a much larger eigen value than the other two, so only this one is used, and the standard deviation for the scale factor is set to 0.5. In practice this amounts to a brightness adjustment.

The first four of these are affine transformations, which can be collapsed into a single transformation together with the one used for preprocessing. This means that the data augmentation step has no noticeable computational cost. To maximize the effect of data augmentation, we randomly perturb the images on demand during training, so the models were never presented with the exact same training example more than once.

By using the data augmentation we get an additional images of 15,000. All the images were in the size of 69 x 69 pixels with all the processing done. So, now we are having a total of 40,188 for training and 6,298 for validation. We choose the 6 classes as final from the 37 answers as enumerated:

1) cigar shaped
2) completely round
3) in between
4) on edge
5) spiral
6) spiral barred

E. Network Architecture

All viewpoints were presented to the network as 45 by 45 by 3 arrays of RGB values, scaled to the interval [0; 1], and processed by the same convolutional architecture. The resulting feature maps were then concatenated and processed by a stack of three fully connected layers to map them to the 37 answer probabilities. There are six convolutional layers with square filters. The rectification non-linearity is applied after each layer[4]. The 37 values that the network produces for an input image are converted into a set of probabilities. First, the values are passed through a rectification non-linearity, and then normalized per question to obtain a valid categorical probability distribution for each question. Valid probability distributions could also be obtained by using a softmax function per question, instead of rectification followed by normalization. However, this decreased the overall performance since it was harder for the network to predict a probability of exactly 0 or 1.

During training, we used dropout[2] in all three dense layers. Using dropout was essential to reduce overfitting to manageable levels. We used 80 epochs of training in order to get the required level of accuracy.

V. RESULTS

The model training and validation accuracies are depicted in the graphs as shown in the figure 4. The model loss is shown in the figure 5. We can observe from the plots that the training accuracy increased upto a certain number of epochs (30 epochs) and then there is no improvement in the accuracy. But from the validation plot, the validation accuracy goes on increasing till the 80 epochs of training. The precision, recall, f1-score and support are calculated using the standard formulas. Their corresponding values are depicted in the plots shown in figure 6. The newtwork architecture after the modelling is shown in the figure 7. The confusion matrix is calculated after modelling the network which is shown in the figure 8. We also performed the testing using the testing images data and the accuracy of the model is 82

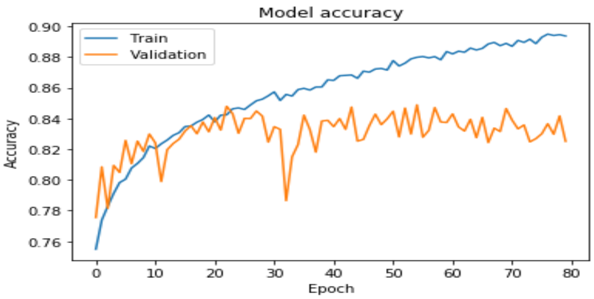


Fig. 4: Model Accuracy

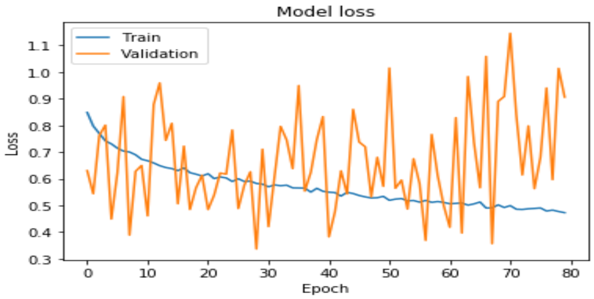
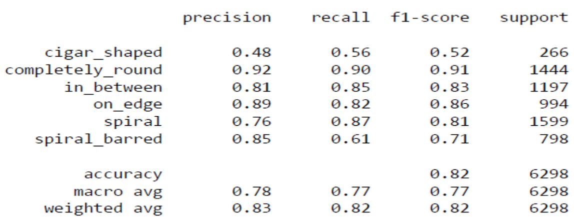


Fig. 5: Model Loss

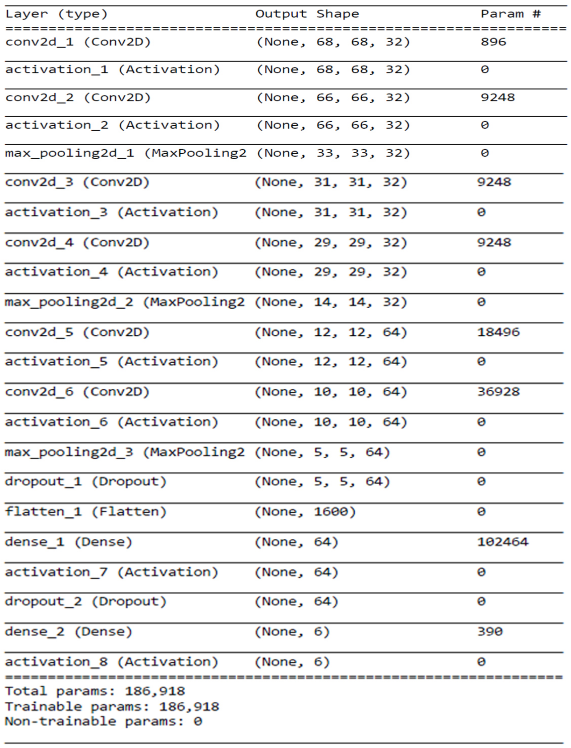


```
                  precision    recall  f1-score   support

    cigar_shaped       0.48      0.56      0.52       266
completely_round       0.92      0.90      0.91      1444
      in_between       0.81      0.85      0.83      1197
         on_edge       0.89      0.82      0.86       994
          spiral       0.76      0.87      0.81      1599
   spiral_barred       0.85      0.61      0.71       798

        accuracy                           0.82      6298
       macro avg       0.78      0.77      0.77      6298
    weighted avg       0.83      0.82      0.82      6298
```

Fig. 6: Parameters Table



```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 68, 68, 32)        896
_________________________________________________________________
activation_1 (Activation)    (None, 68, 68, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 66, 66, 32)        9248
_________________________________________________________________
activation_2 (Activation)    (None, 66, 66, 32)        0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 33, 33, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 31, 31, 32)        9248
_________________________________________________________________
activation_3 (Activation)    (None, 31, 31, 32)        0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 29, 29, 32)        9248
_________________________________________________________________
activation_4 (Activation)    (None, 29, 29, 32)        0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 32)        0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 12, 12, 64)        18496
_________________________________________________________________
activation_5 (Activation)    (None, 12, 12, 64)        0
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 10, 10, 64)        36928
_________________________________________________________________
activation_6 (Activation)    (None, 10, 10, 64)        0
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
dropout_1 (Dropout)          (None, 5, 5, 64)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 1600)              0
_________________________________________________________________
dense_1 (Dense)              (None, 64)                102464
_________________________________________________________________
activation_7 (Activation)    (None, 64)                0
_________________________________________________________________
dropout_2 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_2 (Dense)              (None, 6)                 390
_________________________________________________________________
activation_8 (Activation)    (None, 6)                 0
=================================================================
Total params: 186,918
Trainable params: 186,918
Non-trainable params: 0
_________________________________________________________________
```

Fig. 7: Network Architecture

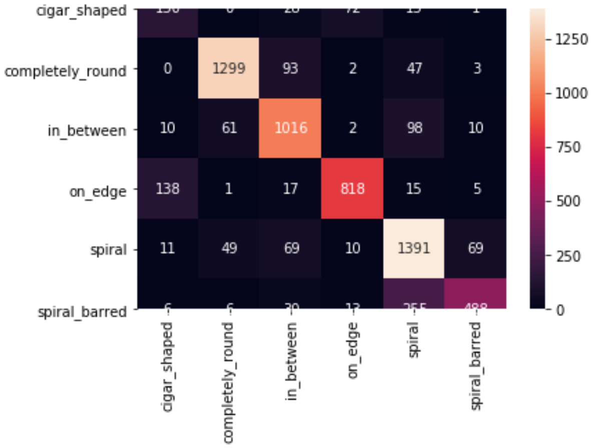


Fig. 8: Confusion Matrix

VI. CONCLUSION AND FUTURE WORK

We present a convolutional neural network for fine-grained galaxy morphology prediction, with a novel architecture that allows us to exploit rotational symmetry in the input images. The network was trained on data from the Galaxy Zoo 2 project. It can automatically annotate large collections of images, enabling quantitative studies of galaxy morphology on an unprecedented scale. Our future work includes working on larger data sets with extra features being added. Next important task is to remove red shift effect on data classification. This would help us in generalizing the model to a huge set of available unannotated data.

References

[1] E Bertin. Classification of astronomical images with a neural network. *Astrophysics and Space Science*, 217(1-2):49–51, 1994.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[3] Evan Kuminski and Lior Shamir. A computer-generated visual morphology catalog of 3,000,000 sdss galaxies. *The Astrophysical Journal Supplement Series*, 223(2):20, 2016.

[4] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[5] Lior Shamir, Nikita Orlov, David Mark Eckley, Tomasz J Macura, and Ilya G Goldberg. Iicbu 2008: a proposed benchmark suite for biological image analysis. *Medical & biological engineering & computing*, 46(9):943–947, 2008.