Classification of the most challenging LOFAR radio galaxies with Convolutional Networks: Final Report

G039 (s1771389, s1453370, s1432223)

Abstract

This project presents a convolutional neural network (CNN) tasked with classifying radio images of galaxies into one of two classes — compact or extended sources — a crucial step towards multi-wavelength cross-identification, and currently a time intensive manual task. The network architectures presented have been closely adapted from existing radio galaxy classifiers designed for similar tasks. In addition, an ensemble classification method was tested, as well as a model which includes image metadata in its classification. The data was augmented by flipping and rotating in order to achieve a balanced set of classes and make the classifier more rotation invariant. The metadata model ultimately achieved a peak test set accuracy of 93.48% on the augmented dataset compared to 89% under our hyperparameter-tuned baseline model.

1. Introduction

Radio astronomy has been embarking on a revolutionary era. The biggest radio telescope ever built, the Square Kilometer Array (SKA) is expected to be due in mid 2020's, and it will produce an unprecedented number of sources. This will allow to study the Universe as never before and it will revolutionise our understanding of galaxy formation and evolution. During the past decade, several SKA 'pathfinders' have been developing technologies to handle large amounts of data, while addressing important scientific questions. In this work we use radio sources from the LOW Frequency Array (LOFAR, van Haarlem et al., 2013). This SKA pathfinder is a phased-array telescope that spans 200 km in diameter in the Netherlands, and has stations in six other European countries, including the UK. LOFAR is conducting a wide survey across all the Northern sky, called LoTSS (LOFAR two-meter Sky Survey, Shimwell et al., 2017) which, when completed, is expected to have detected around 15 million sources. When cross-matched with other multi-wavelength surveys, radio data enables us to perform detailed statistical studies, and allows a much more complete understanding of the radio source populations (e.g. Best et al., 2005; Sabater et al., 2018, see also LoTSS-DR1 publications[1]). However, one of the major challenges that radio surveys are facing today is precisely the crucial step of cross-matching the radio sources with their optical counterparts, in particular for larger and more complex sources, where statistical matching is not reliable and the cross-matching currently requires visual inspection (e.g. Banfield et al., 2015; Williams et al., 2018).

In LoTSS-DR1 (Shimwell et al., 2018), the bulk of the sources were cross-matched using the likelihood ratio (LR) technique (explained in detail in Williams et al., 2018). This well established procedure is suitable for compact sources, such as Star Forming Galaxies or compact Active Galactic Nuclei (AGNs), where the flux-weighted mean position of the radio emission is an accurate estimate of the location at which the radio source originates, and is therefore coincident with the optical emission. While is it relatively easy to cross-match compact sources, complex-structured sources with multi-components are difficult to cross-identify and are sent to classification via visual inspection through a "private" project called LOFAR Galaxy Zoo (LGZ), in which each source was viewed by at least 5 people. Most of these sources are non-compact AGNs, i.e. galaxies that host a supermassive black hole in their centre, which produces jets that can extended hundreds of parsecs from the center of the galaxy, and that can been seen on the radio as twin-lobed sources.

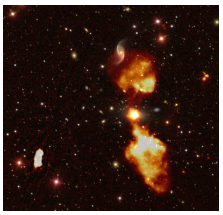

Figure 1. LOFAR image of a radio galaxy overlaid on an optical image of the sky. Credits: Cyril Tasse and the LOFAR surveys team.

In LoTSS-DR1, source detection is performed using the Python Blob Detector and Source Finder (PyBDSF, Mohan & Rafferty, 2015), which fits gaussians to pixel islands, generating components rather than true sources (i.e. physically connected sources). For that reason, these components needed to be first associated before subsequently being cross-matched with their optical counterparts. Both of these tasks were performed on LGZ (see figure 2 for an example).

The selection of the sources that could be cross-identified through LR techniques from the ones that required visual analysis was done using a decision tree. This was based on

[1] https://lofar-surveys.org/publications.html

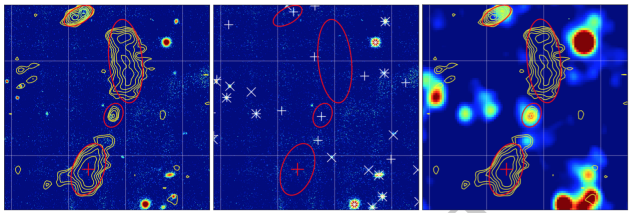

Figure 2. LGZ example of a radio source from LoTSS-DR1 composed of multiple gaussians (see Williams et al., 2018). Left: optical and radio emission; centre: PyBDSF gaussians; right: the previous two plus infrared. The yellow contours correspond to LOFAR 150-MHz flux and the green ones to FIRST 1.4-GHz emission.

the characteristics of the sources (e.g. size, number of gaussians that compose a source and distance between them, value of the LR match, etc. See Williams et al., 2018). This is a time consuming step that can be improved with a machine learning classifier. In this project we aim to improve the selection of LoTSS-DR1 most complex radio sources using a Convolutional Neural Network (CNN). These are sources that require association and which currently can only be reliably cross-matched visually. These are also potentially the most interesting sources from a scientific point of view. Furthermore, as we achieve deeper observations, we probe greater numbers of faint sources (e.g. Whittam et al., 2017), increasing therefore also the number of complex sources. Selecting and cross-identifying these sources with traditional astronomy methods will be impracticable for future surveys, in particular for sources composed by multiple gaussian components, where the robustness of LR decreases.

The research questions we tackle here are to identify the radio sources which are the most difficult to identify, as well as to compare the performance of architectures used in related galaxy classification tasks on our dataset (sec.2). In this project we explore different ways of achieving our goal by redesigning different existing classifiers and implementing a new one which incorporates PyBDSF sources metadata in the CNN architecture (see sec.3). We test the performance of the independent classifiers as well as that on an ensemble of these classifiers (sec.4). Related work and conlusions are presented on sec.5 and sec.6, respectively.

2. Data set and task

Our data was selected from the 325,694 PyBDSF sources that make up LoTSS-DR1 (Shimwell et al., 2018). The sources were selected using the value added catalogues described in (Williams et al., 2018): the original PyBDSF catalogues have the original PyBDSF source positions and the optical crossed matched catalogues the final source identification. The PyBDSF–final optical association selection process is explained in detail in https://laraalegre.github.io/lotss-dr1.

In this project we are mainly concerned with non-compact AGNs, which appear on the radio as multiple component PyBDSF sources, where different components require association (9007 multiple PyBDSF components sources). To have a balanced dataset we created a second class with the same number of sources that did not require association. For that reason, our original dataset has a relatively small number of LoTSS-DR1 sources (18014 sources). Nevertheless, we decided to include also a small group of blended sources (880 sources) that were perceived by PyBDSF as being only one source. All the sources on the class 'single' are drawn randomly from LoTSS-DR1 catalogue. An example of these sources can be seen next.

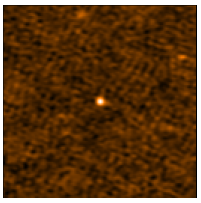

Figure 3. Compact single source

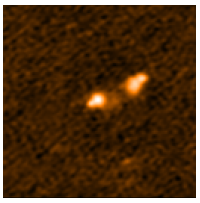

Figure 4. Multi-component source

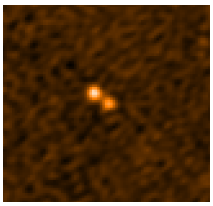

Figure 5. Blended source

These sources are then explored using 3 different partitions:

(A) 2-class partition, where blended sources are grouped with randomly selected single PyBDSF sources into one class, leaving the other class for the multiples.

(B) 2-class partition, where blended sources are grouped with multi-PyBDSF sources into one class, and single PyBDSFs into another class.

(C) 3-class partition, where multi-component sources, singles PyBDSF and blended sources make up each one of the classes.

Our motivation with partition A is to distinguish between sources that are required to be grouped visually. In Williams et al. (2018) deblending was performed using a 'deblending workflow', and tracking these sources among the singles should be easier if the multi-component sources can be identified first (even thought these sources may still require visual analysis). In partition B, we consider that efforts of visually classifying a small number of extra blended galaxies is worthwhile if we achieve better accuracy. In practice, and since the blended sources may still require visual analysis, we consider that these sources are also 'challenging' sources, and with this dataset partition we are able to tackle them. In partition C, we explore the ability to distinguish between these 3 types of sources.

In this work, we present results on the experiments on partitions B and C only.

The images were cut from the LoTSS-DR1 mosaics [2] using Montage[3] to create 128x128 pixel size images centered on the PyBDSF source positions, where each pixel corresponds

[2]Publicly available since 19th February 2019, https://lofar-surveys.org/releases.html

[3]http://montage.ipac.caltech.edu/

to 1.5 arcseconds in the sky. During the cropping procedure 17 sources failed to crop correctly, leading to a dataset of 17997 images (8121 single sources, 8996 multi and 880 deblends), for our baseline experiments.

With the results of our preliminary 'baseline' experiments on dataset A and B (where we were mainly focused on hyperparameter optimisation), we decided to carry out our experiments on revised and augmented data partitions B and C, described in sec. 3.5, as we decided these partitions have the most practical significance. We present our results in the form of accuracy percentages and confusion matrices.

3. Methodology

In this section we explain the models we use in this work and the data augmentation technique we performed. We also explain the adaptations we made to the existing classifiers used for the classifications.

3.1. Baseline - FIRST classifier

After a review of the literature, a baseline model was constructed based on the FIRST classifier implemented by Alhassan et al. (2018) – a paper whose objective we found to be most closely related to that of this report (as mentioned in fig.2). Alhassan et al. (2018) aim to classify radio images of galaxies into one of four classes. Due to the different radio wavelength targeted by the telescope used in their paper, different regions of the observed galaxies are more prominent in their images than those in this report. Their solution to this task uses a shallow CNN architecture in parallel with small amounts of data augmentation, and appears to achieve very good results. Over the four classes targeted by this classifier, it achieves an average f1-score of 0.97 (and a minimum of 0.95).

An outline of their architecture is presented in figure 6. The model is optimised by RMSprop, and uses dropout exclusively as a regularisation method.

The primary goal of the baseline model is to classify images into one of two classes, which we examine using the two dataset partitions as described in sec. 2. The only difference between the two architectures, aside from the reduced number of target classes, is that the images used for the baseline model have a size of 128×128, whereas those used by the FIRST classifier have a size of 150×150. This difference has a knock-on effect on the sizes of the subsequent feature matrices. The hyperparameters used in training the baseline model are presented in table 1. These parameters are identical to those used by Alhassan et al. (2018), except the number of epochs which has been reduced from 400 to 100 in order to decrease training time. As the baseline appears to converge to some extent by around 100 epochs, this was judged not to have a significant impact on model accuracy. We also experimented with other optimization methods, such as Adam, achieving worse results, so we used the original method.

In addition, there were several model parameters which

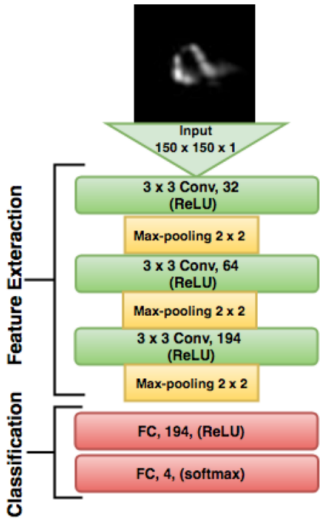

Figure 6. Network architecture of the FIRST classifier. Our adaption to the model takes 128x128 input images and outputs 2 or 3 classifications, depending on the partition we are performing the experiments on.

Hyperparameter	Value
Learning rate	0.0001
Batch size	128
Epochs	100
RMSprop β	0.9
Dropout rate	0.5

Table 1. Baseline hyperparameter values

were not explicitly stated by the authors or able to be discovered through the classifier inference source code which they provided. In this baseline, we made what we believe to be reasonable assumptions about these values. They are specified in table 2.

Hyperparameter	Value
Pool layer stride	2
Conv layer padding	1 (zero padding)
Conv layer stride	1

Table 2. Additional hyperparameter values

The convolutional layer padding and stride values of 1 are chosen because the authors mention only max pooling layers as a sub-sampling technique, implying that the convolutional layers perform no feature size reduction. The authors also make no mention of the stride of the max pooling layers in their model. We selected a stride equal to the pool size as it is a very commonly chosen value.

3.2. Radio Galaxy Zoo (RGZ/Lukic) classifier

We implemented Lukic et al. (2018) best CNN architecture classifier, used to differentiate between different radio

morphological types. These are FIRST sources (as in the work of Alhassan et al. (2018), but were classified in the Radio Galazy Zoo (RGZ). Lukic et al. (2018) achieves an accuracy of 97.4% on a 2-class problem (differentiating between compact and extended sources), and 93.5% for a 4-class problem, where the classification is made on base of the different number of PyBDSF components that make up a source. They explore exhaustively the different factors that affect the performance of the CNN, including data augmentation, and they found that a three convolutional layer architecture with bigger kernel sizes on the first convolutions (without pooling) achieves the best results. The architecture and the layer parameters used in this project can be found in figure 7 and table 3, respectively. We used our baseline hyperparameter values (table 1), instead of Lukic et al. (2018), which comprised a batch size of 8 and running for 100 epochs. We also kept our learning rate constant. The authors of this classifier also do not refer to any optimization method, so we used RMSprop, and to avoid overfitting they use dropout of 50% on the dense layers only, which we included on our classifier. ReLUs were applied after each convolutional layer.

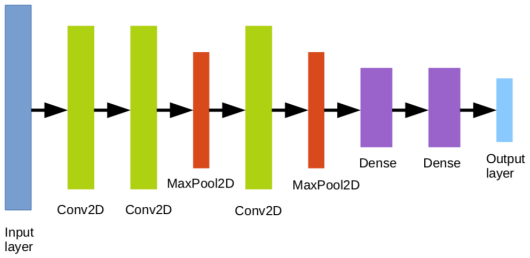

Figure 7. Lukic et al. (2018) best model architecture, used in our experiments.

Layer	Depth	Filter size	Stride
Conv2D	16	8x8	3
Conv2D	32	7x7	2
MaxPool2D	32	3x3	3
Conv2D	64	2x2	1
MaxPool2D	64	2x2	2
FC	1024	-	-
FC	1024	-	-
Softmax	2	-	-

Table 3. Layer parameters used in this experiment. We assumed a stride of the size of the kernel on the maxpooling operations, since the authors do not refer this value. We also do not use padding.

3.3. Aniyan & Thorat (A&T/Aniyan) classifier

We also performed experiments using an adaptation of Aniyan & Thorat (2017) classifier. The authors of this paper again use radio galaxies from FIRST, which they classify into different morphological types, using an architecture which outputs classification probabilities for 2 classes. They achieve a good classification accuracy of around 90-95% for 2 specific classes of radio sources (FR-I and bent-tailed), while achieving worse classification on the FR-II (i.e. sources which have independent radio components). The architecture we used on our experiments is similar to the original one (fig.8), except we removed the second convolutional layer and the subsequent max pooling layer, and we used a different number of filters. This was done to decrease memory requirements of the model. The layer parameters can be seen on table 4.

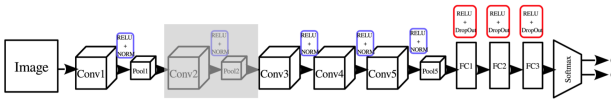

Figure 8. Aniyan & Thorat (2017) model architecture, used in our experiments, where we removed the second convolutional layer (grey shade) and max pooling layer. We also do not perform normalisation.

Layer	Depth	Filter size	Stride
Conv2D	12	11x11	1
MaxPool2D	12	3x3	3
Conv2D	64	3x3	1
Conv2D	64	3x3	1
Conv2D	32	3x3	1
MaxPool2D	32	3x3	3
FC	1024	-	-
FC	1024	-	-
FC	1024 x 2	-	-
Softmax	2	-	-

Table 4. Table of layer parameters for the adapted Aniyan & Thorat (2017) architecture used in this work.

Here, we used a number of filters of 16, 64 and 32 (instead of the original architecture ones of 96, 384 and 256, on the first, third and fourth layer, respectively). We do not present the second layer parameters, which we removed. On the FC layers we used a size of 1024 (instead of 4096). We also used padding of 1 in the first convolutional operation to maintain a correct output size for the next layers, where we use 128x128 image sizes, instead of 150x150 used in Aniyan & Thorat (2017). Here we use 50% dropout on the fully connected layers of the CNN only, and we do not perform normalisation. Other hyperparameter values are on table 1. All of the adaptations we made to the original architecture save computing memory while achieving good performance.

3.4. Metadata model

The LOFAR images which are classified in this report are accompanied by headers which contain a vast amount of metadata which is associated with each source. As the CNN-based classifiers discussed thus far operate solely on image data, this metadata has been discarded, but it could provide a useful source of information for classification purposes. We therefore designed a multi-component clas-

sifier network composed of a CNN architecture for image classification, and a multi-layer perceptron for metadata classification. The metadata we used consists of PyBDSF major and minor axis, radio fluxes (total and peak) and the number of gaussians that compose a PyBDSF source.

The versions of the FIRST classifier and the A&T classifier presented in sec.3 were used as the CNN components of this model. In both models, everything up until the final FC layer, which produces the input to the softmax layer, are taken.

The multi-layer perceptron consists of two affine layers, each followed by a ReLU layer. The outputs of both affine layers are 1024 elements wide. The output of the final ReLU layer of the MLP and that of the CNN are concatenated to form a vector. This vector is 2048 elements wide in the case of the A&T classifer, and 1218 in the case of the FIRST classifier. This vector is subsequently condensed into a 1024-element vector through an affine layer and ReLU layer, before being fed through an affine layer and softmax layer, as is done at the end of the FIRST and A&T classifiers. This is illustrated in Figure 9. All hyperparameters of this model are identical to the baseline, and dropout is performed on each of the newly introduced layers.

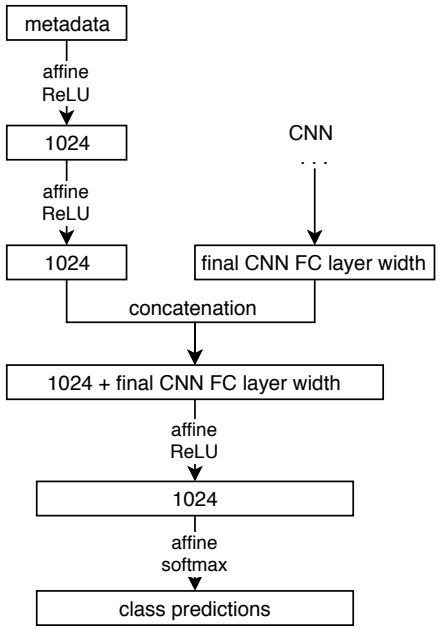

Figure 9. Incorporation of metadata into a CNN classifier. Used in combination with the FIRST and Aniyan classifiers.

3.5. Data augmentation

In order to rectify the our unbalanced classes, data augmentation has been done on the classes with smaller number of examples. With augmentation we are able to obtain a dataset many times larger than our original dataset. Performing random rotations without the presence of black corners is possible due to the fact that our original images are obtained from a catalogue and can be regathered so that they are bigger (for example from 128x128 to 256x256) without loss of resolution but by including a larger part of the sky. After a random rotation, the black corners can be removed by cropping the image back to 128x128, because we only care about the centre object. This way we can increase our dataset without including noise from the corners and without the possibility of our classifier learning to correlate the black corners with a particular class. We are also able to introduce varying level of noise, although the original images did have quite a large variation due to different observation times of the telescope and the different noise levels for the various parts of the sky. Therefore, no noise augmentation was done.

In order to balance the dataset, random rotations and flips were performed on the deblend images until the number of deblends reached 8000 from the original 880. For each extra image required, a random image from the dataset was taken, was flipped with a 50% probability, and was rotated by a uniformly randomly chosen angle in $[0, 360)$. This could be done by extracting larger (256×256) images from the raw mosaics, performing the required rotations on the larger images, and cropping the central 128×128 pixels, thus avoiding blank corners appearing in the images as a result of these rotations.

After augmenting the 'deblends' we created dataset C with approximately 8000 'singles', about 8000 'multi', and about 8000 'deblends'(there was a very small number of corrupted images which had to be removed).

Dataset B included approximately 8000 'singles', about 4000 'multi', and about 4000 'deblends'.

We used training, validation and test sets of 80%, 10% and 10%, respectively for both partitions, resulting in 12672/1536/1536 split for dataset B and 19072/2304/2304 split for C.

4. Experiments

4.1. Baseline experiments

A series of experiments were performed on the baseline model in order to investigate a range of possible hyperparameter values. These included modifications to the learning rate, batch size, dropout rate, and type of pooling layer. For each experiment, a single hyperparameter was adjusted from the baseline values, in order to avoid performing an expensive grid search, and was executed on both the A and B partitions of the non-augmented dataset, described in sec. 2. After the data augmentation and revision, the hyperparameter configuration with the highest performance was trained on partitions B and C, achieving a test set accuracy of 88.99% on B and 79.08% on C.

4.2. Experiments using other existing classifiers

In addition to the FIRST classifier, we examine the performance of the Lukic et al. (2018) and Aniyan & Thorat (2017) classifiers on LOFAR data. These classifiers perform well on similar tasks on FIRST data (Becker et al.,

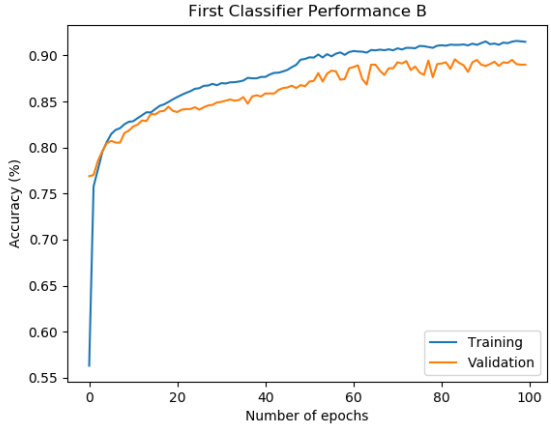

Figure 10. Baseline FIRST network training on partition B.

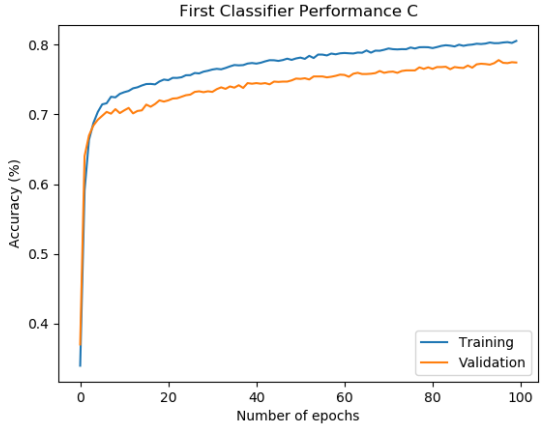

Figure 11. Baseline FIRST network training on partition C.

1995). Despite being also radio data, FIRST probes higher energies, which are usually associated with radio emission from the AGN core, while lower frequencies from LOFAR are associated also with lobe emission. For that reason, we aim to understand how well they perform on our data and for our specific task. Our results for a 2-class and a 3-class problem are presented next.

The Lukic et al. (2018) classifier was also used on the augmented dataset. This architecture achieved the same results as the FIRST classifier on B, giving a test set accuracy of 89.0% and slightly worse than FIRST with 78.3% on C, using the same baseline hyperparameters. However, it showed potential for slight improvement on C, as the training curves for C haven't completely flattened out at the end of the 100 epochs.

The classifier based on that of Aniyan & Thorat (2017) performed the best of all the three pure CNN architectures examined, resulting in a test set accuracy of 91.4% on partition B, and 85.8% on partition C. This would imply that this task can benefit from a deeper network than those used by Lukic et al. (2018) and Alhassan et al. (2018). It may also have benefited from the much larger kernel sizes of this network.

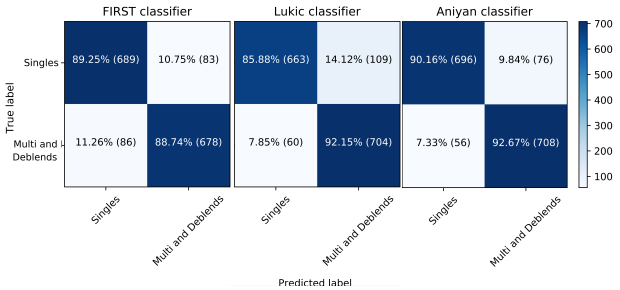

Figure 12. Confusion matrix of the 3 classifiers on partition B.

In this experiment we also used an ensemble classifier, by taking the majority of the votes (Dietterich, 2000) for the B partition. Here we combine the binary predictions obtained with these 3 classifiers to obtain a final prediction. We expected that this ensemble would improve the robustness of the classification, while allowing to identify odd inputs. However, the results are slightly worse than when using the Anyian classifier alone, probably due to influence from the RGZ and FIRST classifiers, which perform worse on this task, in particular when single sources are misclassified as multiples or deblends. Nevertheless, these are the misclassifications which give less concern, since these sources would be sent to visual analysis and get the correct classifications anyway. On the other hand, when a multiple PyBDSF source or a blended source is misclassified as a single, it will be sent to statistical techniques, and will never be visually inspected. In this case the FIRST classifier performs the worst of the 3. Somehow, the predictions are not so biased on the ensemble due to the votes of the RGZ and A&T classifier. The ensemble's accuracy on the test set is 90.54%.

We then analyse the results for data partition C, and the ability of distinguish between the 3 different types of sources we have on our dataset. The results are presented in Figure 14.

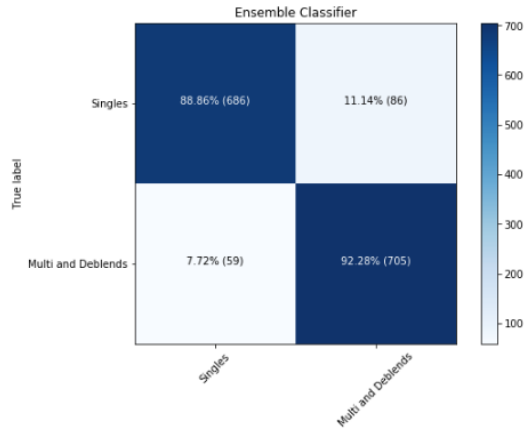

Figure 13. Confusion matrix of the ensemble of the 3 classifiers on partition B.

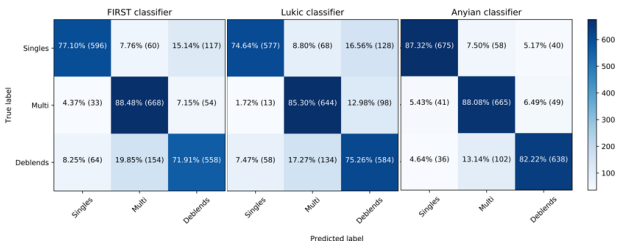

Figure 14. Confusion matrix of the 3 classifiers on partition C.

For each of the classifiers, a blended source has much greater probability of being classified as a multiple component source than as a single source. Again, the FIRST classifier has the worse performance and the Anyian classifier the best one. In the Lukic classifier most of the misclassified multiple sources are classified as blended sources and just a few percent as singles. This is an encouraging result, single blended sources will be analysed further. The Anyian classifier is the only one achieving classifications above 80% in all the classes, and it is interesting to note that the Lukic and the FIRST classifiers are significantly weaker classifiers for single and blended sources but not for multi-component sources.

4.3. Metadata model

From each LOFAR image file, 5 metadata elements were extracted and turned into a feature vector which acts as a second input to the classifier for that image. These features were the major and minor axes of the source in question, the peak and average flux of the image, and the number of gaussians detected in the image by PyBDSF. The inclusion of this metadata improved the test set accuracy on both partition B and C for the best performing A&T classifier, raising them from 91.41% to 93.49% and from 85.85% to 89.63% respectively. The training and validation set accuracies for the Aniyan version are plotted in fig. 15, and its confusion matrices in fig. 16 and 17. The validation set accuracy is clearly far more unstable than that of the baseline, and it appears that it could even benefit from further training, although to keep measurements consistent, we only trained for 100 epochs.

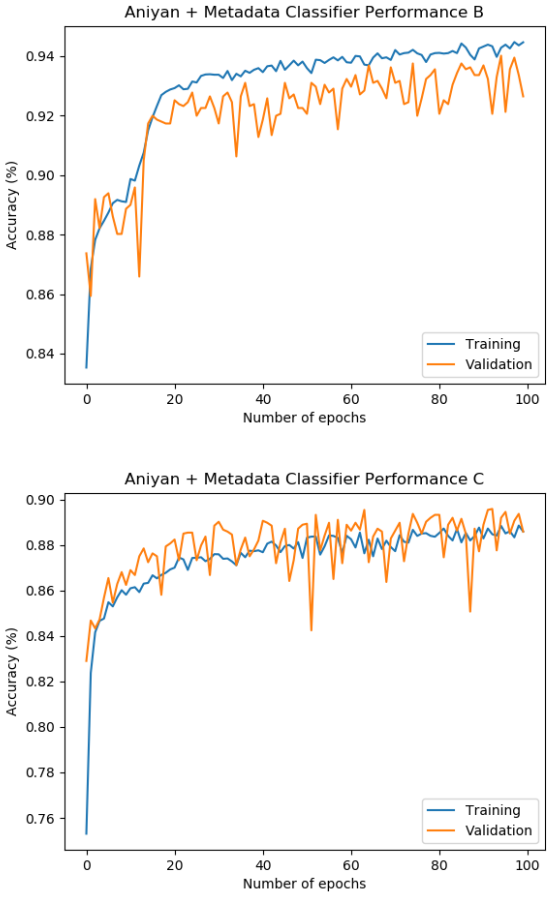

Figure 15. A&T + metadata training on partitions B and C.

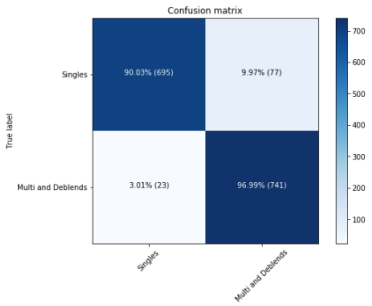

Figure 16. Confusion matrix for the A&T classifier with metadata on partition B.

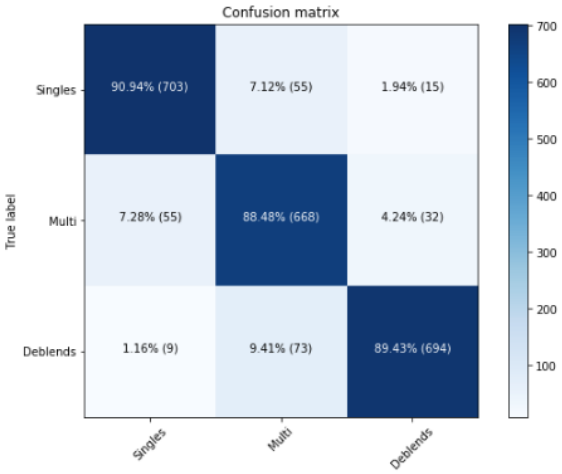

Figure 17. Confusion matrix for the A&T classifier with metadata on partition C.

Concatenating the metadata also showed improvement when using the FIRST classifier.

5. Related work

Deep Learning (CNN based) sources finders already exist (such as Vafaei Sadr et al., 2019; Gheller et al., 2018), outperforming PyBDSF in general. Classifiers that perform both source identification and classification, such as Wu et al. (2018), show the feasibility of applying computer vision and object detection techniques to radio data, in particular with training with a large sample of extended galaxies. However, they still face the same challenges of PyBDSF, in particular for multi-component sources, where their capability of separating unrelated close sources from multi-component sources needs improvement.

CNNs have been also applied to radio morphology classification, but they are focused mainly on differentiate between different morphological types. These CNNs are in general shallow, having 3 or 4 convolutional layers. For example, Aniyan & Thorat (2017) used an ensemble of CNN classifiers to predict three distinct classes of extended radio sources from FIRST, which they classify into 3 distictnt morphological types: FR-I, FR-II and bent-tailed. However, they fail on correctly classifying one type of twin-lobed sources (FR-II), and they do not attempt to distinguish between extended and point sources. Alhassan et al. (2018) used a CNN to distinguish between the same three classes mentioned above, but they include compact sources. They achieve great results, except for the FR-II class. While in FR-I the radio jet emission is connected to the galaxy core, in FR-II it is not. This can be interpreted by the CNN as being more than one source, which leads to classification failure.

(Lukic et al., 2018) used a CNN for classifying radio galaxies from RGZ. Their goal was to distinguish between compact and extended sources with multi-components directly from the radio images, however their CNN was not able to classify the multi-component extended sources effectively.

Overall source finder and classification algorithms for radio data are not reliable yet. Automatic galaxy classification systems are particularly more mature for optical wavelengths than for radio, achieving human accuracy classification (see, e.g Dieleman et al., 2015). On the radio the most challenging sources will still require visual inspection. Currently there is no work published on LoTSS galaxy identification and/or classification using machine learning.

6. Conclusions

Our experiments showed that the baseline CNN-based classifier presented in the interim report of this project could be improved upon by examining deeper networks and incorporating metadata into training and classification. Through these techniques, test case performance was raised from 91.4% to 93.48% in the 2-class partition B task, and from 85.85% to 89.62% in the 3-class partition C task.

The work presented in this report could possibly be extended by testing the performance of more architectures, increasing the number of rotated and flipped training images, or changing the ways in which these augmentations were performed. In particular, ideas could be taken from the model presented by Dieleman et al. (2015) for the morphological classification of images of galaxies, which uses a novel data augmentation methodology to achieve state of the art performance on that particular task.

The results in this work are summarised below:

CLASSIFIER	Dataset B	Dataset C
FIRST	89.0%	79.08%
RGZ	89.0%	78.34%
A&T	91.4%	85.85%
Ensemble(FIRST/RGZ/A&T)	90.56%	N/A
A&T + Metadata	**93.48%**	**89.62%**

Table 5. Classifier accuracies.

CLASSIFIER	Singles	Multi	Deblends
FIRST	77.1%	88.48%	71.91%
RGZ	74.64%	85.30%	75.26%
A&T	87.32%	88.08%	82.22%
A&T + Metadata	90.94%	88.48%	89.43%
Average	82.5%	87.585%	79.705%

Table 6. Source type accuracies across the different classifiers on dataset C and their average.

References

Alhassan W., Taylor A. R., Vaccari M., 2018, Monthly Notices of the Royal Astronomical Society, 480, 2085
Aniyan A. K., Thorat K., 2017, ApJS, 230, 20
Banfield J. K., et al., 2015, MNRAS, 453, 2326

Becker R. H., White R. L., Helfand D. J., 1995, ApJ, 450, 559

Best P. N., Kauffmann G., Heckman T. M., Brinchmann J., Charlot S., Ivezić Ž., White S. D. M., 2005, MNRAS, 362, 25

Dieleman S., Willett K. W., Dambre J., 2015, Monthly notices of the royal astronomical society, 450, 1441

Dietterich T. G., 2000, in International workshop on multiple classifier systems. pp 1–15

Gheller C., Vazza F., Bonafede A., 2018, Monthly Notices of the Royal Astronomical Society, 480, 3749

Lukic V., Brüggen M., Banfield J. K., Wong O. I., Rudnick L., Norris R. P., Simmons B., 2018, MNRAS, 476, 246

Mohan N., Rafferty D., 2015, Astrophysics Source Code Library

Sabater J., et al., 2018, arXiv preprint arXiv:1811.05528

Shimwell T. W., et al., 2017, A&A, 598, A104

Shimwell T., et al., 2018, arXiv preprint arXiv:1811.07926

Vafaei Sadr A., Vos E. E., Bassett B. A., Hosenie Z., Oozeer N., Lochner M., 2019, Monthly Notices of the Royal Astronomical Society, 484, 2793

Whittam I. H., Jarvis M. J., Green D. A., Heywood I., Riley J. M., 2017, Monthly Notices of the Royal Astronomical Society, 471, 908

Williams W., et al., 2018, arXiv preprint arXiv:1811.07927

Wu C., et al., 2018, Monthly Notices of the Royal Astronomical Society, 482, 1211

van Haarlem M. P., et al., 2013, A&A, 556, A2