Pulsar Candidate Selection by Assembling Positive Sample Emphasized Classifiers

Conference Paper · December 2016
DOI: 10.1109/CIS.2016.0036

Ping Guo
Beijing Normal University

Pulsar Candidate Selection by Assembling Positive Sample Emphasized Classifiers

Yao Yao*, Xin Xin*, Ping Guo*†,
*School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China
Email: {bityaoyao, xxin}@bit.edu.cn
†Image Processing and Pattern Recognition Laboratory, Beijing Normal University, Beijing 100875, China
Email: pguo@ieee.org

***Abstract*—Pulsar candidate selection identifies prospective observations of modern radio pulsar surveys for further inspection in search of real pulsars. Typically, human experts visually select valuable candidates and eliminate radio frequency interference or other noises. Recently, machine learning methods are adopted to automate this task, which saves human labor and makes it possible for processing millions of observations efficiently. Considering the number of positive training samples are relatively too small and the cost of incorrectly labeling a real pulsar candidate as negative is large, we propose a novel hierarchical candidate-sifting model by emphasizing the cost of incorrect prediction of positive samples and assembling multiple classifiers trained with different weighting parameters. Experiments on three pulsar selection datasets demonstrate our proposed method improves the pulsar-sifting performance a lot according to several standard evaluation metrics.**

***Keywords*-Pulsar candidate selection; positive sample emphasized; assemble classifiers;**

I. Introduction

Searching for real pulsars among millions of pulsar candidates is an important and meaningful task in astronomy. As radio pulsars are used as probes to study physics and astrophysics, the more pulsars are discovered, the more we are able to study [1]. Modern radio pulsar surveys, such as the Pulsar Arecibo L-band Feed Array [2], the Green Bank North Celestial Cap [3], and the High Time Resolution Universe [4], produce millions of pulsar candidates. Candidates are denoted as a set of diagnostic values and two-dimensional graphical representations. Fig. 1 gives examples of a real pulsar and a non-pulsar candidate. And most of these candidates are just human-created radio frequency interferences (RFI) or other forms of noises. Only a small fraction of them are real pulsars. Apparently, it is impractical for human experts to visually inspect all these millions of candidates.

Fortunately, a range of scoring or learning methods have been developed to select prospective candidates automatically for further inspection. For example, Lee *et al*. [1] linearly combine six different quality factors to compute a score for each candidate. Zhu *et al*. [5] take raw candidates plots as features, train multiple classifiers including support vector machines (SVM), artificial neural networks (ANN) and convolutional neural network (CNN) and combine these classifiers with a logistic regression classifier. Morello *et al*. [6] design six features and train an ANN for binary classification. These methods save a lot of human labor and help astronomers discovering some new real pulsars.

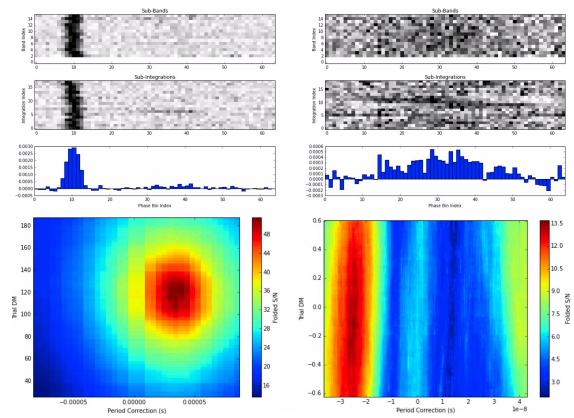

Figure 1. An illustration of pulsar and non-pulsar candidate raw representations. The left column illustrates a diagnostic histogram and three two-dimensional graphical representations of a real pulsar. The right column illustrates these of a non-pulsar candidate.

But all these previous learning-based methods leave out an important fact that incorrectly labeling a real pulsar candidate as negative costs more than incorrectly labeling a RFI or noise candidate as positive. Predicting an interference candidate as positive would increase some amount of human inspection labor. However, classifying a real pulsar as negative may lead to severe outcome of missing new discoveries, which should be avoided.

In this work, we propose a novel hierarchical candidate-sifting model (HCSM): In the preprocessing step, hand-crafted features are extracted from raw representations for all candidates. In the first layer, a set of classifiers are trained by emphasizing the cost of incorrect prediction of positive samples by weighting more in the loss function. In the second layer, an assembling classifier is trained by regarding output of each classifier in the first layer as a feature. During testing scenario, a candidate is first predicted by the classifiers in the first layer. A prediction value vector is formed and then feed into the assembling classifier in the second layer, which determines the final prediction of the candidate.

978-1-5090-4840-3/16 $31.00 © 2016 IEEE
DOI 10.1109/CIS.2016.35

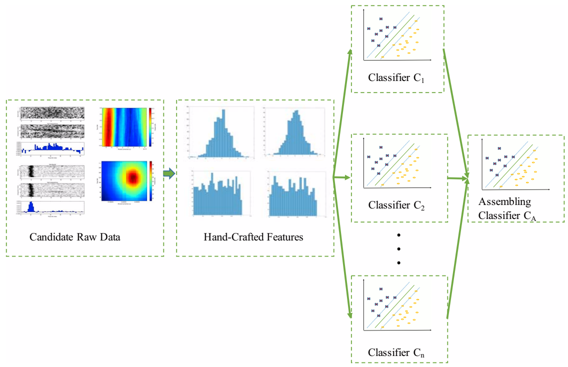

Figure 2. An illustration of our proposed hierarchical candidate-sifting model: 1) Hand-crafted features are extracted from raw candidate representations; 2) A set of positive samples emphasized classifiers $(C_1, C_2, \ldots, C_n)$ are trained on the training set with hand-crafted features; 3) An assembling classifier C_A is trained by taking output of each classifier in the previous layer as a feature.

Our proposed model is illustrated in Fig. 2. Specifically, raw candidate representations are firstly processed into some discriminative hand-crafted features. These features are usually statistics obtained from raw representations shown in Fig. 1. For example, Lyon *et al.* [7] designed eight candidate features from diagnostic histograms. Thornton *et al.* [8] proposed 22 features for training candidate selection models. In the first layer of our hierarchical model, a group of classifiers are learned. Each classifier is trained with a different weight on positive samples in the loss function. Weights of positive samples are set to be larger than these of negative ones, which makes the classifiers in the first layer emphasize more on positive training samples. The second layer of our hierarchical model assembles the outputs of the first layer and determines the final predictions of candidates.

Experiments on three pulsar candidate datasets, *HTRU 1, HTRU 2* and *LOTASS 1* [7], demonstrated that our proposed hierarchical candidate-sifting model largely improves candidate selection performance according to several standard evaluation metrics. The main contributions of our work are summarized as follow: 1) A positive sample emphasized classifiers for pulsar candidate selection is proposed. As incorrect prediction of positive samples costs more in loss function, our positive sample emphasized classifiers are less likely to lose prospective real pulsars. 2) A hierarchical candidate-sifting model assembles outputs of a group of positive sample emphasized classifiers, which further improves performance of pulsar candidate selection.

II. Related Work

Pulsar Candidate Selection: In order to discover new pulsars, modern radio pulsar surveys are designed to search periodic broadband signals which exhibits signs of dispersion. These signals are recorded as a group of diagnostic histograms or graphical representations, and referred to as a pulsar candidate. Pulsar candidate selection is the process of determining prospective candidates for further investigation and excluding RFI and other forms of noises. Recently, automatic candidate-sifting methods based on scoring or machine learning show good performance. Lee *et al*. [1] proposed Pulsar Evaluation Algorithm for Candidate Extraction, which combines six different quality factors to compute a score, and then ranks the candidates based on scores. Zhu *et al*. [5] take diagnostic plots as input features, train multiple classifiers including SVM, ANN and CNN, and combine these classifier with a final logistic regression classifier. Morello *et al*. [6] proposed six features and an ANN for binary classification. Thornton [8] designed 22 features and trained an ANN for pulsar candidate selection.

Candidate Features: Machine learning algorithms take feature representations of samples as input, and learn models for prediction. In pulsar candidate selection task, a variety of features have been proposed in the literature. Zhu *et al*. [5] regard diagnostic plots, such as summed profile, time versus phase plot, frequency versus phase plot and dispersion-measure (DM) curve, as features of samples. These features are first down-sampled or interpolated to a uniform size, then normalized to an appropriate numeric range and finally used for training SVM, ANN and CNN. Morello *et al*. [6] created six features: signal-to-noise ratio (SNR) of the folded profile, intrinsic equivalent duty cycle of the pulse profile, ratio between barycentric period and dispersion measure, validity of optimized dispersion measure, persistence of signal through the time domain and root-mean-square distance between the folded profile and the sub-integrations. Lyon *et al.* [7] designed eight statistic features from folded profile and DM-SNR curve. Thornton *et al.* [8] obtained 22 statistic features from diagnostic plots.

Biased Classifiers: Xing *et al.* [9] proposed a formulation applied to software reliability prediction which controls two types of error ratio by adjusting corresponding penalty factors. Our formulation of positive sample emphasized classifiers is very similar to this one. But in our work, the main difference is that we proposed a novel hierarchical model which assembles a group of classifiers. The model not only improves the performance but also solves the difficulty of choosing the best weight parameter.

III. Method

In this section, our HCSM will be described in detail, including positive sample emphasized classifier formulation, and final classifier assembling.

A. Positive Sample Emphasized Classifiers

In pulsar candidate selection task, incorrectly classifying a real pulsar as negative leads to missing discovery of a new pulsar. However, incorrectly labeling a non-pulsar candidate as positive increases some amount of human labor for further identification. Intuitively, missing a new pulsar is a much worse outcome than adding some human investigations. Motivated by this, we propose Positive Sample Emphasized Classifier (PSEC) to distinguish real pulsars from RFI or noises.

The basic idea of our PSEC is realized by weighting more for positive samples in the loss function in the learning process. Given positive training data feature vectors $\mathbf{x}_i \in \mathbf{R}^d, i = 1, \ldots, l_+$, whose labels are $y_i = 1, i = 1, \ldots, l_+$, and negative training data $\mathbf{x}_j \in \mathbf{R}^d, j = 1, \ldots, l_-$, whose labels are $y_j = -1, j = 1, \ldots, l_-$. d is the feature vector size. l_+ and l_- are the number of positive and negative training samples, respectively. Based on Support Vector Machine [10] large margin classification, PSEC loss function is defined. Then the optimization problem of learning a PSEC is formulated as follow:

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi,\zeta} \frac{1}{2}\mathbf{w}^T\mathbf{w} + w_1 C \sum_{i=1}^{l_+} \xi_i + w_2 C \sum_{j=1}^{l_-} \zeta_j, \\
s.t. y_i \left(\mathbf{w}^T \phi\left(\mathbf{x}_i\right) + b\right) \geq 1 - \xi_i, \\
y_j \left(\mathbf{w}^T \phi\left(\mathbf{x}_j\right) + b\right) \geq 1 - \zeta_j, \\
\xi_i \geq 0, i = 1, \ldots, l_+, \\
\zeta_j \geq 0, j = 1, \ldots, l_-,
\end{aligned} \tag{1}
$$

where $\mathbf{w}, b$ are model parameters supposed to be learned, $C > 0$ is the regularization parameter, $\phi(\mathbf{x})$ explicitly maps $\mathbf{x}$ into a higher-dimensional feature space, ξ_i and ζ_j indicate the tolerance of incorrectly predicting positive and negative samples, respectively. $w_1 > 0$ and $w_2 > 0$ are weights of positive and negative samples in the loss function, respectively.

The optimization problem can be solved in the same way as that of SVM learning. It's a classic quadratic problem. During training, parameters C, w_1 and w_2 are given first. Here, PSEC sets $w_1 > 1$ and $w_2 = 1$. With larger w_1, positive training samples are emphasized more in the learning process.

B. Hierarchical Candidate-Sifting Model

A hierarchical candidate-sifting model is proposed to combine a committee of PSEC's for pulsar candidate selection. Instead of choosing a best w_1 for a single PSEC by cross-validation on a specific training set, we first train a group of PSEC's with different w_1's, respectively. Then this group of PSEC's are combined together to make an assembled final prediction.

1) Framework: The framework of our HCSM is illustrated in Fig. 2. In preprocessing, hand-crafted features are extracted from candidate raw data for each sample. These hand-crafted features are designed to be discriminative for pulsar candidate selection with machine learning models. They are usually statistics, such as mean or standard deviation of histograms [7] and correlation coefficients of two-dimensional raw representations [8]. In the first layer of HCSM, a group of PSEC's, $C_1, \cdots, C_n$, are learned on the training data with a set of different w_1's. These PSEC's form a committee of models for predicting. In the second layer of HCSM, an assembling classifier is trained by taking the outputs of the first layer as input features. The output of the second layer determines the candidate's type.

2) Assembling Classifiers: The final assembling method is a linear classifier trained on the outputs of the first layer. In our HCSM, a linear SVM is adopted.

C. Learning and Inference

The learning of HCSM is implemented in a layer-by-layer way. In the first stage, PSEC's in the first layer are trained on the same training set with different given w_1's. As $w_2 = 1$, the last parameter C is chosen by cross-validation on a validation set for each PSEC, respectively. After the PSEC's are obtained in the first layer. Then, the assembling linear SVM can be learned by regarding the first layer outputs as features. The parameter for the final classifier can also be selected by cross-validation technique.

The inference is a forward propagation process. A pulsar candidate generates hand-crafted features. The features are feed into the learned HCSM model. Then, the model predicts whether this puslar candidate is a promising one or RFI noises.

IV. Experiment

In this section, we'll introduce experiments and discuss results on three pulsar candidate datasets. First, some preliminaries are described, including datasets, evaluation metrics and some implementation details. Then, results are demonstrated and analyzed.

A. Preliminaries

1) Datasets: The models are trained and tested on three pulsar candidate datasets. *HTRU 1*, a candidate dataset from the intermediate Galactic latitude area of the HTRU survey, was introduced by Morello *et al*. [6]. It precisely comprises 1,196 known pulsar candidate observations (denoted as positive) from 521 distinct pulsars, and 89,996 non-pulsar candidates (denoted as negative). *HTRU 2*, which consists of 1,639 pulsar and 16,259 non-pulsar candidates, was produced by Lyon *et al*. [7]. *LOTAAS 1*, a relatively smaller dataset [7], contains 66 pulsar and 4,987 non-pulsar candidates. The number of examples in each dataset are listed in Table. I.

2) Evaluation Metrics: In our experiments, we follow the evaluation metrics in [7]. They are *Accuracy, False Positive Rate(FPR), G-mean, Precision, Recall, F-score*, and *Specificity*. As a basis, the confusion matrix of binary classification is presented in Table. II. *Accuracy*, $(TN + TP)/(TN + FP + FN + TP)$, defines the overall classification accuracy. When the number of non-pulsar negative examples in the test set is overwhelming, high accuracy

Table I
NUMBER OF EXAMPLES IN EACH PULSAR CANDIDATE DATASET USED

Dataset	#Positive	#Negative	#Total Examples
HTRU 1	1,196	89,996	91,192
HTRU 2	1,639	16,259	17,898
LOTAAS 1	66	4,987	5,053

Table II
BINARY CLASSIFICATION CONFUSION MATRIX

Outcomes	Prediction -	Prediction +
Groundtruth -	True Negative (TN)	False Positive (FP)
Groundtruth +	False Negative (FN)	True Positive (TP)

can also be achieved by just labeling all as non-pulsar. That's why other metrics are needed. *FPR*, $FP/(FP+TN)$, is the fraction of negative examples incorrectly predicted as positive. Smaller FPR is better, as less human labor is required to investigate on non-pulsar candidates. *G-mean*, $\sqrt{(TP/(TP+FN))*(TN/(TN+FP))}$, is the ratio between positive and negative accuracy. *Precision*, $TP/(TP+FP)$, describes the ratio of retrieved examples that are positive. *Recall*, $TP/(TP+FN)$, describes the ratio of positive examples retrieved in the test set. *F-score*, $2*(Precision*Recall)/(Precision+Recall)$, considers precision and recall together. *Specificity*, $TN/(TN+FP)$, defines the ratio of negative examples retrieved in the test set. All metrics range from $[0,1]$. Larger numbers mean better performance, except for FPR which is the opposite.

3) Implementation Details: Considering there are more negative examples in each dataset, we first randomly sample the same number of negative examples as positive ones. We use 40% of all positive and selected negative examples for training, 30% for validation and 30% for testing. And the unselected negative ones are collected into the testing set. Eight features in [7] and 22 features in [8] are adopted and concatenated into a 30-dimensional feature vector for each example. Feature scaling technique is also used to rescale each feature into $[0,1]$ range. Pulsar Feature Lab Tools [7] are leveraged to extract these hand-crafted features. Then these 30-dimensional features are explicitly mapped into a higher space by explicit feature maps [11] with Chi-Square kernel. The parameters for training PSEC's are C, w_1 and w_2. We set $w_2=1$. In the first layer of HCSM, we train a group of PSEC's with distinct $w_1 \in \{2,3,4,5,10,15,20,25,30,100,150,200,400\}$. For each PSEC, we choose the best C from $\{1e-5,1e-4,1e-3,1e-2,1e-1,1,1e1,1e2,1e3,1e4,1e5\}$ by cross-validation technique on the validation set, respectively. The parameter C of the assembling linear SVM in the second layer of HCSM is also chosen is this way. The LIBSVM library [12] is used in our experiments.

B. Experimental Results

The main results on the three datasets are shown in Table. III. On each dataset, evaluation metrics of six models are demonstrated in the table. A linear SVM is learned as a baseline, where $w_1=1$. Performance of four representative PSEC models are presented, where model $PSEC\_N$ denotes a PSEC model with $w_1=N$. And at last, evaluations of our proposed HCSM model is listed is the table, the boldface font is used to represent the best results.

1) Performance of Positive Sample Emphasized Classifiers: Most of the PSEC models outperform the baseline SVM a lot on each dataset, according to *F-score*, which considers *precision* and *recall* together. For example, on *HTRU 1*, all listed models, *PSEC_2, PSEC_4, PSEC_10, PSEC_100*, have higher *F-scores*. However, for some models with a bad choice of w_1, such as *PSEC_15* on *HTRU 2* and *PSEC_2* on *LOTAAS 1*, perform worse than the baseline. So the choice of w_1 is important for a single PSEC learning.

In our experiments, we empirically tried a branch of $w_1 \in \{2,3,4,5,10,15,20,25,30,100,150,200,400\}$ to train PSEC classifiers. From Table. III, we can see that the number of recalled positive samples increases as w_1 becomes larger on all three datasets. The best *F-scores* are achieved when $w_1=2$ on *HTRU 1* and *HTRU 2*, and $w_1=15$ on *LOTASS 1*. The best w_1 for a single PSEC varies on different datasets.

2) Performance of Hierarchical Candidate-Sifting Models: The performance of our proposed HCSM model on three datasets is better than all other models, with respect to *accuracy, FPR, precision, F-score, and specificity*. For example, on *LOTASS 1*, HCSM reaches the best results with *accuracy=1, FPR=0, precision=1, F-score=1, and specificity=1*. The HCSM model hierarchically combines a committee of PSEC classifiers, which are learned with different choices of w_1's. As the value of w_1 is significant to the classification capability of PSEC models, HCSM first learns a group of PSEC's with $w_1 \in \{2,3,4,5,10,15,20,25,30,100,150,200,400\}$. Then HCSM model improves performance over every single PSEC by training an assembling classifier to combine all PSEC's in the committee.

V. CONCLUSION

Motivated by the characteristic of pulsar candidate selection that incorrect prediction of positive examples costs more, we proposed the Positive Sample Emphasized Classifiers and Hierarchical Candidate-Sifting Models. By weighting more on positive examples in the learning process with the right choices of weight w_1, PSEC works better than the baseline SVM. Our HCSM combines weak learners with an assembling classifier. The proposed hierarchical model not only improves the pulsar candidate selection performance, but also avoid the necessity of choosing the best w_1 for a single PSEC.

Table III

EVALUATION METRICS OF DIFFERENT MODELS ON THREE DATASETS. SVM IS A BASELINE LINEAR SVM, WHERE $w_1 = 1$. $PSEC\_N$ IS A PSEC MODEL, WHERE $w_1 = N$. $HCSM$ IS OUR PROPOSED HIERARCHICAL MODEL ASSEMBLING A GROUP OF PSEC MODELS. THE NUMBERS INDICATING BEST PERFORMANCE ARE HIGHLIGHTED ON EACH DATASET, RESPECTIVELY.

Datasets	Models	Accuracy(%)	FPR(%)	G-mean(%)	precision(%)	recall(%)	F-score(%)	Specificity(%)
HTRU 1	*SVM*	99.75	0.25	99.60	61.76	99.44	76.20	99.75
	PSEC_2	99.98	0.01	99.02	97.51	98.05	97.78	99.99
	PSEC_4	99.97	0.02	99.15	95.41	98.33	96.84	99.98
	PSEC_10	99.96	0.03	99.56	92.23	99.16	95.57	99.97
	PSEC_100	99.91	0.09	**99.82**	82.30	**99.72**	90.18	99.91
	HCSM	**99.99**	**0**	98.74	**99.15**	97.49	**98.31**	**100.00**
HTRU 2	*SVM*	97.72	2.09	**94.83**	58.80	**91.85**	71.70	97.91
	PSEC_2	99.03	0.57	92.88	83.20	86.76	84.95	99.43
	PSEC_4	98.71	0.95	93.35	75.13	87.98	81.05	99.05
	PSEC_10	98.34	1.41	94.53	67.63	90.63	77.46	98.59
	PSEC_15	97.48	2.33	94.61	56.11	91.65	69.61	97.67
	HCSM	**99.14**	**0.38**	91.76	**87.74**	84.52	**86.10**	**99.62**
LOTAAS 1	*SVM*	99.96	0.02	97.46	95.00	95.00	95.00	99.98
	PSEC_2	99.90	0.02	89.43	94.12	80.00	86.49	99.98
	PSEC_10	99.96	0.02	97.46	95.00	95.00	95.00	99.98
	PSEC_15	99.98	0.02	99.99	95.24	100.00	97.56	99.98
	PSEC_20	99.96	0.04	99.98	90.91	100.00	95.24	99.96
	HCSM	**100.00**	**0**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**

ACKNOWLEDGMENT

This work is fully supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61375045, Beijing Natural Science Foundation (4142030), the Joint Research Fund in Astronomy (U1531242) under cooperative agreement between the NSFC and Chinese Academy of Sciences (CAS). Prof. Ping Guo is the author to whom all the correspondence should be addressed.

REFERENCES

[1] K. Lee, K. Stovall, F. Jenet, J. Martinez, L. Dartez, A. Mata, G. Lunsford, S. Cohen, C. Biwer, M. Rohr *et al.*, "peace: pulsar evaluation algorithm for candidate extraction–a software package for post-analysis processing of pulsar survey candidates," *Monthly Notices of the Royal Astronomical Society*, p. stt758, 2013.

[2] J. M. Cordes, P. Freire, D. R. Lorimer, F. Camilo, D. J. Champion, D. J. Nice, R. Ramachandran, J. Hessels, W. Vlemmings, J. Van Leeuwen *et al.*, "Arecibo pulsar survey using alfa. i. survey strategy and first discoveries," *The Astrophysical Journal*, vol. 637, no. 1, p. 446, 2006.

[3] J. Boyles, R. S. Lynch, S. M. Ransom, I. H. Stairs, D. R. Lorimer, M. A. McLaughlin, J. W. Hessels, V. M. Kaspi, V. I. Kondratiev, A. Archibald *et al.*, "The green bank telescope 350 mhz drift-scan survey. i. survey observations and the discovery of 13 pulsars," *The Astrophysical Journal*, vol. 763, no. 2, p. 80, 2013.

[4] M. Keith, A. Jameson, W. Van Straten, M. Bailes, S. Johnston, M. Kramer, A. Possenti, S. Bates, N. Bhat, M. Burgay *et al.*, "The high time resolution universe pulsar survey–i. system configuration and initial discoveries," *Monthly Notices of the Royal Astronomical Society*, vol. 409, no. 2, pp. 619–627, 2010.

[5] W. Zhu, A. Berndsen, E. Madsen, M. Tan, I. Stairs, A. Brazier, P. Lazarus, R. Lynch, P. Scholz, K. Stovall *et al.*, "Searching for pulsars using image pattern recognition," *The Astrophysical Journal*, vol. 781, no. 2, p. 117, 2014.

[6] V. Morello, E. Barr, M. Bailes, C. Flynn, E. Keane, and W. van Straten, "Spinn: a straightforward machine learning solution to the pulsar candidate selection problem," *Monthly Notices of the Royal Astronomical Society*, vol. 443, no. 2, pp. 1651–1662, 2014.

[7] R. Lyon, B. Stappers, S. Cooper, J. Brooke, and J. Knowles, "Fifty years of pulsar candidate selection: From simple filters to a new principled real-time classification approach," *Monthly Notices of the Royal Astronomical Society*, vol. 459, no. 1, pp. 1104–1123, 2016.

[8] D. Thornton, "The High Time Resolution Radio Sky," Ph.D. dissertation, University of Manchester, November 2013.

[9] F. Xing, P. Guo, and M. Lyu, "Comparison of methods of controlling two types of error ratio applied to software reliability prediction," *Journal of Nanjing University (In Chinese)*, vol. 41, pp. 644–649, 2005.

[10] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[11] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 3, pp. 480–492, 2012.

[12] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.