Detecting Ventricular Beats with Machine Learning Models

Stojancho Tudjarski*, Aleksandar Stankovski*, Marjan Gusev*
*Ss. Cyril and Methodius University, Faculty of Computer Science and Engineering, Skopje, North Macedonia
{stojancho.tudjarski,aleksandar.stankovski}@students.finki.ukim.mk, marjan.gushev@finki.ukim.mk

***Abstract*—This paper aims at modeling a classifier of Ventricular heartbeats by experimenting with the most advanced classic binary classifiers in different scenarios for feature engineering. *Methodology*: The results were acquired based on experimenting with XGBoost and Random Forest algorithms, as two of the most advanced classifiers not based on neural networks. Although the annotated ECG data sets contain records with several heartbeat classes, we focus on a model that would distinguish V from others (Non-V heartbeats). Considering that we are dealing with a highly imbalanced data set, we applied the SMOTE algorithm for data enrichment to provide a better-balanced data set for training the model. To acquire better results, we added new calculated features, with and without feature selection. For feature selection, we used the Fisher Selector algorithm. *Data*: We used MIT-BIH Arrhythmia benchmark database, with train/test split according to the patient-oriented splitting approach that separates the original dataset into two subsets with approximately equal sizes and distribution of heartbeat types. *Conclusion*: The best results are achieved with XGBoost algorithm with original feature set. We achieved precision of 91.36%, recall of 88.31% and F1 score of 89.81%. Results showed that oversampling does not provide significantly better overall model performance. Still, we would recommend this approach since in practice, when dealing with imbalanced data sets, this leads to more robust models that perform better with data outside the training and test sets, such as when the model is used in production.**

***Keywords*—ECG, XGBoost, machine learning, binary classification**

I. Introduction

Heartbeats can be classified in different ways [1]. The Association for Advancement of Medical Instrumentation (AAMI) [2] categorized heartbeats into 5 classes: Normal(N), Supraventricular (S) ectopic, Ventricular (V) ectopic, Fusion (F) and Unknown (Q) beats, also categorized by the IEC standard 60601-2-47 [3]. Ventricular are extra beats of the heart which originate in the bottom heart chambers (the ventricles). A regular heartbeat originates from the top of the heart (the atria). Ventricular ectopic beats occur prematurely and cause the heart to beat differently for that beat. Experiencing ventricular ectopics makes people more likely to develop other heart rhythm problems. [4]

The goal of this research is to find a Machine Learning (ML) model to detect V beats from an electrocardiogram (ECG). Heartbeat classification is a vital step in detecting different types of cardiovascular diseases (CVDs) that affect millions of people worldwide. A single CVD cannot be detected on a single heartbeat ECG recording. However, detecting heartbeat types according to some classification specifications is essential for detecting CVD. It is a good starting point to further classification of series of heartbeats into episodes that can be labeled with some specific CVD. This approach makes automatic heartbeat episodes and CVDs much easier for training an ML model to do it automatically. Doctors can read ECS recordings and diagnose CVDs and related episodes directly, but for the software, it is easier to have the heartbeat accordingly classified first.

The data set is highly imbalanced since the number of V heartbeats, both in training and test sets, is relatively small compared to the total number of beats. Training and test sets are extracted from the MIT-BIH Arrhythmia benchmark database (MITDB) [5], split in a patient-wise manner, according to de Chazal, as DS1 and DS2 separate sets [6]. The input is a list of values extracted from the ECG using signal processing methods, QRS morphology classification, and separate feature space detection to calculate the values of different features accompanied to a recognized single heartbeat, applying the improved Hamilton method [7].

We conducted several experiments with different ML algorithms and various feature selections to provide the best ML model for recognizing V beats. During the research work, we tried to provide answers to address the following Research Questions (RQ):

- **RQ1**: What is the optimal ML algorithm for detecting V heartbeats?
- **RQ2**: Which method improves the model's performance, analyzing the enlarging a specific class to balance the dataset, adding new calculated features and reduced feature set approaches?

The paper is organized according to the following structure: Section II presents the related work. Section III describes the methods used for the research and development of ML algorithms, and classification models are described in Section IV. Results are presented in Section V and Section VI gives insights in some other results for heartbeat type classification. Section VII is devoted to conclusions and directions for future work.

II. Related Work

Many authors/data scientists have worked on the concrete problem of classifying heartbeats. As a result, there are a lot of different approaches to this problem.

1) Ensemble models: Bagging as a meta-algorithm is used by Zeng et al. [8] using the Selecting Base Classifiers on Bagging for heartbeat classification to select an optimal set of classifiers among all candidates through an optimization process, based on the criteria of accuracy and diversity.

This approach is expanded by one magnitude by Yang et al. [9] as they build the hyper-ensemble model as an ensemble of decision trees of ensembles of kernel-based models.

A human electrocardiogram (ECG) identification system based on ensemble empirical mode decomposition (EEMD) is designed by Zhao et al. [10]. A robust preprocessing method comprising noise elimination, heartbeat normalization, and quality measurement is applied to eliminate the effects of noise and heart rate variability.

2) Wavelets: Mondéjar-Guerra et al. [11] use higher-order statistics and SVM over different features based on wavelets, local binary patterns LBP, and several amplitude values.

Higher order statistics (HOS) of wavelet packet decomposition (WPD) coefficients for the purpose of automatic heartbeat recognition by Kutly at al. [12].

Electrocardiogram beat classification based on wavelet transformation and probabilistic neural network is the way how Yu et al. [13] classified heartbeats.

3) SVM: Zhang et al. [14] proposes disease-specific feature selection consisting of a one-versus-one (OvO) features ranking stage and a feature search stage wrapped in the same OvO-rule support vector machine (SVM) binary classifier.

GDA and SVM-based cardiac arrhythmia classification algorithm is applied to input HRV signals by Asl at al. [15]. He obtained from the MIT-BIH arrhythmia database to discriminate six different types of cardiac arrhythmia classes, including normal sinus rhythm, premature ventricular contraction, atrial fibrillation, sick sinus syndrome.

Based on the different characteristics of VEB and SVEB, a novel hierarchical heartbeat classification system was constructed by Huang at al. [16]. First, random projection and support vector machine (SVM) ensemble were used to detect VEB. Then, the ratio of the RR interval was compared to a predetermined threshold to detect SVEB. The optimal parameters for the classification models were selected on the training set and used in the independent testing set to assess the final performance of the classification system.

Can Ye well cover morphological and dynamic features at al. [17] with wavelet transform and independent component analysis applied separately to each heartbeat to extract morphological features, together with RR interval information computed to provide dynamic features. Afterward, the extracted features are processed by a Support Vector Machine (SVM) classifier. SVM classifier approach is also used by Garcia at al. [18] analyzing the vectorcardiogram for extracting features and fine-tuning in feature selection with a particle swarm optimization algorithm.

4) Neural Networks: Neural networks are used to classify heartbeats by Guler at al. [19].

Genetic algorithm for the optimization of features and neural networks in ECG signals classification is the core idea of the research performed by Li et al.[20].

A novel ECG representation based on vectorcardiogram (VCG), called temporal vectorcardiogram (TVCG), along with a complex network for feature extraction, is presented by Garcia et al. [18].

5) Nearest Neighbours: A comparative study of the learning capacity and the classification abilities of four classification methods – Kth nearest neighbor rule, neural networks, discriminant analysis, and fuzzy logic is presented by Jekova et al. [21].

Christov applies premature ventricular contraction classification by the Kth nearest-neighbours rule et al. [22].

6) Morphological Features: De Chazal et al. [6] extracts several feature sets based on ECG morphology and RR intervals and applies statistical modeling over them. The final classification out of a single feature set is then obtained by choosing the class with the highest posterior probability.

Pointing out that the feature extraction from ECG recordings is a vital part of the heartbeat classification problem, Tadekjko et al. [23] compare two strategies for classification of annotated QRS complexes: based on original ECG morphology features and proposed new approach - based on preprocessed ECG morphology features.

7) Feature Selection: Distinguishing feature selection as a preprocessing step is found as performed in the following cases:

Using Scala language and Apache Spark framework distinguishes Alarsan et al. [24] from other researchers. They extract features as a combination of action impulse waveforms produced by specialized cardiac heart tissues.

Normalized RR interval lengths are the basis for the work of Saenz-Cogollo at al. [25] by selecting features with a filter method based on the mutual information ranking criterion on the training set and then applying the Random Forest classifier. They conclude that normalized beat-to-beat RR intervals and features relative to the width of the ventricular depolarization waves (QRS complex) are the most discriminating ones.

Improving generalization capability is the focus of the work of Llamedo et al. [26] validating a simple heartbeat classifier based on ECG feature models.

An adapted state-of-the-art method of processing information known as Reservoir Computing is used to show its utility on the open and time-consuming problem of heartbeat classification, by Escalona at al. [27]

Optimal feature selection with random forests is the main effort of Saens et al. [25]. He then uses the random forest algorithm to perform his heartbeat classification.

8) Open Issues: Analysis of related work shows that it is almost impossible to compare the results due to use of different datasets, number of classes and train/test splitting methodology.

III. Experimental Methods

A. Datasets

The standard MITDB benchmark [5] contains 48 half-hour excerpts of two-channel ambulatory ECG recordings, collected from 47 persons between 1975 and 1979. Twenty-three recordings were chosen randomly from a set of 4000 24-hour ambulatory ECG recordings collected from a mixed population of inpatients (about 60%) and outpatients (about 40%) at Boston's Beth Israel Hospital. The remaining 25 recordings were selected from the same set to include less common but clinically significant arrhythmias that would not be well-represented in a small random sample.

The recordings were digitized at 360 Hz per channel with 11-bit resolution over ten mV. Two or more cardiologists independently annotated each record, and disagreements were resolved to obtain the computer-readable reference annotations for each beat (approximately 110,000 annotations in all) included with the database.

B. Features

Features are extracted to represent the morphological characteristics of the captured heartbeat. This feature set extraction is based on an improved Hamilton method for beat detection [7]. Domazet and Gusev [28] specify over thirty features that correspond to measurements of the baseline deflection, heights and widths of the QRS, their relative ratio, and ratio with the average of last beats, RR interval, and relative ratio versus the average of last RR intervals, triangular similarity index, QRS type determined by the deflections of R and S waves, etc. For example, $fQSc$ means measure between two points calculated on 5% of the max height, $fQSa$ average of last five $fQSc$, $fQSd$ difference between $fQSc$ and $fQSa$, SB height between S and baseline point, QSc means measure between Q and S points, QSa average of last five QSc, QSd difference between QSc and QSa, QSe average of input values in the interval (energy), $TriSim$ triangular similarity between QSc and QSa, $Type$ of the current, previous and last normal QRS, Th morphology of the T wave (normal, inverse and biphasic), RRl and RRa as last RR interval and average of last five RR intervals, RRr ratio between RRl and RRa, $QRca$ and $RSca$, deviation between QR and RS heights from the previous and current QRS, ST height of the ST segment, etc.

C. Evaluation Metrics

True positives (TP) is the number of correctly labeled items belonging to the positive class, while false positives (FP) are items incorrectly labeled as belonging to the positive class. False negatives (FN) are items that were not labeled as belonging to the positive class but should have been, and true negatives (TN) are correctly labeled to belong to negatives.

We used the following accuracy scores:

- **Precision** is TP divided by the total number of elements belonging to the positive class (i.e. the sum of TP and FP).
- **Recall** is TP divided by the total number of elements that actually belong to the positive class (i.e. the sum of TP and FN).
- **F1 score** is a measure that combines precision and recall as their harmonic mean.

Due to its irrelevancy in imbalanced datasets, we did not use the most common score in ML problems: accuracy. Let us say one trains a model for detecting fraudulent transactions that occur in 0.1% of situations, and one chooses the dumb model scenario that classifies all the transactions to non-fraudulent. Therefore, one has a model with 99.9% of accuracy that did not catch any fraudulent transaction. Such a model is useless, even with pretty high accuracy.

A confusion matrix presents TP, FP, FN and TN, as a widely recognized and extremely valuable information.

In addition, we present the improvement factor IF calculated by (1) that compares the F1 performance prior and after applying a certain method in developing the model.

$$IF = \frac{F1_{prior}}{F1_{after}} \quad (1)$$

IV. ML modeling

A. Dataset Train-Test Split

The analyzed classes were V and NonV, consisting of all other heartbeat types. To conduct proper training of our models, the train-test split was done such that the patients do not overlap in the training and testing data sets, meaning that the model will be tested over data that the model was never exposed to during the training.

Instead of using an obvious choice of the commonly used 70%-30% split of shuffled data, we used a patient-wise split of the data, as proposed by de Chazal [6]. According to his methodology, the MITDB dataset is split into two datasets, DS1 and DS2, with recordings from 22 patients, and four paced recordings are not included in the split datasets. DS1 contains 51029 annotated heartbeats, and DS2 contains 49711 annotated heartbeats.

Table I presents the number of V beats and the high imbalance of the training set DS1 and test set DS2.

TABLE I: Distribution V beats in the training set DS1 and test set DS2

Dataset	Total beats	V beats	V beats%
DS1	51300	4142	8,07%
DS2	49665	3218	6,48%

B. Handling Imbalanced Dataset

Since we are dealing with a highly imbalanced dataset with an imbalance ratio NonV vs. V class with a value of 11.385 times in favor of NonV class, we applied several oversampling techniques that would automatically generate artificial examples of the V beats, similar to the existing ones, and select methods that might lead to a better performing trained model.

In a classic oversampling technique, the minority data is duplicated from the minority data population. While it increases the number of data, it does not give any new information or variation to the machine learning model.

The oversampling technique we used is SMOTE, implemented within imblearn Python package. This algorithm randomly chooses two points from the minor-class labeled samples and places a new example in a random point between these two points. It is important to note that SMOTE technique is used only over the training set, thus extending/balancing only DS1 dataset, effectively canceling any possibility for occurring data leakage.

C. Feature Engineering

Next thing worth to try out is feature engineering. We added new calculated features (FT), one per column calculated by (2).

$$FT_{new} = mean(FT_{existing} - abs(FT_{existing})) \quad (2)$$

This is performed for all numerical (non-categorical) features. The goal we want to achieve is to introduce features that are in monotonic relationship with the target of the prediction, and our assumption is that the breaking point is the near the mean of the values of a feature. This is just a single example how new calculated features can be introduced to the model. The reason why we need monotonic relationship with the target variable is the fact that all of the ML algorithms work better with such variables.

Also, we used the standard data normalization as a data preprocessing step. The idea is to try to capture the importance of the new features in situations where the features are not monotonically dependent on the target values.

D. Feature Selection

Feature selection is a process where we select a subset of all available features that leads to a better performing model. Therefore, we filter out features that do not provide additional information about the training examples according to the target values that we predict, and we do that in a systematic manner.

Fisher score is one of the most frequently used feature ranking approaches used for feature selection. It allows sorting the features in descending order and choosing top N features. Here we choose the value of N as a value which further increase does not significantly improve the model's accuracy, and according to our processing power. Training a model with more features is more time consuming, and produces a model with slower inference. According to Gu at al. [29], the vital idea of the Fisher score is to find a subset of features, such that in the data space spanned by the selected features, the distances between data points in different classes are as large as possible, while the distances between data points in the same class are as small as possible.

TABLE II: Confusion matrices for initial models

	RF		XGB		FFNN	
	NonV	**V**	**NonV**	**V**	**NonV**	**V**
NonV	43168	2878	45779	267	45637	409
V	76	3123	374	2825	548	2651

TABLE III: Classification scores of initial models

Score type	RF	XGB	FFNN
Precision	52.04%	91.36%	86.60%
Recall	97.62%	88.31%	85.46%
F1	67.89%	89.81%	86.03%

V. Results from ML modelling

A. Initial Model Selection

Considering the best classifiers involving neural networks (NN), we chose to start the model selection with Random Forest (RF), XGBoost (XGB) and Fully-connected plain feed-forward NN (FFNN).

Table II presents the confusion matrices and Table III the corresponding classification scores. The best results with this algorithm are achieved when max_features value is set to 70% of all features.

The optimal results using this approach with one hidden layer, with the same number of neurons as the number of input neurons. Fewer neurons in the hidden layer lead to worse performing models. More neurons in the hidden layer also lead to worse-performing models due to overfitting. According to this, the best performing algorithm is XGB achieving an F1 score of 89.81%.

B. Handling Training Set Imbalance

Since we did enough experiments to find the most optimal model for our binary classification problem, we went further with experiments on how properly handled imbalance in the training set data would affect the results. After enriching the dataset with enough V-like labeled records according to SMOTE methodology, we effectively injected another 44239 V-labeled artificial training examples, reaching the number of 93484 examples in total. The resulting confusion matrices applying the SMOTE method to deal with the imbalance on the XGBoost model training are presented in Table IV, and the performance results in Table V. We observe that F1 score is dropping down after applying this operation. Therefore, it does not improve the model's performance.

C. Feature Engineering

Two feature engineering techniques were used:

TABLE IV: Confusion matrices applying the SMOTE imbalance handling

	Before		After	
	NonV	**V**	**NonV**	**V**
NonV	45711	335	45637	409
V	267	2932	548	2651

TABLE V: Classification scores applying the SMOTE imbalance handling

Score type	Before	After	IF%
Precision	91.36%	86.60%	-5.21%
Recall	88.31%	85.46%	-3.22%
F1	89.81%	86.03%	-4.21%

TABLE VI: Confusion matrices applying data normalization

	Before		After	
	NonV	**V**	**NonV**	**V**
NonV	45711	335	38350	7696
V	267	2932	137	3062

TABLE VII: Classification scores applying data normalization

Score type	Before	After	IF%
Precision	91.36%	28.46%	-68.85%
Recall	88.31%	95.72%	+8.39%
F1	89.81%	43.88%	-51.14%

1) Data Normalization: Data normalization is a standard step in the data preprocessing phase, where we prepare the dataset to be better fitted for the training process. A type of data normalization we used is MinMaxScaler provided by sklearn Python library, which scales all the values within all the features in a range of [0, 1].

We observe a considerable decrease of sensitivity which causes that F1 score to drop down after applying the data normalization. Therefore it does not improve the model's performance. After analyzing the results before and after data normalization, we concluded that the reason of the drop of the classification accuracy occurs because of the outliers that appear with high values. The result is shrinking the majority of the values into very narrow range, and that affected the training procedure with negative consequences.

2) Adding New Calculated Features: Another step of feature engineering was to add new features, calculated from the existing non-categorical ones, according to (2). The idea driving this formula was to allow the model to capture information from new features. The way the new features are composed makes them better suited for the model training since they preserve a monotonic relationship with the target that is to be predicted.

The results from this experiment are presented in Table VIII and Table IX. We conclude that F1 score drops after applying a method that calculates new features; therefore, it does not improve the model's performance.

TABLE VIII: Confusion matrices applying new calculated features

	Before		After	
	NonV	**V**	**NonV**	**V**
NonV	45711	335	45935	111
V	267	2932	741	2458

TABLE IX: Classification scores applying new calculated features

Score type	Before	After	IF%
Precision	91.36%	95.68%	+4.73%
Recall	88.31%	76.84%	-12.99%
F1	89.81%	85.23%	-5.1%

TABLE X: Confusion matrices applying feature selection

	Before		After	
	NonV	**V**	**NonV**	**V**
NonV	45711	335	45886	160
V	267	2932	796	2403

TABLE XI: Classification scores applying feature selection

Score type	Before	After	IF%
Precision	91.36%	93.76%	+2.63%
Recall	88.31%	75.12%	-14.94%
F1	89.81%	83.41%	-7.13%

D. Feature Selection

A feature selection experiment was conducted over the original feature set using the Fisher selector with 20 as an input value for the feature we want to see extracted as the most important ones.

Applying the Fisher method for feature selection, we conclude that Therefore, the F1 score is dropping down and does not improve the model's performance. In our experiment, we used 13 of the chosen maximum of 20 features were chosen among the newly added ones.

VI. DISCUSSION

The performance analysis of the related work is presented in Table XII. De Chazal et al. [6] achieved a sensitivity of 75.9%, a positive predictivity of 38.5%, and a false positive rate of 4.7% for the SVEB class. The most relevant result for our writing is the VEB class scores: the sensitivity was 77.7%, the positive predictivity was 81.9%, and the false positive rate was 1.2%. Garcia at al. [18] achieved 87.3% of Sensitivity (Se) for the Ventricular ectopic beat (V) class. Yang at al. [9] achieved sensitivity ventricular ectopic beats of 94.4%. Ince at. [30] reaches accuracy of 98.3% and sensitivity of 84.6% for VEB detection.

Our approach uses a method where the training and test data correspond to different patients, as used in some other approaches that reach higher recall and precision, which is expected in cases of testing of what has the classifier trained for. Some of the related work provide incomplete performance measures so they can. not be

TABLE XII: Performance comparison to classify V beats

Score type	Precision	Recall	F1
De Chazal at al. [6]	77.7%	81.9%	79.7%
Zhang et al. [14]	85.5%	92.8%	89.0%
Llamedo et al. [26]	81.0%	87.0%	83.9%
Escalona at al. [27]	84.8%	88.8%	86.8%
This paper	91.36%	88.31%	89.81%
Huang at al. [16]	93.9%	90.9%	92.4%
Kutly at al. [12]	90.0%		
Saenz et al. [25]			90.8%
Garcia at al. [18]	87.3%		
Yang at al. [9]	94.4%		
Ince at. [30]	84.6%		

directly compared. In addition, our method achieves higher precision, as one of the goals we used in the optimization.

VII. Conclusion

Analyzing RF, XGB and FFNN for RQ1, we conclude that the best performing algorithm is the XGB, which is not surprisingly, since it is already well known as the best classification and regression algorithm for classic machine learning problems.

Referring to RQ2, we conclude that none of the approaches to enlarge the training dataset with artificially injected V-like data points, adding new calculated features and feature selection to 20 top-ranked features does not improve the classification performance. V heartbeat data points are grouped well enough that the classification boundary can be drawn well sufficient without data augmentation. Decision tree-based models are otherwise proven to be very robust to the characteristics of the features they are operating with since they do not necessarily require a monotonic relationship with the target, even that may help occasionally.

References

[1] J. He, L. Sun, J. Rong, H. Wang, and Y. Zhang, "A pyramid-like model for heartbeat classification from ecg recordings," *PloS one*, vol. 13, no. 11, p. e0206593, 2018.

[2] Association for the Advancement of Medical Instrumentation (AAMI), "ANSI/AAMI EC57:2012 testing and reporting performance results of cardiac rhythm and ST segment measurement algorithms." [Online]. Available: https://webstore.ansi.org/Standards/AAMI/ANSIAAMIEC572012EC57

[3] International Electrotechnical Commission, "IEC 60601-2-47:2012 medical electrical equipment - part 2-47: Particular requirements for the basic safety and essential performance of ambulatory electrocardiographic systems." [Online]. Available: https://webstore.iec.ch/publication/2666

[4] London Heart Clinic, "Ventricular ectopics." [Online]. Available: https://theheartclinic.london/conditions-treatment/ventricular-ectopics-treatment/

[5] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[6] P. De Chazal, M. O'Dwyer, and R. B. Reilly, "Automatic classification of heartbeats using ecg morphology and heartbeat interval features," *IEEE transactions on biomedical engineering*, vol. 51, no. 7, pp. 1196–1206, 2004.

[7] E. Domazet and M. Gusev, "Improving the qrs detection for one-channel ecg sensor," *Technology and Health Care*, vol. 27, no. 6, pp. 623–642, 2019.

[8] X. D. Zeng, S. Chao, and F. Wong, "Ensemble learning on heartbeat type classification," in *Proceedings 2011 International Conference on System Science and Engineering*. IEEE, 2011, pp. 320–325.

[9] P. Yang, D. Wang, W.-B. Zhao, L.-H. Fu, J.-L. Du, and H. Su, "Ensemble of kernel extreme learning machine based random forest classifiers for automatic heartbeat classification," *Biomedical Signal Processing and Control*, vol. 63, p. 102138, 2021.

[10] Z. Zhao, L. Yang, D. Chen, and Y. Luo, "A human ecg identification system based on ensemble empirical mode decomposition," *Sensors*, vol. 13, no. 5, pp. 6832–6864, 2013.

[11] V. Mondéjar-Guerra, J. Novo, J. Rouco, M. G. Penedo, and M. Ortega, "Heartbeat classification fusing temporal and morphological information of ecgs via ensemble of classifiers," *Biomedical Signal Processing and Control*, vol. 47, pp. 41–48, 2019.

[12] Y. Kutlu and D. Kuntalp, "Feature extraction for ecg heartbeats using higher order statistics of wpd coefficients," *Computer methods and programs in biomedicine*, vol. 105, no. 3, pp. 257–267, 2012.

[13] S.-N. Yu and Y.-H. Chen, "Electrocardiogram beat classification based on wavelet transformation and probabilistic neural network," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1142–1150, 2007.

[14] Z. Zhang, J. Dong, X. Luo, K.-S. Choi, and X. Wu, "Heartbeat classification using disease-specific feature selection," *Computers in biology and medicine*, vol. 46, pp. 79–89, 2014.

[15] B. M. Asl, S. K. Setarehdan, and M. Mohebbi, "Support vector machine-based arrhythmia classification using reduced features of heart rate variability signal," *Artificial intelligence in medicine*, vol. 44, no. 1, pp. 51–64, 2008.

[16] H. Huang, J. Liu, Q. Zhu, R. Wang, and G. Hu, "A new hierarchical method for inter-patient heartbeat classification using random projections and rr intervals," *Biomedical engineering online*, vol. 13, no. 1, pp. 1–26, 2014.

[17] C. Ye, B. V. Kumar, and M. T. Coimbra, "Heartbeat classification using morphological and dynamic features of ecg signals," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2930–2941, 2012.

[18] G. Garcia, G. Moreira, D. Menotti, and E. Luz, "Inter-patient ecg heartbeat classification with temporal vcg optimized by pso," *Scientific reports*, vol. 7, no. 1, pp. 1–11, 2017.

[19] İ. Güler and E. D. Übeylı, "Ecg beat classifier designed by combined neural network model," *Pattern recognition*, vol. 38, no. 2, pp. 199–208, 2005.

[20] H. Li, D. Yuan, X. Ma, D. Cui, and L. Cao, "Genetic algorithm for the optimization of features and neural networks in ecg signals classification," *Scientific reports*, vol. 7, no. 1, pp. 1–12, 2017.

[21] I. Jekova, G. Bortolan, and I. Christov, "Assessment and comparison of different methods for heartbeat classification," *Medical Engineering & Physics*, vol. 30, no. 2, pp. 248–257, 2008.

[22] I. Christov, I. Jekova, and G. Bortolan, "Premature ventricular contraction classification by the kth nearest-neighbours rule," *Physiological measurement*, vol. 26, no. 1, p. 123, 2005.

[23] P. Tadejko and W. Rakowski, "Mathematical morphology based ecg feature extraction for the purpose of heartbeat classification," in *6th International Conference on Computer Information Systems and Industrial Management Applications (CISIM'07)*. IEEE, 2007, pp. 322–327.

[24] F. I. Alarsan and M. Younes, "Analysis and classification of heart diseases using heartbeat features and machine learning algorithms," *Journal of Big Data*, vol. 6, no. 1, pp. 1–15, 2019.

[25] J. F. Saenz-Cogollo and M. Agelli, "Investigating feature selection and random forests for inter-patient heartbeat classification," *Algorithms*, vol. 13, no. 4, p. 75, 2020.

[26] M. Llamedo and J. P. Martínez, "Heartbeat classification using feature selection driven by database generalization criteria," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3, pp. 616–625, 2011.

[27] M. A. Escalona-Morán, M. C. Soriano, I. Fischer, and C. R. Mirasso, "Electrocardiogram classification using reservoir computing with logistic regression," *IEEE Journal of Biomedical and health Informatics*, vol. 19, no. 3, pp. 892–898, 2014.

[28] E. Domazet and M. Gusev, "An electrocardiogram beat classification algorithm for wearable sensors," University Sts Cyril and Methodius, Faculty of Computer Science and Engineering, Tech. Rep. FINKI-2019-12, 2019.

[29] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," *arXiv preprint arXiv:1202.3725*, 2012.

[30] T. Ince, S. Kiranyaz, and M. Gabbouj, "A generic and robust system for automated patient-specific classification of ecg signals," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1415–1426, 2009.