Arrhythmia detection using resampling and deep learning methods on unbalanced data

E.Y. Shchetinin [1], *A.G. Glushkova* [2]
[1] *Financial University under the government of the Russian Federation,*
125993, Moscow, 49 Leningradsky Prospekt, Russia;
[2] *Endeavor, London W4 5HR, Chiswick Park, 566 Chiswick High Road, United Kingdom*

Abstract

Due to cardiovascular diseases millions of people die around the world. One way to detect abnormality in the heart condition is with the help of electrocardiogram signal (ECG) analysis. This paper's goal is to use machine learning and deep learning methods such as Support Vector Machines (SVM), Random Forests, Light Gradient Boosting Machine (LightGBM), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BLSTM) to classify arrhythmias, where particular interest represent the rare cases of disease.

In order to deal with the problem of imbalance in the dataset we used resampling methods such as SMOTE Tomek-Links and SMOTE ENN to improve the representation ration of the minority classes. Although the machine learning models did not improve a lot when trained on the resampled dataset, the deep learning models showed more impressive results. In particular, LSTM model fitted on dataset resampled using SMOTE ENN method provides the most optimal precision-recall trade-off for the minority classes Supraventricular beat and Fusion of ventricular and normal beat, with recall of 83 % and 88 % and precision of 74 % and 66 % for the two classes respectively, whereas the macro-weighted recall is 92 % and precision is 82 %.

Keywords: machine learning, deep learning, ECG, resampling, arrhythmia.

Citation: Shchetinin EY, Glushkova AG. Arrhythmia detection using resampling and deep learning methods on unbalanced data. Computer Optics 2022; 46(6): 980-987. DOI: 10.18287/2412-6179-CO-1112.

Introduction

Cardiovascular diseases belong to a group of diseases of the heart and blood vessels. Based on report of Word Health Organization (WHO) heart attacks and strokes cause 80 % of deaths [1]. The greatest damage is dealt to the impoverished, in particular, 81 % of the 17.9 million of people died in 2019 in the developing world. Besides that, many other people at least once experienced abnormal heartbeats, which raises the importance of arrhythmias detection.

Effectiveness of the treatment often depends on timely diagnostics of the illness, which can be done with the help of electrocardiogram (ECG) analysis. It reveals heart disturbances and can be used in adjustment of the current treatment. However, automation of this process is complicated, because of existence of various wave types. In addition, experts may misinterpret or miss important information. Additionally, in situations where medical help is almost unavailable it is essential to develop computer-based instruments which can be used in arrhythmias detection.

This paper shows methodology of detection of a heart disease in an unbalanced dataset using resampling techniques and machine learning and deep learning algorithms.

1. Literature review

Accurate and quick ECG analysis can save lives, and with modern computational tools this can be done around the world. While many issues with arrhythmia detection were grappled by previous research, some challenges remain.

Most of the past research used kernel-based classifiers and neural networks. According to one review of the methods used to tackle the ECG classification problem, Convolutional neural network and Recurrent neural network used together may lead to the best performance, although some problems with interpretability, scalability, efficiency still need to be discussed [2].

A lot of papers mentioned the dataset and what problems it may cause, for example, since the most popular class represent the normal heartbeat, some algorithms may experience problems in distinguishing the disease cases due to low number of such examples [3]. Then, one of the most popular metric, accuracy, should not be used since it calculates weighted average for each class, so the weights are higher for the majority classes. Nevertheless, a lot of authors use accuracy to evaluate models' performance and compare them. Others used custom class weights as well as resampling and constructed a Random Forest and a Convolutional neural network, evaluating the final models using recall, precision, f1-score.

Another possible issue is the high number of features that the model must consider, which increases the computational time. Dimensionality reduction or feature selection are the common methods used to deal with this problem [4], however, deep learning models can also be applied [2] since they can show appropriate results without the need for feature engineering, because they learn the patterns during training [5].

In this paper we proposed a combination of up-sampling and down-sampling techniques to deal with the problem of imbalanced dataset. Then, machine learning and deep learning models were applied, and a comparative analysis was carried out. To further improve the performance, an attention model was implemented and evaluated using a variety of metrics for proper analysis of the minority classes. As a result, we came up with a model which could classify the disease cases with high accuracy.

2. Data and proposed methodology

2.1. Data description

As mentioned earlier, electrocardiogram (ECG) analysis is often used in detection of arrhythmia. It measures how long one electrical phase in a heartbeat lasts by perceiving electrical activity of a heart using special sensors which are fixed on a patient's chest. In particular, if the problem is not obvious from the ECG, then it is recommended to wear a portable ECG device for a while. Thus, the detection of arrhythmias can be problematic, since sometimes it is required to analyze ECG records for a long time, and human factor can also affect accuracy of the detection. Hence, a necessity for automatic detection of the illness exists.

One of the most commonly used databases in arrhythmias research is the MIT-BIH Arrhythmia Database. It has 48 half-hour recordings of two channel electrocardiogram recordings collected from 47 patients, which were digitized at 360 samples per second per channel with 11-bit resolution over a 10mV range. The sampling frequency of the channels is 125 Hz and they were segmented so that a heartbeat is represented by each segment.

For the purposes of the paper, we are using the preprocessed dataset [6], since it has the properties useful for the modelling. First, a heartbeat is represented as an observation, where a feature is a segment of it, and each signal has equal length of 187 features. The features belong to a range from 0 to 1. Overall, there are 87,554 samples in training and 21,892 samples in the testing dataset.

All the observations correspond to one of the following classes: Normal beat (N or 0), Supraventricular premature beat (S or 1), Premature ventricular contraction (V or 2), Fusion of ventricular and normal beat (F or 3), Unclassifiable beat (Q or 4) as shown in fig. 1 and fig. 2.

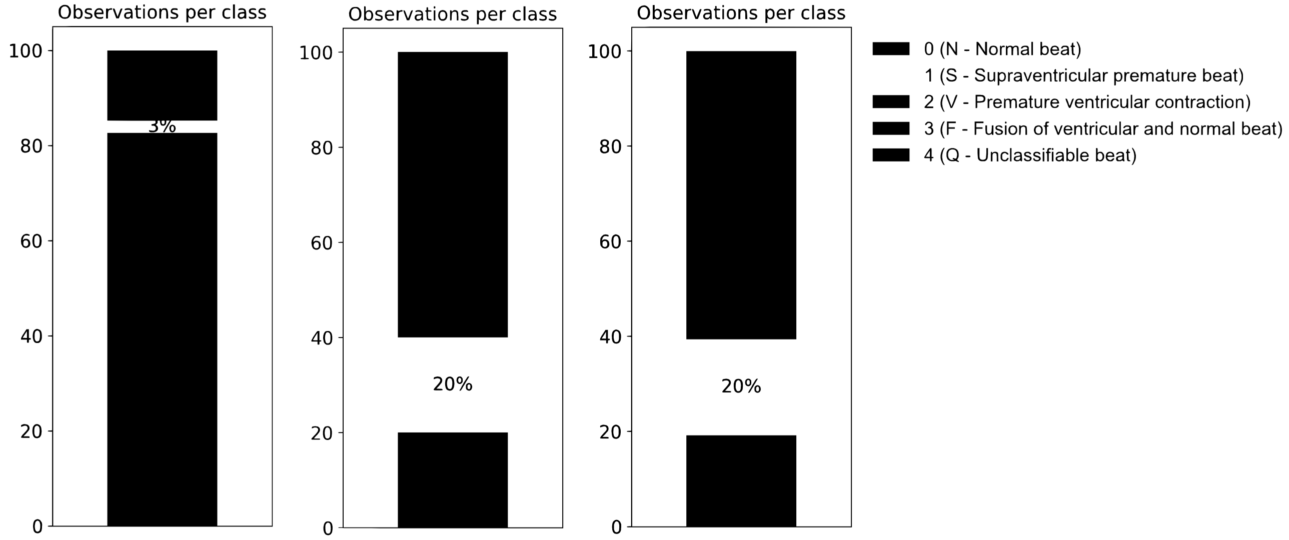

Fig. 1. Observations per class in percentages for the training data in the original dataset, in the dataset resampled using SMOTE Tomek-Links, in the dataset resampled using SMOTE ENN

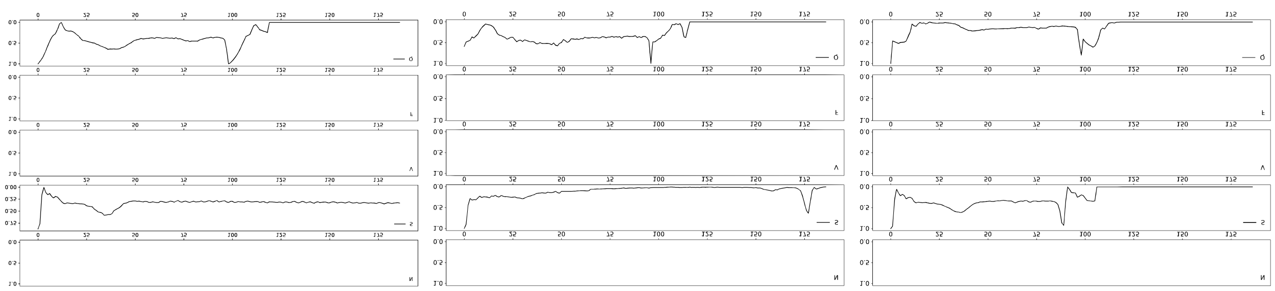

Fig. 2. Samples from different classes (Normal beat (N), Supraventricular premature beat (S), Premature ventricular contraction (V), Fusion of ventricular and normal beat (F), Unclassifiable beat (Q)) from the original dataset, the dataset resampled using SMOTE Tomek-Links, the dataset resampled using SMOTE ENN

2.2. Over- and undersampling as a method to cope with unbalanced dataset

One of the common approaches to using imbalanced dataset is to implement resampling methods first. Re-sampling changes the dataset so that a balanced distribution of classes is created, which is easier to deal with for the classifiers. In general, two methods are used: undersampling and oversampling. Under-sampling involves removal of some of the observations, while basic oversampling is performed by duplicating existing observations. While undersampling has a consequence that important information may be removed, over-sampling does not add new information, thus, there is a necessity to use advanced methods to improve the result.

A following approach to multiclass classification was suggested. Firstly, using SMOTE technique we created synthetic observations to upsample the minority classes and removed some observations from the popular classes using Tomek-Links or ENN.

Synthetic minority oversampling technique (SMOTE) adds new data points based on the distance of each sample and its closest neighbor, thus introducing new information that the algorithms can use in training [7, 8]. The synthetic observations are created from the minority class. One problem is that the use of SMOTE may lead to creation of noise in the data.

Studies show that if undersampling and oversampling methods are used together, then better results can be achieved. As SMOTE is one of the most popular techniques, authors often use it in combination with undersampling methods [9]: first, SMOTE method is applied to increase the number of observations belonging to the minority classes, then either Tomek-Links or ENN is used to undersample the majority class.

Tomek-Links technique identifies samples which are close neighbors and belong to different classes, and which an algorithm finds difficult to classify. Then the method removes samples from a more popular class [3]. SMOTE method together with Tomek-Links first creates synthetic samples from the minority class and then removes one sample from a pair of samples belonging to different classes.

Alternatively, Edited Nearest Neighbors (ENN) is an undersampling technique that removes misclassified samples using K neighbors [3]: if the majority class of the closest points is not the same as the class of the sample in question, then the sample and its neighbors are deleted from the dataset. Using this method with SMOTE leads to more clear class separation, as samples from different classes are deleted.

Thus, machine learning and deep learning models can be trained on the resampled dataset to discover if resampling can improve performance.

2.3. Machine learning algorithms

We fitted SVM, LightGBM, Random Forest, CNN, LSTM, BLSTM algorithms on three variations of the original dataset.

Support Vector Machines (SVM) is a supervised algorithm the goal of which is to find a hyperplane in a multidimensional space, where number of dimensions correspond to number of features, that can successful classify the datapoints. Support vectors are used to create hyperplane. Since multiple hyperplanes can exist, the algorithm tries to choose one which maximizes the margin between the points of different groups.

LightGBM (Light Gradient Boosted Machine) is a boosting algorithm which uses tree-based algorithms. It is similar to Gradient boosting and XGBoost, but it also trains faster and is more efficient and can handle large datasets, while using less memory.

Another ensemble algorithm, random forest, constructs several decision trees, and the output for classification is the class for which most of the trees voted. This algorithm reduces problem of overfitting which is common in decision trees.

2.4. Deep learning algorithms

Convolutional Neural Networks (CNN) are deep neural networks usually applied in image analysis and classification. They can be viewed as regularized versions of fully connected neural networks. Due to their ability to extract patterns and capture spatial and temporal dependencies, usually less preprocessing is required compared to other algorithms [10]. In this paper one-dimensional CNN (1D CNN) is implemented. A randomized search cross-validation algorithm was used to determine the optimal number of layers. In the end, minimum value of the loss function was achieved with three convolutional and two fully-connected layers. The architecture also included batch normalization and one-dimensional max-pooling. The model was fitted using batch size of 32.

Long Short-Term Memory networks (LSTM) can process data sequences, for example time series [11, 12]. They were created to cope with the with the vanishing gradient problem which is typical for the standard recurrent neural networks (RNN). The optimal model included two blocks of LSTM and Dropout layers and two blocks of Dense layers.

While LSTM is not always effective in using new datapoints to make predictions, Bidirectional LSTM (BLSTM), which uses forward and backward computing, preforms better, in particular in sentence representations for document-level sentiment classification [13, 14]. The trained model consists of 5 blocks of BLSTM and Dropout layers.

3. Evaluation

3.1. Hyperparameter tuning

The training dataset was divided into training and validation datasets (80 % and 20 % respectively) and the testing dataset was used to evaluate the tuned model. The optimal hyperparameters for the machine learning models were discovered using a five-fold grid search cross validation, and for the deep learning models three-fold randomized search cross validation.

3.2. Classification metrics

In order to evaluate classification quality of the models, a detailed analysis of the models' performance was conducted, where multiple metrics were calculated and compared, such as macro-averaged recall, precision, f1-score, confusion matrices, ROC and PR curves [3]. Accuracy is calculated as a proportion of correct predictions to total number of predictions:

$$Accuracy = \frac{Correct\ predictions}{Total\ Predictions}. \quad (1)$$

However, accuracy should not be used alone in multiclass classification, because it does not provide enough representation of the minority class.

Precision is the proportion of correct positive observations to total positive observations:

$$Precision = \frac{Correctly\ predicted\ positive\ observations}{Total\ predicted\ positive\ observations}. \quad (2)$$

Recall can be used to judge how accurately a model can detect a positive label:

$$Recall = \frac{Correctly\ predicted\ positive\ observations}{Total\ observations\ in\ a\ class}. \quad (3)$$

F1-score is a harmonic mean of precision and recall:

$$F1 - score = 2 \times \frac{Recall \times Precision}{Recall + Precision}. \quad (4)$$

This paper uses macro-averaging in calculating the above metrics, where the final metric is the average of independently computed individual metrics for each class.

Averaged metrics for the models fitted on the testing dataset can be seen in tab. 1. The values in tab. 1 are sorted by recall.

Tab. 1 shows that deep learning models are among the best performers judging by recall and precision as they benefit from resampling in contrast to the machine learning models, since recall and ROC AUC scores are higher. In particular, the highest macro-weighted recall of 92.2 % is achieved by the CNN model trained on dataset resampled using SMOTE Tomek-Links. While BLSTM and LSTM have better performance when trained on dataset resampled using SMOTE ENN, CNN and Random Forest models show better performance after SMOTE Tomek-Links resampling.

Tab. 1 shows that deep learning models are among the best performers judging by recall and precision as they benefit from resampling in contrast to the machine learning models, since recall and ROC AUC scores are higher. In particular, the highest macro-weighted recall of 92.2 % is achieved by the CNN model trained on dataset resampled using SMOTE Tomek-Links. While BLSTM and LSTM have better performance when trained on dataset resampled using SMOTE ENN, CNN and Random Forest models show better performance after SMOTE Tomek-Links resampling.

Tab. 1. Comparison of the testing metrics (accuracy, ROC AUC, macro-averaged f1-score, precision, recall) for the best models trained on original data, on data, resampled using SMOTE Tomek-Links method (SMOTE TL) and on data, resampled using SMOTE ENN method

Algorithm	Dataset	F1-score	Precision	Recall	Accuracy	ROC-AUC
CNN	Original	90.7 %	93.9 %	88.1 %	98.4 %	93.4 %
CNN	SMOTE TL	88.4 %	85.7 %	92.2 %	97.7 %	95.5 %
CNN	SMOTE ENN	86.8 %	83.9 %	91.2 %	97.3 %	94.9 %
BLSTM	Original	90.1 %	90.9 %	89.7 %	98.4 %	94.2 %
BLSTM	SMOTE TL	88.2 %	85.6 %	91.6 %	97.2 %	95.2 %
BLSTM	SMOTE ENN	85.2 %	80.6 %	92.1 %	96.7 %	95.4 %
LSTM	Original	89.7 %	90.9 %	88.8 %	98.2 %	93.7 %
LSTM	SMOTE TL	89.8 %	88.4 %	91.5 %	98.0 %	95.2 %
LSTM	SMOTE ENN	86.3 %	82.4 %	91.9 %	97.2 %	95.4 %
SVM	Original	80.4 %	75.3 %	91.5 %	95.7 %	94.9 %
SVM	SMOTE TL	76.9 %	71.3 %	91.4 %	93.8 %	94.7 %
SVM	SMOTE ENN	76.0 %	70.1 %	91.4 %	93.2 %	94.7 %
LightGBM	Original	81.0 %	75.1 %	91.0 %	95.1 %	94.6 %
LightGBM	SMOTE TL	83.9 %	79.7 %	90.3 %	96.3 %	94.3 %
LightGBM	SMOTE ENN	90.1 %	89.6 %	90.6 %	97.9 %	94.7 %
Random Forest	Original	78.3 %	76.1 %	86.7 %	95.0 %	92.1 %
Random Forest	SMOTE TL	77.2 %	73.8 %	88.8 %	94.3 %	93.2 %
Random Forest	SMOTE ENN	75.8 %	71.5 %	88.7 %	93.6 %	93.1 %

Tab. 2 and 3 allow us to get a better view on the models' performance, since we can compare recall and precision for each class.

Although most deep learning models trained on resampled data have higher recall, the resampling method which improved performance is not universal. Since it is important that the models classify the minority classes correctly, CNN fitted on data resampled using SMOTE Tomek-Links method is not as appropriate as SVM or LSTM. Thus, based on recall for the classes 1 and 3, which have the smallest number of observations, LSTM trained on the data resampled using SMOTE Tomek-Links (LSTM SMOTE TL) approach with recall of 87 % and 88 % for classes 1 and 3 is preferable, followed by SVM SMOTE ENN, LSTM SMOTE ENN, CNN SMOTE TL.

Tab. 2. Comparison of the classification results based on recall

Model	**Dataset**	**Recall**				
		Class 0	**Class 1**	**Class 2**	**Class 3**	**Class 4**
LSTM	Original	99 %	81 %	93 %	80 %	99 %
LSTM	SMOTE TL	98 %	87 %	95 %	88 %	99 %
LSTM	SMOTE ENN	98 %	83 %	95 %	88 %	99 %
SVM	Original	99 %	71 %	92 %	73 %	97 %
SVM	SMOTE TL	94 %	82 %	94 %	85 %	98 %
SVM	SMOTE ENN	99 %	84 %	95 %	85 %	98 %
CNN	Original	100 %	77 %	94 %	79 %	98 %
CNN	SMOTE TL	99 %	83 %	94 %	88 %	98 %
CNN	SMOTE ENN	99 %	82 %	96 %	82 %	99 %
BLSTM	Original	100 %	77 %	94 %	80 %	98 %
BLSTM	SMOTE TL	98 %	82 %	95 %	85 %	98 %
BLSTM	SMOTE ENN	97 %	83 %	94 %	87 %	99 %
LightGBM	Original	95 %	82 %	95 %	85 %	98 %
LightGBM	SMOTE TL	97 %	79 %	94 %	84 %	98 %
LightGBM	SMOTE ENN	97 %	79 %	95 %	81 %	98 %
Random Forest	Original	96 %	71 %	89 %	83 %	94 %
Random Forest	SMOTE TL	95 %	76 %	90 %	87 %	95 %
Random Forest	SMOTE ENN	94 %	77 %	90 %	87 %	95 %

Tab. 3. Comparison of the classification results based on precision

Model	**Dataset**	**Precision**				
		Class 0	**Class 1**	**Class 2**	**Class 3**	**Class 4**
CNN	Original	99 %	91 %	97 %	83 %	99 %
CNN	SMOTE TL	99 %	72 %	97 %	61 %	99 %
CNN	SMOTE ENN	99 %	78 %	94 %	78 %	98 %
BLSTM	Original	99 %	89 %	98 %	69 %	100 %
BLSTM	SMOTE TL	99 %	73 %	95 %	63 %	99 %
BLSTM	SMOTE ENN	99 %	61 %	91 %	54 %	98 %
LSTM	Original	99 %	84 %	97 %	72 %	99 %
LSTM	SMOTE TL	99 %	63 %	95 %	49 %	98 %
LSTM	SMOTE ENN	99 %	74 %	94 %	66 %	96 %
SVM	Original	99 %	81 %	95 %	77 %	99 %
SVM	SMOTE TL	99 %	41 %	88 %	39 %	97 %
SVM	SMOTE ENN	99 %	39 %	85 %	38 %	98 %
LightGBM	Original	99 %	50 %	86 %	46 %	95 %
LightGBM	SMOTE TL	99 %	58 %	91 %	53 %	98 %
LightGBM	SMOTE ENN	99 %	80 %	95 %	76 %	98 %
Random Forest	Original	98 %	69 %	92 %	25 %	97 %
Random Forest	SMOTE TL	98 %	60 %	92 %	22 %	96 %
Random Forest	SMOTE ENN	98 %	53 %	90 %	22 %	94 %

On the other hand, we can notice that models fitted on the original dataset have higher precision. As it is desirable to maximize both recall and precision, only LSTM SMOTE ENN has the most optimal precision-recall trade-off, which is also confirmed by the high values of macro-averaged recall and ROC AUC (tab. 1) as well as the highest area under PR curves for the classes 1 and 3 (fig. 4).

Thus, this model is the best choice for classification of the disease cases which are the minority classes. Its architecture is sequential and is shown in fig. 3.

The tuned hyperparameters can be found in tab. 4.

Here we see that in contrast to the deep learning models, the machine learning algorithms chose the same optimal hyperparameters regardless of the type of resampling.

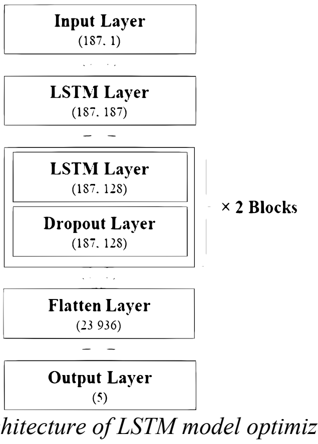

Fig. 3. Architecture of LSTM model optimized by the randomized cross-validation search algorithm. Shape of the output of a layer (or a block of layers) is specified in parenthesis

Tab. 4. Tuned hyperparameters for the resampled datasets. The hyperparameters not specified in the table have the default values

Algorithm	Dataset	Tuned hyperparameters
SVM	SMOTE TL	class_weight = "balanced", gamma = 0.1
SVM	SMOTE ENN	class_weight = "balanced", gamma = 0.1
LightGBM	SMOTE TL	class_weight = "balanced", learning_rate = 0.05, max_depth = 10, reg_alpha = 0.07, reg_lambda = 0.03, subsample = 0.7
LightGBM	SMOTE ENN	class_weight = "balanced", learning_rate = 0.05, max_depth = 10, reg_alpha = 0.07, reg_lambda = 0.03, subsample = 0.7
Random Forest	SMOTE TL	max_depth = 10, min_samples_leaf = 5, oob_score = True
Random Forest	SMOTE ENN	max_depth = 10, min_samples_leaf = 5, oob_score = True
CNN	SMOTE TL	strides = 2, pool_size = 2, padding = 'same', kernel_size = 6, layers = 3, filters = 128, dense_neurons = 128, dense_layers = 2
CNN	SMOTE ENN	strides = 2, pool_size = 2, padding = 'same', kernel_size = 3, layers = 3, filters = 128, dense_neurons = 64, dense_layers = 2
LSTM	SMOTE TL	layers = 5, learning_rate = 0.001, units = 128
LSTM	SMOTE ENN	layers = 2, learning_rate = 0.001, units = 128
BLSTM	SMOTE TL	layers = 2, learning_rate = 0.001, units = 64, dense_layers = 2, dense_neurons = 128
BLSTM	SMOTE ENN	layers = 5, learning_rate = 0.001, units = 64, dense_layers = 2, dense_neurons = 64

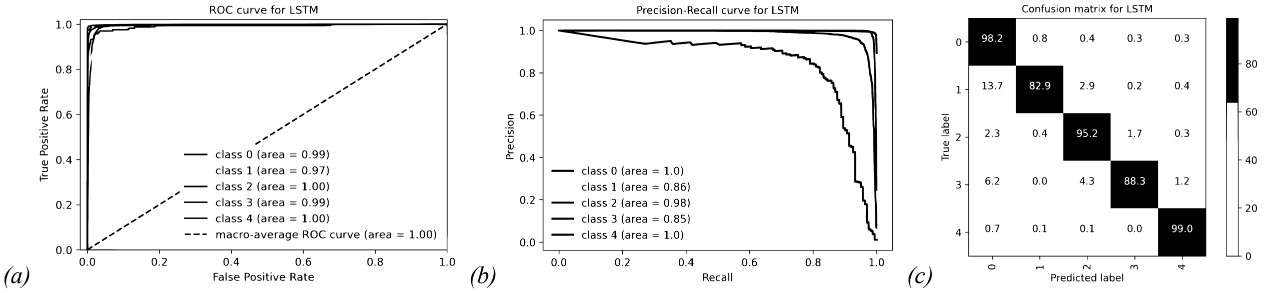

Fig. 4. (a) ROC curve, (b) PR curve and (c) confusion matrix for the LSTM model fitted on data resampled using SMOTE ENN method

As a comparison, Figure 5 shows SVM model which is the best performer among the machine learning algorithms according to tab. 1.

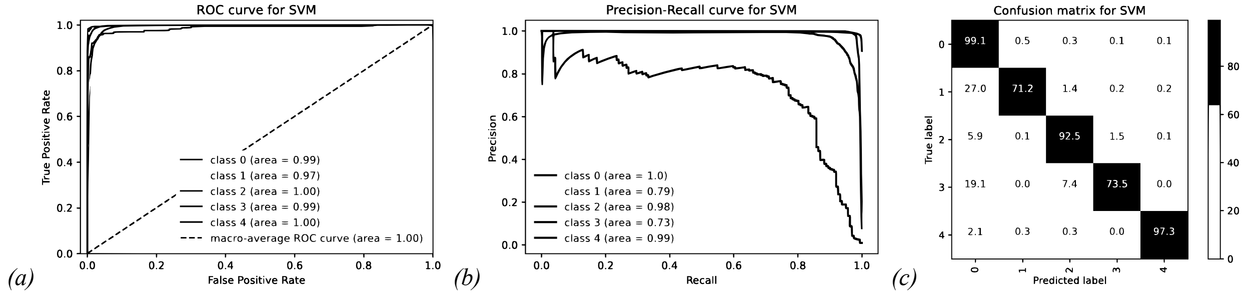

Fig. 5. (a) OC curve, (b) PR curve and (c) confusion matrix for the SVM model fitted on original data

As we see, it has the lowest precision value, so it is not appropriate for the task as it does not minimize precision-recall trade-off as effectively as the LSTM model described in fig. 3.

4. Discussion

While most of the models classified the largest classes quite successfully, they were not as good in detecting the minority classes 1 and 3, although class 2, to which 6.6 % of the data belong, was classified more accurately.

An essential conclusion of this project is that machine learning algorithms trained on the resampled data did not show significantly improved metrics, in particular recall. One of the causes of this behavior is that synthetic samples which were added to the data could not contribute to the successful classification, because the algorithm itself was not appropriate for the imbalanced dataset [9]. At the same time, deep learning models were able to use the new data successfully and showed improved metrics for both original and resampled datasets. As a result, a model of a simple structure capable of classification of the minority classes was developed.

Tab. 5 and 6 show comparison of the model performance results as described by different authors.

Tab. 5. Comparison of ECG classification results. This paper used macro-weighted f1-score, precision, recall. Some papers used different preprocessing and feature selection methods

Paper	Model	Recall	Accuracy
This paper	SMOTE ENN LSTM	91.94 %	97.22 %
Acharya et al. [14]	CNN	96.01 %	93.47 %
Martis et al. [15]	PCA & LS-SVM	99.46 %	93.76 %
Li et al. [16]	Random Forest	-	94.60 %
Shoughi et al. [17]	SMOTE & CNN-BLSTM	-	98.71 %

Tab. 6. Comparison of ECG classification results by class

Paper	Model	Recall				
		Class 0	Class 1	Class 2	Class 3	Class 4
This paper	SMOTE ENN LSTM	98 %	83 %	95 %	88 %	99 %
Shoughi et al. [17]	SMOTE & CNN-BLSTM	99 %	93 %	97 %	83 %	100 %

Thus, the main contributions of this study are the following:

- An approach to multiclass classification based on resampling and a deep learning model. This approach can be used to solve tasks other than ECG classification;
- LSTM model fitted on resampled data is the best classifier when applied to the imbalanced dataset.

In the future, more advanced algorithms can be used to create synthetic samples such as Generative adversarial networks in order to deal with imbalanced datasets.

Conclusion

As described earlier, deep learning models show improved performance when trained on the resampled dataset in case of severe imbalance in the original dataset. Hence, the approach to multiclass classification which consists of up-sampling and down-sampling of original data and application of a deep learning model could be used for any imbalanced dataset.

One of the problems which was not covered by this paper is that neural networks and machine learning models are susceptible to adversarial attacks. For example, if noise or a non-human heartbeat is introduced, then the algorithm may still use it as if it belonged to a human and the result would be different. Thus, it is important to create neural network that are robust to such cases.

To conclude, the methodology developed in this paper could potentially improve cardiac arrhythmias detection, given limited medical help. Future research could attempt to assess model's generalizability by testing the developed model against data from another database [18]. It is also possible to implement model stacking to improve the quality of the classification [19].

Acknowledgements

The authors would like to acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work: http://dx.doi.org/10.5281/zenodo.22558. Specifications: https://www.arc.ox.ac.uk/arc-systems.

References

[1] WHO list of priority medical devices for management of cardiovascular diseases and diabetes. In Book: WHO medical device technical series. Geneva: World Health Organization; 2021. License: CC BY-NC-SA 3.0 IGO. ISBN: 978-92-4-002797-8. Source: ⟨https://www.who.int/publications/i/item/9789240027978⟩.

[2] Hong S, Zhou Y, Shang J, Xiao C, Sun J. Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review. Comput Biol Med 2020; 122: 103801. DOI: 10.1016/j.compbiomed.2020.103801.

[3] Haibo H, Yunqian M, eds. Imbalanced learning: Foundations, algorithms, and applications. Hoboken, New Jersey: John Wiley & Sons Inc; 2013. ISBN: 978-1-118-07462-6. DOI: 10.1002/9781118646106.

[4] Murat F, Yildirim O, Talo M, Baloglu UB, Demir Y, Acharya UR. Application of deep learning techniques for heartbeats detection using ECG signals-analysis and review. Comput Biol Med 2020; 120: 103726. DOI: 10.1016/j.compbiomed.2020.103726.

[5] Hannun AY, Rajpurkar P, Haghpanahi M, Tison GH, Bourn C, Turakhia MP, Ng AY. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nat Med 2019; 25(1): 65-69. DOI: 10.1038/s41591-018-0268-3.

[6] Kachuee M, Fazeli S and Sarrafzadeh M. ECG Heartbeat classification: A deep transferable representation. IEEE Int Conf on Healthcare Informatics (ICHI), New York City, NY, USA 2018: 443-444. DOI: 10.1109/ICHI.2018.00092.

[7] Zhong ZX, Michael AJ, Lun ZJ, Yue DH. ECG classification using machine learning techniques and smote oversampling technique. 2nd Int Conf on Image Processing and Machine Vision (IPMV 2020) 2020: 10-13. DOI: 10.1145/3421558.3421560.

[8] Chawla N, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic minority over-sampling technique. J Artif Intell Res 2002; 16: 321-357. DOI: 10.1613/jair.953.

[9] He H, Garcia EA. Learning from imbalanced data sets. IEEE Trans Knowl Data Eng 2009; 21(9): 1263-1284. DOI: 10.1109/TKDE.2008.239.

[10] Avanzato R, Beritelli F. Automatic ECG diagnosis using convolutional neural network. Electronics 2020; 9(6): 951. DOI: 10.3390/electronics9060951.

[11] Greff K, Srivastava R, Koutnik J, Steunebrink BR, Schmidhuber J. LSTM: A search space odyssey. IEEE Trans Neural Netw Learn Syst 2017; 28(10): 2222-2232. DOI: 10.1109/TNNLS.2016.2582924.

[12] Liu G, Guo J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. Neurocomputing 2019; 337(C): 325-338. DOI: 10.1016/j.neucom.2019.01.078.

[13] Rao G, Huang W, Feng Z, Cong Q. LSTM with sentence representations for document-level sentiment classification. Neurocomputing 2018; 308: 49-57. DOI: 10.1016/j.neucom.2018.04.045.

[14] Acharya UR, Oh SL, Hagiwara Y, Tan JH, Adam M, Gertych A, Tan RS. A deep convolutional neural network model to classify heartbeats. Comput Biol Med 2017; 89: 389-396. DOI: 10.1016/j.compbiomed.2017.08.022.

[15] Martis RJ, Acharya UR, Lim CM, Mandana K, Ray AK, Chakraborty C. Application of higher order cumulant features for cardiac health diagnosis using ECG signals. Int J Neural Syst 2013; 23(04): 1350014. DOI: 10.1142/S0129065713500147.

[16] Li T, Zhou M. ECG classification using wavelet packet entropy and random forests. Entropy 2016; 18(8): 285. DOI: 10.3390/e18080285.

[17] Shoughi A, Dowlatshahi MB. A practical system based on CNN-BLSTM network for accurate classification of ECG heartbeats of MITBIH imbalanced dataset. 2021 26th Int Computer Conf Computer Society of Iran (CSICC) 2021: 1-6. DOI: 10.1109/CSICC52343.2021.9420620.

[18] Schetinin E. Automatic arrhythmia detection based on the analysis of electrocardiograms with deep learning. Herald of Computer and Information Technologies 2021; 18(5): 18-27. DOI: 10.14489/vkit.2021.05.pp.018-027.

[19] Gaowei X, Tianhe R, Yu C, Wenliang C. A one-dimensional CNN-LSTM model for epileptic seizure recognition using EEG signal analysis. Front Neurosci 2020; 14: 578126. DOI: 10.3389/fnins.2020.578126.

Authors' information

Eugene Yurievich Shchetinin (b. 1962), graduated from Moscow State University in 1985, majoring in Applied Mathematics. Currently he works as a professor of Mathematics department at Financial University under the Government of the Russian Federation. Research interests are data analysis, machine learning, computer vision.
E-mail: *riviera-molto@mail.ru* .

Anastasia Gennadievna Glushkova (b. 1996), graduated from University of Oxford in 2021, majoring in Statistical Science. Currently she works as Senior Data Analyst at Endeavor. Research interests are data analysis, deep learning, computer vision. E-mail: *aglushkova@endeavorco.com* .

Code of State Categories Scientific and Technical Information (in Russian – GRNTI)): 28.17.37
Received February 21, 2022. The final version – June 13, 2022.