Atrial Fibrillation Classification from a Short Single Lead ECG Recording Using Hierarchical Classifier

Erin E Coppola, Prashnna K Gyawali, Nihar Vanjara, Daniel Giaime, Linwei Wang

Rochester Institute of Technology, Rochester, USA

Abstract

Atrial fibrillation (AF), one of the most common cardiac arrhythmias, can be diagnosed using electrocardiography. We present a data-driven model to automatically detect the occurrence of atrial fibrillation on a single lead electrocardiogram (ECG). Our model incorporates a wide range of features including heart rate variability in the time and frequency domain, spectral power analysis and statistical modeling of atrial activity. We use an over-sampling strategy to balance the dataset across different categories. We design a hierarchical classification model to predict an ECG signal as either AF, normal, noisy or an alternative rhythm. The best performance was achieved with a hierarchical bagged ensemble classifier, with an average F_1 score of 0.7855 over all samples.

1. Introduction

Atrial fibrillation (AF) is a very common arrhythmia associated with serious heart-related complications including stroke and heart failure [1]. The incidence of AF increases with age and presence of chronic health conditions. Electrocardiography is currently the gold standard in the diagnosis of AF, since it accurately captures the electrical activity of the heart [2]. It is very difficult to diagnose AF during routine in-office visits, since symptoms occur in episodes [1].

Recent approaches have been considered to address the high mortality rate and low efficacy in detection of AF [2]. AF detectors allow for earlier screening and identification of AF compared to manual methods. Current algorithms are mostly based on ventricular response and/or atrial activity analysis. Recent work has found several features that characterize AF including heart rate variability, wavelet entropy, and p-wave detection [3]. However, the application of current AF detection methods to clinical settings are limited [4]. In previous studies, classification was performed only on clean data. However, noise is inevitable in continuous-monitoring settings, due to lead detachment, respiration, or motion. In addition, such classification was performed to only distinguish AF signals from normal signals [4]. Since AF is often misdiagnosed as other arrhythmia types, classification of AF against an alternative rhythm would help in making the detector more robust.

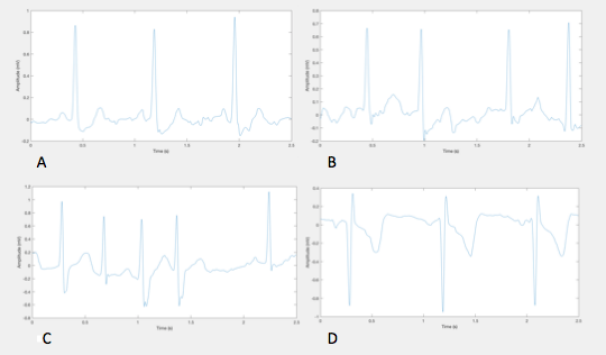

Figure 1. Representative ECG waveforms for each category: (A) normal rhythm, (B) atrial fibrillation rhythm, (C) alternative rhythm and (D) noisy signal.

We use the data provided by the AliveCor device, available from PhysioNet [5]. In this data set, 8,658 single lead ECG signals were collected lasting from 9 s to 60 s. The signals were recorded at a rate of 300 Hz and have been bandpass filtered. The data is imbalanced in regards to the number of signals per classification category. The largest categories include normal rhythm and alternative rhythm signals, and only a limited number of signals are included in AF and noisy categories. This sample distribution reflects a real-world data set, where only a small percentage of abnormal examples are available. We use a synthetic minority over-sampling technique (SMOTE) to oversample the minority class/es, thereby increasing the sensitivity of our classifier [6]. Figure 1 shows representative signals from each of the classification categories.

2. Methods

A machine learning model is presented for the classification of provided ECG signals into four different rhythm types. In this section, we first elucidate our strategy to extract different features and then describe how we are ad-

 ISSN: 2325-887X DOI:10.22489/CinC.2017.354-425

dressing the issue of class imbalance. Finally, a description of the proposed model is presented to classify extracted features into four different classes.

2.1. Feature Extraction

A set of numeric features are extracted from the raw ECG signals. We broadly categorize feature extraction into three main feature subsets: ventricular response, atrial activity, and raw ECG signal features. 17 ventricular response features are selected based on time and frequency domain analysis of RR-intervals. 4 atrial activity features are selected based on the p-wave morphological analysis. Raw ECG signals are then processed into 7 features. Additional statistical features (maximum, minimum, mean, variance) are also included. In total, 60 features are extracted from the ECG signal.

2.1.1. Ventricular response features

We extract ventricular response features to address the irregular nature of atrial fibrillation rhythm. These are extracted from the timing of heartbeats (RR-intervals) in ECG signal. Extracted time-domain features for characterizing ventricular responses:

- Average of all heartbeats (RR-intervals)
- Ratio of heartbeats considered normal
- Standard deviation of all heartbeats
- RMS difference between heartbeat intervals
- Ratio of heartbeat intervals differing by > 50 ms
- Maximum heartbeat length
- Minimum heartbeat length
- Average peak of heartbeat peaks
- Standard deviation of all heartbeat peaks

Extracted frequency-domain features characterizing ventricular responses:

- Total spectral power, up to 0.04 Hz
- Total spectral power, 0 to 0.003 Hz
- Total spectral power, 0.003 to 0.04 Hz
- Total spectral power, 0.04 to 0.15 Hz
- Total spectral power, 0.15 to 0.40 Hz
- Ratio of high to low frequency power
- Average spectral power of RR-intervals

Besides these features, we also incorporate multiscale entropy analysis of RR-intervals to account for signal complexity.

2.1.2. Atrial activity features

As atrial fibrillation is characterized by the lack of visible p-waves [7], we extract atrial activity features from the morphology of p-waves. Statistical modeling of p-wave features is used to quantify morphological features and detect p-wave absence in atrial fibrillation. The p-waves are segmented into six pieces and the mean amplitude is calculated. Statistical features calculated from the extracted p-wave amplitudes are:

- Variance of p-wave segment means
- Skewness of p-wave segment means
- Kurtosis of p-wave segment means
- Average of p-wave peaks

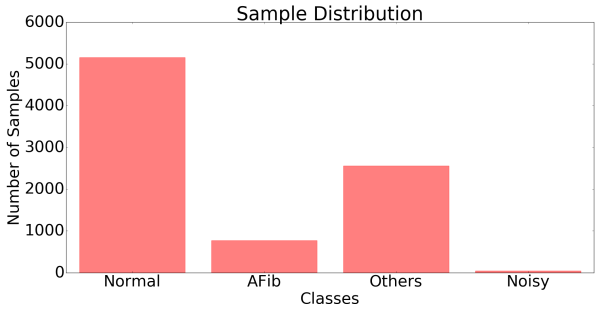

Figure 2. ECG sample distribution plot, showing the number of ECG samples present in the data subset of each class.

2.1.3. Other ECG features

We extract different features relating to power spectrum from the signal amplitude as the ECG signal features.

- Average spectral power from signal
- Variance of spectral power from signal
- Average spectral power from signal
- Variance of spectral power from signal

We also calculate root mean square fluctuation for the integrated and detrended time series. This reveals the long-range correlations in the raw ECG time series. Finally, fast Fourier transform is also calculated from the raw signal.

- Root mean square fluctuation of time series
- Average total power of time series
- Variance of total power of time series

2.2. Class imbalance

Building machine learning models using skewed datasets is a challenging task. In the current task of AF classification, the class imbalance prevalent in our dataset can be visually seen in Fig 2. Different techniques [8], [6], have been proposed to address the issue of class imbalance where most of them are by re-sampling the dataset to offset the imbalance. Depending upon the task in hand, majority class(es) are down-sampled or minority class(es) are up-sampled. Each of these techniques has their own merits and demerits, but one striking disadvantage might be the loss of information when down-sampling a dataset. As such we are using SMOTE (Synthetic Minority Over-sampling Technique) [6] for increasing the sample size of the minority classes. By generating such samples, the

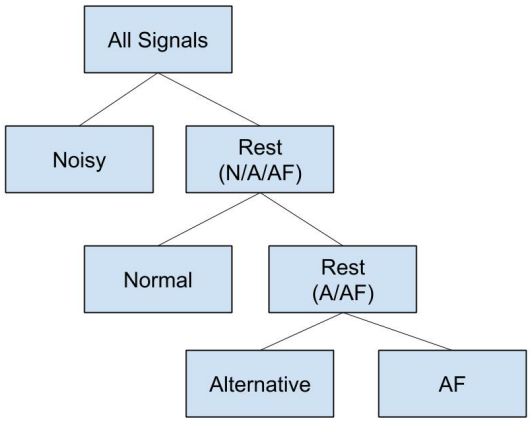

Figure 3. Illustrative diagram of the proposed hierarchical architecture. Binary classification is performed at each hierarchy stage.

learner or classifiers is able to broaden its decision regions for the minority class.

We apply SMOTE by making the pairs involving each minority class with majority class. Let X_i represents the sample from class c_i. If c_k represents the majority class then pairs like (X_k, X_i) such that $k \neq i$ are formed and SMOTE is applied on each pair. At the end of this process, each of the minority classes will have a same number of samples as the majority class c_k.

2.3. Hierarchical classification

Hierarchical classification is relevant when some classes of similar objects are related to one another compared to other classes. Classification between one class and the remaining classes is performed in each hierarchy and after each stage, the lone class is dropped as shown in Fig3. As such, each classifier is responsible for binary classification. In this setup, we need to train $k - 1$ different models for classifying the k number of classes.

Three models are trained independently for our extracted feature data set. A hierarchical model is a stack of these independently trained classifiers, as shown in Fig 3. At each stage, the classifier divides the data into two parts. Let X_n^i, represents data where n is the number of samples and i is the set of classes present in X. M_k represents the model used at the k^{th} stage of the hierarchy. When X_n^i is the input for the k^{th} stage of the hierarchical model, the data is divided into two parts by model M_k: $X_{n_1}^p$ and X_{n2}^q. Here, $n_1 + n_2 = n$ and $p + q = i$. Also, p contains only one of the classes present in set i whereas q contains all of the remaining classes at that hierarchy stage. After extensive experiments with various classifiers, we achieved the best results with AdaBoost, Bagging and Robust Boost respectively in each of the three hierarchy

AdaBoost: AdaBoost [9] is a classification technique which uses a combination of weak classifiers to create a robust classifier. AdaBoost is much more resistant to over fitting than other simple classifiers, making it one of the ideal choices for classification. AdaBoost works by initially assigning every sample the same weight. A base classifier then classifies the samples and assigns the higher weight to misclassified samples. Finally, a new base learner is used to classify the samples with new weights and the process continues until a high accuracy is achieved or until the model's performance saturates.

Bagging: Bagging, also known as Bootstrap Aggregation [10], is a technique for generating multiple versions of the classifier model and uses a majority vote to classify samples. It also helps in reducing errors generated by fluctuations in the training samples. Given a dataset with n samples, bagging samples m new datasets having n' samples. For each new dataset, a model is generated. Finally, the m models are aggregated by taking the majority vote.

Robust Boost: Robust boost [11] is able handle data consisting of noisy labels better than other boosting algorithms (i.e. AdaBoost). The objective of Robust boost is to minimize the margin based cost function. Robust boost is much less sensitive to noise when compared to other boosting algorithms.

3. Experiments

Signal analysis and feature extraction is performed using the WFDB Software Package applications [12]. First, the QRS complexes within the electrocardiographic signals are annotated using the *gqrs* application. P-waves are annotated using the *ecgpuwave* application. Heart rate variability analysis is performed using the HRV Toolkit. Multiscale entropy analysis is performed using the *mse* and *sampen* applications. The spectral analysis utilized the DFA software [13] for detrended fluctuation analysis and the *fft* application for fast Fourier transformation. QRS waveform boundary recognition is performed using *ecgpuwave* to find the onset and offset of p-waves in each signal. Power spectrum analysis is carried out using the *lomb* and *memse* of WFDB applications. SMOTE, used to address class imbalance, was performed using a Python-based library, *imbalanced-learn* [14]. Finally, the proposed hierarchical model was designed using MATLAB.

We trained various hierarchical models using different combinations of classifiers throughout the hierarchy. The best hierarchical model is chosen based on F_1 score performance. The hierarchical model is also compared against the four-class classification model (*flat classification*). Since binary classification is performed at each hierarchical stage, the classifiers that performed best could not be directly applied to perform multi-class classification.

Table 1. F-score for each individual classes along with average score for all the classes.

Method	Noise	Normal	AF	Other	Combined
Flat classification	0.75	0.8049	**0.8276**	0.5038	0.7216
Hierarchical	0.8254	**0.8690**	0.7250	0.6286	0.7620
Hierarchical + SMOTE	**0.8522**	0.7826	0.7023	**0.8049**	**0.7855**

3.1. Model training

The provided challenge dataset [5] includes both the training and validation set. However, the validation set was provided only to verify the implementation correctness during the submission and includes the same data from the training set itself. As such, we manually removed the validation set from the training set to ensure the testing of our model is performed on a held-out set.

3.2. Results

The results of the experiments are recorded in Table 1. The presented results include F_1 score for each individual class along with the average score for all the classes. The average F_1 score recorded using the hierarchical model is 0.7620, 5% increment over the result from the flat classification. Further improvement of around 3% is observed after addressing class imbalance (with SMOTE) taking average F_1 score to 07855. Comparing individual F_1 score, we can observe increment in predictability for noisy signals and an alternative rhythm. This however, seems to be affecting the prediction ability for normal and AF rhythm.

4. Discussion

In this current work, we are using a total of 60 features with all features having equal weight. In future work, feature importance analysis could be performed to drop or add more extracted features in order to improve the predicting capability of the model. In addition, analyzing a subset of features important at each hierarchy would help in creating better models. We also plan to investigate the drop in performance for AF rhythm as seen by using hierarchical models. Finally, automatic feature extraction can be performed using deep networks. We believe augmenting such automatically extracted features with our extracted feature set would further improve the models' performance.

5. Conclusion

In this work, we present a hierarchical classifier to classify atrial fibrillation from a short single lead ECG recording. We incorporate a wide variety of features to perform classification between four different rhythm categories. The extracted features are then used in the proposed hierarchical classifier to perform binary classification at each hierarchy stage.

Acknowledgements

This work is done as a part of the PhysioNet Computing in Cardiology Challenge 2017 [5].

References

[1] DT L. Accurate, automated detection of atrial fibrillation in ambulatory recordings. Cardiovasc Eng Technol 2016; 7(2):182–189.

[2] Rienstra M1 Lubitz SA MSMJea. Symptoms and functional status of patients with atrial fibrillation: state of the art and future research opportunities. Circulation 2000; 125(23):2933–2943.

[3] Garca M Rdenas J AReа. Application of the relative wavelet energy to heart rate independent detection of atrial fibrillation. Comput Methods Programs Biomed 2016; 131(7):157–168.

[4] Huang C Ye S CHea. A novel method for detection of the transition between atrial fibrillation and sinus rhythm. IEEE Trans Biomed Eng 2011;58(4):1113–1119.

[5] Gari Clifford Chengyu Liu BMISQLAJ, Mark R. Af classification from a short single lead ecg recording: the physionet computing in cardiology challenge 2017. Computing in Cardiology Rennes IEEE 2017. In Press.;.

[6] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research 2002;16:321–357.

[7] S Ladavich BG. Rate-independent detection of atrial fibrillation by statistical modeling of atrial activity. Biomedical Signal Processing and Control 2015;18:274–281.

[8] Seiffert C, Khoshgoftaar TM, Van Hulse J, Napolitano A. Rusboost: Improving classification performance when training data is skewed. In Pattern Recognition, 2008. ICPR 2008. 19th International Conference on. IEEE, 2008; 1–4.

[9] Freund Y, Schapire R, Abe N. A short introduction to boosting. Journal Japanese Society For Artificial Intelligence 1999;14(771-780):1612.

[10] Breiman L. Bagging predictors. Machine learning 1996; 24(2):123–140.

[11] Freund Y. A more robust boosting algorithm. arXiv preprint arXiv09052138 2009;.

[12] Silva I MG. An open-source toolbox for analysing and processing physionet databases in matlab and octave. Journal of Open Research Software 2014;2(1):27.

[13] Peng C-K Halvin S SHea. Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series. Chaos 1995;5:82–87.

[14] Lemaître G, Nogueira F, Aridas CK. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. J Machine Learning Research 2017; 18(17):1–5.

Address for correspondence:

Name: Erin E Coppola

Full postal address: 6000 Reynolds Drive, Rochester, NY, 14623

E-mail address: eec3769@rit.edu