# A Similarity Constrained Signal Selection Algorithm for Contrastive Learning In The Detection of Atrial Fibrillation

Alejandro Navarro Luna, Karen Andrea Fonseca and Carlos A. Fajardo
Department of Electrical Engineering, Universidad Industrial de Santander
alnavluna@gmail.com, karen.fonseca@correo.uis.edu.co and cafajar@uis.edu.co

## Abstract

*Recent advances in deep learning have led to several techniques for accurately detecting arrhythmias. However, these techniques depend highly on labeled data; obtaining these labels is challenging and expensive. Recently, a self-supervised learning technique called contrastive learning has emerged as a promising method to use unlabeled data to increase model accuracy. This technique is usually used as a pre-training stage for deep learning models. In this study, we propose incorporating a clustering stage before contrastive learning to select a set of unlabeled ECG signals that strengthen the pre-training stage in scenarios with a shortage of labeled data. Our results suggest that our method outperforms the baseline strategy that only uses contrastive learning. Specifically, for a model trained with only 5k labeled data, our approach improves it by 4.9 on the 10-fold mean F1-score and 0.4 on the AUC metric compared to the baseline model.*

## 1. Introduction

The most prevalent cardiac arrhythmia is atrial fibrillation (AFib). This arrhythmia causes high mortality, morbidity, and higher healthcare expenses and is characterized by disorganized and erratic atrial beats [11, 17]. Therefore, it is pertinent to detect arrhythmias promptly to minimize these complications and reduce the risk of death.

The electrocardiogram (ECG) is a well-established and widely used technique for monitoring heartbeats. These ECGs are interpreted by expert physicians, cardiologists, or electrophysiologists to determine many heart diseases. In the context of machine learning, a consensus of experts determine the labeled version of the ECG data so that a machine learning model can learn from it [13].

In the recognition and classification of data, deep learning has achieved significant advances [15], including ECG signals [9, 18–20]. However, these deep learning solutions need a large amount of labeled data, which requires much human effort for production. Labeled medical data is time-consuming, susceptible to mislabeling, and expensive. On the other hand, unlabeled medical data, which corresponds to data not tagged for its classification, is easier and cheaper to obtain. Large volumes of ECG data are recorded by remote monitoring equipment.

On the other hand, unsupervised learning can also be used as a deep learning method. Self-supervised learning involves finding good representations from unlabeled data and can be deemed a subclass of unsupervised learning. However, unsupervised learning centers around clustering, grouping, and dimensionality reduction of the data [7]. In recent years, self-supervised learning methods that use a contrastive loss have improved supervised learning methods [1, 2, 8]. Contrastive learning (CL) is a self-supervised learning technique that involves finding good representations from unlabeled data, taking advantage of the accessibility of this type of data.

We propose to use another stage previous to CL which seeks to improve the performance of a CNN for the classification of atrial fibrillation. We seek to enhance the CNN found in [5] because it performs well in challenging scenarios, such as using few-labeled data and CNNs with fewer parameters. This stage consists of a data selection or manual clustering algorithm, proposed in [12], to select the more relevant samples of an unlabeled dataset to have an optimal pre-trained model.

Clustering is one of the most fundamental tools in unsupervised learning. This technique groups data into different clusters without any labels. In our case, we used the similarity-constrained selection algorithm proposed in [12] to test which set of signals selected would improve the CL-based pre-training and, ultimately, the performance of the trained CNN on the test set. To the best of our knowledge, this is the first work that seeks to use clustering techniques to select the unlabeled signals to be used in CL to detect AFib.

The rest of this work is ordered as follows: Section 2 describes the proposed approach and its stages. Section 3 shows the results of the different implementations using various metrics. Section 4 discusses and compares our results with other works. Section 5 concludes the results.

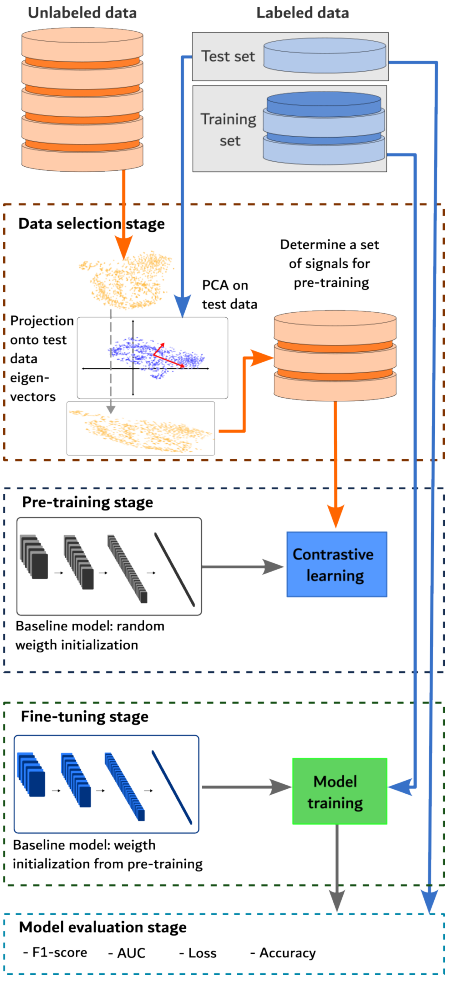


Figure 1. The proposed stages for obtaining a trained CNN model for classifying cardiac arrhythmias.

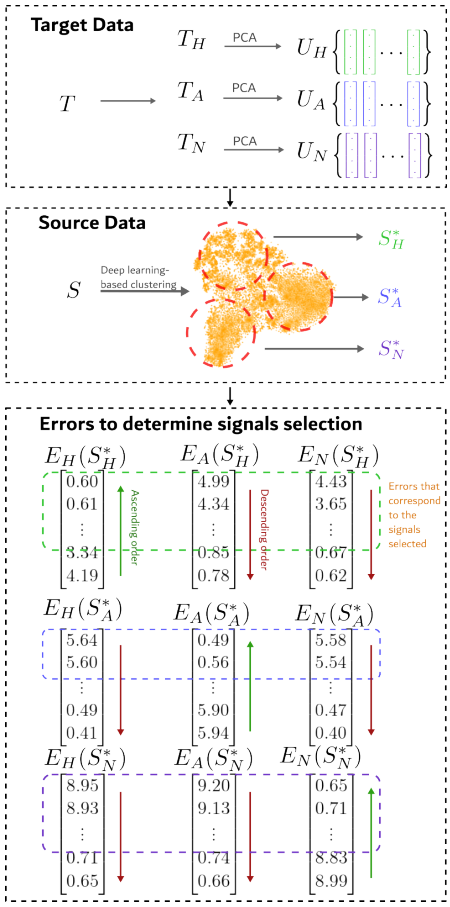


Figure 2. Similarity constrained data selection algorithm for ECG signals with three labels normal (H), arrhythmia (A), and noise (N). PCA is applied to the target data on each class ($T_H$, $T_A$, and $T_N$) to obtain a base for each class subspace. Clustering is applied to the unlabeled source data to discover and select the subset of signals of each class. Nine errors are obtained to determine the signals in the source data most similar to the test data.

## 2. Proposed Approach

The general workflow consists of four main stages; data selection, pre-training, fine-tuning, and model evaluation (Figure 1). The data selection stage is a clustering algorithm corresponding to a similarity-constrained data selection algorithm proposed in [12]. Next, the pre-training stage consists of using the set of ECG signals selected in the previous step as the unlabeled dataset for CL-based pre-training of the model. The guidelines for the CL algorithm implementation were taken from [5]. Afterward, The model is trained using labeled data. Finally, the results from each training section are evaluated according to their classification of the test set data.

### 2.1. Data Selection Stage

The algorithm proposed in [12] aims to select data points from a source set that have the most similarity with data points in a target set. We utilize the same algorithm, adapt it to ECG signals, and use three classes: normal, cardiac arrhythmias (AFib or Atrial Flutter (AF)), and noise. We use the test and unlabeled datasets as the target and source sets, respectively.

Figure 2 shows the detailed steps of the data selection stage. The target and source datasets are denoted by $T$ and $S$, respectively. Next, let us denote by $T_H \subseteq T$, $T_A \subseteq T$, and $T_N \subseteq T$ the subsets of ECG signals in the target set labeled as normal (H), arrhythmia (A), or noise (N), respectively. Every target data from $T_H$, $T_A$, and $T_N$ is normalized to have zero mean and unit standard deviation. PCA is applied to each subset in the target data, and $d$ eigenvectors corresponding to the $d$ largest eigenvalues are chosen as a base for the subspace of each subset. In the next step, since the source data is unlabeled, clustering is performed to find and select the corresponding subsets. Variational Deep

Embedding (VaDE) [6] is used as the clustering algorithm. VaDE uses variational autoencoders and has been applied for ECG analysis [10]. This algorithm is comparable to the conventional Gaussian mixture models (GMM) clustering algorithm, except the feature space is automatically learned. These subsets are determined with clustering and will, unfortunately, contain some data points corresponding to different classes. Therefore, the corresponding subsets of $S$ are denoted with an asterisk (*). Hence, $S_H^* \subseteq S$, $S_A^* \subseteq S$, and $S_N^* \subseteq S$ are the subsets of the ECG signals in the source set. In the last step, each subset found from the source data is projected onto each set of eigenvectors for each class in the test dataset. This projection is denoted by $Proj_{U_i} S_i^*$, i.e., $Proj_{U_H} S_H^*$ is $S_H^*$ projected onto $T_H$. This results in 9 different projection errors. Based on these projections, the following similarity function considering the projection errors is defined:

$$E_i S_j^* = \left\| Proj_{U_i} S_j^* - S_j^* \right\|_2 \quad (1)$$

where $\|*\|_2$ stands for the $\ell_2$-norm. Also, $i$ and $j$ can be any of the classes as mentioned earlier labels (H, A, or N). Equation 1 projects all signals in $S_H^*$, $S_A^*$, and $S_N^*$ and builds nine errors. The errors where $i = j$ are sorted in ascending order to obtain the most similar signals (smallest error) in the same class. The errors where $i \neq j$ are sorted in descending order for a similar reason as before, which is to obtain the signals that are the most different (biggest error) between different classes. Finally, each error vector's first $k$ values are selected and used as input for the pre-training.

### 2.2. Pre-training Stage

Deep learning scenarios that require a lot of data can benefit from Data Augmentation (DA) [14]. In the medical field, comparable circumstances and short time scales may be shared by physiological recordings from the same patient that was gathered [3]. The same signal can be transformed to produce positive sets of ECG recordings. Figure 3 shows the case where double data (used for DA) is obtained from the ECG recordings. The CL model consists of the encoder and the projection layer. The encoder is a CNN that learns to extract features from the input data. Next, the projection layer brings these features into an $n$-dimensional embedding space where the transformed pairs can be compared. Finally, the contrastive loss function teaches the model that positive pairs are closer together and farther away from negative pairs. The next step is to test the model on a contrastive validation set. Eventually, this hands us the contrastive metrics: contrastive loss and accuracy (from the contrastive validation step).

### 2.3. Training Stage

The sparse categorical cross-entropy loss function and the Adam optimizer are both used during the training procedure with the default settings. We take the pre-trained encoder and add a linear probe layer after finishing the pre-training phase. Lastly, we use a certain number of signals to train the encoder. The classification metrics are the F1 score, the area under the curve (AUC), loss, and accuracy.

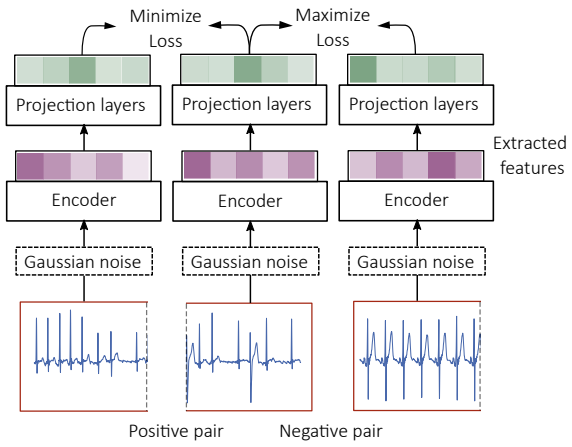


Figure 3. Contrastive learning applied to ECG signals [5]. ECG signals are passed through the model to obtain a feature vector representing the input signal.

## 3. Results

In this section, we will present the experiments performed to validate our method, a database description, and the training conditions.

### 3.1. Experiment setting

We train the model proposed in [4] by modifying the size of the initial convolutional filter and the model depth. The metrics used are the F1-score and AUC on the test set. Additionally, we use 10-Folds to calculate the mean and the standard deviation on the test set. All the simulations ran on Nvidia V100 SXM2 GPUs. For the Pre-training and training stage, we set a batch size of 256 ECG signals. The amount of unlabeled data used for the pre-training stage ranged between 0, 5k, 10k, 15, 20k, 25k, and 50k.

We use the Icentia11k [16] dataset, the largest public ECG dataset of continuous raw signals for representation learning. This dataset corresponds to 11000 patients and is classified according to rhythm type: NSR (Normal Sinus Rhythm), AFib, AF, and Noise.

### 3.2. Experiments

We performed experiments using two different data selection procedures: Random Selection (RS) and Similarity Constrained Selection (SCS). Both procedures involved selecting $K$ double-length unlabeled signals and both were followed by the Contrastive Learning algorithm. Besides,

we also compare our results with Random Initialization (RI), which involves no pre-training process.

Tables 1 and 2 present the F1-scores and AUC values for each model on the test set. The models were initialized using different methods, including Random initialization (RI), Random Selection in Contrastive Learning (RS+CL), and Similarity Constrained Selection in Contrastive Learning (SCS + CL). The training process is repeated ten times, and the resulting F1-score and AUC are averaged.

We evaluated each case on three different models of varying sizes. The smallest model is referred to as M1, the medium-sized model as M2, and the largest model as M3. Each model increases in depth and complexity of the CNN.

| Model | Init. method | Labeled data 5k | 10k | 15k |
|---|---|---|---|---|
| M1 (66k param.) | RI | $50.4 \pm 9.8$ | $70.3 \pm 4.3$ | $77.6 \pm 4.1$ |
| | RS+CL [5] | $64.9 \pm 5.5$ | $79.1 \pm 3.2$ | $82.1 \pm 1.5$ |
| | SCS+CL (Ours) | $\mathbf{69.9 \pm 0.4}$ | $\mathbf{83.3 \pm 2.3}$ | $\mathbf{84.2 \pm 0.1}$ |
| | $\Delta_1$ | **19.4** | 13.0 | 6.6 |
| | $\Delta_2$ | **4.9** | 4.2 | 2.1 |
| M2 (250k param.) | RI | $47.9 \pm 14$ | $78.7 \pm 3.5$ | $82.5 \pm 3.3$ |
| | RS+CL [5] | $66.2 \pm 6.6$ | $79.6 \pm 2.6$ | $82.0 \pm 1.3$ |
| | SCS+CL (Ours) | $\mathbf{67.7 \pm 5.8}$ | $\mathbf{81.5 \pm 2.6}$ | $\mathbf{82.9 \pm 1.4}$ |
| | $\Delta_1$ | **19.8** | 2.8 | 0.4 |
| | $\Delta_2$ | **0.2** | $-0.1$ | 0.1 |
| M3 (1M param.) | RI | $46.0 \pm 11$ | $76.2 \pm 10$ | $\mathbf{84.1 \pm 1.5}$ |
| | RS+CL [5] | $63.7 \pm 5.2$ | $75.6 \pm 4.6$ | $78.9 \pm 3.2$ |
| | SCS+CL (Ours) | $\mathbf{64.3 \pm 4.6}$ | $\mathbf{77.4 \pm 3.9}$ | $79.3 \pm 3.2$ |
| | $\Delta_1$ | **18.3** | 1.2 | $-4.8$ |
| | $\Delta_2$ | 0.6 | **1.9** | 0.4 |

Table 1. Average F1-scores on the Icentia11k [16] test set. $\Delta_1$ and $\Delta_2$ is the difference between our method compared with RI and RS+CL, respectively.

## 4. Discussion and Analysis

Our experimental results suggest that the proposed approach is highly effective for small models, particularly with few labeled data scenarios where the improvement is most significant. As observed in table 1 and 2, our strategy outperforms RS+CL by up to 4.9% on the F1-score and 0.4 on the AUC metric using a model with 66k parameters. We argue that this is due to the combination of the low complexity of the model with the initialization process, which acts as a regularization mechanism during the learning process. This leads to improved generalization performance and better robustness against overfitting.

On the other hand, our experimental results indicate that performance improvement is not always significant for larger models. Nevertheless, we observed that our pre-trained approach still provides benefits by improving the stability of the model results. We attribute this stability to our approach's ability to select the most suitable data for pretraining. These findings suggest that our approach can help improve the training of larger models, even when data is abundant, by providing better initialization.

| Model | Init. method | Labeled data 5k | 10k | 15k |
|---|---|---|---|---|
| M1 (66k param.) | RI | $89.1 \pm 7.4$ | $97.1 \pm 0.5$ | $97.9 \pm 0.2$ |
| | RS+CL [5] | $96.2 \pm 0.6$ | $97.6 \pm 0.2$ | $97.9 \pm 0.1$ |
| | SCS+CL (Ours) | $\mathbf{96.6 \pm 0.4}$ | $\mathbf{98.0 \pm 0.2}$ | $\mathbf{98.1 \pm 0.1}$ |
| | $\Delta_1$ | **7.5** | 0.9 | 0.2 |
| | $\Delta_2$ | 0.4 | **0.4** | 0.2 |
| M2 (250k param.) | RI | $88.6 \pm 8.3$ | $\mathbf{97.7 \pm 0.3}$ | $\mathbf{98.1 \pm 0.2}$ |
| | RS+CL [5] | $96.0 \pm 0.7$ | $97.4 \pm 0.3$ | $97.8 \pm 0.1$ |
| | SCS+CL (Ours) | $\mathbf{96.2 \pm 0.5}$ | $97.6 \pm 0.2$ | $97.9 \pm 0.1$ |
| | $\Delta_1$ | **7.6** | 0.2 | $-0.2$ |
| | $\Delta_2$ | **0.2** | $-0.1$ | 0.1 |
| M3 (1M param.) | RI | $87.0 \pm 6.6$ | $97.0 \pm 1.4$ | $\mathbf{98.2 \pm 0.1}$ |
| | RS+CL [5] | $\mathbf{95.4 \pm 0.8}$ | $96.8 \pm 0.4$ | $97.3 \pm 0.3$ |
| | SCS+CL (Ours) | $95.1 \pm 0.8$ | $\mathbf{97.0 \pm 0.4}$ | $96.4 \pm 0.2$ |
| | $\Delta_1$ | **8.1** | $\sim 0$ | $-0.8$ |
| | $\Delta_2$ | $-0.3$ | **0.2** | 0.1 |

Table 2. Average AUC on the Icentia11k [16] test set. $\Delta_1$ and $\Delta_2$ is the difference between our method compared with RI and RS+CL, respectively.

One potential limitation of this work is that all the data used in our experiments belonged to the same dataset collected from a specific type of portable device. This may introduce biases that limit the generalizability of our findings to other devices and datasets. To address this limitation, future work should test our approach on a more extensive and diverse set of datasets collected from different sources and types of devices.

## 5. Conclusions

This work focuses on enhancing contrastive learning to improve the detection of cardiac arrhythmias using unlabeled data. The proposed approach uses a similarity-constrained data selection algorithm to determine the signals used in pre-training a CNN. Our experimental results suggest that our proposed approach outperforms the CL baseline. Our results suggest that the proposed approach is a promising solution for small models in a few data scenarios. The effectiveness of the proposed method decreases as the deep learning model size and amount of labeled data used in training increase. However, the method improves the training robustness in almost all cases.

## References


[1] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020. 1

[2] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. 1

[3] Nathaniel Diamant, Erik Reinertsen, Steven Song, Aaron Aguirre, Collin Stultz, and Puneet Batra. Patient Contrastive Learning: a Performant, Expressive, and Practical Approach to ECG Modeling. *arXiv preprint arXiv:2104.04569*, pages 1–17, 2021. 3

[4] Carlos Fajardo. Compressed deep learning models to detect cardiac arrhythmias. https://github.com/cafajar-uis/. 3

[5] Karen Fonseca, Sergio Osorio, Jeyson Castillo, and Carlos Fajardo. Contrastive learning for atrial fibrillation detection in challenging scenarios. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 1218–1222, 2022. 1, 2, 3, 4

[6] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. Variational deep embedding: A generative approach to clustering. *CoRR*, abs/1611.05148, 2016. 3

[7] Alamira Jouman Hajjar. In-depth guide to self-supervised learning: Benefits & uses, 2022. 1

[8] Prannay Khosla et al. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020. 1

[9] Bhekumuzi Mathunjwa, Yin-Tsong Lin, Chien-Hung Lin, Maysam Abbod, and Jiann-Shing Shieh. Ecg arrhythmia classification by using a recurrence plot and convolutional neural network. *Biomedical Signal Processing and Control*, 64:102262, 02 2021. 1

[10] João Pereira and Margarida Silveira. Unsupervised representation learning and anomaly detection in ecg sequences. *International Journal of Data Mining and Bioinformatics*, 22:389–407, 08 2019. 3

[11] James A Reiffel. Atrial fibrillation and stroke: epidemiology. *The American journal of medicine*, 127(4):e15–e16, 2014. 1

[12] Karen Sanchez, Carlos Hinojosa, Henry Arguello, Denis Kouamé, Olivier Meyrignac, and Adrian Basarab. Cx-dagan: Domain adaptation for pneumonia diagnosis on a small chest x-ray dataset. *IEEE Transactions on Medical Imaging*, 41(11):3278–3288, 2022. 1, 2

[13] Jürg Schläpfer and Hein J. Wellens. Computer-interpreted electrocardiograms: Benefits and limitations. *Journal of the American College of Cardiology*, 70(9):1183–1192, 2017. 1

[14] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019. 3

[15] Ajay Shrestha and Ausif Mahmood. Review of deep learning algorithms and architectures. *IEEE access*, 7:53040–53065, 2019. 1

[16] Shawn Tan, Guillaume Androz, Ahmad Chamseddine, Pierre Fecteau, Aaron Courville, Yoshua Bengio, and Joseph Paul Cohen. Icentia11k: An unsupervised representation learning dataset for arrhythmia subtype discovery, 2019. 3, 4

[17] Mintu P. Turakhia et al. Rationale and design of a large-scale, app-based study to identify cardiac arrhythmias using a smartwatch: The apple heart study. *American Heart Journal*, 207:66–75, 2019. 1

[18] Tao Wang, Changhua Lu, Yining Sun, Mei Yang, Chun Liu, and Chunsheng Ou. Automatic ecg classification using continuous wavelet transform and convolutional neural network. *Entropy*, 23(1), 2021. 1

[19] Ahmet Çınar and Seda Arslan Tuncer. Classification of normal sinus rhythm, abnormal arrhythmia and congestive heart failure ecg signals using lstm and hybrid cnn-svm deep neural networks. *Computer Methods in Biomechanics and Biomedical Engineering*, 24(2):203–214, 2021. PMID: 32955928. 1

[20] Özal Yıldırım, Paweł Pławiak, Ru-San Tan, and U. Rajendra Acharya. Arrhythmia detection using deep convolutional neural network with long duration ecg signals. *Computers in Biology and Medicine*, 102:411–420, 2018. 1