DiffECG: Diffusion Model-Powered Label-Efficient and Personalized Arrhythmia Diagnosis

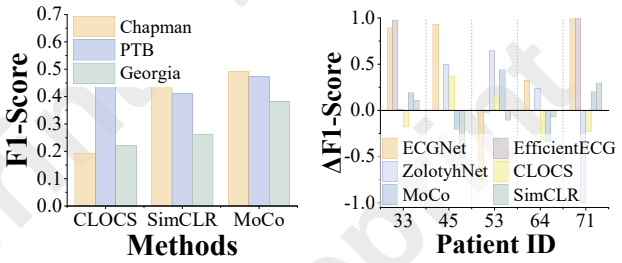
Tianren Zhou , **Zhenge Jia*** , **Dongxiao Yu** and **Zhaoyan Shen**
School of Computer Science and Technology, Shandong University
trzhou@mail.sdu.edu.cn, {zhengejia, dxyu, shenzhaoyan}@sdu.edu.cn

Abstract

Arrhythmia diagnosis using electrocardiogram (ECG) is critical for preventing cardiovascular risks. However, existing deep learning-based methods struggle with label scarcity and contrastive learning-based methods suffer from false-negative samples, which lead to poor model generalization. Besides, due to inter-subject variability, pre-trained models cannot achieve even performance across individuals. Conducting model fine-tuning for each individual is computationally expensive and does not guarantee improvement. We propose DiffECG, a diffusion-based self-supervised learning framework for label-efficient and personalized arrhythmia detection. Our method utilizes a diffusion model to extract robust ECG representations, coupled with a novel feature extractor and a multi-modal feature fusion strategy to obtain a well-generalized model. Moreover, we propose an efficient model personalization mechanism based on zeroth-order optimization. It personalizes the model by tuning the noise-adding step t in the diffusion process, significantly reducing computational costs compared to model fine-tuning. Experimental results show that our proposed method outperforms the SOTA method by 37.9% and 23.9% in terms of generalization and personalization performance, respectively. The source code is available at: https://github.com/Auguuust/DiffEC.

1 Introduction

Arrhythmia, characterized by irregular heart rhythms, can lead to cardiovascular diseases such as stroke, heart failure, and sudden cardiac arrest [Association, 2022]. Accurate arrhythmia detection is challenging due to its sporadic occurrence and asymptomatic nature. In recent years, deep learning has been widely adopted in arrhythmia detection. Existing approaches leverage either the temporal [Krasteva *et al.*, 2020; Xu and Liu, 2020; Jia *et al.*, 2021] or spectral [Wasimuddin *et al.*, 2021; Jun *et al.*, 2018; Huang *et al.*, 2019] forms of ECG time-series data to train the deep neural network and achieve promising detection performance.



(a) Generalization evaluation. (b) Personalization evaluation.

Figure 1: Evaluation results of existing methods. (a) F1 scores on public datasets of models pre-trained with three different self-supervised learning methods. (b) The changes in F1 scores on individuals of different pre-trained models after being fine-tuned on subject-specific data.

These supervised learning methods generally require extensive data with precise annotations. However, annotating ECG data is labor-intensive and costly. To reduce the dependency on annotations, contrastive learning methods such as CLOCS [Kiyasseh *et al.*, 2021], SimCLR [Mehari and Strodthoff, 2022] and MoCo [Nakamoto *et al.*, 2022] are applied for arrhythmia detection. These methods perform self-supervised learning by pulling similar sample pairs closer in the feature space while pushing dissimilar pairs apart.

Nevertheless, there are still two main challenges: 1) Existing contrastive learning methods cannot guarantee the satisfactory generalization (ability to perform well on unseen data) of the deep model. The performance of these methods could be impacted by false-negative samples. In contrastive learning, false-negative samples refer to instances of the same category that are incorrectly treated as dissimilar pairs, which are particularly common in arrhythmia detection due to the similarity of ECG patterns across different rhythm types. As shown in Fig. 1(a), the models pre-trained on the large public ECG dataset all achieve F1 scores below 50% when applied to different downstream tasks; 2) The pre-trained models perform unevenly across different individuals due to the inter-subject variability in ECG morphological characteristics. A general solution is fine-tuning the pre-trained model using a small portion of an individual's labeled data. However, as

*Corresponding author.

shown in Fig. 1(b), existing methods could not achieve improvements across all individuals after fine-tuning.

To address the above challenges, we propose DiffECG, a diffusion-based self-supervised learning framework for label-efficient and personalized arrhythmia detection. While the diffusion model focuses on multi-scale noise prediction rather than classification, we decompose its structure and design a new feature extraction network for downstream arrhythmia detection tasks. Additionally, we design a multi-modal fusion strategy that incorporates temporal, spectral, and domain knowledge to further enhance the model's generalization. To achieve efficient model personalization, we analyze the impact of noise-adding steps t of the diffusion model on individual performance and propose a zeroth-order optimization-based personalization mechanism. It enhances model personalization by tuning t, greatly reducing computational overhead compared to full model fine-tuning. We evaluate our proposed method on five public ECG datasets and get satisfactory results. In terms of model generalization, the proposed method outperforms existing self-supervised learning approaches by up to 37.9% in F1 score. For model personalization, our method surpasses the SOTA method by 23.9% and 13.9% in F1 score and Accuracy, respectively. The main contributions of the work are summarized as follows:

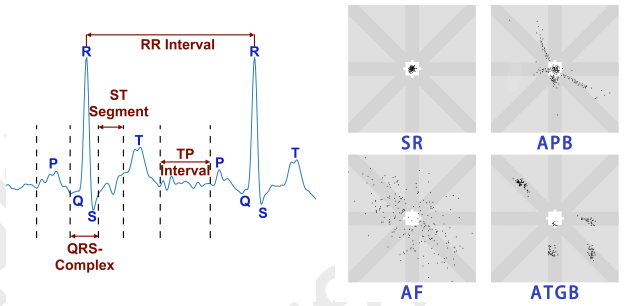
- We propose DiffECG, a diffusion model-based self-supervised learning framework for arrhythmia detection, achieving high label efficiency and detection performance.
- We design a new feature extractor structure and a multi-modal feature fusion strategy to enhance the model's generalization.
- We investigate the correlation between the noise-adding step t and individual performance, proposing an efficient personalization mechanism that tunes t without extra computation costs.
- Experimental results show that DiffECG surpasses the SOTA method by 37.9% and 23.9% in generalization and personalization performance, respectively.

2 Background

2.1 Arrhythmia and Diagnosis

Arrhythmia is a condition characterized by irregularities in the heart's rhythm, which can show as a heartbeat that is too fast (tachycardia), too slow (bradycardia), or erratic [Kanna and Eliyas, 2023]. Clinicians primarily analyze electrocardiograms (ECGs) or corresponding Lorenz plots to conduct the diagnosis. Lorenz plots could visualize the nature of the variability, which is particularly useful in the study of heart rate dynamics and variability. As shown in Fig. 2(a), the ECG segment contains two heartbeats, each of which consists of a P-wave, a QRS-Complex (consisting of Q-wave, R-peak, and S-wave), and a T-wave. It could also reveal information about the association between heartbeats, such as the RR interval, TP interval, and ST segment.

Cardiologists also use the Lorenz plots, a two-dimensional scatter plot that shows the relationship between successive R-R intervals in an ECG, for arrhythmia diagnosis. Each point's



(a) ECG. (b) Lorenz plots.

Figure 2: Different representations of cardiac signals. (a) ECG shows two consecutive cardiac cycles. (b) Lorenz plots correspond to four cardiac rhythm patterns.

coordinate in Lorenz plots is calculated using Equation (1),

$$\begin{cases} x_i = dRR_i = RR_i - RR_{i-1} \\ y_i = dRR_{i-1} = RR_{i-1} - RR_{i-2}, \end{cases} \tag{1}$$

where RR_i is defined as a series of RR intervals by locating all R-peaks in the ECG segment. Fig. 2(b) shows the Lorenz plots of four different cardiac rhythms with significant differences in scatter distribution, which could aid cardiologists in diagnosing arrhythmias.

There are automatic arrhythmia diagnostic algorithms based on the above characteristics, such as diagnosing atrial fibrillation based on the pattern of change of RR intervals in the ECG segments [Lian *et al.*, 2011], or furthermore, combining the analysis of P waves [Hindricks *et al.*, 2010; Pürerfellner *et al.*, 2014]. However, these methods heavily depend on expert and clinical experience, which makes it difficult to obtain the best parameter setting by manual tuning.

2.2 Deep Learning-Based Arrhythmia Detection

Deep learning has been widely adopted to conduct arrhythmia detection due to its ability to train models using only labeled data [Haleem *et al.*, 2019; Holmes *et al.*, 2004], thereby minimizing the dependency on domain-specific expertise. Convolutional neural networks [Rajpurkar *et al.*, 2017; Tan and Le, 2019] and variational autoencoder [Kuznetsov *et al.*, 2020] are used for arrhythmia detection based on features extracted over the temporal form of ECG data.

There are also studies that leverage spectral features from time-frequency analysis for arrhythmia detection [Huang *et al.*, 2019]. Time-frequency analysis refers to examining a non-stationary signal in the frequency domain, enabling a clear description of how the signal's frequency components evolve. The Short-Time Fourier Transform (STFT) is a widely utilized time-frequency analysis method, the process of which can be described by Equation (2),

$$STFT(x) = X(i,g) = \sum_{m=-\infty}^{\infty} x(m)g(i-m)e^{-j\omega m}, \tag{2}$$

where $X()$ is a two-dimensional function defined on the time and frequency, $x(m)$ denotes the ECG segment. $g()$ denotes

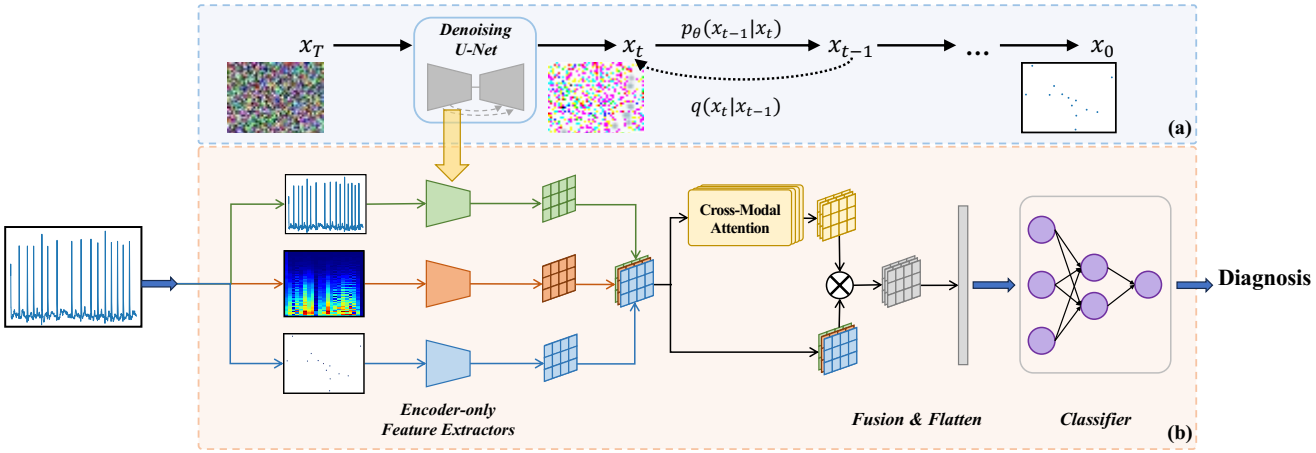


Figure 3: The architecture of the diffusion-based arrhythmia detection model. (a) The forward and reverse processes of diffusion models. The original diffusion model training process is utilized during the pre-training phase. (b) The inference process of the proposed self-supervised learning framework. During fine-tuning for the downstream arrhythmia detection task, only the classifier parameters are updated, while the remaining parameters are frozen.

the window function, which is used to reduce spectral energy leakage. The Hanning window is commonly selected as the window function, as shown in Equation (3), where M is the number of sampling points and n is the window length.

$$g(n) = \begin{cases} 0.5[1 - cos(\frac{2\pi n}{M-1})], & 0 \leq n \leq M-1 \\ 0, & otherwise. \end{cases} \tag{3}$$

In addition, some works adopt self-supervised learning frameworks to reduce the dependency on annotations. Classic contrastive learning methods, such as SimCLR and MoCo, have been applied to ECG signal classification [Mehari and Strodthoff, 2022; Nakamoto *et al.*, 2022]. Other studies design methods based on the unique characteristics of ECG signals. CLOCS [Kiyasseh *et al.*, 2021] introduces a contrastive learning method across space, time, and patients. CLECG [Chen *et al.*, 2021] creates different views of an ECG segment using wavelet transforms and segmented random cropping. Positive samples are obtained from these views, while other segments are used as negative samples for contrastive learning.

3 Methodology

3.1 Diffusion-Based Arrhythmia Detection Model

We propose a diffusion-based self-supervised learning framework for accurate arrhythmia detection. As shown in Fig. 3, we first perform self-supervised pre-training on the diffusion model. The denoising network of the diffusion model then serves as a feature extractor for downstream arrhythmia detection tasks, which processes the input data to generate features for a classifier. Our proposed diffusion-based arrhythmia detection model achieves high detection performance with only a few fine-tuning steps while keeping the feature extractor frozen.

The main process of the diffusion model is illustrated in part (a) of Fig. 3. It primarily involves two stages: diffusion and reverse. In the diffusion process, Gaussian noise is gradually added to the input sample x_0 by time step t and finally gets $x_t \sim \mathcal{N}(0, 1)$, which could be defined by Equation (4),

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{4}$$

where $\beta_1, ..., \beta_t$ are some fixed variance schedules. It is worth noting that every noisy sample x_t under step t can be directly obtained from the original sample x_0 in the diffusion process:

$$x_t = \sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, 1), \tag{5}$$

where $\alpha_t := 1 - \beta_t, \overline{\alpha} := \prod_{s=1}^{t} \alpha_s$.

The reverse process of diffusion models transforms noise $x_T \sim \mathcal{N}(0, I)$ to the x_0 through gradually denoising x_T to less noisy samples x_t by step t. The reverse process could be defined as follows :

$$p_\theta(x_{t-1}|x_t) := \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{6}$$

The noise prediction network $\epsilon(x_t, t)$ in the reverse process predicts the noise added at step t from x_{t-1} to x_t, which typically employs the UNet structure.

In our framework, the input data undergoes a t-step noise addition, where t is a pre-specified hyperparameter. All the noise-added input samples are processed through the UNet for feature extraction.

3.2 Network Structure Search and Multi-Modal Feature Fusion

The structure of UNet is illustrated in part (a) of Fig. 4, it comprises a contracting path (encoder), an expanding path (decoder), bottleneck layers, and skip connection structures. This structure efficiently captures features at different granularities. However, it may lead to performance degradation in classification tasks due to the introduction of high-frequency noise and mismatches in semantic abstraction.

To fully utilize the robust feature extraction capabilities of the diffusion model without sacrificing performance, we apply neural architecture search (NAS) to identify the optimal

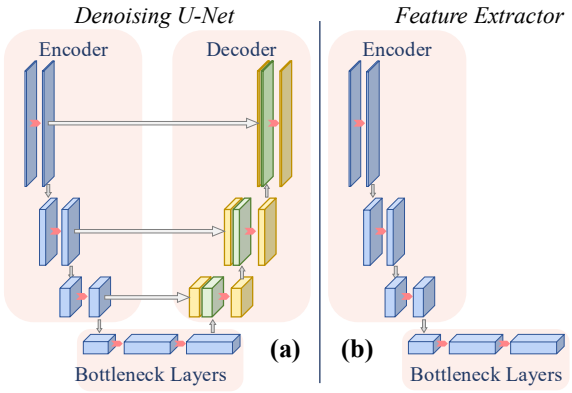


Figure 4: Network structures. (a) Architecture of the denoising U-Net used in the diffusion model, comprising an encoder for feature extraction, a bottleneck for intermediate representation, and a decoder for image reconstruction. (b) The optimized feature extractor discovered through NAS, which preserves the encoder and bottleneck components of the original U-Net while omitting the decoder, focusing solely on efficient feature representation.

structural configurations. The search space includes multi-scale feature extraction modules, hierarchical abstraction layers, and information propagation pathways derived from the original UNet architecture in the diffusion model. Through several optimization cycles, the search algorithm removes redundant components while preserving essential information pathways. This process converges on the optimized structure shown in Fig. 4 (b). The final structure includes the contracting path of the UNet and two bottleneck layers with attention modules, which capture the most abstract and high-level features from the input data while consolidating rich hierarchical features. Since the optimal feature extractor structure identified through NAS can be integrated into a full UNet, we use a UNet with this searched structure as a subnetwork for denoising during the diffusion model training phase. Afterward, we extract the feature extractor's partial parameters and structure for the downstream arrhythmia detection task, eliminating the need for additional model training.

To further enhance the detection performance of the model, we design an attention-based multi-modal feature fusion strategy. As shown in part (b) of Fig. 3, each raw ECG signal segment is transformed into two formats: the spectrogram and the Lorenz plots. The model uses these two formats, along with the original ECG signal segment, as inputs for the proposed diffusion-based arrhythmia detection model. The feature maps from each modality are stacked along the channel dimensions and then weighted and fused by a cross-modal attention module. This approach enables the model to adaptively adjust the weight of each modality for different subjects during the fine-tuning stage.

3.3 t-Based Efficient Personalization Mechanism

To ensure robust arrhythmia detection across subjects, we analyze the factors influencing the model's detection efficacy. The analysis reveals that the noising-adding step t, which introduces noise to the raw input data, significantly affects detection performance. This effect is particularly pronounced when the sample distribution across different categories is highly imbalanced.

Guided by prior knowledge, performing a grid search for the step t can undoubtedly find the optimal set of t values. However, this approach incurs significant time and computational costs. Therefore, we propose a zero-order optimization-based personalization mechanism, along with an original optimization objective, to determine the optimal set of t values.

Bayesian optimization estimates the probability distribution of the objective function by constructing a surrogate model. It then determines the location of the next evaluation point using an acquisition function. This point is evaluated on the true objective function, and the surrogate model is updated accordingly. The process iteratively continues, gradually converging to the optimal solution.

We employ Gaussian processes as the surrogate function for Bayesian optimization, which is shown as Equation (7):

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')), \tag{7}$$

where $m(x)$ denotes the mean function and $k(x, x')$ denotes the covariance function. In this paper, we choose the Radial Basis Function (RBF) as the covariance function, which could be described as Equation (8):

$$k(x, x') = exp(-\frac{\|x - x'\|}{2l^2}), \tag{8}$$

where $\|x-x'\|$ denotes the Euclidean distance between x and x' and l denotes the length scale for the smoothness control of the function.

For the acquisition function, we choose the Expected Improvement (EI) method, which is described as Equation (9):

$$EI(x) = \mathbb{E}[max(f(x) - f(x^+), 0)], \tag{9}$$

where x^+ denotes the optimal solution position selected by the acquisition function.

Accuracy or F1-score is commonly used as the optimization objective in Bayesian optimization. However, these metrics are not meaningful until the model approaches convergence, which is time-consuming and computationally expensive. Therefore, we propose a new metric, $\boldsymbol{\gamma}$, to assess the model's convergence ability based solely on the initial training epochs. This metric is defined as follows:

$$\gamma = \frac{\Delta \mathcal{L}}{\Delta \omega} = \frac{\mathcal{L}(\omega_r) - \mathcal{L}(\omega_{r+1})}{\|\omega_{r+1} - \omega_r\|}, \tag{10}$$

where the ω denotes the weight parameters of the neural network model (or model itself) and $\mathcal{L}(\omega)$ denotes the loss function. γ represents the sensitivity of the model's loss value to parameter updates, which intuitively could reflect the model's convergence ability.

Starting from the objective function of neural networks, the feasibility of using γ as the optimization objective is analyzed as follows. Equation (11) shows the optimization object of the neural network,

$$\min_{\omega} \mathcal{L}(\omega), \tag{11}$$

where ω, denotes the neural network model and $\mathcal{L}(\omega)$ denotes the loss function. Given that neural networks have a

large number of parameters and complex nonlinear transformations, finding their analytical solutions is very difficult. Therefore, it is impossible to directly obtain a set of parameters ω that minimizes the Loss value. Instead, iterative methods could be used. Equation (12) shows the first-order Taylor series approximation to calculate $\mathcal{L}(\omega)$ at ω_{r+1}:

$$\mathcal{L}(\omega_{r+1}) \approx \mathcal{L}(\omega_r) + \nabla\mathcal{L}(\omega_r)^T(\omega_{r+1} - \omega_r), \tag{12}$$

where ω_r represents the model's parameters after rth update. According to the optimization object shown in Equation (11), the loss value should decrease with each iteration, meaning the following inequality holds constant:

$$\Delta\mathcal{L} = \mathcal{L}(\omega_{r+1}) - \mathcal{L}(\omega_r) \approx \nabla\mathcal{L}(\omega_r)^T(\omega_{r+1} - \omega_r) < 0. \tag{13}$$

Obviously, for Inequality (13) to always hold, $\nabla\mathcal{L}(\omega_r)$ and $(\omega_{r+1} - \omega_r)$ must have opposite signs, thus define $(\omega_{r+1} - \omega_r)$ as follows:

$$\begin{aligned} \Delta\omega = (\omega_{r+1} - \omega_r) &= -\eta\nabla\mathcal{L}(\omega_r) \\ \Rightarrow \omega_{r+1} &= \omega_r - \eta\nabla\mathcal{L}(\omega_r), \end{aligned} \tag{14}$$

where the second line of Equation 14 is the definition of the parameter update process in gradient descent, validating the previous derivation. Further, with the combination of Equations (14), (10), and Inequality (13), the γ would be transformed into the following form:

$$\begin{aligned} \gamma &= \frac{\mathcal{L}(\omega_r) - \mathcal{L}(\omega_{r+1})}{\|\omega_{r+1} - \omega_r\|} \\ &\approx \frac{-\nabla\mathcal{L}(\omega_r)^T(-\eta\nabla\mathcal{L}(\omega_r))}{\|\omega_{r+1} - \omega_r\|} \\ &\approx \frac{\eta\|\nabla\mathcal{L}(\omega_r)\|^2}{\eta\|\nabla\mathcal{L}(\omega_r)\|} \\ &\approx \|\nabla\mathcal{L}(\omega_r)\|, \end{aligned} \tag{15}$$

which shows that the proposed γ approximates the paradigm of the gradient, and could reflect the change ratio of the loss function $\mathcal{L}(\omega)$ in the parameter space. Using γ, we aim to find a set of t values that maximize γ, thereby achieving the largest gradient within 0th to rth training epochs. At this point, the model could approximate the optimal solution in the early phases of training. Compared to using the changes in accuracy, F1-score, loss value, or mixed definitions of these metrics, γ neither requires extensive training epochs for the model to converge nor produce excessively large or small values, making it more reliable for evaluating the model's performance.

4 Experiments

4.1 Experimental Setup

Dataset. We evaluate all the methods based on five public ECG datasets. **CPSC** [Goldberger *et al.*, 2000; Liu *et al.*, 2018] consists of ECG records with 9 different types of arrhythmias ranging from 6 to 60 seconds.**Chapman** [Zheng *et al.*, 2020] consists of ECG records with 11 different types of arrhythmias with a 10-second duration for each record. We group these labels into 4 major types suggested by the dataset. **PTB** [Bousseljot *et al.*, 1995; PhysioBank, 2000] dataset contains 549 records ranging from 1 to 5 seconds. We conduct a binary classification for atrial fibrillation (AF). **Georgia** [Alday *et al.*, 2020] consists of ECG records with 56 different types of arrhythmias, with a 10-second duration for each record. We eliminate the types with the number of data samples under 500 and group the rest into 4 major types suggested by the dataset. The above 4 datasets have rich types of arrhythmia that are suitable for generalization evaluation. **LTAF** [Petrutiu *et al.*, 2007; Goldberger *et al.*, 2000] includes 84 ECG records of subjects with paroxysmal or sustained AF. Each record has a duration of 24 or 48 hours, which is suitable for long-term detection.

Implementation Details. We conduct experiments to evaluate the **generalization** and **personalization** performance of our method. We report the generalization performances in terms of the detection performances achieved by models on the unseen subjects' data. Specifically, we first split each dataset (i.e., CPSC, Chapman, PTB, and Georgia) into fine-tuning and testing sets subject-wisely (with a splitting ratio of 2:8) to ensure the subject's data is not mixed between the fine-tuning and testing sets. Next, we pre-train the model using the entire dataset from one source (e.g., CPSC or Chapman) and fine-tune this pre-trained model with the fine-tuning sets of the remaining datasets. During the fine-tuning phase, feature extractors of all self-supervised methods are frozen. Finally, we demonstrate the generalizability of different approaches by evaluating the detection performance of the model on the testing set of each dataset.

We report personalization performance based on the individual detection accuracy achieved by the model when personalized with subject-specific data. Specifically, we first split each subject's data in the LTAF dataset with a 1:9 ratio, using 10% for fine-tuning and 90% for testing. Next, we pre-train all methods on the CPSC dataset and then personalize the detection model using the 10% subject-specific data. The personalized model is then tested on the remaining 90% of each subject's data. The personalization performance is reported in terms of macro accuracy and F1 score.

We use PyTorch for all methods to build networks, train models, and report detection performance. The training is conducted on a server equipped with four NVIDIA RTX 4090 GPUs, an Intel Xeon Platinum 8480+ CPU, and 1 TB of memory.

Baselines. We evaluate the proposed diffusion-based detection method, DiffECG, against 11 different baseline methods. For generalization evaluation, we implement 3 supervised learning methods: ECGNet [Jun *et al.*, 2018], EfficientECG [Akkuzu *et al.*, 2023], ZolotyhNet [Kuznetsov *et al.*, 2020]; and 3 self-supervised learning methods: CLOCS [Kiyasseh *et al.*, 2021], SimCLR [Mehari and Strodthoff, 2022], MoCo [Nakamoto *et al.*, 2022]. To ensure consistency, we integrate the multi-modal feature fusion strategy into other self-supervised learning baseline methods except CLOSC, since it is specifically designed for time-domain data and can only utilize raw ECG as inputs. All self-supervised learning methods adopt the same feature ex-

Methods	CPSC			Chapman		
	Chapman	PTB	Georgia	CPSC	PTB	Georgia
ECGNet [Jun *et al.*, 2018]	.888	.546	.710	.280	.530	.618
EfficientECG [Akkuzu *et al.*, 2023]	.283	.468	.161	.064	**.648**	.182
ZolotyhNet [Kuznetsov *et al.*, 2020]	.656	.579	.484	.304	.587	.517
CLOCS [Kiyasseh *et al.*, 2021]	.541	.289	.369	.179	.546	.396
SimCLR [Mehari and Strodthoff, 2022]	.761	.492	.512	.158	.475	.503
MoCo [Nakamoto *et al.*, 2022]	.556	.468	.392	.264	.468	.472
DiffECG (uni-temporal)	.906	.522	.627	.426	.573	.594
DiffECG (uni-spectral)	.890	.544	.726	.441	.521	.724
DiffECG (uni-domain knowledge)	.788	.550	.595	.273	.558	.606
DiffECG	**.913**	**.598**	**.735**	**.537**	.599	**.727**

Table 1: The macro F1 score of generalization evaluation. All methods are pre-trained using the CPSC and Chapman datasets, respectively, fine-tuned on the other datasets with 20 epochs. The noise-adding step t of our method is fixed to 20.

tractor structure as our proposed method. We also evaluate our proposed method using three uni-modal features respectively as the ablation experiments. For personalization evaluation, we implement all the abovementioned methods and 2 additional medical knowledge-based algorithms: VCL-Evidence [Pürerfellner *et al.*, 2014], Pwave-Evidence [Sarkar *et al.*, 2017].

4.2 Experimental Results

Genelization Performance. Table 1 shows the generalization performance of all methods. When methods are pre-trained on CPSC and fine-tuned on the Chapman dataset, where the distribution of arrhythmia types is relatively balanced, almost all methods demonstrate high detection performance. Our proposed DiffECG under the multi-modal feature fusion strategy achieves the best performance with the F1 score of 91.3%. It outperforms other self-supervised learning methods, CLOCS, SimCLR, and MoCo, by 37.2%, 15.2%, and 35.7%, respectively. In contrast, when methods are pre-trained on Chapman and fine-tuned on CPSC, all baseline methods show poor performance with F1 scores falling below 50%. This is because the CPSC dataset contains more heart rhythm types than Chapman, which imposes higher generalization demands on the model. In this scenario, our proposed method achieves an F1 score of 53.7% using the multi-modal feature fusion strategy, outperforming other self-supervised learning methods by up to 37.9%.

When fine-tuning on downstream datasets with imbalanced arrhythmia types distributions, some supervised methods exhibited a significant decline in performance. The EfficientECG gets very low F1 scores on the Georgia dataset, specifically 16.1% and 18.2% when pre-trained on CPSC and Chapman. This suggests that in cases of highly unbalanced data, labeling an uneven number of samples can shift from being beneficial for network convergence to a contributing factor in network overfitting towards a specific class of samples. On the other hand, some self-supervised learning-based methods also show significant performance degradation. This is primarily caused by the presence of false-negative samples during the pre-training phase, which limits its feature extraction capability. In contrast, our DiffECG achieves the highest performance among all self-supervised learning methods, surpassing the best-performing self-supervised learning method, SimCLR, by 10.6%, 22.3%, 12.1%, and 22.4% on the PTB and Georgia datasets when pre-trained on CPSC and Chapman, respectively. It is worth noting that our proposed DiffECG outperforms all self-supervised learning baseline methods, even when relying solely on uni-modal features.

Method	F1	ACC
VCL-Evidence [Pürerfellner *et al.*, 2014]	.649	.818
Pwave-Evidence [Sarkar *et al.*, 2017]	.384	.485
ECGNet [Jun *et al.*, 2018]	.769	.964
EfficientECG [Akkuzu *et al.*, 2023]	.476	.915
ZolotyhNet [Kuznetsov *et al.*, 2020]	.500	.649
CLOCS [Kiyasseh *et al.*, 2021]	.566	.915
SimCLR [Mehari and Strodthoff, 2022]	.607	.798
MoCo [Nakamoto *et al.*, 2022]	.579	.911
DiffECG (uni-temporal)	.806	.992
DiffECG (uni-spectral)	.827	.987
DiffECG (domain knowledge)	.642	.948
DiffECG	**.846**	**.991**

Table 2: Personalization performances on dataset LTAF. For our method using the uni-modal feature, the noising-adding step t is fixed at 20, whereas our method using the multi-modal feature fusion strategy employs the proposed method to automatically determine ts' value.

Personalization Performance. As shown in Table 2, the proposed DiffECG achieves the highest F1 score and demonstrates excellent personalization performance. It surpasses

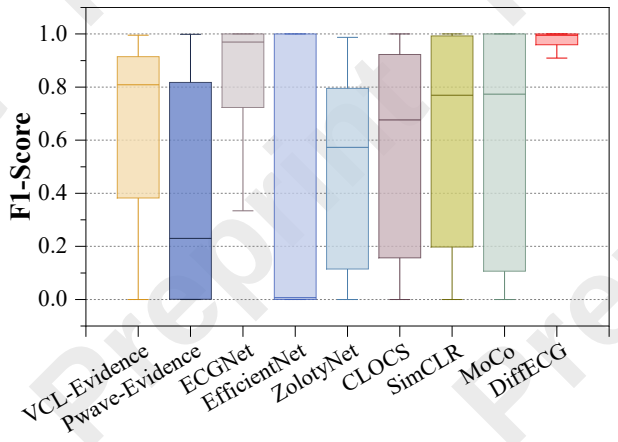


Figure 5: Box plots of performance on individual testing subject's fine-tuned model of all methods.

	t=20	t=30	t=40	t=50	t=60	t=70	**γ-opt**
F1	.731	.730	.719	.718	.714	.691	**.846**
ACC	.950	.962	.959	.955	.953	.947	**.991**

Table 3: Personalization performances on dataset LTAF through different t settings.

the medical knowledge-based method VCL-evidence by 19.7% in terms of F1 score and 17.3% in terms of accuracy. Notably, both medical knowledge-based methods perform poorly. This is because their parameters require adjustment when employed on different subjects, which heavily rely on domain knowledge and clinical experience. On the other hand, our method also outperforms the other self-supervised learning methods CLOCS, SimCLR, and MoCo by 28.0%, 23.9%, and 26.7% in terms of the F1 score, respectively.

Fig. 5 illustrates the distribution of the F1 scores on all individual subjects of each method. The F1 scores for testing subjects fluctuate between 0% to 100% with both medical knowledge-based methods. While the self-supervised learning baseline methods exhibit higher F1 scores compared to most supervised methods, there remains a large performance gap across subjects. For example, SimCLR achieves the highest lower quartile and median values among them, which are only about 20% and 76%, respectively. In contrast, the proposed DiffECG achieves the best performance, with F1 scores ranging from approximately 91% to 100%, and exhibits the shortest interquartile range, indicating evenly high detection performance across all subjects.

Zeroth-Order Optimization Performance. To validate the original optimization objective-based zero-order optimization strategy, we evaluate the AF detection performance of the proposed DiffECG on the LTAF dataset under different settings of t values. The first six columns of Table 3 show the average F1-scores when the t values of three modalities of all testing subjects are set to the same fixed values of 20, 30, 40, 50, 60, and 70, respectively. The last column shows the

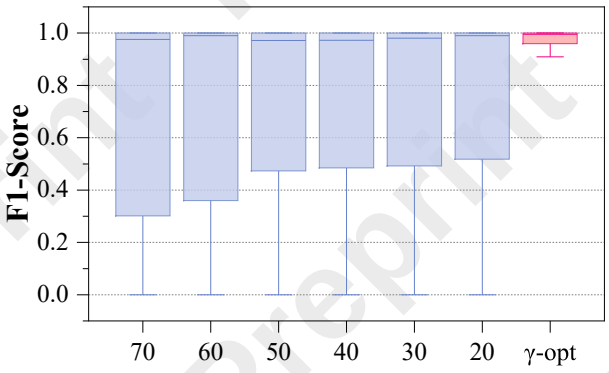


Figure 6: Box plots of performance on individual subjects of DiffECG under different settings of t.

	Δ ACC	Δ F1	Δ Loss	Δ Mix	γ
F1	.809	.807	.810	.809	**.846**
ACC	.965	.962	.969	.949	**.991**

Table 4: Personalization performances on dataset LTAF under different optimization objective settings.

average F1-score achieved by using γ as the optimization objective to personalize the t's value for each modality of each testing subject. It is evident that as t (i.e., the noise level) increases, the model's performance shows a downward trend. Compared to the best-performing fixed value of $t = 20$, our zero-order optimization strategy improves the F1-score by 11.5%. On the other hand, Fig. 6 illustrates the distribution of F1 scores for all testing subjects under different settings of t. The interquartile range shows significant variation across the different settings. Our $gamma$-based zero-order optimization setup exhibits a notably shorter interquartile range, with even the minimum value being significantly higher than the lower quartile value when t is fixed at 20.

In addition, as shown in Table 4, we also evaluate the effect of using different optimization objectives to search for t values. The optimization objectives include the differences in accuracy, F1 score, loss value, and Mix between two epochs, as well as the proposed γ. The Mix is defined as a weighted mixture of the cross-entropy loss function and the F1 score. It is evident that using our proposed γ as the optimization objective achieves the best search results, with the F1 scores higher than the second highest $\Delta Loss$ by 3.6%.

5 Conclusion

In this paper, we propose a diffusion-based self-supervised learning framework with an original optimization object-based zeroth-order optimization strategy to achieve both high label efficiency and personalization performance in arrhythmia detection. The incorporation of the novel feature extractor structure and the multi-modal feature fusion strategy further enhances the detection performance of our proposed method. The proposed diffusion-based model demonstrates strong generalization and personalization performances through evaluation.

Acknowledgments

The work described in this paper is partially supported by Shandong Provincial Natural Science Foundation under Grants ZR2022LZH010, the National Natural Science Foundation of China under Grant U24B20149, and the Taishan Scholars Program under Grant tsqn202408009.

References

[Akkuzu *et al.*, 2023] Nida Akkuzu, Murat Ucan, and Mehmet Kaya. Classification of multi-label electrocardiograms utilizing the efficientnet cnn model. In *2023 4th International Conference on Data Analytics for Business and Industry (ICDABI)*, pages 268–272. IEEE, 2023.

[Alday *et al.*, 2020] Erick A Perez Alday, Annie Gu, Amit J Shah, Chad Robichaux, An-Kwok Ian Wong, Chengyu Liu, Feifei Liu, Ali Bahrami Rad, Andoni Elola, Salman Seyedi, et al. Classification of 12-lead ecgs: the physionet/computing in cardiology challenge 2020. *Physiological measurement*, 41(12):124003, 2020.

[Association, 2022] American Heart Association. What is an arrhythmia? https://www.heart.org/en/health-topics/arrhythmia/about-arrhythmia, 2022. Accessed: 2024-07-25.

[Bousseljot *et al.*, 1995] Ralf Bousseljot, Dieter Kreiseler, and Allard Schnabel. Nutzung der ekg-signaldatenbank cardiodat der ptb über das internet. *Biomedical Engineering / Biomedizinische Technik*, 1995.

[Chen *et al.*, 2021] Hui Chen, Guijin Wang, Guodong Zhang, Ping Zhang, and Huazhong Yang. Clecg: A novel contrastive learning framework for electrocardiogram arrhythmia classification. *IEEE Signal Processing Letters*, 28:1993–1997, 2021.

[Goldberger *et al.*, 2000] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

[Haleem *et al.*, 2019] Abid Haleem, Mohd Javaid, and Ibrahim Haleem Khan. Current status and applications of artificial intelligence (ai) in medical field: An overview. *Current Medicine Research and Practice*, 9(6):231–237, 2019.

[Hindricks *et al.*, 2010] Gerhard Hindricks, Evgueny Pokushalov, Lubos Urban, Milos Taborsky, Karl-Heinz Kuck, Dmitry Lebedev, Guido Rieger, and Helmut Pu¨rerfellner. Performance of a new leadless implantable cardiac monitor in detecting and quantifying atrial fibrillation results of the xpect trial. *Circulation: Arrhythmia and Electrophysiology*, 3(2):141–147, 2010.

[Holmes *et al.*, 2004] J Holmes, L Sacchi, R Bellazzi, et al. Artificial intelligence in medicine. *Ann R Coll Surg Engl*, 86:334–8, 2004.

[Huang *et al.*, 2019] Jingshan Huang, Binqiang Chen, Bin Yao, and Wangpeng He. Ecg arrhythmia classification using stft-based spectrogram and convolutional neural network. *IEEE access*, 7:92871–92880, 2019.

[Jia *et al.*, 2021] Zhenge Jia, Yiyu Shi, Samir Saba, and Jingtong Hu. On-device prior knowledge incorporated learning for personalized atrial fibrillation detection. *ACM Transactions on Embedded Computing Systems (TECS)*, 20(5s):1–25, 2021.

[Jun *et al.*, 2018] Tae Joon Jun, Hoang Minh Nguyen, Daeyoun Kang, Dohyeun Kim, Daeyoung Kim, and Young-Hak Kim. Ecg arrhythmia classification using a 2-d convolutional neural network. *arXiv preprint arXiv:1804.06812*, 2018.

[Kanna and Eliyas, 2023] DN Kanna and M Mohammed Eliyas. Arrhythmia heart syndrome-a silent killer. *Current Research in Life Sciences*, page 15, 2023.

[Kiyasseh *et al.*, 2021] Dani Kiyasseh, Tingting Zhu, and David A Clifton. Clocs: Contrastive learning of cardiac signals across space, time, and patients. In *International Conference on Machine Learning*, pages 5606–5615. PMLR, 2021.

[Krasteva *et al.*, 2020] Vessela Krasteva, Sarah Ménétré, Jean-Philippe Didon, and Irena Jekova. Fully convolutional deep neural networks with optimized hyperparameters for detection of shockable and non-shockable rhythms. *Sensors*, 20(10):2875, 2020.

[Kuznetsov *et al.*, 2020] VV Kuznetsov, VA Moskalenko, and N Yu Zolotykh. Electrocardiogram generation and feature extraction using a variational autoencoder. *arXiv preprint arXiv:2002.00254*, 2020.

[Lian *et al.*, 2011] Jie Lian, Lian Wang, and Dirk Muessig. A simple method to detect atrial fibrillation using rr intervals. *The American journal of cardiology*, 107(10):1494–1497, 2011.

[Liu *et al.*, 2018] Feifei Liu, Chengyu Liu, Lina Zhao, Xiangyu Zhang, Xiaoling Wu, Xiaoyan Xu, Yulin Liu, Caiyun Ma, Shoushui Wei, Zhiqiang He, et al. An open access database for evaluating the algorithms of electrocardiogram rhythm and morphology abnormality detection. *Journal of Medical Imaging and Health Informatics*, 8(7):1368–1373, 2018.

[Mehari and Strodthoff, 2022] Temesgen Mehari and Nils Strodthoff. Self-supervised representation learning from 12-lead ecg data. *Computers in biology and medicine*, 141:105114, 2022.

[Nakamoto *et al.*, 2022] Mitsuhiko Nakamoto, Satoshi Kodera, Hirotoshi Takeuchi, Shinnosuke Sawano, Susumu Katsushika, K Ninomiya, H Akazawa, and I Komuro. Self-supervised contrastive learning for electrocardiograms to detect left ventricular systolic dysfunction. In *Proceedings of the Annual Conference of JSAI, vol. JSAI2022*, pages 1–7, 2022.

[Petrutiu *et al.*, 2007] Simona Petrutiu, Alan V Sahakian, and Steven Swiryn. Abrupt changes in fibrillatory wave

characteristics at the termination of paroxysmal atrial fibrillation in humans. *Europace*, 9(7):466–470, 2007.

[PhysioBank, 2000] PhysioToolkit PhysioBank. Physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.

[Pürerfellner *et al.*, 2014] Helmut Pürerfellner, Evgeny Pokushalov, Shantanu Sarkar, Jodi Koehler, Ren Zhou, Lubos Urban, and Gerhard Hindricks. P-wave evidence as a method for improving algorithm to detect atrial fibrillation in insertable cardiac monitors. *Heart Rhythm*, 11(9):1575–1583, 2014.

[Rajpurkar *et al.*, 2017] Pranav Rajpurkar, Awni Y Hannun, Masoumeh Haghpanahi, Codie Bourn, and Andrew Y Ng. Cardiologist-level arrhythmia detection with convolutional neural networks. *arXiv preprint arXiv:1707.01836*, 2017.

[Sarkar *et al.*, 2017] Shantanu Sarkar, Daniel L Hansen, Grant A Neitzell, Jerry D Reiland, and Ryan Wyszynski. Method and apparatus for atrial arrhythmia episode detection, 2017.

[Tan and Le, 2019] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[Wasimuddin *et al.*, 2021] Muhammad Wasimuddin, Khaled Elleithy, Abdelshakour Abuzneid, Miad Faezipour, and Omar Abuzaghleh. Multiclass ecg signal analysis using global average-based 2-d convolutional neural network modeling. *Electronics*, 10(2):170, 2021.

[Xu and Liu, 2020] Xuexiang Xu and Hongxing Liu. Ecg heartbeat classification using convolutional neural networks. *IEEE Access*, 8:8614–8619, 2020.

[Zheng *et al.*, 2020] Jianwei Zheng, Jianming Zhang, Sidy Danioko, Hai Yao, Hangyuan Guo, and Cyril Rakovski. A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients. *Scientific data*, 7(1):1–8, 2020.