UNLOCKING THE SPECIALIZATION CAPABILITY OF PPG FOUNDATION MODELS FOR WEARABLE SIGNALS THROUGH STRUCTURED PRUNING AND ADAPTIVE FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Photoplethysmography (PPG) plays a central role in wearable health monitoring and clinical decision support. However, existing approaches to universal PPG representation learning are predominantly trained on high-quality public benchmark datasets and often struggle to adapt to sparse and low-quality real-world datasets, thereby limiting their generalization to complex clinical tasks and heterogeneous patient populations. Surprisingly, even after full fine-tuning, PPG foundation models cannot consistently outperform general-purpose time-series models. Therefore, a key question is how to effectively adapt PPG foundation models to target prediction tasks. Through empirical studies on various PPG foundation models, we find that pretrained models often exhibit inherent sparsity and computational redundancy, suggesting that PPG foundation models have learned to activate task-relevant network substructures for different prediction tasks. To preserve this valuable prior knowledge, we propose a structured pruning-and-fine-tuning framework, **SPAFT**, which regularizes the adaptive fine-tuning process by constraining it to a more relevant and compact parameter space based on parameter importance and structural redundancy. Extensive experiments across four PPG foundation models, five benchmark datasets, and ten downstream tasks demonstrate that SPAFT achieves win rates of 82.8% and 87.5% on regression and classification tasks, respectively. This "prune-then-finetune" paradigm often enables PPG foundation models to achieve state-of-the-art performance, laying a promising foundation for the real-world clinical deployment of PPG foundation models and the development of personalized PPG foundation models.

1 INTRODUCTION

Photoplethysmography (PPG) in wearable devices has become a widely used sensing modality due to its ability to assess physiological states non-invasively without requiring firm attachment to the skin (Charlton et al., 2023). By measuring blood volume changes through optical sensors, PPG enables the non-invasive assessment of various physiological indicators, including heart rate (Demirel & Holz, 2026), atrial fibrillation (Pereira et al., 2020), blood pressure (Wang et al., 2023), blood glucose (Saha et al., 2025), and respiratory rate (He et al., 2026; Kazemi et al., 2025). In recent years, the growing availability of large-scale PPG data has driven the development of Physiological Foundation Models. Compared with training separate models from scratch for each clinical task, PPG foundation models provide a promising solution for leveraging knowledge acquired through large-scale pretraining while reducing the dependence of downstream tasks on labeled data.

Despite the rapid development of PPG foundation models (PPG FMs), existing studies (Zhang et al., 2026; Chen et al., 2026) have primarily focused on improving the representation capabilities of FMs through larger-scale pretraining data, more effective pretraining objectives, and cross-modal physiological information, while research on how to effectively adapt pre-trained models to different downstream tasks remains very limited. Furthermore, learning high-quality general-purpose PPG representations through pretraining does not necessarily mean that these representations can directly meet the requirements of all downstream clinical tasks. PPG downstream applications exhibit

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

substantial task heterogeneity, with different prediction targets relying on distinct physiological information embedded in the waveforms. For example, heart rate prediction primarily depends on the periodicity of pulse signals, whereas atrial fibrillation detection focuses more on rhythm irregularities. Moreover, clinically collected waveform data are often limited in scale, suboptimal in quality, and sparsely labeled (Xie et al., 2025). Therefore, *how to effectively adapt pretrained PPG FMs to specific downstream tasks is a key challenge in transforming their general-purpose representations into task-specific representations.*

To address this challenge, we further investigate the internal representations of PPG FMs. Please note that we focus on convolutional neural network-based PPG FMs for two main reasons: 1) convolution-based PPG FMs achieve the best performance in most cases; and 2) most Transformer-based PPG FMs are not open-sourced. Our analysis reveals that pre-trained PPG FMs typically exhibit inherent sparsity, with a certain degree of redundancy in their parameters. For example, during downstream task execution, up to 50% of filter outputs can be nearly negligible, while up to 80% of intermediate filters remain inactive. These sparse patterns vary across datasets, suggesting that PPG FMs activate different subnetworks depending on the specific task. Although conventional fine-tuning preserves the original model architecture, indiscriminately updating all parameters may disrupt these task-relevant activation patterns. This not only increases the risk of overfitting but also fails to fully exploit the valuable task-specific substructures formed during pre-training.

In this paper, we propose **SPAFT**, a filter-based adaptive pruning-and-fine-tuning framework consisting of two stages: *structural pruning* and *adaptive fine-tuning*. During the structural compression stage, we first freeze the encoder and fit the task head to estimate the importance of individual units. We then construct an intra-layer convolutional parameter graph to quantify structural redundancy and accordingly allocate the pruning budget: the redundancy determines which layers should be pruned, while the importance determines which units should be removed within each layer. During the fine-tuning stage, we re-estimate the importance of individual units in the compressed model and adaptively allocate the trainable-unit budget for fine-tuning according to the relative concentration of highly scored units across different layers. Furthermore, to address individual heterogeneity, we explore *personalized training* to further improve the predictive performance of PPG FMs. The key contributions of SPAFT are:

❶ **Identifying the specialization bottleneck of PPG FMs.** We conduct comprehensive and fair benchmark comparisons, showing that even after fine-tuning, PPG FMs still struggle to achieve strong generalization performance across different downstream tasks, highlighting PPG FM specialization as a critical unresolved challenge.

❷ **Uncovering the inherent sparsity of PPG FMs.** Our empirical analysis reveals that pretrained PPG FMs activate only task-relevant subsets of parameters during downstream inference. We argue that these specialized model components represent valuable prior knowledge acquired during pretraining and should therefore be preserved when adapting PPG FMs to specific downstream PPG data.

❸ **Pioneering fine-tuning with pruning for PPG FMs.** We propose a novel prune-then-finetune paradigm for post-training PPG FMs. To the best of our knowledge, this is the first work to improve predictive performance through pruning and personalized training of PPG FMs. Remarkably, we find that optimal performance can be achieved by fine-tuning as little as approximately 1% of the model parameters.

❹ **Achieving simultaneous improvements in performance and efficiency.** We conduct extensive experiments across four PPG FMs, four public datasets, and a new dataset collected using smart rings in one of the largest hospitals in Asia. The results demonstrate that our method enables PPG FMs to outperform state-of-the-art baselines across diverse downstream task settings.

2 Related work

2.1 PPG Foundation Models

In the field of PPG, existing foundation models primarily leverage the intrinsic characteristics of the signals, including inter-individual variability, waveform morphology, susceptibility to noise and artifacts, and temporal predictability (Abbaspourazad et al., 2024; Pillai et al., 2025; Ding et al., 2024).

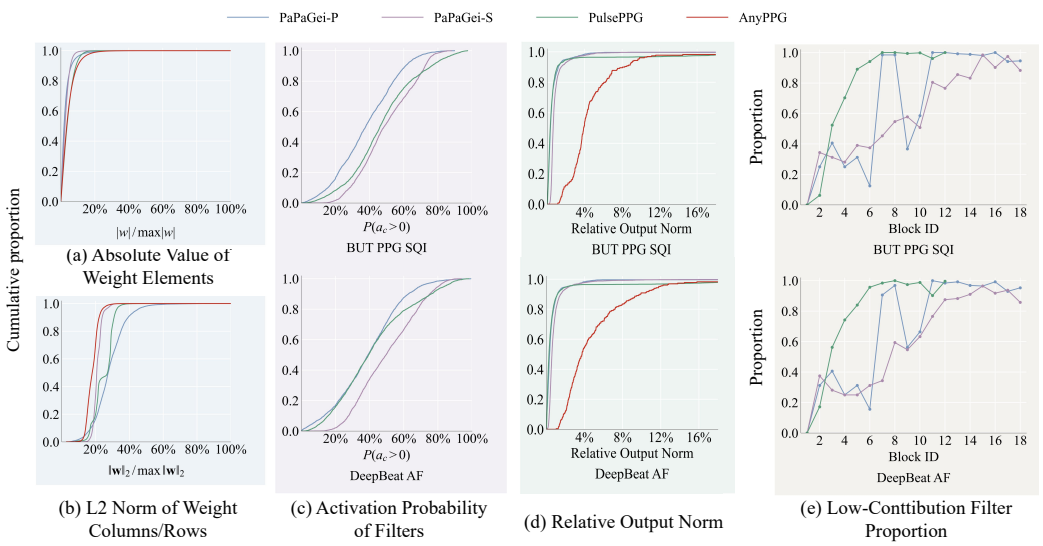


Figure 1: Empirical analysis. More results and analyses are provided in Appendix B.

PulsePPG (Saha et al., 2025) further incorporates waveform morphology by designing contrastive learning objectives based on physiological indices or motif-level similarity, enabling the extraction of physiologically meaningful features. In contrast, GPT-PPG (Chen et al., 2025) and PPGPT (Zhang et al., 2026) explore generative pretraining approaches inspired by autoregressive language models to learn representations from PPG time series. Meanwhile, another line of research on foundation models for wearable sensing leverages multimodal data, including PPG or its derived signals (Erturk et al., 2025; Narayanswamy et al., 2025), such as AnyPPG (Nie et al., 2026) and LeJEPA (Chen et al., 2026). However, these approaches are not specifically designed for PPG and therefore differ fundamentally in their modeling objectives from PPG-focused foundation models. A few studies (Xiang et al., 2026; Ni et al., 2025) have begun to explore the distillation and compression of PPG FMs, aiming to facilitate their deployment on edge devices. Despite the significant progress made by these studies, they have yet to explore the generalization capability of PPG FMs across downstream tasks, particularly on wearable data collected in real-world clinical settings, leaving a critical research gap that remains to be addressed.

2.2 Model Pruning

CNN filter pruning methods can be broadly divided into two categories: data-dependent filter pruning and data-independent filter pruning (He et al., 2019; He & Xiao, 2023). Since the use of training data introduces additional computational overhead, data-independent methods are generally more efficient than data-dependent methods. **Data-Dependent Filter Pruning**. Some filter pruning methods (Zhuang et al., 2018; Suau et al., 2018) require training data to determine which filters should be pruned. (He et al., 2020) designed a differentiable pruning criterion sampler, which is optimized using the validation loss of the pruned network obtained from the sampled criteria. **Data-Independent Filter Pruning**. Several data-independent filter pruning strategies (Ye et al., 2018; Sui et al., 2021; Park et al., 2024) have also been explored. REPrune (Park et al., 2024) first employs agglomerative clustering to identify similar kernels within each filter. It then selects filters that maximize the inclusion of representative kernels by optimizing the maximum cluster coverage problem. However, these techniques have not yet been applied to PPG FMs to investigate their generalizability.

3 Preliminary

3.1 Empirical Analysis of PPG FMs

To investigate the internal mechanisms of PPG FMs, we study them from the perspective of convolutional filters. We conduct experiments on the BUT PPG (Nemcova et al., 2021) and Deep-

Beat (Torres-Soto & Ashley, 2020) datasets using four pretrained PPG FMs, including AnyPPG, PaPaGei-P (Pillai et al., 2025), PaPaGei-S (Pillai et al., 2025), and Pulse-PPG (Saha et al., 2025).

(1) Insignificant Weight Magnitudes and Outputs of filters. We first examine at the parameter level whether PPG FMs contain a large proportion of low-magnitude parameters, providing initial evidence that the models may exhibit substantial redundancy. For each filter, we separately compute its parameter magnitude in the following two ways: (1) *the normalized absolute values of the weight elements*; and (2) *the normalized L_2-norm of the weight vector*.

As shown in Figure 1(a) and (b), a substantial proportion of the parameter weights in the four PPG FMs have small magnitudes, exhibiting similarly skewed distributions. Notably, parameter magnitude-based pruning is a task-agnostic approach and may lead to suboptimal performance on downstream tasks. Therefore, we use changes in the loss to compute importance scores. We further measure the magnitude of each filter's output relative to the residual representation to quantify the extent to which that filter actually modifies the residual component. Specifically, we define it as follows.

Definition 1 *Given the output of the i-th filter, $o_i \in \mathbb{R}^d$, and the residual representation $x \in \mathbb{R}^d$, the relative output norm of the i-th filter is defined over the downstream dataset as $\mathbb{E}_x \left(\|o_i\|_2 / \|x\|_2 \right)$.*

Figure 1(d) illustrates the different distributions of relative output norms. Compared with AnyPPG, where all filters actively contribute at least 1.8% relative norm, some filters in the other three PPG FMs produce outputs with nearly zero norms. Moreover, the degree of sparsity is affected by both downstream tasks and model architectures.

(2) Sparse Activations in filters. To further verify whether the low-contribution phenomenon arises from the model's selective use of filters, we measure and analyze the activation patterns of each filter.

Definition 2 *The activation probability of the i-th intermediate filter in PPG FMs is defined as the frequency with which its activation value a_i is positive over the entire dataset, i.e., $P(a_i > 0)$.*

Figure 1(c) shows that PPG FMs exhibit a high degree of sparsity. For example, in the PaPaGei-P model, approximately 71%–78% of filters have an activation probability below 50%.

(3) Task-Specific Subnetwork. We further investigate whether the structural sparsity of PPG FMs is a systematic characteristic that persists across different blocks[1]. We define blocks with an average relative output norm below 1% as unimportant blocks. Figure 1(e) further demonstrates that significant sparsity patterns are widely present in PPG FMs.

Based on the above analysis, most pre-trained PPG FMs exhibit inherent sparsity during inference, with a substantial proportion of parameters being either redundant or inactive. Moreover, they exhibit different sparsity patterns across different datasets. This motivates us to further explore more generalizable pruning-and-fine-tuning methods.

4 METHODS

Given a pretrained PPG encoder $E_{\boldsymbol{\theta}}$, a task head $h_{\boldsymbol{\phi}}$ and a downstream dataset $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, where y_n denotes either a continuous physiological measurement or a binary classification label, the complete prediction model is defined as $F_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}) = h_{\boldsymbol{\phi}}(E_{\boldsymbol{\theta}}(\mathbf{x}))$. Given a global pruning ratio ρ, our goal is to remove computational units from the encoder that contribute less to the current task and exhibit strong structural redundancy, thereby obtaining a compact encoder $E_{\boldsymbol{\theta}^\star}$ and its task head $h_{\boldsymbol{\phi}^\star}$, which are then fine-tuned to improve downstream task performance.

The overall pipeline is illustrated in Figure 2. First, we perform a single forward pass over the unpruned model to estimate the task importance of all nodes[2]. We then construct an intra-layer structural graph based on the cosine relationships among convolutional kernel parameters and allocate pruning opportunities across layers according to their structural redundancy. Based on the computed structural redundancy and task importance, we perform node pruning planning and subsequently conduct adaptive fine-tuning on the compact model.

[1] A block refers to a ResNet layer in the PPG FMs.

[2] The nodes refer to filters in PPG FMs.

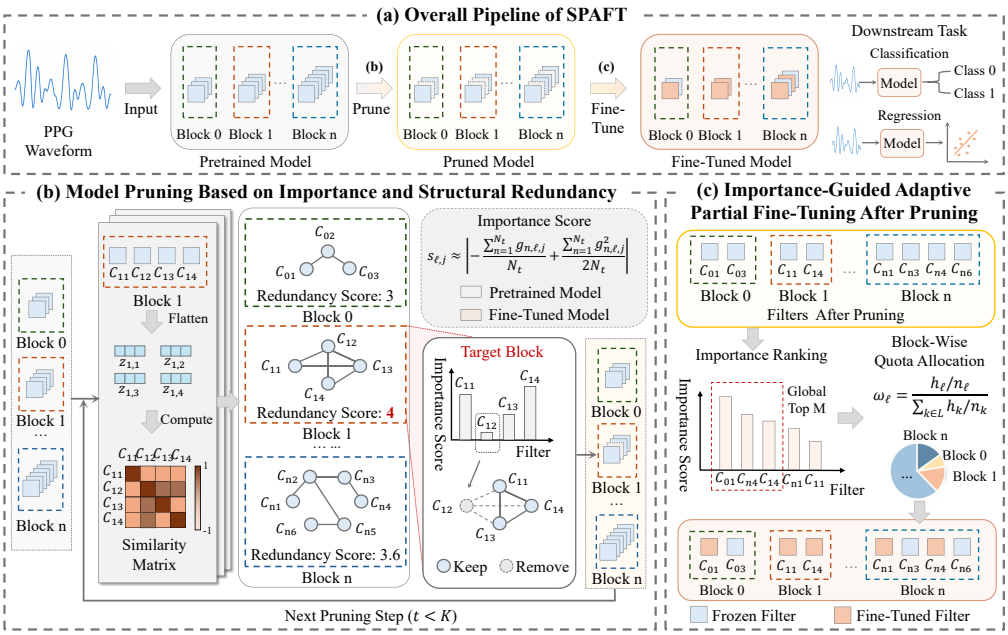


Figure 2: Architecture of Our Proposed SPAFT Framework.

4.1 Model Pruning Based on Importance and Structural Redundancy

Task Importance Estimation Before Pruning. We first freeze the encoder and then fit a linear regression head or a logistic classification head according to the task requirements. Next, we perform a one-time importance estimation on the unpruned model using the training set $\mathcal{D}_t$. The loss change caused by removing a specific node is

$$\Delta_{\ell,j} = \left|\mathcal{L}(\mathbf{m} - \mathbf{e}_{\ell,j}) - \mathcal{L}(\mathbf{m})\right|, \quad \mathcal{L}(\mathbf{m}) = \mathcal{L}\big(F^{\mathbf{m}}_{\boldsymbol{\theta},\boldsymbol{\phi}}(\mathbf{x}), y\big)\,. \tag{1}$$

where $\mathbf{m}$ denotes the gate vector for all nodes, and $\mathbf{e}_{\ell,j}$ denotes an indicator vector that takes the value 1 only at node (ℓ, j), corresponding to the j-th filter in the ℓ-th block. A smaller $\Delta_{\ell,j}$ indicates that the corresponding node has a weaker influence on the current task. However, explicitly computing this loss difference for each node requires a large number of repeated inference passes. To address this issue, we approximate the loss change caused by node removal at $\mathbf{m}_{\ell,j} = 1$. Let the gradient of the n-th sample with respect to the node gate be

$$g_{n,\ell,j} = \left(\partial \mathcal{L}_n(\mathbf{m}) / \partial \mathbf{m}_{\ell,j}\right)|_{\mathbf{m}_{\ell,j}=1}. \tag{2}$$

By applying a second-order Taylor expansion to $\Delta_{\ell,j}$ and approximating the diagonal entries of the Hessian matrix using the squared per-sample first-order gradients, the node importance is given by

$$s_{\ell,j} \approx \left|-\mathbf{e}_{\ell,j}^{\top}\nabla_{\mathbf{m}}\mathcal{L} + \frac{1}{2}\mathbf{e}_{\ell,j}^{\top}\left(\nabla^2_{\mathbf{m}}\mathcal{L}\right)\mathbf{e}_{\ell,j}\right| \approx \left|-\frac{\sum_{n=1}^{N_t} g_{n,\ell,j}}{N_t} + \frac{\sum_{n=1}^{N_t} g^2_{n,\ell,j}}{2N_t}\right|, \tag{3}$$

where the first term describes the first-order loss change induced by removing the gate, while the second term provides a second-order correction based on an empirical Fisher information approximation. N_t denotes the number of samples in the training set. Note that, for efficiency, s remains fixed throughout the subsequent pruning steps and is not recomputed.

Structural Redundancy on the Cosine Graph. Task importance characterizes the relationship between individual nodes and the downstream target, but it alone cannot determine how many nodes can be removed from different layers (Wang et al., 2021). To address this issue, we represent each block as an undirected graph and use graph structural redundancy to characterize the substitutability of nodes within the block.

Specifically, the structural feature $\mathbf{z}_{\ell,j}$ of a node is the flattened parameter vector of the corresponding output convolution kernel in the first convolutional layer. We normalize the node features and construct the current adjacency relations according to the cosine threshold τ:

$$\bar{\mathbf{z}}_{\ell,j} = \mathbf{z}_{\ell,j}/\|\mathbf{z}_{\ell,j}\|_2, \quad A^{(t)}_{\ell,ij} = \mathbb{I}\left[\bar{\mathbf{z}}^{\top}_{\ell,i}\bar{\mathbf{z}}_{\ell,j} \geq \tau\right], \quad i,j \in V^{(t)}_{\ell},\ i \neq j\,, \tag{4}$$

where $A^{(t)}_{\ell,ii} = 0$, $V^{(t)}_\ell$ denote the set of active nodes in block ℓ after t global pruning steps. The cosine metric removes the influence of the overall magnitude of the parameter vectors, while the threshold τ controls edge formation.

Based on the constructed cosine graph, we combine connected components and local coverage (Kolmogorov & Tikhomirov, 1959) to measure structural redundancy. Let $k^{(t)}_\ell$ denote the number of connected components, and define a closed ball of radius r using the graph distance $d_{G^{(t)}_\ell}$: $\mathcal{B}^{(t)}_{\ell,r}(v) = \left\{u \in V^{(t)}_\ell : d_{G^{(t)}_\ell}(u,v) \le r\right\}, r \in \{1,2\}$. For each radius, among the uncovered nodes, we select the node with the largest degree in the current induced graph and mark its closed ball as covered, repeating this procedure until the entire graph is covered. Let the resulting greedy cover number be $c^{(r)}_{\ell,t}$, and define $\widehat{N}^{(t)}_{\ell,1c} = (c^{(1)}_{\ell,t} + c^{(2)}_{\ell,t})/2$. The structural redundancy is defined as:

$$R^{(t)}_\ell = \frac{N^{(t)}_\ell}{w_1 k^{(t)}_\ell + w_2 \widehat{N}^{(t)}_{\ell,1c}}, \qquad N^{(t)}_\ell = |V^{(t)}_\ell|\,, \tag{5}$$

where w_1, w_2 are the probability weights. Redundancy is higher when a larger number of nodes can be represented by fewer structural families and local centers.

Task-Structure Collaborative Node Pruning. We use structural redundancy to allocate pruning opportunities across layers and frozen importance scores to select nodes within each layer. This hierarchical decision mechanism avoids directly comparing importance scores across layers and does not require predefined layer-wise pruning ratios.

To prevent any residual branch from degenerating into an empty branch, only blocks that retain at least two nodes are considered eligible for pruning. At step t, we first select the block ℓ_t with the highest current structural redundancy. Subsequently, among the currently active nodes in the selected block, we remove the actual node j_t with the lowest task importance, thereby updating the active node sets. Then, only the induced graph of block ℓ_t over $V^{(t)}_{\ell_t}$ is updated, and the number of connected components, coverage estimate, and structural redundancy of that block are recomputed, while the graph states of all other blocks remain unchanged. Repeating the above steps for K iterations yields the final node set for each block: $V^\star_\ell = V^{(K)}_\ell$.

4.2 Importance-Guided Adaptive Fine-Tuning After Pruning

Node removal changes the predictive function, and the importance of the remaining nodes may change accordingly. To allocate the limited update budget to paths [3] that remain highly task-sensitive after pruning, we re-estimate importance on the compact model and allocate fine-tuning quotas according to the concentration of high-scoring nodes within each block.

Post-Pruning Importance Re-estimation. Generate gating masks based on the final active set $V^\star_\ell$, and slice the convolutional and normalization tensors to obtain the compact encoder parameters θ_c. We reintroduce gates $\widetilde{m}_{\ell,j}$ initialized to 1 on the remaining nodes, freeze the compact encoder, the original task head, and the BN states, and reuse the losses from Section 4.1. Furthermore, we can compute

$$\widetilde{g}_{n,\ell,j} = \left.\frac{\partial \ell\Big(F^{\widetilde{\mathbf{m}}}_{\boldsymbol{\theta}_c,\quad \boldsymbol{\phi}}(\mathbf{x}_n), y_n\Big)}{\partial \widetilde{m}_{\ell,j}}\right|_{\widetilde{\mathbf{m}}=\mathbf{1}}, \quad \widetilde{s}_{\ell,j} = \left|-\frac{\sum_{n=1}^{N_f}\widetilde{g}_{n,\ell,j}}{N_f} + \frac{\sum_{n=1}^{N_f}\widetilde{g}^2_{n,\ell,j}}{2N_f}\right|, \quad j \in V^\star_\ell. \tag{6}$$

The pre-pruning scores $s_{\ell,j}$ determine which nodes are removed, whereas the post-pruning scores $\widetilde{s}_{\ell,j}$ determine which nodes are selected for fine-tuning; they respectively characterize task sensitivity in the original and compact networks.

Layer-Width-Normalized Budget Allocation. Since different blocks may have different widths after pruning, directly allocating the budget based on the number of high-scoring nodes would be influenced by block size. Therefore, we measure the contribution of each layer using the proportion of high-scoring nodes within that block. The specific formulation is given as follows:

$$h_\ell = |j : (\ell, j) \in \mathcal{I}_{\mathrm{M}}|\,, \qquad d_\ell = \frac{h_\ell}{n_\ell}, \qquad \omega_\ell = \frac{d_\ell}{\sum_{k\in L} d_k}\,, \tag{7}$$

[3] The path refers to the connection between two blocks in PPG FMs, including BatchNorm (BN) layers.

where the set $\mathcal{I}_{\mathrm{M}}$ contains the $\mathrm{M} = \lceil \gamma N_{\mathrm{c}} \rceil$ highest-scoring remaining nodes, and γ is the filter ratio of the PPG FMs to be fine-tuned, N_{c} is the total number of remaining nodes, $\lceil \cdot \rceil$ denotes the ceiling operation, L is the number of blocks in the PPG FMs. To preserve an adaptation path for every block, we first reserve one node for each block and then allocate the remaining $\mathrm{M} - L$ quotas according to ω_ℓ. The final quota satisfies

$$q_\ell = 1 + \mathrm{M}\omega_\ell, \qquad 0 \le \mathrm{M}\omega_\ell \le n_\ell - 1, \qquad \sum_{\ell \in L} q_\ell = \mathrm{M}\,. \tag{8}$$

Constrained Optimization Along Filter Paths. For each selected node, the convolutional parameters along its dependency path are made trainable. For classification tasks, the fine-tuning objective is formulated as

$$\min_{\boldsymbol{\theta}, \boldsymbol{\phi}'} \quad \frac{1}{N_f} \sum_{(\mathbf{x}_n, y_n) \in \mathcal{D}_f} \ell_{\mathrm{fin}} \left(F_{\boldsymbol{\theta}, \boldsymbol{\phi}'}(\mathbf{x}_n), y_n \right) \,. \tag{9}$$

For regression tasks, the prediction residual is normalized by the target standard deviation σ_f of the fitting set, and the Smooth L1 loss (Girshick, 2015) is used:

$$e_n = \frac{\hat{y}_n - y_n}{\sigma_f}, \quad \ell_{\mathrm{rec}}(\hat{y}_n, y_n) = \begin{cases} e_n^2/2, & |e_n| < 1, \\ |e_n| - 1/2, & |e_n| \ge 1. \end{cases} \tag{10}$$

5 Experiments

In this section, we conduct extensive experiments under both data-wise and subject-wise splitting settings to evaluate the effectiveness of our method for adapting PPG FMs to downstream tasks, as well as its effectiveness for personalized training. Additional experimental results are provided in Appendix C, including subnetwork transfer analysis (Appendix C.3), inference efficiency analysis (Appendix C.4), adaptivity analysis (Appendix C.5), and parameter analysis (Appendix C.6).

5.1 Experimental Settings

Datasets. To evaluate the effectiveness of SPAFT, we conduct extensive experiments across multiple datasets and tasks. Our evaluation covers both clinical settings and portable or wearable device scenarios. For regression tasks, we use UCI-BP (Kachuee et al., 2015), PPG-DaLiA (Reiss et al., 2019), BUT PPG, and the Ring-PPG Dataset; for classification tasks, we use BUT PPG and DeepBeat. Further details are provided in Appendix A.1.

Baselines. To comprehensively evaluate SPAFT, we compare it with fourteen representative baseline methods from five categories: (1) **conventional adaptation methods**: Linear Probe, Full-FT, LoRA, and QLoRA; and (2) **pruning-based adaptation methods**: PFEC (Li et al., 2017), Taylor-FO (Molchanov et al., 2019), FPGM (He et al., 2019), SRR-GR (Wang et al., 2021), and Less-is-More (Zhao et al., 2026); (3) **Distillation-based Methods**: PPG-DISTIL (Ni et al., 2025); (4) **Time-Series Foundation Models**: Chronos (Ansari et al., 2024) and MOMENT (Goswami et al., 2024); (5)**Transformer-based PPG FMs**: GPT-PPG (Chen et al., 2025) and SIGMA-PPG (Guo et al., 2026). The details of the baselines are provided in Appendix A.3.

Implementation Details. We evaluate SPAFT using pretrained AnyPPG, PaPaGei-P, PaPaGei-S, and Pulse-PPG. The adaptive fine-tuning stage uses AdamW with an effective batch size of 1,024 for at most 10 epochs and early-stopping patience of three. Task-specific search spaces cover the cosine threshold τ, global pruning ratio ρ, fine-tuning ratio γ, and learning rate. Specific details are provided in Appendix A.2.

5.2 Overall Performance

Tables 1 and 2 report the regression and classification results, respectively, across diverse downstream PPG tasks. We have the following observations: **(1) Overall, SPAFT achieves the strongest aggregate performance across both regression and classification tasks.** SPAFT achieves win rates of 82.8% and 87.5% on regression and classification tasks, respectively. Its superior performance demonstrates that SPAFT can effectively specialize general-purpose PPG representations to

Table 1: Model performance on regression tasks. We mark the winner in **bold**, and the runner-up is indicated by an underline. The complete results are provided in Table 16 of Appendix C.1.

Data	Tasks	FM	Linear Probe		Full-FT		LoRA		QLoRA		PFEC		Taylor-FO		FPGM		SRR-GR		Less-is-More		Ours	
			MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓	MSE ↓	MAE ↓
UCI-BP	SBP	AnyPPG	379.531	15.388	221.626	10.804	327.334	13.870	392.230	15.488	460.250	16.821	191.174	9.871	223.532	10.885	338.435	14.133	379.268	15.086	**143.083**	**8.345**
		PaPaGei-P	430.231	16.542	371.494	14.974	379.325	15.183	471.292	17.481	482.716	17.735	371.448	14.924	365.637	14.757	482.458	17.775	434.871	16.591	**245.788**	**11.438**
		PaPaGei-S	461.856	17.295	383.028	15.299	422.433	16.301	438.100	16.600	492.882	17.987	393.232	15.628	380.783	15.240	445.977	16.893	387.265	15.478	**259.401**	**11.858**
		Pulse-PPG	417.647	16.263	256.021	11.825	356.951	14.652	354.624	14.660	335.480	14.175	257.260	11.845	254.844	11.788	312.325	13.553	277.007	12.581	**183.581**	**9.661**
	DBP	AnyPPG	87.673	6.881	59.591	5.156	78.304	6.224	92.546	7.243	82.621	6.564	53.857	4.816	78.587	6.081	76.814	6.256	80.273	6.475	**41.276**	**4.111**
		PaPaGei-P	98.548	7.436	87.802	6.735	88.378	6.752	92.338	7.192	88.084	6.814	90.350	6.846	87.061	6.659	99.238	7.458	89.353	6.992	**62.908**	**5.317**
		PaPaGei-S	104.376	7.697	82.521	6.446	95.727	7.113	99.174	7.196	88.765	6.918	85.060	6.608	84.573	6.579	95.835	7.156	90.817	6.973	**64.756**	**5.420**
		Pulse-PPG	95.357	7.283	66.772	5.575	82.859	6.460	82.715	6.559	96.364	7.074	66.458	5.550	67.699	5.617	78.050	6.202	91.655	6.907	**49.865**	**4.609**
PPG-DaLiA	HR	AnyPPG	92.093	6.505	82.302	5.597	87.603	6.304	242.786	11.836	151.705	7.873	75.095	5.385	164.685	7.387	142.293	7.181	145.387	7.923	**71.870**	**5.066**
		PaPaGei-P	196.916	10.282	222.989	11.029	190.257	10.043	263.867	12.824	212.701	11.105	212.615	11.002	259.243	12.630	190.598	10.054	174.190	9.386	**108.284**	**6.818**
		PaPaGei-S	228.658	11.563	258.853	12.017	217.661	11.195	277.062	13.108	229.755	11.384	174.363	9.736	311.547	13.493	215.235	11.246	285.047	13.502	**130.235**	**7.977**
		Pulse-PPG	149.143	8.996	144.507	8.707	145.238	8.869	199.532	10.704	221.702	11.086	130.593	8.291	121.318	7.922	242.504	11.332	253.971	12.197	**106.274**	**7.191**
Ring-PPG	HR	AnyPPG	74.642	6.167	80.774	7.376	77.026	6.446	62.443	6.479	75.636	7.004	76.961	6.427	92.972	7.778	64.864	6.469	59.285	6.371	**58.447**	**5.717**
		PaPaGei-P	78.552	6.645	91.423	7.289	80.245	6.703	74.180	7.021	**58.648**	**6.147**	76.612	6.611	154.472	10.378	60.984	6.424	59.671	6.283	70.638	6.296
		PaPaGei-S	92.354	7.999	129.305	9.622	153.571	10.458	94.347	8.019	99.860	8.328	209.814	12.192	109.040	8.859	102.507	8.464	94.254	7.893	**77.574**	**7.278**
		Pulse-PPG	76.184	6.715	**69.563**	**6.531**	81.410	7.009	80.308	7.259	93.583	7.684	77.627	6.544	164.395	10.448	97.916	7.799	86.114	7.228	75.062	6.697
	SBP	AnyPPG	335.459	14.787	555.882	19.105	323.997	14.672	214.162	12.225	234.743	12.668	321.300	13.957	333.044	14.664	224.976	12.301	199.295	11.839	**155.748**	**10.751**
		PaPaGei-P	389.351	16.079	391.395	16.431	358.480	15.566	212.676	12.178	219.713	12.468	388.049	16.048	239.318	13.021	223.084	12.487	188.714	11.984	**162.895**	**10.843**
		PaPaGei-S	391.653	15.372	427.919	16.231	339.616	14.218	189.743	11.814	204.748	12.191	580.246	19.623	230.095	12.836	197.943	11.873	191.154	11.752	**152.555**	**10.454**
		Pulse-PPG	321.497	14.497	382.435	16.023	291.316	13.543	237.512	12.598	271.118	13.579	418.904	16.017	300.412	14.384	309.956	14.512	194.090	13.322	**136.191**	**9.417**
	DBP	AnyPPG	245.139	11.525	285.319	12.774	248.110	11.537	220.622	12.007	329.623	14.543	199.901	11.371	213.504	11.639	363.325	15.538	309.329	14.167	**153.459**	**10.195**
		PaPaGei-P	336.107	15.038	356.623	15.129	334.480	14.987	216.306	12.184	211.046	12.168	311.798	14.176	228.868	12.446	200.726	12.153	218.867	12.320	**186.804**	**11.337**
		PaPaGei-S	287.337	13.617	412.088	15.609	288.868	13.626	207.111	**11.881**	213.396	12.157	551.298	18.605	362.254	14.935	260.305	13.131	299.418	13.876	**193.974**	11.898
		Pulse-PPG	239.984	12.160	277.450	13.754	240.546	12.233	608.723	18.827	547.154	16.620	244.191	12.772	285.420	13.961	353.672	15.587	752.999	22.112	**176.193**	**11.405**
	SpO_2	AnyPPG	1.787	1.061	2.231	1.200	2.080	1.151	1.638	1.047	1.795	1.098	1.827	1.107	1.488	0.985	1.789	1.073	1.288	0.937	**0.898**	**0.763**
		PaPaGei-P	2.093	1.182	2.998	1.453	2.205	1.208	1.626	1.089	1.719	1.137	1.927	1.133	1.774	1.121	1.710	1.124	1.665	1.079	**0.793**	**0.749**
		PaPaGei-S	2.015	1.134	2.417	1.315	2.728	1.299	1.918	1.190	1.869	1.160	7.452	2.301	2.506	1.366	1.857	1.163	1.665	1.094	**0.935**	**0.797**
		Pulse-PPG	2.211	1.226	2.443	1.296	2.239	1.245	1.622	1.076	2.498	1.281	1.976	1.176	2.480	1.376	2.267	1.278	1.693	1.094	**0.872**	**0.758**
BUT PPG	HR	AnyPPG	202.635	10.675	204.919	10.632	203.507	10.615	212.448	11.282	187.598	10.222	198.694	10.550	337.001	13.130	185.012	10.208	191.083	10.107	**179.911**	**9.931**
		PaPaGei-P	212.401	11.336	236.696	11.803	214.771	11.282	206.196	11.037	203.959	**10.917**	225.336	11.347	214.450	11.324	203.093	11.007	**201.520**	10.937	207.383	11.265
		PaPaGei-S	237.665	12.054	255.788	12.136	242.740	12.066	214.732	**11.140**	216.468	11.422	236.288	11.675	233.095	11.814	215.611	11.479	**211.060**	11.257	215.597	11.318
		Pulse-PPG	228.978	11.830	229.720	12.059	231.559	11.812	269.051	12.855	265.401	12.305	223.648	11.600	**207.564**	**11.135**	250.855	12.435	303.459	13.693	208.575	11.205
Win Rate			0%		3.1%		0%		3.1%		4.7%		0%		3.1%		0%		3.1%		82.8%	

* "Win Rate" denotes the percentage of MSE and MAE comparisons in which a method achieves the best result.

Table 2: Model performance on classification tasks and the complete results are provided in Table 17 of Appendix C.1.

Data	Tasks	FM	Linear Probe		Full-FT		LoRA		QLoRA		PFEC		Taylor-FO		FPGM		SRR-GR		Less-is-More		Ours	
			F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑	F1 ↑	AUROC ↑
BUT PPG	SQI	AnyPPG	0.622	0.799	0.599	0.724	0.645	0.793	0.579	0.626	0.678	**0.811**	0.632	0.734	0.622	0.755	0.697	0.800	0.695	0.799	**0.707**	0.806
		PaPaGei-P	0.535	0.736	0.595	0.682	0.475	0.685	0.558	0.579	0.585	0.628	0.599	0.712	0.544	0.690	0.581	0.598	0.599	0.633	**0.669**	**0.766**
		PaPaGei-S	0.544	0.714	0.606	0.708	0.491	0.697	0.629	0.686	0.614	0.645	0.612	0.701	0.594	0.695	0.600	0.633	0.616	0.675	**0.686**	**0.801**
		Pulse-PPG	0.514	0.718	0.562	0.724	0.515	0.717	0.517	0.575	0.629	0.717	0.572	0.738	0.540	0.764	0.630	0.749	0.566	0.734	**0.638**	**0.766**
DeepBeat	AF	AnyPPG	0.544	0.708	0.651	0.838	0.468	0.525	0.609	0.754	0.642	0.799	0.646	0.833	0.653	0.834	0.638	0.800	0.640	0.796	**0.684**	**0.844**
		PaPaGei-P	0.441	0.608	0.624	0.785	0.432	0.545	0.627	0.754	0.649	0.781	0.623	0.769	0.599	0.805	**0.677**	0.800	0.662	0.788	0.649	**0.831**
		PaPaGei-S	0.444	0.610	0.596	0.789	0.428	0.406	0.585	0.671	0.617	0.738	0.594	0.775	0.589	0.785	0.618	0.735	0.604	0.727	**0.638**	**0.809**
		Pulse-PPG	0.496	0.653	0.625	0.808	0.525	0.600	0.648	0.716	0.633	0.753	0.618	0.803	0.609	0.806	0.657	0.807	0.643	0.809	**0.671**	**0.841**
Win Rate			0%		0%		0%		0%		6.3%		0%		0%		6.3%		0%		87.5%	

meet the requirements of diverse downstream tasks. **(2) Compared with conventional adaptation methods.** SPAFT achieves better results in 60 out of 64 regression comparisons and 15 out of 16 classification comparisons. In particular, Full-FT also underperforms SPAFT in most settings, indicating that updating a larger parameter space does not necessarily lead to better task specialization. This observation is consistent with our motivation that indiscriminate model adaptation may disturb task-relevant substructures acquired during pretraining. In contrast, SPAFT provides a more targeted adaptation process for downstream task specialization. **(3) Compared with pruning-based adaptation methods.** SPAFT outperforms the best-performing pruning baseline in 54 out of 64 regression comparisons and 13 out of 16 classification comparisons, indicating that its performance gains do not simply stem from introducing pruning before fine-tuning. Although Less-is-More also adopts a task-aware prune-then-finetune strategy, it directly fine-tunes the retained compact backbone after pruning. In contrast, SPAFT adapts only the selectively identified parameter paths after compression, thereby further constraining downstream optimization to task-relevant substructures.

5.3 Personalized Training

To further investigate the personalized modeling capability of PPG FMs, we evaluate personalized training with SPAFT, LoRA, and Less-is-More on the temporal PPG-DaLiA split. Each method starts from its corresponding task-adapted checkpoint. For each subject, the earlier half of the original test samples is divided chronologically into training and validation sets at an 8:2 ratio, while the later half is used for evaluation. As shown in Figure 3, SPAFT achieves lower post-personalization test MSE than LoRA for 10 of 15 subjects and Less-is-More for 12 subjects on PaPaGei-S. Its own MSE decreases for all 15 subjects, with particularly visible improvements for S5 and S9. Across the same evaluation samples, the overall MSE decreases from 96.643 to 56.648 for SPAFT (41.38%), compared with 110.504 to 73.173 for LoRA (33.78%) and 146.365 to 127.635 for Less-is-More (12.80%) (see Table 12 in Appendix C.2). These results illustrate the potential of the parameter paths selected by SPAFT for personalized adaptation, as SPAFT consistently improves performance across all subjects and achieves the lowest overall MSE after personalized training on PaPaGei-S. Full settings and results for all four PPG FMs are provided in Appendix C.2.



Figure 3: Performance of PaPaGei-S with and without personalized training.

5.4 Ablation Studies

To better understand the contribution of each design in SPAFT, we conduct ablation studies on UCI-BP DBP and BUT PPG SQI. We individually remove importance-guided filter deletion (**W/O IFD**), SRR-guided block selection (**W/O SRRBS**), selective filter fine-tuning (**W/O SFF**), and adaptive blockwise fine-tuning allocation (**W/O ABFA**). As shown in Table 3, SPAFT achieves the best mean performance across all metrics for both PPG FMs, while removing any component consistently degrades performance. For the structural pruning stage, removing IFD causes clear performance drops. Removing SRRBS leads to inferior results. This suggests that structural redundancy provides useful guidance for allocating pruning opportunities across blocks, complementing task importance for within-block filter selection. For the fine-tuning stage, removing SFF consistently leads to performance degradation. Moreover, without ABFA, the MSE on Pulse-PPG increases to 59.584, while the AUROC on PaPaGei-P decreases to 0.664. These results demonstrate the effectiveness of reallocating the fine-tuning budget according to the task importance estimated after pruning.

Table 3: SPAFT component ablation study.

Model	Variant	UCI-BP DBP		BUT PPG SQI	
		MSE ↓	MAE ↓	F1 ↑	AUROC ↑
PAPAGEI-P	w/o IFD	65.089±0.416	5.441±0.054	0.595±0.069	0.726±0.009
	w/o SRRBS	65.746±0.147	5.506±0.034	0.572±0.100	0.737±0.007
	w/o SFF	87.165±0.225	6.746±0.016	0.512±0.049	0.627±0.078
	w/o ABFA	66.072±3.360	5.483±0.196	0.607±0.017	0.664±0.061
	SPAFT	63.760±0.730	5.373±0.052	0.640±0.057	0.765±0.021
Pulse-PPG	w/o IFD	55.500±2.761	4.936±0.141	0.457±0.000	0.718±0.020
	w/o SRRBS	55.203±0.912	4.908±0.046	0.551±0.033	0.716±0.060
	w/o SFF	56.042±3.107	4.923±0.192	0.487±0.023	0.722±0.047
	w/o ABFA	59.584±2.840	5.170±0.163	0.554±0.074	0.740±0.036
	SPAFT	50.747±1.465	4.659±0.086	0.589±0.052	0.768±0.008

5.5 Case Study

We further analyze the representations learned by different methods on the BUT PPG SQI test set. We select representative samples from the two classes for waveform visualization. Sample 1 is a low-quality waveform exhibiting pronounced morphological distortions and transient artifacts, whereas Sample 2 is a high-quality waveform showing clearer and more regular periodic pulse patterns. As shown in Figure 4, representative methods such as QLoRA fail to distinguish the two waveforms, with substantial inter-class overlap remaining in their feature spaces. In contrast, SPAFT produces more compact intra-class clusters and clearer inter-class separation. Samples with these distinct waveform morphologies are mapped to different class regions in our representation space, suggesting that the learned representations can effectively encode waveform differences associated with signal quality.

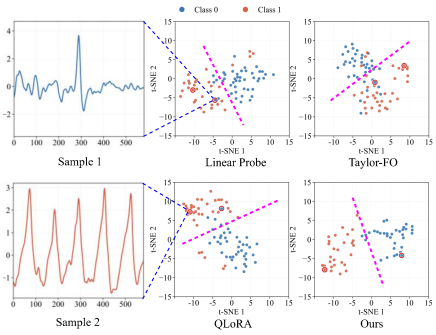


Figure 4: t-SNE visualization on SQI.

6 Conclusion

This work reveals a key challenge: how to effectively adapt pre-trained PPG FMs to achieve improved performance on downstream tasks. To address this challenge, we propose a "prune-then-finetune" framework, SPAFT, which identifies and extracts critical subnetworks through structured pruning. Extensive experiments demonstrate that our method can significantly improve predictive performance. To the best of our knowledge, this is the first work to primarily employ structured pruning to improve model performance through architectural specialization, providing a promising pathway toward unlocking the full potential of PPG FMs in real-world applications.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language-related tasks, including language polishing, grammar correction, and Chinese-to-English translation. We did not use generative AI tools to generate research ideas, design experiments, analyze data, develop the core methodology, or draw scientific conclusions.

ETHICS STATEMENT

We adhere to the ICLR Code of Ethics. This study uses only de-identified data, with no direct interaction with human subjects. No personally identifiable information (PII/PHI) was accessed or disclosed, and no attempt at re-identification was made. This work is for research purposes only and is not a medical device. No conflicts of interest or sensitive sponsorships are present.

REFERENCES

Salar Abbaspourazad, Oussama Elachqar, Andrew Miller, Saba Emrani, Udhyakumar Nallasamy, and Ian Shapiro. Large-scale training of foundation models for wearable biosignals. In *International Conference on Learning Representations*, volume 2024, pp. 3158–3181, 2024.

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Peter H Charlton, John Allen, Raquel Bailón, Stephanie Baker, Joachim A Behar, Fei Chen, Gari D Clifford, David A Clifton, Harry J Davies, Cheng Ding, et al. The 2023 wearable photoplethysmography roadmap. *Physiological measurement*, 44(11):111001, 2023.

Zhaoliang Chen, Cheng Ding, Saurabh Kataria, Runze Yan, Minxiao Wang, Randall Lee, and Xiao Hu. Gpt-ppg: a gpt-based foundation model for photoplethysmography signals. *Physiological Measurement*, 46(5):055004, 2025.

Zhaoliang Chen, Saurabh Kataria, Minxiao Wang, and Xiao Hu. Bridging ecg and ppg: Latent-space prediction for robust physiological analysis. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 10693–10703, 2026.

Berken Utku Demirel and Christian Holz. Continuous heart rate variability estimation from ppg via state-space modeling. *IEEE Transactions on Biomedical Engineering*, 2026.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115, 2023.

Cheng Ding, Zhicheng Guo, Zhaoliang Chen, Randall J Lee, Cynthia Rudin, and Xiao Hu. Siamquality: a convnet-based foundation model for photoplethysmography signals. *Physiological Measurement*, 45(8):085004, 2024.

Eray Erturk, Fahad Kamran, Salar Abbaspourazad, Sean Jewell, Harsh Sharma, Yujie Li, Sinead Williamson, Nicholas J Foti, and Joseph Futoma. Beyond sensor data: Foundation models of behavioral data from wearables improve health predictions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 15516–15541, 2025.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. In *International Conference on Machine Learning*, 2024.

Zongheng Guo, Tao Chen, Yang Jiao, Yi Pan, Xiao Hu, and Manuela Ferrario. SIGMA-PPG: Statistical-prior informed generative masking architecture for PPG foundation model. In *Forty-third International Conference on Machine Learning*, 2026.

Chenyang He, Xinyi Shao, Shun Huang, Bosong Huang, Daoqiang Zhang, Ming Jin, and Cheng Ding. Cap: Towards ppg universal representation learning with patient-level supervision. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 11050–11060, 2026.

Yang He and Lingao Xiao. Structured pruning for deep convolutional neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(5):2900–2919, 2023.

Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 4335–4344. IEEE, 2019.

Yang He, Yuhang Ding, Ping Liu, Linchao Zhu, Hanwang Zhang, and Yi Yang. Learning filter pruning criteria for deep convolutional neural networks acceleration. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 2006–2015. IEEE, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Mohamad Kachuee, Mohammad Kiani, Hoda Mohammadzade, and Mahdi Shabany. Cuff-less blood pressure estimation. *UCI Machine Learning Repository*, 10:C5B602, 2015.

Kianoosh Kazemi, Iman Azimi, Pasi Liljeberg, and Amir M Rahmani. Respiration rate estimation via smartwatch-based photoplethysmography and accelerometer data: A transfer learning approach. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(1):1–24, 2025.

Andrei Nikolaevich Kolmogorov and Vladimir Mikhailovich Tikhomirov. ε-entropy and ε-capacity of sets in function spaces. *Uspekhi Matematicheskikh Nauk*, 14(2):3–86, 1959.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations*, 2017.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11256–11264. IEEE, 2019.

Girish Narayanswamy, Xin Liu, Kumar Ayush, Yuzhe Yang, Xuhai Xu, Shun Liao, Jake Garrison, Shyam Tailor, Jacob Sunshine, Yun Liu, et al. Scaling wearable foundation models. In *International Conference on Learning Representations*, volume 2025, pp. 59382–59413, 2025.

Andrea Nemcova, Enikö Vargova, Radovan Smisek, Lucie Marsanova, Lukas Smital, and Martin Vitek. Brno university of technology smartphone ppg database (but ppg): Annotated dataset for ppg quality assessment and heart rate estimation. *BioMed Research International*, 2021(1): 3453007, 2021.

Juntong Ni, Saurabh Kataria, Shengpu Tang, Carl Yang, Xiao Hu, and Wei Jin. PPG-distill: Efficient photoplethysmography signals analysis via foundation model distillation. In *NeurIPS 2025 Workshop on Learning from Time Series for Health*, 2025.

Guangkun Nie, Xiaocheng Fang, Gongzheng Tang, Yujie Xiao, Jun Li, Bo Liu, Hongyan Li, and Shenda Hong. Anyppg: An ecg-guided ppg foundation model trained on over 100,000 hours of recordings for holistic health profiling. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 11717–11727, 2026.

Mincheol Park, Dongjin Kim, Cheonjun Park, Yuna Park, Gyeong Eun Gong, Won Woo Ro, and Suhyun Kim. Reprune: Channel pruning via kernel representative selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 14545–14553, 2024.

Tania Pereira, Nate Tran, Kais Gadhoumi, Michele M Pelter, Duc H Do, Randall J Lee, Rene Colorado, Karl Meisel, and Xiao Hu. Photoplethysmography based atrial fibrillation detection: a review. *NPJ digital medicine*, 3(1):3, 2020.

Arvind Pillai, Dimitris Spathis, Fahim Kawsar, and Mohammad Malekzadeh. Papagei: Open foundation models for optical physiological signals. In *International Conference on Learning Representations*, volume 2025, pp. 48230–48261, 2025.

Attila Reiss, Ina Indlekofer, and Philip Schmidt. Ppg-dalia. *UCI Machine Learning Repository*, 10: C53890, 2019.

Mithun Saha, Maxwell A Xu, Wanting Mao, Sameer Neupane, James M Rehg, and Santosh Kumar. Pulse-ppg: An open-source field-trained ppg foundation model for wearable applications across lab and field settings. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 9(3):1–35, 2025.

Xavier Suau, Luca Zappella, Vinay Palakkode, and Nicholas Apostoloff. Principal filter analysis for guided network compression. *arXiv preprint arXiv:1807.10585*, 2, 2018.

Yang Sui, Miao Yin, Yi Xie, Huy Phan, Saman Aliari Zonouz, and Bo Yuan. Chip: Channel independence-based pruning for compact neural networks. *Advances in neural information processing systems*, 34:24604–24616, 2021.

Jessica Torres-Soto and Euan A Ashley. Multi-task deep learning for cardiac rhythm detection in wearable devices. *NPJ digital medicine*, 3(1):116, 2020.

Weinan Wang, Pedram Mohseni, Kevin L Kilgore, and Laleh Najafizadeh. Pulsedb: A large, cleaned dataset based on mimic-iii and vitaldb for benchmarking cuff-less blood pressure estimation methods. *Frontiers in Digital Health*, 4:1090854, 2023.

Zi Wang, Chengcheng Li, and Xiangyang Wang. Convolutional neural network pruning with structural redundancy reduction. In *CVPR*, pp. 14908–14917, 2021.

Wei Xiang, Jian Liu, Shuaicong Hu, Haihui Zhang, Chao Huang, and Cuiwei Yang. An ultra-efficient edge-based wearable system for real-time and remote blood pressure monitoring. *Engineering Applications of Artificial Intelligence*, 166:113603, 2026.

Hongwei Xie, Longpeng Yang, Binbin Jiang, Zhenlong Huang, and Yuan Lin. State-of-the-art wearable sensors for cardiovascular health: a review. *npj Cardiovascular Health*, 2(1):53, 2025.

Jianbo Ye, Xin Lu, Zhe Lin, and James Z. Wang. Rethinking the smaller-norm-less-informative assumption in channel pruning of convolution layers. In *Proceedings of International Conference on Learning Representations*, 2018.

Zexing Zhang, Huimin Lu, and Qingxin Zhao. Ppgpt: transferring next-token modeling from language to ppg signals. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28573–28581, 2026.

Lifan Zhao, Yanyan Shen, Zhaoyang Liu, Jiaji Deng, et al. Less is more: Unlocking specialization of time series foundation models via structured pruning. *Advances in Neural Information Processing Systems*, 38:171877–171902, 2026.

Zhuangwei Zhuang, Mingkui Tan, Bohan Zhuang, Jing Liu, Yong Guo, Qingyao Wu, Junzhou Huang, and Jinhui Zhu. Discrimination-aware channel pruning for deep neural networks. *Advances in neural information processing systems*, 31, 2018.