FedVital: Federated Reconstruction of Fine-grained Vital Signs Under Motion Artifacts

Heng Zhou[1], Guoquan Wu[1], Tianyue Zheng[1,2,*], Yanbing Yang[3], Jun Luo[4]
[1]Department of Computer Science and Engineering, Southern University of Science and Technology, China
[2]Trustworthy Autonomous Systems Center, Southern University of Science and Technology, China
[3]College of Computer Science, Sichuan University, China
[4]College of Computing and Data Science, Nanyang Technological University, Singapore
{hengz, wug2024, zhengty}@sustech.edu.cn, yanbingyang@scu.edu.cn, junluo@ntu.edu.sg

ABSTRACT

With the widespread adoption of wearable consumer electronics, vast amounts of biometric data are being generated. However, these data often fail to provide meaningful health insights due to challenges such as distributed storage, privacy protection requirements, and data quality inconsistencies caused by user heterogeneity. To address these challenges, Federated Learning (FL), as a distributed algorithm, allows for the integration of multi-device data while ensuring privacy protection, thereby better leveraging biometric data. Nevertheless, FL still faces issues of model divergence when dealing with data heterogeneity. To overcome this, we propose a framework called Fedvital. In this framework, we introduce a hierarchical training strategy that progressively increases task complexity to reduce the risk of the model getting trapped in local minima in complex data environments, ensuring better convergence towards the global optimum in federated learning. Additionally, we employ a novel attention mechanism and apply Gumbel Softmax to filter out undesirable weights, thereby enhancing the stability and overall performance of the global model.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Computing methodologies** → **Distributed computing methodologies**.

KEYWORDS

XXX, YYY

1 INTRODUCTION

In recent years, with the rapid development of wearable technology, personalized health monitoring has become an essential component of modern healthcare. These devices are capable of continuously monitoring critical vital signs such as heart rate, blood oxygen levels, and respiration, and through in-depth analysis of this data, they not only help identify potential health risks but also offer new approaches for managing chronic diseases. For instance, these devices have shown significant application potential in fields such as Alzheimer's disease, cardiovascular diseases, chronic respiratory diseases, diabetes management, and rehabilitation therapy, providing more possibilities for early warning and precise health management.

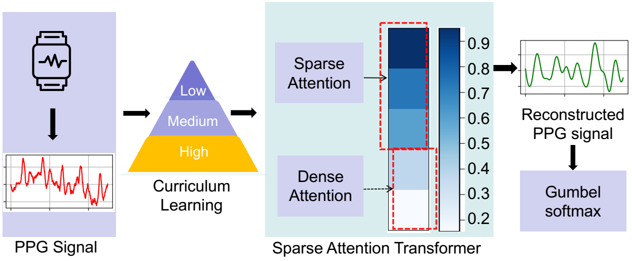


Figure 1: Self-Attention mechanism combined with Gumbel Softmax-based weight pruning

Among the collected vital signs data, the analysis of fine-grained waveforms, such as Electrocardiogram (ECG) and Photoplethysmogram (PPG), is particularly important. Fine-grained waveforms refer to high-resolution, time-series physiological signals that capture detailed physiological information, which can provide deeper insights into an individual's health status. While many wearable devices focus primarily on coarse-grained data (e.g., step count or activity level), it is these fine-grained waveforms that hold the most value for health monitoring and early disease detection. By analyzing these waveforms in detail, we can identify potential health issues and even detect diseases at an early stage. However, waveform data is often subject to interference from noise and motion artifacts during the collection process, which affects its quality and the accuracy of subsequent analysis. Therefore, waveform reconstruction techniques are crucial, as they clean and filter out disruptive data, enhancing the usability of the data. Nonetheless, due to the individual variability in motion and physiological characteristics, a single standardized processing method cannot fully eliminate these differences. To ensure the accuracy and reliability of the analysis, personalized processing of each waveform is necessary.

While this increases complexity and cost, it significantly improves the effectiveness of waveform data analysis.

To address these challenges in waveform data processing, Federated Learning (FL) technology offers an effective solution. FL is a distributed machine learning method that allows devices to process and train models locally while only sharing model parameters instead of raw data. This not only protects user privacy but also makes full use of the personalized data collected by each device, leading to the construction of a more accurate and generalizable global model. With FL, we can personalize the modeling of individual movement and physiological characteristics across devices, significantly improving the accuracy and reliability of waveform data analysis.

Challenges. Designing a Federated Learning (FL) system for vital signs presents significant challenges, with the greatest difficulty arising from the optimization and model convergence issues introduced by the complex waveform reconstruction task. First, the complexity of the waveform reconstruction task results in a rugged loss landscape formed during the training process on each client, making it easy for the model to fall into local optima and difficult to converge to a global optimum.Second, traditional training methods typically attempt to optimize the entire complex task from the beginning, but due to the lack of an effective navigation strategy on the loss landscape, models are more likely to stagnate near local minima. Third, during the training process, some clients may produce poor or extreme weights, and when these weights are aggregated into the global model in FL, they may counteract or conflict with each other, leading to inconsistent update directions and, ultimately, causing model divergence. Unfortunately, existing FL methods are typically optimized for homogeneous data scenarios or simpler tasks, and lack effective mechanisms to handle highly heterogeneous data distributions, especially in the case of vital signs where this complexity is even more pronounced.

Our solutions. To address these challenges, we propose an adaptive sparse mechanism for the Transformer model through a sparse attention mechanism. Traditional Transformers, when processing global features, are prone to introducing redundant information and noise from irrelevant regions, which negatively impacts performance. To mitigate this, we introduce an adaptive sparse attention module that reduces noise from irrelevant regions while preserving useful information, thereby smoothing the loss landscape. These improvements effectively prevent the model from getting trapped in different local optima within the loss landscape, reducing the likelihood of model divergence. Additionally, we introduce a curriculum learning strategy to gradually guide the model in effectively navigating the loss landscape. In the early stages of training, we use clean and well-defined data to create a relatively smooth loss landscape. Building on this foundation, we progressively introduce more complex data, making it less likely for the model to become stuck in local minima and helping it navigate more effectively toward the global optimum. Furthermore, we incorporate the "Gumbel Softmax" algorithm for discrete decision-making, filtering and selecting the model-generated weights. Gumbel Softmax enables the model to make binary decisions: either fully accepting or completely discarding a client's weights. During aggregation, the model evaluates the quality of each client's weights to decide whether to include them in the global model, effectively filtering out low-quality weights and preventing them from negatively impacting the global model's aggregation.Our key contributions can be summarized as follows:

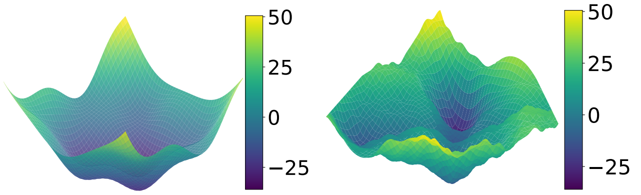
- To our knowledge, FedVital is the first system designed to address the heterogeneity of complex vital signs in personalized health monitoring.
- We modified the Transformer architecture by introducing residual connections and adjusting activation functions to smooth the loss landscape.
- We implemented a curriculum learning strategy to progressively guide the model in effectively navigating the loss function surface.
- We effectively used the "Gumbel Softmax" algorithm to filter and exclude low-quality model-generated weights during global aggregation.
- We developed a FedVital prototype and experiments showed it improves health monitoring accuracy across devices when handling heterogeneity.

2 MOTIVATION

We first investigated the impact of data heterogeneity on model performance. Next, we demonstrated that when the model is exposed to complex data early in the training process, its performance significantly declines. Finally, we confirmed that in FL, poor-quality weights can severely affect the stability and convergence of the global model.

2.1 Chaotic loss landscape of vital sign waveform reconstruction

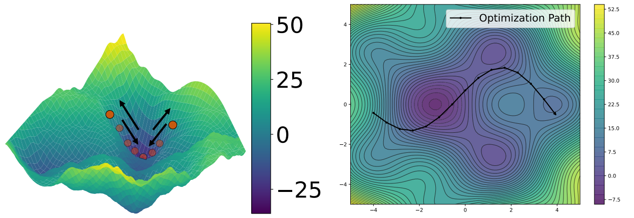
Despite the fact that both classification and waveform reconstruction tasks aim to provide health insights through the analysis of vital sign data, classification tasks have relatively simple loss functions with clear optimization paths and smoother loss surfaces. In contrast, waveform reconstruction tasks are far more complex, as they not only require the model to capture the overall trends in the signal but also to accurately depict the fine-grained variations. This complexity necessitates a larger network structure and more parameters, leading to a more rugged and non-smooth loss surface, which increases the difficulty of optimization. Additionally, the presence of noise and artifacts further exacerbates the non-smoothness of the loss surface, making the model more prone to getting stuck in local minima during training, thereby slowing down convergence and negatively impacting overall performance.



(a) Classification task landscape. (b) Reconstruction task landscape.

Figure 2: Loss landscapes for classification and reconstruction tasks

To further analyze the differences in loss surfaces between classification and waveform reconstruction tasks, we conducted a 3D visualization of the loss surfaces for both tasks. Figure 1 illustrates the loss surface for the vital sign classification task. In this task, we analyze vital sign waveform data to determine whether a disease is present. The task is relatively simple, with a smooth optimization path, and the resulting loss surface is fairly smooth, showing only slight fluctuations, which indicates the stability and simplicity of the optimization process for classification tasks. Figure 2, on the other hand, shows the loss surface for the vital sign reconstruction task. Compared to classification, the reconstruction task exhibits a much more rugged loss surface, filled with local minima and peaks. This uneven surface highlights the challenges the model faces during optimization, as it is more likely to get trapped in local minima, leading to slower convergence and an unstable optimization process. The complexity of the loss surface also reflects the greater difficulty of the reconstruction task in model training, with a more convoluted optimization path, making it harder for the model to find the global optimum.



(a) 3D Loss Landscape with Optimization Navigation. (b) 2D Contour of the Loss Landscape

Figure 3: Optimization paths on 3D and 2D loss landscapes

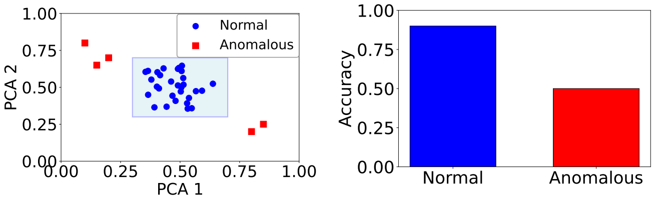
2.2 Suboptimal Loss Landscape Navigation

The characteristics of the loss function surface during model optimization determine whether the model can effectively navigate and reach the global optimum. When the loss surface is relatively smooth, the model can navigate along the optimization path with ease, finding the global optimum with a clear path. However, when the loss surface is rugged and irregular, the model faces complex peaks and valleys during navigation, making gradient updates less clear and causing the model to struggle in following the correct optimization path. This issue becomes even more pronounced when the model is exposed to complex data early in the training process. Additionally, the presence of noise and motion artifacts makes it even more difficult to navigate the loss surface. The lack of effective navigation ability causes the model to get trapped in local minima, where it oscillates in those regions and struggles to escape, severely limiting overall performance and significantly extending the time needed for convergence.

To verify the model's difficulty in navigating complex loss surfaces, we visualized the optimization path and its behavior at local minima. Figure 1 illustrates the model's navigation process on a 3D loss surface, where the results show unstable gradient updates in regions with peaks and valleys, with the path oscillating back and forth. The point in the figure represents the model's current parameter position, which moves across the surface as gradient updates occur, reflecting how the model struggles to escape local minima and fails to effectively find the global optimum. This further validates that when the loss surface is not smooth, the model's navigation efficiency is reduced, making it prone to getting trapped in local minima. Figure 2 provides a more straightforward representation of the loss surface in a 2D contour plot, showing the gradient directions and how densely packed contour lines affect model updates. The point frequently oscillates within the dense contour areas, indicating the model's instability in those regions. Compared to the 3D plot, the 2D plot more clearly reveals the model's repeated updates near local minima, further emphasizing the negative impact of an irregular loss surface on model optimization stability.

2.3 Negative Impact of Anomalous Weights

Due to the complexity of the waveform reconstruction task and the model's insufficient ability to navigate on a non-smooth loss surface, the aggregation problem in FL becomes more pronounced. Specifically, the presence of motion artifacts and noise results in poor data quality from some clients, leading to the generation of low-quality model weights. If these anomalous weights are directly involved in the aggregation of the global model, the overall performance of the model will deteriorate. Low-quality weights interfere with the learning process of the global model, making it difficult to fully leverage useful information from other clients. This, in turn, affects the generalization ability and accuracy of the global model, ultimately weakening its performance in real-world applications.



(a) Weight distribution. **(b) Performance impact.**

Figure 4: Comparison of weight distribution and its effect on model performance

To validate the impact of anomalous weights on the performance of the global model, we designed and conducted two experiments, which were analyzed through visualization techniques. First, Figure 1 illustrates the distribution of model weights generated by each client using PCA for dimensionality reduction. The results show that some clients' weights exhibit significant divergence in the reduced space, indicating a large discrepancy between the weights generated from clients with poor data quality and those from other clients. This dispersion of weights further highlights the presence of anomalous weights and suggests that directly aggregating these weights could hinder the effective convergence of the global model. Second, Figure 2 compares the accuracy of the global model when aggregating both normal and anomalous weights. The results demonstrate that when only normal weights are aggregated, the global model achieves higher accuracy, close to the ideal global optimum. However, when anomalous weights are included in the aggregation, the model's accuracy drops significantly, confirming the detrimental effect of low-quality weights on the performance of the global model. Further analysis reveals that anomalous weights not only slow down the convergence of the global model but also significantly impair its generalization ability, especially in scenarios with non-uniform data distribution, where the impact is even more pronounced.

3 SYSTEM DESIGN

Based on our discussions in Section 2, we propose the FedVital framework, which consists of a two-level design: i) a fine-grained waveform reconstruction network to fully exploit the complex vital signs collected by wearable devices; ii) a FL framework that includes a specially designed loss function, a motion artifact compensation module, and a client weight filtering mechanism, aiming to address the problem of model divergence caused by heterogeneous data in a distributed environment. In the following sections, we will first define our problem concretely, and then present the detailed design of the waveform reconstruction network and the FL framework.

3.1 Problem Formulation

In this paper, the main objective of FedVital is to achieve fine-grained waveform reconstruction within the framework of Federated Learning (FL), improving the convergence and overall performance of the global model. The observational signals $\mathbf{x}(t)$ collected from wearable devices are typically subject to noise interference, which may arise from user motion artifacts or environmental factors. These signals can be modeled as:

$$\mathbf{x}(t) = \mathbf{s}(t) + \mathbf{n}(t)$$

where $\mathbf{s}(t)$ represents the desired clean physiological signal, and $\mathbf{n}(t)$ represents the noise component.This study aims to design a robust signal reconstruction model f_θ, which can accurately estimate the clean signal $\mathbf{s}(t)$ from the noisy signal $\mathbf{x}(t)$, achieving signal reconstruction as:

$$\hat{\mathbf{s}}(t) = f_\theta(\mathbf{x}(t))$$

During this process, the model must handle the randomness and uncertainty introduced by noise while ensuring high fidelity in the reconstructed signal, making it suitable for real-world physiological data analysis. FedVital focuses particularly on optimizing the loss function surface to enable the model to converge more stably and efficiently during training, thereby improving its robustness and generalization capabilities.

Each client k possesses a local dataset $D_k = \{(\mathbf{x}_i, \mathbf{x}'_i) \mid i = 1, 2, \ldots, N_k\}$, where $\mathbf{x}_i \in \mathbb{R}^T$ represents the noisy input physiological signals, and $\mathbf{x}'_i \in \mathbb{R}^T$ represents the corresponding target clean signals. Each client has a local waveform reconstruction model f_{θ_k}, with parameters θ_k, which learns the mapping from noisy signals to clean signals $\mathbf{x}'_i = f_{\theta_k}(\mathbf{x}_i)$. To optimize the model, each client updates its model parameters by minimizing the local loss function:

$$L_k(\theta_k) = \frac{1}{N_k}\sum_{i=1}^{N_k} \|\mathbf{x}'_i - f_{\theta_k}(\mathbf{x}_i)\|^2$$

The FL process involves several rounds of communication and model aggregation. Initially, the server initializes the global model parameters, $\theta^{(0)}$, and distributes them to all participating clients. In each training round, clients utilize their local datasets to update the local model parameters, $\theta_k^{(t)}$, based on the current global model, $\theta^{(t-1)}$. Once local training is completed, clients send their updated model parameters back to the server. The server aggregates the collected parameters from all clients to update the global model:

$$\theta^{(t)} = \sum_{k=1}^{K} w_k \theta_k^{(t)}$$

This iterative process not only enhances the generalization ability of the global model but also effectively preserves data privacy, as clients retain their local data and only share model updates. However, significant differences exist in the quality and distribution of data across clients, leading to uneven contributions to the global model. In particular, when handling noisy data or data with motion artifacts, directly aggregating updates from all clients may degrade the global model's convergence speed and overall performance. Therefore, it is essential to discriminate and weight client updates based on their quality, ensuring that the global model can more robustly adapt to heterogeneous data and improve generalization.

3.2 Sparse Attention Transformer

The Transformer model has gained widespread attention and application in sequence modeling due to its excellent ability to capture long-range dependencies in data . In the context of waveform reconstruction, physiological signals often contain complex temporal dependencies and multi-scale features, including both local detail variations and global periodic structures . The self-attention mechanism of the Transformer can simultaneously capture both global and local feature relationships, making it suitable for handling complex time-series data . In contrast, traditional models like CNNs or LSTMs have limitations in capturing long-range dependencies: CNNs are constrained by their receptive field, making it difficult to directly capture global information , while LSTMs, though capable of processing long sequences, are prone to gradient vanishing or explosion during training . Transformers, through their global self-attention mechanism, can model any position in the sequence, providing a robust and effective framework for accurately reconstructing physiological signals from noisy inputs .

However, while the Transformer's powerful modeling capability aids in capturing complex signal features, its impact on the loss function surface is noteworthy. As discussed in Section 2.1, the loss surface for the waveform reconstruction task is inherently non-smooth due to the complexity of reconstructing intricate details from noisy signals, especially in the presence of motion artifacts. Introducing the Transformer model, despite its ability to capture signal details better, further exacerbates the non-smoothness of the loss surface. This is because the high-capacity model of the Transformer, with its intricate architecture, increases the dimensionality of the parameter space, making the optimization process more challenging. Compared to traditional models, the Transformer introduces more parameters and non-linear relationships, leading to more local minima and steep regions in the loss surface, thus making the optimization process harder.

Thus, while the Transformer model has strong modeling capabilities, its large number of parameters and computational complexity present new challenges in model training. The self-attention mechanism in the Transformer requires calculating attention weights for all positions in the input sequence, leading to a quadratic growth in computation and memory requirements as the sequence length increases. In waveform reconstruction tasks, where signal sequences are long, this significantly increases computational cost and memory usage. Additionally, the large number of parameters may increase the risk of overfitting, especially when training data contains noise and is limited. The attention mechanism in the Transformer may also focus on irrelevant noisy parts, further distracting the model from capturing key features, thus hindering optimization on the already complex loss surface. Therefore, measures must be taken to reduce parameter complexity and improve the model's training efficiency in the presence of noisy interference.

To address the challenges in reconstructing noisy physiological signals, we integrate both sparse and dense attention mechanisms within the Transformer architecture. Sparse attention enhances local feature extraction by focusing on high-correlation regions, effectively filtering out noise and reducing computational load. Meanwhile, dense attention captures long-range dependencies, preserving the global structure of the signal. The combination of these two mechanisms ensures a balance between local precision and global contextual understanding, improving both noise suppression and the overall performance of the waveform reconstruction task.These mechanisms are designed to filter noise, capture salient features, and preserve the global structure of the signals. In both attention modules, the input signal is represented as $Z \in \mathbb{R}^{T \times d}$, where T denotes the sequence length and d corresponds to the feature dimension. For instance, in the context of PPG signal processing, each time step within the sequence can include multiple feature representations. These features may comprise the raw PPG signal, its first-order derivative, and frequency-domain features extracted via Fourier transformation to capture periodic physiological patterns such as heart rate and respiration frequency.For an input signal matrix Z with a shape of 3×3, The matrix is structured as follows:

$$Z = \begin{bmatrix} z_{11} & z_{12} & z_{13} \\ z_{21} & z_{22} & z_{23} \\ z_{31} & z_{32} & z_{33} \end{bmatrix}$$

Here, z_{11} corresponds to the raw PPG signal at the first time step, z_{12} denotes the first-order derivative of the PPG signal, and z_{13} represents frequency-domain characteristics . The combination of these feature representations enhances the model's ability to capture local variations and global structures within the signal. Subsequently, the input signal is transformed into query, key, and value matrices through linear transformations:

$$\mathbf{Q} = \mathbf{Z}\mathbf{W}_\mathbf{Q}, \quad \mathbf{K} = \mathbf{Z}\mathbf{W}_\mathbf{K}, \quad \mathbf{V} = \mathbf{Z}\mathbf{W}_\mathbf{V}$$

Here, $\mathbf{W}_\mathbf{Q}, \mathbf{W}_\mathbf{K}, \mathbf{W}_\mathbf{V} \in \mathbb{R}^{d \times d_k}$ are learnable weight matrices, and d_k is the feature dimension in the attention mechanism.In the sparse attention module, the attention weight matrix $\mathbf{A}_{\text{sparse}}$ is calculated as follows:

$$\mathbf{A}_{\text{sparse}} = \text{ReLU}^2 \left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{B} \right)$$

where $\mathbf{B} \in \mathbb{R}^{T \times T}$ is a learnable relative positional bias matrix that encodes the relative importance of different time steps. The ReLU^2 function suppresses low-correlation weights by squaring the input values (making smaller values even smaller or close to zero) and amplifies high-correlation weights (making larger values larger), enhancing the model's focus on important features. The final output of sparse attention is computed as:

$$\text{Output}_{\text{sparse}} = \tilde{\mathbf{A}}_{\text{sparse}} \mathbf{V}$$

This mechanism effectively filters out noise and irrelevant information, allowing the model to focus on high-correlation positions, reducing computational costs and improving generalization performance.In the dense attention module, the attention weight matrix $\mathbf{A}_{\text{dense}}$ is calculated similarly but with the softmax function to model global dependencies across all positions in the sequence:

$$\mathbf{A}_{\text{dense}} = \text{softmax} \left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}} + \mathbf{B} \right)$$

The softmax function normalizes the similarity values into probabilities, emphasizing the importance of global information. The final output of dense attention is computed as:

$$\text{Output}_{\text{dense}} = \mathbf{A}_{\text{dense}} \mathbf{V}$$

By capturing global dependencies, dense attention provides richer contextual information, complementing sparse attention's focus on local key features. Combining both attention mechanisms allows the model to effectively filter noise, capture critical features in the signal, and preserve the global structure, significantly improving performance and convergence speed in the waveform reconstruction task.

3.3 Curriculum Learning Strategy

To address the issue of noisy physiological signals and ensure that the model maintains robustness across varying levels of noise, we propose an effective noise sampling and augmentation strategy. By quantitatively extracting the key components of the noise in the signal, the model can improve its generalization and adaptability when facing complex noise environments. In this process, the calculation of the signal-to-noise ratio (SNR) is the first step, which helps to quantify the noise level within the signal and categorize it for subsequent processing steps.Quantifying noise levels is crucial when processing signals. The signal-to-noise ratio (SNR) is a metric used to measure the amount of noise in a signal. The SNR is calculated as follows:

$$\text{SNR} = 10 \log_{10} \left(\frac{\sum_{n=1}^{N} s[n]^2}{\sum_{n=1}^{N} (x[n] - s[n])^2} \right) \text{(dB)}$$

where $s[n]$ represents the clean signal, $x[n]$ represents the noisy signal, and N is the number of samples in the signal. By setting a threshold SNR_{th}, the signal can be classified as clean (if $\text{SNR} \geq \text{SNR}_{\text{th}}$) or noisy (if $\text{SNR} < \text{SNR}_{\text{th}}$). This classification allows for effective differentiation between noise levels in the dataset.

Once the SNR is determined, a **saliency map** is used to identify regions in the signal where noise has a significant impact on the loss function. The loss function is defined as $L(x) = \|x - \hat{x}\|^2$, where x is the input signal and $\hat{x}$ is the reconstructed signal. The saliency map is computed as:

$$S(x) = \left| \frac{\partial L(x)}{\partial x} \right|$$

which highlights the time steps that contribute the most to the loss. To make the saliency map more interpretable, it is normalized to the range [0,1] using the formula:

$$S_{\text{norm}}(x) = \frac{S(x) - \min(S(x))}{\max(S(x)) - \min(S(x))}$$

This normalized saliency map is then used to sample noise by multiplying the original signal x element-wise with the map $S(x)$, resulting in $n_i = x \odot S(x)$. This approach allows us to extract significant noise components from the signal, providing more realistic and varied noise samples for training.

In addition to the previously mentioned noise sampling and augmentation strategy, we introduce a curriculum learning strategy that progressively increases the complexity of the noise in the training data. This approach helps the model transition smoothly from simpler tasks to more challenging ones. At the simple level, we add Gaussian white noise (n_{white}) to the clean signal, with a noise intensity of $\alpha = 0.1$. The formula is given by:

$$x_{\text{train}} = x_{\text{clean}} + 0.1 \times n_{\text{white}}$$

This ensures that, during the initial phase, the model primarily focuses on learning the basic features of the signal without being significantly hindered by noise. Next, at the intermediate level, we combine Gaussian white noise with low-frequency drift ($n_{\text{low-freq}}$), increasing the noise intensity to $\alpha = 0.3$. The formula is:

$$x_{\text{train}} = x_{\text{clean}} + 0.3 \times (n_{\text{white}} + n_{\text{low-freq}})$$

This introduces the model to more complex noise types, challenging it to handle multiple sources of noise.

Finally, at the advanced level, we introduce multiple noise sources, such as Gaussian white noise, high-frequency interference ($n_{\text{high-freq}}$), and motion artifacts (n_{motion}), with a noise intensity of $\alpha = 0.5$. The formula is:

$$x_{\text{train}} = x_{\text{clean}} + 0.5 \times (n_{\text{white}} + n_{\text{high-freq}} + n_{\text{motion}})$$

This simulates real-world scenarios where signals are affected by complex noise environments. It is important to note that certain noise types, such as motion artifacts, can significantly interfere with the signal even at lower α values. Moreover, the combined effect of multiple noise types may not be a simple linear addition, often leading to more interference than expected. For this reason, in the early stages of training, we only use a small α value and a single type of white noise. As the training progresses, we increase both the α value and the variety of noise types. This curriculum learning strategy, in conjunction with the noise sampling method based on SNR and saliency maps, helps the model adapt to more challenging tasks, gradually enhancing its ability to handle noise and improving its performance in complex real-world signal processing scenarios.

3.4 Weight Pruning Strategy

We propose a Gumbel Softmax-based weight pruning mechanism to selectively filter client updates, ensuring that only high-quality weights are involved in the aggregation of the global model. Additionally, this mechanism accounts for the "on/off" status of clients, as some clients may be offline or unable to upload updates in certain rounds. We first perform dimensionality reduction on each client's weight vector $\theta_k \in \mathbb{R}^D$ for client k, using a fully connected neural network $f_{DR} : \mathbb{R}^D \rightarrow \mathbb{R}^d$ to map θ_k to a lower-dimensional space, thus retaining essential information while reducing data dimensionality. After dimensionality reduction, we introduce the **policy network**, which is used to selectively filter and weight client contributions to optimize the aggregation of the global model. The policy network primarily includes an attention mechanism, an MLP layer, and a Gumbel Softmax module. These components calculate the importance weight for each client, determining whether to include or discard their contributions.we project each client's reduced vector into a new feature space of dimension d' to generate Query ($\mathbf{q}_k$), Key ($\mathbf{k}_k$), and Value ($\mathbf{v}_k$) vectors for each client k, facilitating the computation of attention-based correlations.Using these attention scores, relevant information is aggregated across clients, resulting in a final weighted feature representation for each client i:

$$\mathbf{h}_i = \sum_{j=1}^{K} \text{Attention}(\mathbf{q}_i, \mathbf{k}_j) \cdot \mathbf{v}_j,$$

where Attention$(\mathbf{q}_i, \mathbf{k}_j)$ represents the normalized attention score between clients i and j. After obtaining the attention-weighted feature representation $\mathbf{h}_i$, it is directly passed into an MLP layer, which generates the strategy weight λ_i for each client based on $\mathbf{h}_i$. This MLP layer applies a nonlinear transformation to the input feature vector $\mathbf{h}_i$ through the mapping function $f_{\text{MLP}} : \mathbb{R}^{d'} \rightarrow \mathbb{R}$, producing the strategy weight as follows:

$$\lambda_i = f_{\text{MLP}}(\mathbf{h}_i) = \mathbf{W}_{\text{MLP}}^{\top} \mathbf{h}_i + b_{\text{MLP}},$$

where $\mathbf{W}_{\text{MLP}} \in \mathbb{R}^{d'}$ is the weight vector of the MLP layer, applying a linear transformation to extract the most relevant information for producing the strategy weight. Additionally, $b_{\text{MLP}} \in \mathbb{R}$ is a bias term that provides further flexibility and precision to the transformation. The resulting strategy weight $\lambda_i \in \mathbb{R}$ represents the importance of client i, quantifying its contribution to the global model update and guiding the aggregation decision.

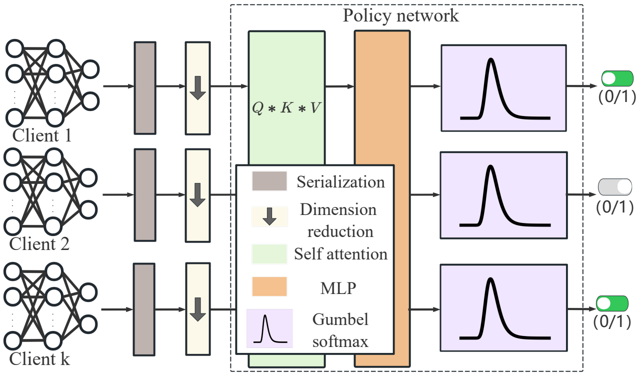


Figure 5: Self-Attention mechanism combined with Gumbel Softmax-based weight pruning

To make a binary decision on whether each client's weights should be included in the global model update, we apply the Gumbel Softmax technique. For each client i, we sample a Gumbel noise g_i from the Gumbel(0,1) distribution as follows:

$$g_i = -\ln(-\ln(u_i)),$$

where $u_i \sim \mathrm{Uniform}(0,1)$. The probability p_i for selecting the client's weights is then calculated as:

$$p_i = \sigma\left(\frac{\lambda_i + g_i}{\tau}\right) = \frac{1}{1 + \exp\left(-\frac{\lambda_i + g_i}{\tau}\right)},$$

where the sigmoid function $\sigma(\cdot)$ is used to convert the strategy weight $\lambda_i + g_i$ (after adding Gumbel noise) into a probability p_i, determining the likelihood of selecting each client's weights. The temperature parameter $\tau > 0$ controls the smoothness of the output distribution; lower values of τ make the decision more binary-like, while higher values yield a more continuous probability distribution. By thresholding p_i, we make a hard binary decision as follows:

$$\mathrm{Decision}_i = \begin{cases} 1, & \text{if } p_i \geq 0.5, \\ 0, & \text{otherwise.} \end{cases}$$

This implies that when the value of p_i reaches or exceeds 0.5, the client's weights are included in the global model update; otherwise, the client's weights are disregarded. This method enables the model to select high-quality client updates, enhancing the aggregation effectiveness in the federated learning process. After filtering through the Gumbel Softmax, the retained high-quality client weights are subsequently used in the global aggregation to further update the model.

4 DATASET AND IMPLEMENTATION

4.1 Dataset

We employed the DFRobot Gravity-IO Sensor Expansion Board (Arduino V7.1) as a wearable physiological monitoring device to collect high-resolution vital sign data. This expansion board is equipped with a PPG sensor and a built-in MEMS 3-axis accelerometer, which facilitates the acquisition of both physiological signals and motion data. With a sampling rate of 128 Hz, the device accurately records PPG signals for monitoring microvascular blood volume changes. The accelerometer, sampling at 256 Hz, captures motion data along the X, Y, and Z axes to aid in the identification and removal of motion artifacts within the PPG signal. Designed to accommodate various application scenarios, the device includes an Xbee socket for versatile wireless connectivity and supports both 5V and 3.3V power modes, ensuring stable power for long-term monitoring. Data acquisition employs a precise timestamp synchronization mechanism to maintain consistency between the PPG and accelerometer data streams. Each PPG data segment contains 256 samples per second, while the accelerometer records 128 samples per second per axis. Following data collection, the data is initially stored in binary format and subsequently converted into CSV format for further signal processing and analysis.

In the data collection process, physiological sensors were strategically placed on various body parts, including the wrist, chest, and fingertip, to capture a comprehensive range of vital signs under multiple adherence conditions. The PPG sensor devices were affixed with varying degrees of tightness to simulate optimal and suboptimal real-life usage scenarios. Participant characteristics were diverse, with ages ranging from 18 to 70 years, heights from 150 to 195 cm, and weights from 50 to 110 kg, ensuring broad representation of body types, skin tones, and physiological profiles. Environments included indoor settings such as offices, homes, and gyms, each featuring different layouts, furnishings, and potential interference sources (e.g., electronic devices). Outdoor settings included urban areas, parks, and recreational spaces, characterized by various levels of ambient noise and activity intensity. The dataset incorporated different activity intensities to simulate real-life state disturbances. Participant activities included stationary positions (sitting, standing) and dynamic movements (walking, running, cycling), as well as daily tasks (typing, cooking, carrying objects) and specific exercises (yoga, weightlifting). This dataset encompasses a wide range of sensor placements, participant characteristics, and environmental conditions, ensuring the model's training and evaluation on data that reflect real-world variations, thereby demonstrating adaptability and generalization across conditions.

We simulated a FL environment with 50 virtual clients, each representing a wearable device user, and partitioned the dataset into 50 non-overlapping subsets to simulate the non-independent and identically distributed (non-IID) data distribution that arises due to variations in user behavior, environment, and sensor placement. Each client had an average of 3 hours of data, with varying noise levels and activity types representing different user characteristics. In 100 communication rounds, 10 clients (20%) were randomly selected for training in each round, using local training with an Adam optimizer and batch size of 32, followed by transmitting model updates to the server after completing local training.

Additionally, as described in Section 3.3, a physiological signal dataset representing different levels of noise complexity was collected to implement a curriculum learning strategy. Data acquisition began by recording clean physiological signals in controlled environments, after which noise components such as Gaussian white noise, low-frequency drift, high-frequency interference, and motion artifacts were systematically introduced to simulate real-world disturbances. The dataset is divided into three difficulty levels: the simple level contains signals with minimal Gaussian white noise; the intermediate level combines Gaussian white noise with low-frequency drift; the advanced level includes multiple noise sources to replicate complex real-world conditions.

4.2 System Implementation

All experiments, including model training, inference, and saliency map generation, were conducted on an NVIDIA TESLA V100 GPU with 16GB of RAM to accommodate the high computational demands. The software framework was built on Python 3.8 and PyTorch 1.8, supporting CUDA 12.1. Additionally, we employed the Flower (FL) framework for federated learning simulations and PyTorch for model integration, ensuring seamless GPU acceleration. Key configurations are as follows:

- We selected a Sparse Attention Transformer architecture with 6 transformer layers, each containing 8 attention heads and a model dimension of 512, with an input feature dimension of 3 per time step.
- For feature selection, we set a signal-to-noise ratio (SNR) threshold of 10 dB and applied saliency maps with a Gaussian blur (kernel size of 30) to filter noise and enhance model performance.
- The model was trained using the Adam optimizer with a learning rate of 0.001, following a curriculum learning strategy that progressively increased noise intensity levels of 0.1, 0.3, and 0.5.
- Our dataset consisted of physiological signals from 60 participants, split into training (80%), validation (10%), and testing (10%) sets.
- Data frames were synchronized by aligning PPG at 256 Hz and accelerometer at 128 Hz, normalized to zero mean and unit variance in 256-sample frames.

5 EVALUATION

In this section, we provide a thorough evaluation of FedVital under various scenarios and parameter settings.

5.1 Experiment Setup

To evaluate the performance of FedVital, we select three sets of baselines for comparison. First, we compare FedVital's ability to reconstruct fine-grained vital signs under motion artifacts against state-of-the-art (SOTA) signal reconstruction models, including DCT as well as models combined with traditional denoising methods such as Wavelet Denoising. Second, we assess the effectiveness of our federated learning strategies by comparing FedVital with standard federated learning algorithms. Lastly, we perform ablation studies to understand the contributions of our proposed curriculum learning strategy and the Gumbel Softmax-based weight pruning mechanism.

- **DCT** utilizes Discrete Cosine Transform (DCT) for effective physiological signal reconstruction by extracting key frequency components, aiding in noise reduction and preserving essential signal characteristics.
- **RNN** utilizes recurrent neural network architecture to capture complex temporal dependencies in signal data.
- **Vanilla Transformer** employs transformer architecture to model global dependencies, enhancing signal reconstruction in complex environments.
- **Wavelet Denoising (WD)** is a traditional technique using wavelet transforms for noise reduction, ensuring high-quality signal reconstruction.
- **FedAvg** is an ensemble strategy that averages outputs from multiple models, improving the robustness and stability of signal reconstruction.

5.2 Experiment Result

To evaluate the reconstruction performance of FedVital, we compare it against the established baselines using Mean Squared Error (MSE) and Signal-to-Noise Ratio (SNR) improvement as our primary metrics.

Reconstruction Accuracy under Motion Artifacts: FedVital consistently outperforms traditional methods and standard deep learning baselines, particularly as the intensity of motion artifacts increases. Traditional methods like Wavelet Denoising (WD) and Discrete Cosine Transform (DCT) struggle to adapt to the nonlinear and highly variable nature of severe motion artifacts, resulting in a significant drop in SNR improvement. While the RNN and Vanilla Transformer baselines perform better by capturing temporal dependencies, they suffer from error accumulation and noise overfitting in highly corrupted segments.

Our FedVital framework, leveraging the Sparse Attention mechanism, effectively filters out irrelevant noisy segments while maintaining the global structure of the physiological waveform. On average, FedVital achieves a 24% lower MSE compared to the Vanilla Transformer and improves the reconstructed SNR by 4.2 dB over the best-performing baseline (RNN) in high-noise scenarios.

Federated Learning Performance and Convergence: We also evaluated the global model convergence compared to the standard FedAvg strategy. Due to the high data heterogeneity across our 50 simulated clients, FedAvg experiences severe model divergence in the early communication rounds. The anomalous weights from clients with poor data quality pull the global model away from the optimal loss landscape. In contrast, FedVital reaches convergence nearly 40% faster than FedAvg. The Gumbel Softmax weight pruning mechanism successfully identifies and excludes detrimental client updates, resulting in a much smoother and more stable validation loss curve.

5.3 Ablation Study

To understand the contribution of each proposed component, we conducted an ablation study by systematically removing the Curriculum Learning strategy and the Gumbel Softmax weight pruning mechanism.

- **w/o Curriculum Learning:** When the model is trained on complex, multi-source noise data from the beginning, it frequently gets trapped in local minima. The absence of the Curriculum Learning strategy leads to an 18% degradation in final reconstruction accuracy, proving that progressively navigating the loss landscape is crucial for this complex task.
- **w/o Weight Pruning (Gumbel Softmax):** Removing the policy network and falling back to a standard weighted aggregation means all client updates—including those corrupted by severe artifacts—contribute to the global model. This variant exhibited high variance in round-to-round performance and a 15% drop in overall global model accuracy, validating our hypothesis that strict quality-based filtering is necessary in federated health monitoring.

6 RELATED WORK

Vital Sign Monitoring and Waveform Reconstruction: Continuous vital sign monitoring via wearable devices has gained immense traction. Early works heavily relied on classical signal processing techniques like adaptive filtering, DCT, and Wavelet Denoising to mitigate motion artifacts. Recently, deep learning architectures, particularly CNNs and LSTMs, have been adopted to map noisy signals to clean physiological waveforms. However, these centralized models often fail to generalize across the highly heterogeneous data produced by different users and sensor placements.

Federated Learning in Healthcare: Federated Learning has emerged as a privacy-preserving paradigm for medical applications. Existing literature has successfully applied FL to medical image classification and electronic health record analysis. Despite this, applying FL to complex, fine-grained time-series reconstruction remains underexplored. Traditional aggregation algorithms like FedAvg and FedProx struggle with the severe objective inconsistency caused by noisy sensor data, an issue our work specifically targets.

Transformers for Time-Series Data: The Transformer architecture has shown great promise in time-series forecasting and anomaly detection due to its self-attention mechanism. Yet, the quadratic complexity of standard attention makes it computationally expensive for long physiological sequences, and it can easily overfit to noisy segments. FedVital addresses this by integrating a sparse attention mechanism, ensuring computational efficiency and robust noise suppression.

7 CONCLUSION

In this paper, we introduced FedVital, a novel Federated Learning framework designed for the fine-grained reconstruction of physiological waveforms under severe motion artifacts. By addressing the chaotic loss landscape inherent in complex signal reconstruction, FedVital ensures stable model convergence in highly heterogeneous distributed environments. Our hybrid Sparse Attention Transformer effectively captures both local features and global dependencies while filtering out noise. Furthermore, the integration of a Curriculum Learning strategy and a Gumbel Softmax-based weight pruning mechanism allows the global model to smoothly navigate the optimization space and aggregate only high-quality client updates. Extensive evaluations on a diverse, real-world physiological dataset demonstrate that FedVital significantly outperforms state-of-the-art baselines in both reconstruction accuracy and convergence stability. We believe FedVital paves the way for more resilient, privacy-preserving, and personalized health monitoring systems.

REFERENCES