Peak-R1: Instruction-Tuned Large Language Models for Robust J-Peak Detection in Cardiomechanical Signals

Anonymous Author(s)
Affiliation
Address
email

Abstract

Cardiomechanical signals, encompassing ballistocardiography (BCG) and the bodyseismogram (BSG), represent a promising modality for unobtrusive and continuous assessment of cardiovascular health. The J-peak, a key fiducial point within the cardiomechanical signal, serves as a robust surrogate for cardiac timing, underpinning heart rate (HR) estimation and hemodynamic modeling. However, precise J-peak localization is frequently confounded by annotation ambiguities, inter-subject signal variability, and motion artifacts. We introduce **Peak-R1**, a novel framework that leverages an instruction-tuned Large Language Model (LLM) for robust J-peak detection. Central to our approach is a peak-extraction front end that transforms raw BCG segments into compact peak sequences. This peak-centric representation reduces noise and introduces a principled inductive bias, guiding the LLM to focus on physiologically meaningful events and thereby improving its reasoning over time-series data. Peak-R1 is trained via a two-stage strategy: (i) supervised fine-tuning (SFT) to establish stable output formatting and baseline signal interpretation, followed by (ii) reinforcement learning (RL) with Group Relative Policy Optimization (GRPO). The RL stage employs a multi-objective reward function to jointly optimize for output validity, HR consistency, absolute localization accuracy, and detection completeness. The framework achieves an F1 score of 0.930 and HR mean absolute error (MAE) of 0.399 BPM on the Kansas dataset, while maintaining robust performance (F1: 0.770, HR MAE: 7.002 BPM) on the more challenging hospital-BSG dataset. Our ablation studies confirm the necessity of the peak-extraction front end and reveal that RL tuning is critical for improving detection accuracy.

1 Introduction

Cardiomechanical signals, including ballistocardiography (BCG) and body seismography (BSG), enable unobtrusive cardiovascular monitoring by capturing mechanical responses to cardiac ejection [1, 2, 3]. The J-peak, a key fiducial point within these signals, serves as a reliable surrogate for cardiac timing and forms the foundation for heart rate estimation and hemodynamic modeling [4, 5]. However, precise J-peak localization in real-world data remains challenging due to annotation ambiguities, inter-subject variability, motion artifacts, and sensor heterogeneity [6, 7, 8].

Traditional signal processing methods, while computationally efficient, are fundamentally limited by their reliance on hand-crafted features, constraining their robustness in noisy environments [9]. Deep learning approaches improve sensitivity but typically employ point-wise binary classification, which suffers from severe class imbalance and lacks precise temporal localization [8].

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

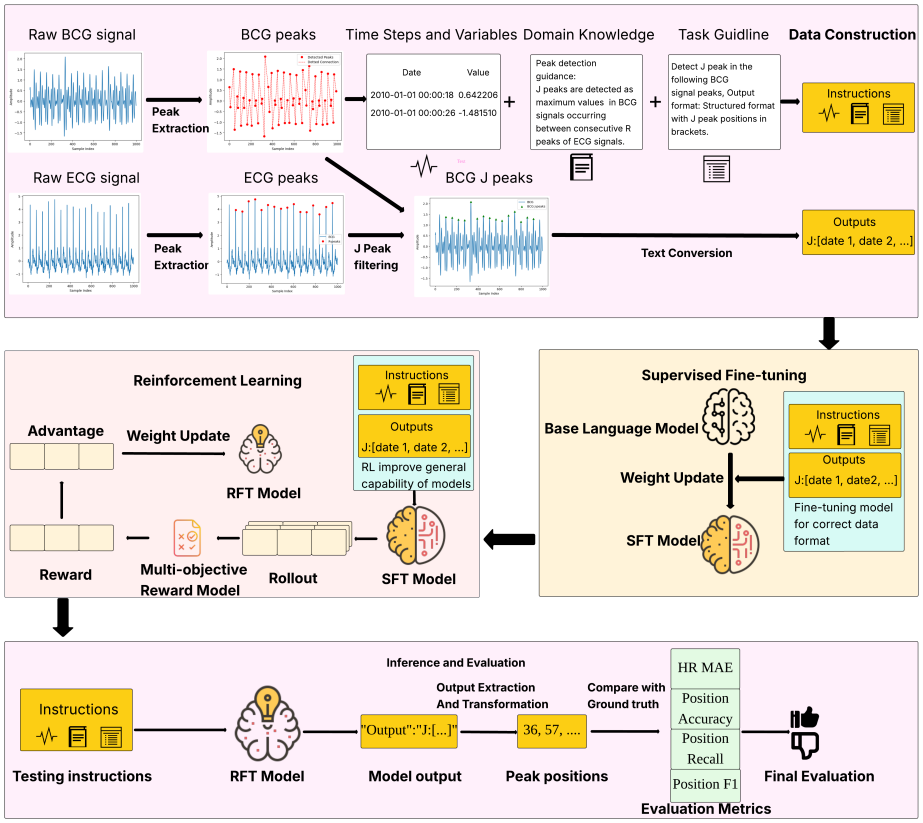


Figure 1: The Peak-R1 framework processes raw BCG signals by first extracting peaks and representing them as timestamp:value pairs. These sequences are formatted into a prompt for a Large Language Model (LLM), which includes domain knowledge and task instructions. The LLM is then trained in two stages: first with supervised fine-tuning to ensure correctly formatted output, and second with reinforcement learning to maximize the robustness of J-peak detection.

While instruction-tuned large language models (LLMs) have shown promise for structured biomedical data [10, 11, 12], their application to high-sampling-rate physiological waveforms is hindered by computationally prohibitive sequence lengths exceeding 1000 time-steps [13]. We hypothesize that an appropriate intermediate representation can unlock LLM potential for robust J-peak detection.

We introduce Peak-R1, which reformulates cardiomechanical segments into compact peak sequences and employs two-stage instruction tuning: supervised fine-tuning (SFT) for schema compliance, followed by reinforcement learning with Group Relative Policy Optimization (GRPO) using multi-objective rewards for format validity, heart-rate consistency, positional accuracy, and detection completeness.

Our contributions are:

- We develop a novel way to represent cardiomechanical signal data that allows large language models to effectively analyze physiological time series.
- We employ a two-stage training approach combining supervised learning and reinforcement learning to optimize objectives that traditional methods cannot handle.
- We achieve best-in-class results on two datasets: Kansas (F1=0.930, HR error=0.399 BPM) and Hospital-BSG (F1=0.779, HR error=7.002 BPM), with experiments confirming each component's importance.

2 Methods

Representation and Training Template. To make BCG signals tractable for LLM processing, we introduce a peak-extraction method that converts each raw segment into a compact, symbolic sequence. The process involves band-pass filtering, segmentation, and robust per-segment normalization, followed by a local-extrema search to identify all candidate positive and negative peaks. This strategy of retaining both positive and negative extrema is critical for maintaining high morphological fidelity while drastically reducing the input sequence length. The retained peaks are then serialized into `[Date:value]` tokens in temporal order, where the `[Date]` is a synthetic calendar timestamp. This proxy time encoding was chosen to leverage the observed proficiency of LLMs with calendrical reasoning [13]. The training template includes: (i) Instructions: including Time steps and variables, domain knowledge, and task guidline. The time steps and variables are delimited by `<TS_START>` and `<TS_END>` from other text (ii) Output: an explicit output schema, `J:[Date1, Date2, ...]`. It is noticed that Ground-truth J-peaks are defined as the maximum BCG peak between consecutive ECG R-peaks, consistent with standard definitions [14, 15, 5]. An example of the template is shown in Appendix H.

Supervised fine-tuning (SFT). We warm-start Qwen2.5-Instruct-3B [16] as the base model, optimizing it to produce strictly valid outputs and learn a mapping from candidate peaks to J-peaks. Given a prompt q and target sequence $y^\star$ encoding `J:[...]`, we minimize token-level cross-entropy:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(q,y^\star)} \sum_t \log \pi_\theta(y_t^\star \mid q, y_{<t}^\star)\,.$$

Reinforcement learning with GRPO. Starting from the SFT model, we optimize a multi-objective reward aligned with task metrics:

$$R(q,o) = \lambda_1\,\gamma_{\text{format}} + \lambda_2\,\gamma_{\text{HR}} + \lambda_3\,\gamma_{\text{pos}} + \lambda_4\,\gamma_{\text{cnt}}.$$

γ_{format} awards syntactically valid `J:[...]` outputs. γ_{HR} rewards agreement between predicted and reference HR. γ_{pos} compute F1 score from one-to-one matching under a fixed tolerance τ . γ_{cnt} encourages correct beat counts with mild penalties for under/over-detection. We use Group Relative Policy Optimization (GRPO): for each prompt, we sample a group of outputs, compute group-mean rewards as a baseline to form sequence-level advantages, and optimize a clipped policy objective with a KL penalty to the SFT policy to stabilize language fidelity. Sampling temperature/top-p, group size, clipping, KL weight, and λ weights are tuned on validation.

Matching and metrics. Predicted times are matched to the ground truth via a one-to-one, minimum-cost bipartite assignment with costs $C_{ij} = |\hat{t}_i - t_j|$. Precision, recall, and F1 are then computed from the counts of matched and unmatched events based on tolerance tau. We also report heart-rate mean absolute error with the unit of BPM under a strict tolerance of $\tau = 0.01\,\mathrm{s}$.

3 Experiments and Results

Setup. We evaluate Peak-R1 on two datasets: the publicly available Kansas dataset for high-quality signals and a proprietary Hospital-BSG dataset representing challenging real-world conditions. All data are split 8:2 (training:testing) at the subject level to ensure generalization and prevent overfitting. Input sequences comprise 1000 timesteps, corresponding to 10-second signal windows sampled at 100 Hz. We report precision, recall, F1-score, and heart-rate mean absolute error (MAE) in BPM. Peak-R1 is benchmarked against classical methods (PINO [17], CHOI [18]) and deep learning baselines including U-Net [19], U-Net++ [15], CNN [20], LSTM [21], and Transformer [22] following [15]. We also evaluate the base qwen2.5-3B-instruct model [16]. All methods use identical preprocessing and evaluation protocols. Ablation studies compare the full model (SFT+RL) against SFT-only and peak-extraction-removed variants. The implementation details for the baselines are listed in Appendix G.

Main results. Table 1 demonstrates Peak-R1's superior performance across both datasets. On the **Kansas dataset**, Peak-R1 achieves state-of-the-art results with F1=0.930 and HR MAE=0.399 BPM, significantly outperforming the best deep learning baseline U-Net++ (F1=0.885, HR MAE=6.300

Table 1: Full results with precision, recall, F1, and HR MAE (BPM) under a fixed matching tolerance τ in Kansas and ICU. Best per block in bold. "-" indicates no valid result.

	Kansas				Hospital-BSG			
Method	Prec.	Rec.	F1	HR MAE	Prec.	Rec.	F1	HR MAE
Large Language Model Approaches								
Peak-R1 (Full)	**0.918**	**0.948**	**0.930**	0.399	**0.750**	0.816	**0.779**	**7.002**
Peak-R1 (SFT-only, ablation)	0.914	0.936	0.922	**0.379**	0.693	**0.819**	0.751	10.600
Peak-R1 (w/o peak extraction)	0.052	0.056	0.054	32.913	0.650	0.779	0.709	20.430
qwen2.5-3B-instruct	-	-	-	-	-	-	-	-
Deep Learning Models								
UNet++ [15]	**0.841**	0.946	**0.885**	**6.300**	**0.617**	**0.853**	**0.711**	23.000
UNet [19]	0.799	0.910	0.841	9.300	0.613	0.859	0.711	24.100
CNN [20]	0.633	0.863	0.719	18.100	0.529	0.761	0.621	26.400
LSTM [21]	0.778	**0.961**	0.856	6.400	0.608	0.860	0.708	24.300
Transformer [22]	0.686	**0.907**	0.772	13.000	0.413	0.644	0.500	36.200
Traditional Algorithms								
PINO [17]	0.252	0.280	0.264	4.300	**0.672**	**0.686**	**0.676**	8.602
CHOI [18]	**0.813**	**0.882**	**0.841**	5.800	0.655	0.651	0.650	**7.522**

BPM) and traditional method CHOI (F1=0.841, HR MAE=5.800 BPM). Peak-R1's high recall (0.948) and exceptionally low HR MAE indicate superior temporal accuracy. On the challenging **Hospital-BSG dataset**, Peak-R1 maintains performance leadership with F1=0.779 and HR MAE=7.002 BPM, demonstrating robustness to clinical noise and artifacts. Baseline methods show notable degradation: U-Net++'s F1 drops to 0.711 (precision falls from 0.841 to 0.617), indicating increased false positives in noisy conditions, while CHOI achieves F1=0.650 and HR MAE=7.522 BPM. Visualization examples and a visual analysis are provided in Appendix F.

Ablations. We evaluated an SFT-only variant that omits RL fine-tuning. On Kansas, it performed comparably to the full model (F1=0.922, HR MAE=0.379 BPM), but significantly degraded on noisy Hospital-BSG data (F1=0.751, HR MAE=10.600 BPM vs. full model's F1=0.779, HR MAE=7.002 BPM). This highlights RL's critical role in enhancing robustness for noisy signals. Removing the peak extraction module caused catastrophic performance collapse: F1 dropped to 0.054 on Kansas and 0.709 on Hospital-BSG. This demonstrates that peak-centric representation is foundational—providing the structured input necessary for effective LLM-based J-peak detection. The base `qwen2.5-3B-instruct` model failed to generate valid outputs when prompted directly, confirming that domain-specific adaptations through both peak representation and fine-tuning are essential for task success.

Limitations While Peak-R1 achieves superior accuracy, it comes at the cost of increased computational complexity, a trade-off detailed in Appendix D. The model's inference throughput of 3.571 Samples per second is orders of magnitude lower than that of the baseline deep learning and traditional algorithms. Despite this, the processing speed is well-suited for real-world deployment. A 10-second window of data can be processed in under 0.3 seconds, ensuring that analysis can proceed in real-time without creating a bottleneck, as this is faster than the data acquisition itself.

4 Conclusion

We introduce Peak-R1, which combines a peak-centric input representation with two-stage instruction tuning—supervised fine-tuning (SFT) followed by GRPO-based reinforcement learning—for J-peak detection from cardiomechanical signals. Across the public Kansas dataset and a self-collected Hospital-BSG dataset, Peak-R1 achieves state-of-the-art event-level accuracy and robust heart-rate estimation, surpassing both traditional and deep-learning baselines under a unified evaluation protocol. Ablation studies indicate that the peak-centric representation is essential and that the RL stage improves absolute temporal localization and robustness to noisy signals beyond SFT alone, while preserving strict schema validity of the outputs. The principal trade-off for this advanced performance is computational cost. Although the current model is fast enough for real-time monitoring, a critical direction for future research is to enhance its inference speed, making high-accuracy analysis more accessible for resource-constrained applications.

References

[1] Jeremy W Gordon. Certain molar movements of the human body produced by the circulation of the blood. *Journal of anatomy and physiology*, 11(Pt 3):533, 1877.

[2] Chang-Sei Kim, Stephanie L Ober, M Sean McMurtry, Barry A Finegan, Omer T Inan, Ramakrishna Mukkamala, and Jin-Oh Hahn. Ballistocardiogram: Mechanism and potential for unobtrusive cardiovascular health monitoring. *Scientific reports*, 6(1):31297, 2016.

[3] Yingjian Song, Bingnan Li, Dan Luo, Glenna S Brewster Glasgow, Bradley G Phillips, Yuan Ke, and Wenzhan Song. Real-time continuous blood pressure estimation with contact-free bedseismogram. In *ICC 2024-IEEE International Conference on Communications*, pages 214–219. IEEE, 2024.

[4] Yandao Huang, Lin Chen, Chenggao Li, Junyao Peng, Qingyong Hu, Yu Sun, Hao Ren, Weimin Lyu, Wen Jin, Junzhang Tian, et al. Ai-driven system for non-contact continuous nocturnal blood pressure monitoring using fiber optic ballistocardiography. *Communications Engineering*, 3(1):183, 2024.

[5] Jae Hyuk Shin and Kwang Suk Park. Hrv analysis and blood pressure monitoring on weighing scale using bcg. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 3789–3792. IEEE, 2012.

[6] Samuel M Pröll, Stefan Hofbauer, Christian Kolbitsch, Rainer Schubert, and Karl D Fritscher. Ejection wave segmentation for contact-free heart rate estimation from ballistocardiographic signals. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3571–3576. IEEE, 2019.

[7] A. Suliman, C. Carlson, C. J. Ade, S. Warren, and D. E. Thompson. Performance comparison for ballistocardiogram peak detection methods. *IEEE Access*, 7:53945–53955, 2019.

[8] Christoph Schranz, Christina Halmich, Sebastian Mayr, and Dominik PJ Heib. Surrogate modelling of heartbeat events for improved j-peak detection in bcg using deep learning. *Frontiers in Network Physiology*, 4:1425871, 2024.

[9] Samuel M Proell, Elias Tappeiner, Stefan Hofbauer, Christian Kolbitsch, Rainer Schubert, and Karl D Fritscher. Heart rate estimation from ballistocardiographic signals using deep learning. *Physiological Measurement*, 42(7):075005, 2021.

[10] Jonathan W Kim, Ahmed Alaa, and Danilo Bernardo. Eeg-gpt: exploring capabilities of large language models for eeg classification and interpretation. *arXiv preprint arXiv:2401.18006*, 2024.

[11] Longfei Liu, Guosheng Cui, Cheng Wan, Dan Wu, and Ye Li. Ecg-llm: Leveraging large language models for low-quality ecg signal restoration. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 3537–3542. IEEE, 2024.

[12] Yubao Zhao, Jiaju Kang, Tian Zhang, Puyu Han, and Tong Chen. Ecg-chat: A large ecg-language model for cardiac disease diagnosis. *arXiv preprint arXiv:2408.08849*, 2024.

[13] Elizabeth Fons, Rachneet Kaur, Soham Palande, Zhen Zeng, Tucker Balch, Manuela Veloso, and Svitlana Vyetrenko. Evaluating large language models on time series feature understanding: A comprehensive taxonomy and benchmark. *arXiv preprint arXiv:2404.16563*, 2024.

[14] Yongfeng Huang, Tianchen Jin, Chenxi Sun, Xueyang Li, Shuchen Yang, and Zhiming Zhang. Efficient j peak detection from ballistocardiogram using lightweight convolutional neural network. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 269–272. IEEE, 2021.

[15] Tengda Zhou, Shaoyang Men, Jingxian Liang, Baoxian Yu, Han Zhang, and Xiaomu Luo. 1d u-net++: an effective method for ballistocardiogram j-peak detection. *Journal of Mechanics in Medicine and Biology*, 21(10):2140058, 2021.

[16] Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

[17] Esteban J Pino, Javier AP Chávez, and Pablo Aqueveque. Noninvasive ambulatory measurement system of cardiac activity. In *2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 7622–7625. IEEE, 2015.

[18] Byung Hun Choi, Gih Sung Chung, Jin-Seong Lee, Do-Un Jeong, and Kwang Suk Park. Slow-wave sleep estimation on a load-cell-installed bed: a non-constrained method. *Physiological measurement*, 30(11):1163, 2009.

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[20] Serkan Kiranyaz, Turker Ince, and Moncef Gabbouj. Real-time patient-specific ecg classification by 1-d convolutional neural networks. *IEEE transactions on biomedical engineering*, 63(3):664–675, 2015.

[21] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[23] Charles Carlson, Vanessa-Rose Turpin, Ahmad Suliman, Carl Ade, Steve Warren, and David E Thompson. Bed-based ballistocardiography: Dataset and ability to track cardiovascular parameters. *Sensors*, 21(1):156, 2020.

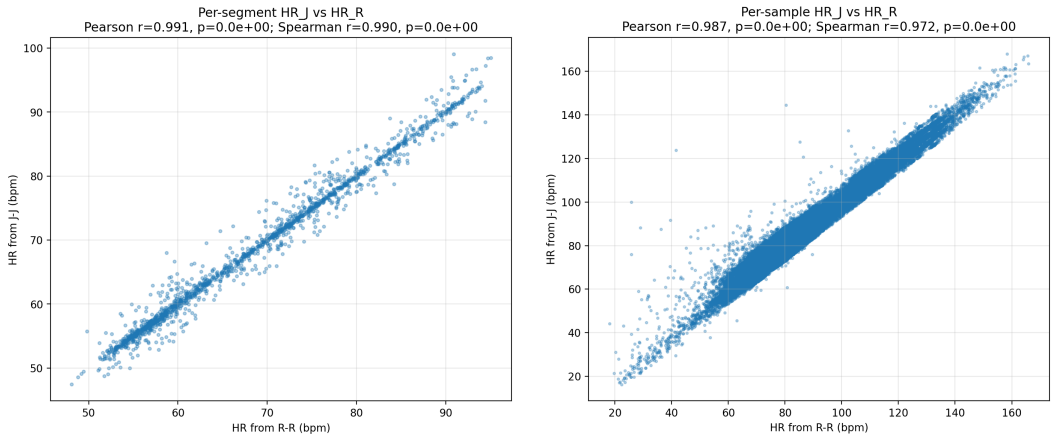


Figure 2: Correlation between J–J and R–R intervals in Kansas (left) and ICU (right).

A J peak definition

In this study, we define ground-truth J-peaks as the maximum BCG amplitude within each R–R interval of the reference ECG. This definition is supported by prior literature and further validated empirically: we quantify the association between successive J–J intervals and the corresponding R–R intervals and observe a strong correlation (Fig. 2). This result indicates that J-peak–derived intervals faithfully track the reference heart rate.

B Reward functions

We use four per-sample rewards that promote (i) schema compliance, (ii) heart-rate consistency, (iii) precise peak localization, and (iv) peak-count completeness. Below, s denotes the model output string.

Format compliance. Admissible outputs are those that match the required schema `J: [ ··· ]` (Eq. (1)); the binary reward is given in Eq. (2).

$$\mathcal{F}_J \;=\; \big\{\, s \in \Sigma^* : s \text{ matches } \texttt{J: [} \cdots \texttt{]} \,\big\}. \tag{1}$$

$$\gamma_{\text{format}}(s) \;=\; \begin{cases} 1, & s \in \mathcal{F}_J, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Heart-rate consistency. A smooth reward penalizes relative HR error (Eq. (3)); H_{pred} and H_{true} are BPM computed from predicted and reference J-peaks.

$$\gamma_{\text{HR}} \;=\; \begin{cases} w\,\exp\!\Big(-k\,\dfrac{|H_{\text{pred}} - H_{\text{true}}|}{H_{\text{true}}}\Big), & \text{if both HRs are available,} \\ 0, & \text{otherwise,} \end{cases} \qquad \text{with } w = 0.15,\ k = 2. \tag{3}$$

Positional accuracy. Let n_p and n_g be the counts of predicted and ground-truth peaks, and let m be the size of a maximum one-to-one matching under a temporal tolerance $\tau = 0.01$ s. The positional reward is the F1 score in Eq. (4).

$$\gamma_{\text{pos}} \;=\; \begin{cases} \dfrac{2m}{n_p + n_g}, & n_p + n_g > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Count completeness. We define summary quantities in Eq. (5), a completeness score in Eq. (6), and the weighted reward in Eq. (7).

$$n_g \;=\; \#\text{GT peaks}, \quad n_p \;=\; \#\text{predicted peaks}, \quad \rho \;=\; \frac{\min(n_p, n_g)}{n_g}, \quad \epsilon \;=\; \frac{n_p - n_g}{n_g}. \tag{5}$$

$$S_{\text{comp}}(n_p, n_g) \;=\; \begin{cases} 0, & n_g = 0 \text{ or } n_p = 0, \\ 1, & n_p = n_g \;(\geq 1), \\ 0.8\,\rho, & 0 < n_p < n_g, \\ \max\!\big(0,\; \rho\,(1 - 0.5\,\epsilon)\big), & n_p > n_g. \end{cases} \tag{6}$$

$$\gamma_{\text{cnt}} \;=\; w_c\, S_{\text{comp}}(n_p, n_g), \qquad w_c = 0.15. \tag{7}$$

C Group Relative Policy Optimization (GRPO)

Group Relative Policy Optimization (GRPO) is a PPO-style policy optimization method for RLHF that compares multiple responses generated for the same prompt and uses *group-relative* advantages (e.g., z-scored rewards within the group). This reduces reliance on calibrated absolute rewards, aligns with comparison-based reward models, and retains PPO's stability via clipping with an explicit KL regularizer to a reference policy.

$$\begin{aligned} \mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\substack{q\sim P(Q) \\ \{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)}} \Bigg[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \min\Big(r_{i,t}\,\hat{A}_{i,t},\; \text{clip}(r_{i,t},\,1-\varepsilon,\,1+\varepsilon)\,\hat{A}_{i,t} \Big) \\ -\;\beta\,\mathrm{D}_{\text{KL}}\big(\pi_\theta(\cdot \mid q)\,\|\,\pi_{\text{ref}}(\cdot \mid q)\big) \Bigg]. \end{aligned} \tag{8}$$

$$r_{i,t} := \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}. \tag{9}$$

where $r_{i,t} = \dfrac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}$, $\hat{A}_{i,t}$ are group-relative advantages (e.g., standardized within the G responses), ε is the clipping parameter, and β weights the KL regularization. Key knobs are the group size G, clip ε, and KL weight β.

D Computational Complexity Analysis

The superior accuracy of Peak-R1 requires significantly more computation than baseline methods, a critical trade-off detailed in Table 2. Computational throughout is measured in samples per second (SPS). The model's training is intensive, with throughputs of 0.397 SPS (SFT) and 0.24 SPS (RL), reflecting the overhead of its 3-billion-parameter architecture compared to the much smaller deep learning models.

For inference, Peak-R1 processes 3.571 SPS. Although this is orders of magnitude slower than deep learning models (up to 52k SPS) and traditional algorithms (over 1M SPS), this speed is well-suited for practical application. Processing a 10-second signal window requires less than 0.3 seconds, substantially faster than the data acquisition time itself. This confirms that Peak-R1's inference speed is tolerable and enables real-time, window-by-window analysis without causing a processing bottleneck.

Table 2: Computational complexity and throughput analysis for all evaluated models. Throughput is measured in samples per second (SPS). For traditional algorithms, which do not have a training phase, throughput reflects their combined processing speed.

Model/Algorithm	**Parameters**	**Training Throughput (SPS)**	**Inference Throughput (SPS)**
Large Language Model			
Peak-R1 (our model)	~3 Billion	SFT: 0.397, RL: 0.24	3.571
Deep Learning Models			
UNetPlusPlus1D	1,790,465	3,414.5	12,379.8
UNet1D	382,849	7,009.5	33,527.0
LSTM1D	133,761	4,595.3	12,372.4
CNN1D	329,057	18,036.8	52,129.6
Transformer1D	8,705	2,664.1	10,676.1
Traditional Algorithms			
PINO	N/A	N/A	1,719,949.2
CHOI	N/A	N/A	4,316,707.8

E Summary of KSU and Hospital-BSG dataset

Kansas dataset is an open-source BCG dataset with 40 subjects and synchronized BCG, ECG, PPG, and ABP signals, sampled at 1000 Hz. heart rate range from 48 to 95 BPM, but with limited BP variability and few patients with cardiovascular disease. This stable dataset provides a baseline for evaluating feature extraction and calibration under lower variability conditions.

Hospital-BSG dataset collected 1120 hours of BSG, ABP, HR, ECG, PPG, and RR data from 52 ICU patients (ages 6–86, 35 males) using our SeismoDot system (100 Hz). Heart rate span 46-262 BPM, reflecting the high BP variability typical in critical care. All data collection was IRB-approved. The inclusion of a broad range of HR values and signal quality in the ICU cohort is essential for assessing model robustness and generalizability to real-world, high-variability clinical settings.

The summary of KSU and Hospital-BSG dataset is shown in Table 3.

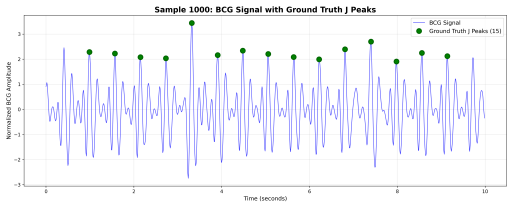
Table 3: Summary of Datasets Used in the Study

Characteristic	**Kansas Dataset [23]**	**Hospital-BSG Dataset**
Source	Kansas State University	Yixing People's Hospital
Number of Subjects	40	52
Gender Distribution	17 males, 23 females	35 males, 17 females
Age Range (years)	18–65	6–86
Sensors	EMFi sensors and load cells	BetDot
Sampling Rate	1000 Hz	100 Hz
Signals	BCG, ECG, PPG, ABP	BSG, ABP, RR, HR, ECG, PPG
Heart Rate Range(BPM)	48–95	46–262
Systolic Pressure Range (mmHg)	58.7–187.0	38–310
Diastolic Pressure Range (mmHg)	44.5–101.0	20–293
Systolic Pressure Dynamics (mmHg)	11–46	53–259
Diastolic Pressure Dynamics (mmHg)	5–27	19–264
Data Duration	$>$4.5 hours	1120 hours
Special Notes	4 subjects with cardiovascular conditions	Hospital ICU patients

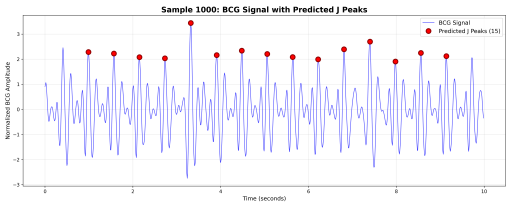
F Visualization

We visualize two representative segments of the ballistocardiogram (BCG) to assess the performance of different models and algorithms: an easy sample and a hard sample (Figure 3 and Figure 4). On the easy sample, most models perform well; however, deep-learning models exhibit slight temporal shifts in peak locations, likely due to the absence of explicit output constraints. On the hard sample, these shifts become more pronounced. While such temporal misalignment has limited impact on heart-rate (HR) estimation—primarily driven by inter-beat intervals—it substantially affects blood-pressure (BP) estimation, which depends on accurately measuring the amplitude difference between the J and K peaks.

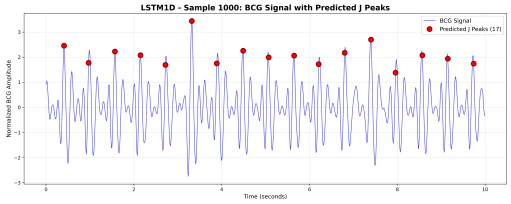
For Peak-R1, most potential J peaks are captured, but their count is occasionally overestimated, which can inflate HR estimates. Nevertheless, because the predicted amplitudes tend to be consistent, the impact on BP estimation can be comparatively limited, even for difficult peaks. In contrast, the PINO and Choi baselines exhibit larger inconsistencies in distorted regions, leading to notable errors in both HR and BP estimates.



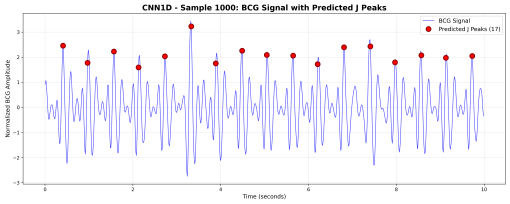
((a)) BCG (ground truth)



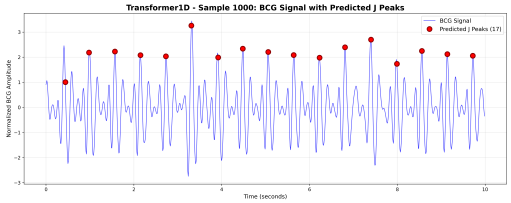
((b)) Peak-R1 (predictions)



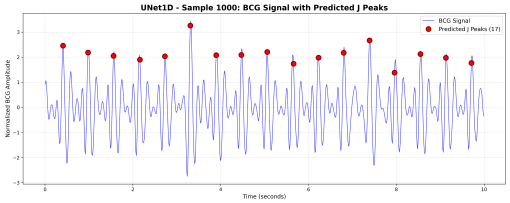
((c)) LSTM (predictions)



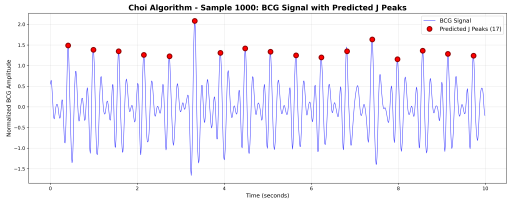
((d)) CNN (predictions)



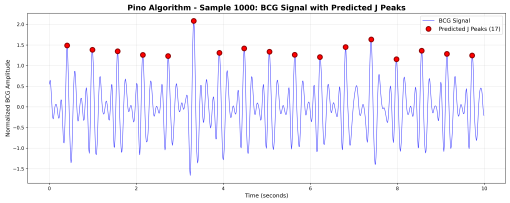
((e)) Transformer (prediction)



((f)) U-Net++ (predictions)

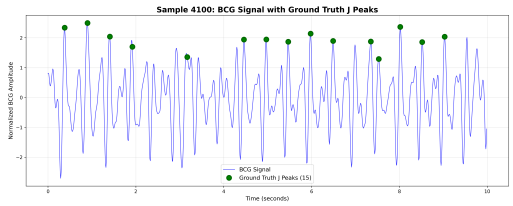


((g)) Choi et al. (predictions)

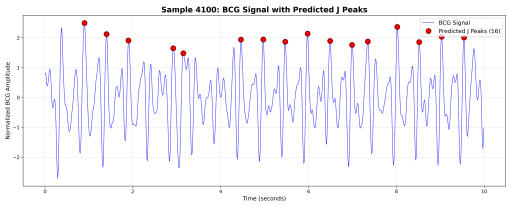


((h)) PINO (predictions)

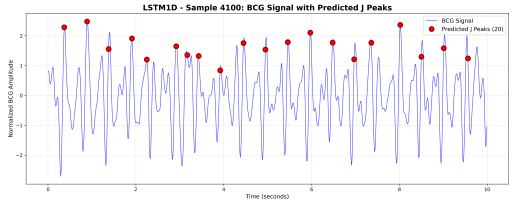
Figure 3: Comparison of ground truth and model outputs for the easy sample (ID 1000). Top row: BCG ground truth and Peak-R1 predictions. Subsequent rows: outputs from deep-learning models (LSTM, CNN, Transformer, U-Net) and baseline methods (Choi, PINO).



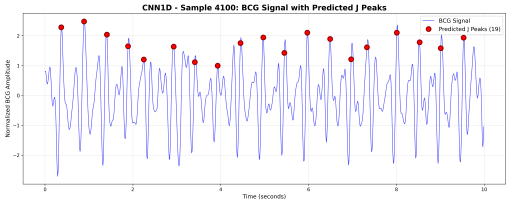
((a)) BCG (ground truth)



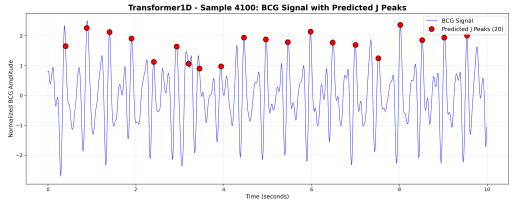
((b)) Peak-R1 (predictions)



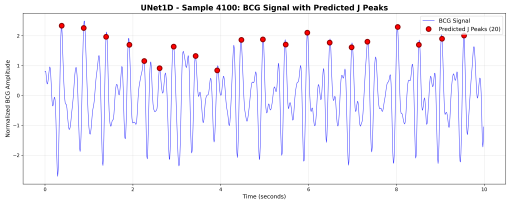
((c)) LSTM (predictions)



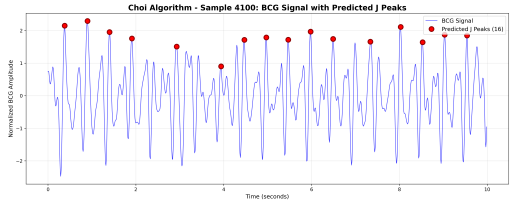
((d)) CNN (predictions)



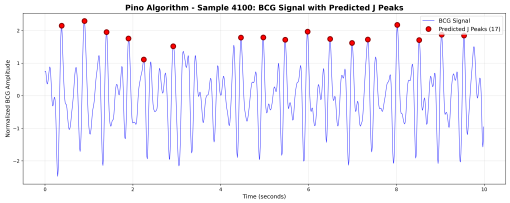
((e)) Transformer (predictions)



((f)) U-Net++ (predictions)



((g)) Choi et al. (predictions)



((h)) PINO (predictions)

Figure 4: Comparison of ground truth and model outputs for the hard sample (ID 4100). Top row: BCG ground truth and Peak-R1 predictions. Subsequent rows: outputs from deep-learning models (LSTM, CNN, Transformer, U-Net) and baseline methods (Choi, PINO).

G Implementation Details

G.1 Model Architectures

We evaluated five distinct deep learning architectures for the task of J-peak detection. The UNet-based models utilize a repeating ‘ConvBlock‘, which consists of two sequential 1D convolutional layers, each followed by a ReLU activation function. The key architectural details for each model are summarized in Table 4.

G.2 Training Procedure

All models were trained using a standardized procedure to ensure consistency and reproducibility across experiments. The training hyperparameters and settings are detailed in Table 5.

Table 4: Detailed Summary of Deep Learning Model Architectures.

Model	Architectural Details
CNN1D	A sequential 1D convolutional network structured as follows: • **Conv1**: 32 filters, kernel size 32, ReLU activation. • **MaxPool1D**: kernel size 2. • **Conv2**: 64 filters, kernel size 32, ReLU activation. • **MaxPool1D**: kernel size 2. • **Conv3**: 128 filters, kernel size 32, ReLU activation. • **Upsampling**: Two sequential upsampling layers (scale factor 2) to restore original length. • **Output**: Final 1x1 convolution to 1 channel, followed by a Sigmoid activation.
LSTM1D	A recurrent architecture designed to capture temporal dependencies: • **BiLSTM**: A two-layer bidirectional LSTM with 64 hidden units in each direction (total 128). • **Output**: A fully connected layer maps the LSTM output to a single channel, followed by a Sigmoid activation.
Transformer1D	An attention-based model for sequence-to-sequence probability mapping: • **Input Projection**: A linear layer projects the input channel to a model dimension (d_{model}) of 32. • **Positional Encoding**: Standard sine-cosine positional encoding is added. • **Encoder**: A single Transformer encoder layer with 4 attention heads, a feedforward dimension of 64, and GELU activation. • **Output**: A final linear layer maps the encoder output to 1 channel, followed by a Sigmoid activation.
UNet1D	A 1D U-Net with a 3-level symmetric encoder-decoder structure (base filters = 16): • **Encoder**: Three 'ConvBlock' layers that progressively halve the sequence length and double the channel count ($16 \rightarrow 32 \rightarrow 64$). • **Decoder**: Three blocks that progressively double the sequence length and halve the channel count. Each block consists of an up-convolution, a skip connection concatenating the output with the corresponding encoder feature map, and a 'ConvBlock'. • **Output**: A 1x1 convolution to 1 channel, followed by a Sigmoid activation.
UNetPlusPlus1D	An advanced U-Net with nested and dense skip connections (base filters = 16): • **Encoder**: A 4-level encoder path ($X^{0,0}$ to $X^{3,0}$) with channel counts ($16 \rightarrow 32 \rightarrow 64 \rightarrow 128$). • **Nested Skip Pathways**: Intermediate 'ConvBlock' layers are connected in a dense, nested pattern. Each node receives concatenated inputs from the previous node in the same level and the up-sampled output from the node in the level below it. • **Output**: The final output is generated from the highest-level node ($X^{0,3}$), which aggregates features from all semantic levels, followed by a 1x1 convolution and a Sigmoid activation.

Table 5: Standardized Training Parameters for all Deep Learning Models.

Parameter	**Value / Setting**
Data Split	80% Training, 20% Validation
Optimizer	Adam
Initial Learning Rate	0.001
Weight Decay	1e-5
Loss Function	Focal Loss ($\gamma = 2$, α=dynamic inverse frequency)
LR Scheduler	ReduceLROnPlateau (Patience=3, Factor=0.5)
Gradient Clipping	Max norm = 1.0
Batch Size	32
Epochs	Up to 150 (terminated by early stopping)

H Prompt-Based Analysis Example

The following code block shows a complete prompt-and-response example, demonstrating how the instruction and input data are used to generate the correct, structured output of identified J-peak timestamps.

Listing 1: Example of a structured prompt for J-peak detection.

```
{
    "instruction": "You are a specialized assistant for biomedical
       signal analysis, specifically trained in BCG J peak detection
       from ICU 3D HR signals. Analyze the following ICU 3D HR signal
        peaks using ECG-guided J peak detection methodology. Peak
       detection guidance: J peaks are detected as maximum values in
       BCG signals occurring between consecutive R peaks. Detect J
       peak in the following BCG signal peaks. Output format:
       Structured format with J peak positions in brackets.",
    "input": "ICU 3D HR signal sampled at 100 Hz with duration of
       10.00 seconds. Detected peaks in signal: <TS_START>\\nDate:
       2020-01-01 00:00:13, Value: 2.059006\\nDate: 2020-01-01
       00:00:23, Value: -1.752526\\n... (remaining peak data) ...\\
       nDate: 2020-01-01 00:16:32, Value: -1.243836\\n<TS_END>",
    "output": "J: [2020-01-01 00:02:45,2020-01-01 00:03:14,2020-01-01
       00:04:17,2020-01-01 00:05:33,2020-01-01 00:07:24,2020-01-01
       00:08:14,2020-01-01 00:09:27,2020-01-01 00:10:41,2020-01-01
       00:12:02,2020-01-01 00:13:23,2020-01-01 00:14:39]"
}
```