4 Hz, 4 Pages: Just-in-Time Substance Use Relapse Risk Detection from Wearable Time Series Data

Abhishek Singh Dhadwal
School of Computing and Augmented Intelligence
Arizona State University
Tempe, AZ, USA
adhadwal@asu.edu

Abstract

Substance use relapse is often preceded by elevated stress and affect states. Wearable sensing offers a pathway for just-in-time adaptive interventions (JITAIs), but practical deployment requires lightweight, deterministic, and CPU-efficient models. We present a minimal pipeline that standardizes multimodal wearable signals to 4 Hz, applies sliding-window segmentation, and evaluates both a MiniRocket+Ridge classifier and a compact feature+logistic baseline. We evaluate across three public stress/affect datasets (WESAD, PhysioNet Stress, CAN-Stress) as proxies for relapse risk, reporting AUPRC, AUROC, F1 at optimal threshold, time-to-detection (TTD) at 80% recall, and per-window latency. Our results show that the approach achieves competitive predictive accuracy with latencies below 2 ms per window on CPU, supporting feasibility for real-time JITAI triggers. The design emphasizes deterministic evaluation and proxy-to-relapse framing, offering a transparent foundation for future clinical validation and uses for global, low resource, remote substance use interventions. We release an open-source, CPU-only implementation of this pipeline as a lightweight command-line tool and library to support reproducible JITAI experimentation on wearable time series.

1 Introduction

Substance use relapse remains a critical public health challenge in North America, with high rates of recurrence even after treatment. Evidence suggests that relapse risk is linked to heightened stress and affective states, which can be continuously monitored using wearable devices. Delivering timely interventions in these moments is the goal of just-in-time adaptive interventions (JITAIs). However, existing approaches for relapse prediction often rely on complex, compute-heavy models or require GPU acceleration, limiting deployability in mobile and edge settings.

To address this gap, we present a lightweight, deterministic pipeline that standardizes multimodal wearable signals to 4 Hz and applies sliding-window modeling entirely on CPU. Rather than proposing a new model family, our contribution is deployment-oriented: we (i) design a minimal, fully deterministic pipeline around MiniRocket plus ridge regression and a compact feature plus logistic/SGD baseline, (ii) systematically quantify the latency–performance trade-offs of these choices on three public stress and affect datasets (WESAD, PhysioNet Stress, CAN-Stress) as practical proxies for relapse risk, and (iii) demonstrate that a CPU-only implementation can achieve millisecond-scale inference with competitive AUROC and AUPRC under leave-one-subject-out evaluation. We also release an open-source implementation of this pipeline as a small library and command-line interface to support reproducible experimentation and future extension.

2 Related Work

Contemporary work on just in time adaptive interventions for substance use sits at the intersection of JITAI design, wearable stress and affect modeling, and lightweight time series methods. In this section, we briefly review prior work along these three axes to clarify how our minimal 4 Hz pipeline fits into the existing landscape.

Wearables for SUD relapse and proxies. Digital phenotyping uses continuous mobile and wearable data to model risk states related to substance use and relapse. Evidence reviews highlight the feasibility of using wrist-worn sensors to track stress, craving, and withdrawal in naturalistic settings, with electrodermal activity (EDA), heart rate (BVP / PPG) and temperature as key physiological channels (e.g. (5; 7; 6; 18)). Given privacy and data-access constraints for true relapse events, studies frequently adopt stress/affect as a *proxy* signal for elevated relapse risk.

Public stress/affect datasets. WESAD provides multimodal wearable signals for 15 participants after sensor-screening (wrist ACC, BVP, EDA, TEMP) (1). PhysioNet hosts stress-induction datasets suitable for benchmarking cross-subject generalization (2). CAN-Stress offers semi-naturalistic recordings for 82 participants (39 cannabis users, 43 non-users) (3). Recent surveys catalog stress datasets and discuss open-access considerations (8; 9).

Lightweight time-series modeling. MiniRocket achieves fast, near-deterministic time-series classification by applying a fixed bank of convolutional kernels and computing simple statistics such as the proportion of positive values, yielding strong accuracy without gradient-based training (4). This makes MiniRocket attractive for deployment, since the transform is deterministic and the downstream classifier can be a simple linear model. Handcrafted, compact feature sets paired with logistic or SGD models remain appealing for on-device and streaming applications; related literature in TinyML and on-device inference reports millisecond-scale execution on constrained hardware (15; 16; 17). Our work sits in this design space, evaluating how a MiniRocket-based pipeline compares to compact-feature baselines under strict latency budgets.

Early-warning metrics and reliability. Relapse prevention is time-sensitive. Beyond AUPRC/AUROC, *time-to-detection* (TTD) captures how quickly the model identifies a risk window. Recent work examines metric behavior under imbalance and temporal precision requirements (10; 11). Label-noise handling and leave-one-subject-out (LOSO) evaluation are emphasized as good practice for robust cross-subject inference (12; 13; 14).

3 Data and Labeling

We evaluate across three de-identified, publicly released datasets serving as relapse-risk proxies:

WESAD. 17 subjects participated; 2 discarded due to sensor malfunction, yielding 15. Wrist-based Empatica E4 signals include ACC (32 Hz), BVP (64 Hz), EDA (4 Hz), and TEMP (4 Hz) (1).

PhysioNet Stress. 36 subjects exposed to stress stimuli with multimodal wrist/chest signals (2).

CAN-Stress. 82 participants (39 cannabis users, 43 non-users) with semi-natural E4 recordings(3).

Labeling. Ground-truth intervals or events are converted to per-window binary labels. To tolerate timing uncertainty between annotated events and physiological responses, we use a default event radius of 60s: a window is labeled positive if an event occurs within 60 seconds of its center. This horizon is intended to capture short-term changes in stress and affect that are actionable for just-in-time intervention, while avoiding very long windows that blur distinct episodes. Heuristic label expansions (for example, top-percentile EDA segments) are supported for development but excluded from reported metrics.

4 Pipeline and Methods

We now describe the end to end pipeline used in our experiments, from ingesting raw multimodal Empatica files through resampling and windowing to model training and streaming inference. Figure 1 summarizes the architecture at a glance; here we highlight the key design choices that affect both detection performance and latency.

Preprocessing: Raw modality files are resampled to 4 Hz and aligned into a tidy table with subject identifier, timestamp, ACC/BVP/EDA/TEMP channels, and binary label. All subsequent steps operate on this standardized representation.

Windowing and horizon: Unless otherwise noted, we use 30 second windows with 10 second stride. A window is labeled positive if an event proxy of relapse risk occurs within a 60 second radius around the window center, consistent with the labeling strategy in §3. This horizon is intended to reflect a short-term intervention window: predictions should arrive within tens of seconds of elevated stress or affect, not hours before or after.

Train/Test Protocol: Evaluation uses leave-one-subject-out (LOSO) cross-validation. For each held-out subject, models are trained on all remaining subjects and evaluated on the held-out subject only. Metrics are aggregated across all folds. We do not report multi-seed repeats; each row in Table 1 corresponds to a single deterministic LOSO run.

Models and Hyperparameters: The MiniRocket + Ridge configuration applies the default MiniRocket transform with ridge regression (scikit-learn, cross-validated regularization; no explicit optimizer or learning rate). The compact-feature baselines extract hand-crafted statistics (for example, means and variability measures) and use either L2-regularized logistic regression (C = 1.0) or SGD with the default learning rate schedule. No batch size is defined beyond the sliding-window segmentation. All runs are deterministic except for online SGD updates.

Online and streaming inference: To simulate online deployment, we implement a "Streamer" that maintains a rolling buffer of the most recent 30 seconds of standardized data at 4 Hz and emits a prediction each time the buffer advances by the chosen stride. Windows are constructed causally from past and present samples only, with no access to future context, so that the evaluation reflects how the model would behave in real time as new wearable samples arrive.

Metrics: We report AUPRC, AUROC, F1 at the optimal threshold, time-to-detection (TTD) at 80 percent recall, and latency (milliseconds per window). Thresholds are selected per fold based on validation performance. Latency is measured as end-to-end CPU time from receiving the last sample in a window to emitting a prediction. All experiments run on a CPU-only Apple M4-based MacBook Pro with 24 GB unified memory and a 1 TB solid-state drive, with no GPU acceleration.

Open Source Implementation: We release an open-source implementation of the full 4 Hz pipeline, including standardization, MiniRocket and compact-feature baselines, LOSO evaluation, and streaming inference, at: `https://github.com/AbhishekSinghDhadwal/TS4H_4Hz_4Pages`. The repository provides a small library and command-line interface for reproducing all experiments and adapting the pipeline to new wearable datasets.

Pipeline schematic

5 Results

Main Table (all runs): Table 1 includes *all* reported runs (batch and online variants). We add a *Config* column for interpretability.

Table 1: Evaluation across runs. Metrics: AUPRC, AUROC, F1 at optimal threshold, time-to-detection (TTD) at 80% recall, and latency (ms/window).

Dataset	Config	AUPRC	AUROC	$\mathrm{F1}_{opt}$	Thr.	TTD (s)	Lat. (ms)
PhysioNet Stress	Batch; win=30s, stride=5s	0.407	0.695	0.382	0.27	0	1.340
PhysioNet Stress	Batch; win=30s, stride=10s	0.418	0.695	0.386	0.29	10	1.260
WESAD	Batch; win=30s, stride=5s	0.325	0.673	0.407	0.70	5	1.430
WESAD	Batch; win=30s, stride=10s	0.337	0.675	0.415	0.69	70	1.350
CAN-Stress	Online+ClassBalanced; stride=10s	0.226	0.466	0.409	0.00	0	0.570
CAN-Stress	Batch; win=30s, stride=10s	0.453	0.713	0.508	0.32	0	0.200
CAN-Stress	Online; win=30s, stride=10s	0.248	0.535	0.434	0.02	0	0.500

Performance Comparison: Where both configurations are available, MiniRocket-based batch runs outperform the online compact-feature models in AUROC and AUPRC, while online models re-

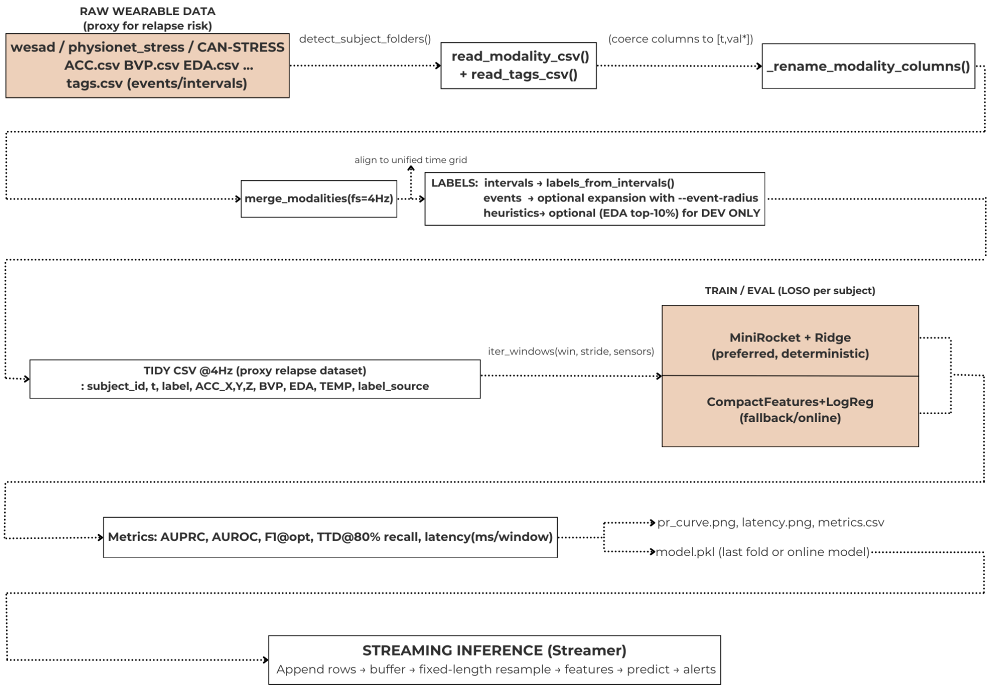
RAW WEARABLE DATA
(proxy for relapse risk)
wesad / physionet_stress / CAN-STRESS
ACC.csv BVP.csv EDA.csv ...
tags.csv (events/intervals)
detect_subject_folders()
read_modality_csv()
+ read_tags_csv()
(coerce columns to [t,val*])
_rename_modality_columns()
align to unified time grid
merge_modalities(fs=4Hz)
LABELS: intervals → labels_from_intervals()
events → optional expansion with --event-radius
heuristics→ optional (EDA top-10%) for DEV ONLY
TRAIN / EVAL (LOSO per subject)
MiniRocket + Ridge
(preferred, deterministic)
CompactFeatures+LogReg
(fallback/online)
iter_windows(win, stride, sensors)
TIDY CSV @4Hz (proxy relapse dataset)
: subject_id, t, label, ACC_X,Y,Z, BVP, EDA, TEMP, label_source
Metrics: AUPRC, AUROC, F1@opt, TTD@80% recall, latency(ms/window)
pr_curve.png, latency.png, metrics.csv
model.pkl (last fold or online model)
STREAMING INFERENCE (Streamer)
Append rows → buffer → fixed-length resample → features → predict → alerts

Figure 1: Pipeline for **4 Hz, 4 Pages**: Just-in-Time Relapse Risk Detection using stress/affect proxies from wearables.

main competitive in F1. For example, on CAN-Stress the batch MiniRocket model (AUROC 0.713, AUPRC 0.453) clearly outperforms the online variants (AUROC 0.466–0.535) while maintaining sub-millisecond latency. Between datasets, WESAD (lab-controlled) exhibits modest predictive performance (AUPRC around 0.33), whereas CAN-Stress (semi-naturalistic) achieves stronger AUPRC despite more noise. This contrast underscores the importance of ecological validity: noisier but realistic datasets may capture stress dynamics that are more relevant to relapse risk than tightly controlled laboratory protocols.

Early-Warning Behavior: The time-to-detection (TTD) metric shows how quickly risk can be flagged relative to an event. For WESAD, TTD ranges from 5s to 70s depending on stride, highlighting sensitivity to windowing parameters. In contrast, CAN-Stress and PhysioNet runs often achieve TTD = 0–10s, suggesting that compact-feature or stride choices can shift detection timing dramatically.

Latency: Figure 2 shows per-window latency for all runs. All models achieve sub-2ms latency, including online variants. The CAN-Stress batch run completes inference in $\sim$0.2ms, effectively instantaneous. These findings demonstrate CPU-only feasibility for real-time relapse-risk detection and JITAI deployment. These latencies are measured on the Apple M4-based laptop described in §4 and leave considerable headroom for additional pre-processing, decision logic, or user interface delays within clinically realistic just-in-time intervention budgets.

6 Discussion

Our study highlights both the promise and the constraints of stress and affect-based proxies for relapse risk detection. Stress and affect datasets provide useful signals related to relapse vulnerability, but they cannot capture the full spectrum of relapse contexts such as craving, environmental cues, or social triggers. Dataset heterogeneity further shapes model behavior: WESAD offers controlled laboratory conditions, CAN-Stress provides noisier semi-naturalistic data, and PhysioNet Stress represents an intermediate case. This diversity introduces generalization challenges but also tests robustness across different stress-induction paradigms. From a modeling perspective, the results show that a deterministic MiniRocket-based pipeline and compact-feature baselines can achieve

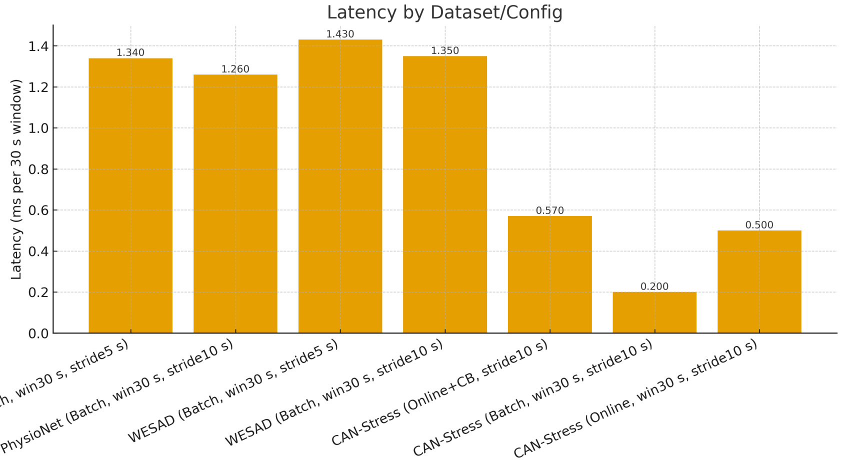

Figure 2: Latency (ms per window) across all runs, grouped by dataset and configuration.

competitive AUROC and AUPRC while maintaining millisecond-scale latency on CPU-only hardware. In many clinical scenarios, relapse-related decisions occur on the order of seconds to minutes, so ultra-low latency is not valuable on its own. Instead, the benefit of such a fast pipeline is that it leaves headroom for additional pre-processing, safety checks, decision policies, and user interface delays while staying within clinically realistic just-in-time intervention budgets. Ethical and privacy considerations remain central for any deployment of continuous wearable monitoring in high-risk substance use populations. Continuous biosignal collection raises concerns about inappropriate secondary use by employers, insurers, or the justice system, and frequent false alarms could erode trust or contribute to stigma. Conversely, missed detections may lead to overconfidence in an automated system. Our design choices emphasize deterministic, CPU-only inference that can in principle run on-device, limiting the need to stream raw signals to cloud services and enabling stronger data minimization. Any real-world deployment should pair such technical safeguards with explicit consent processes, clear communication about how data are used, and involvement of patients and clinicians in co-designing alert thresholds and escalation pathways.

7 Limitations

Labeling uncertainty is a key limitation. Expanding events with a 60 second radius tolerates timing noise between annotated events and physiological responses but introduces subjectivity, and as Table 1 shows, time-to-detection can vary widely depending on window length and stride. Benchmarking across multiple radii and horizons would help standardize evaluation and clarify how much temporal precision is realistically achievable. Another limitation is scalability. While LOSO ensures reproducibility, performance estimates on relatively small stress and affect datasets may not generalize to larger and more diverse populations without careful external validation. Finally, this work infers risk only from affect proxies and is not calibrated on true relapse events; it makes no clinical relapse claims and is not intended for direct clinical decision support. We also do not benchmark large pre-trained or GPU-accelerated deep models; future work should compare such architectures under the same hardware and latency constraints to more fully map the accuracy–efficiency trade-off.

8 Conclusion

We presented a lightweight pipeline for just-in-time relapse risk detection using wearable stress and affect proxies. Evaluation across WESAD, PhysioNet Stress, and CAN-Stress shows that even modest datasets allow MiniRocket and compact baselines to achieve competitive accuracy with latencies below 2 ms. These results demonstrate the feasibility of CPU-only, deterministic inference for JITAI deployment. Future work should validate against real relapse outcomes, integrate behavioral and contextual signals, and balance sensitivity with specificity to mitigate false alarms and missed detections. By providing a transparent pipeline and benchmarking on public datasets, we aim to accelerate translational progress toward just-in-time relapse prevention.

References

[1] Schmidt, P., Reiss, A., Duerichen, R., Marberger, C., & Van Laerhoven, K. (2018). WESAD: A Multimodal Dataset for Wearable Stress and Affect Detection. In *Proc. ICMI* (pp. 400–408). ACM.https://doi.org/10.24432/C57K5T

[2] Goldberger, A. L., et al. (2000). PhysioBank, PhysioToolkit, and PhysioNet. *Circulation*, 101(23), e215–e220. https://doi.org/10.1161/01.CIR.101.23.e215

[3] CAN-STRESS: A Real-World Multimodal Dataset for Cannabis Use and Stress (2025). arXiv preprint. https://arxiv.org/abs/2503.19935

[4] Dempster, A., Petitjean, F., & Webb, G. I. (2021). MiniRocket: A Very Fast (Almost) Deterministic Transform for Time Series Classification. In *Proc. KDD* (pp. 248–257). https://arxiv.org/abs/2012.08791

[5] Hsu, M., et al. (2020). Digital Phenotyping to Enhance Substance Use Disorder Treatment. *JMIR Mental Health*. https://mental.jmir.org/2020/10/e21814/

[6] Alinia, P., et al. (2021). Associations Between Physiological Signals Captured Using Wearables and Substance Use Outcomes. *JMIR Formative Research*. https://formative.jmir.org/2021/7/e27891/

[7] Goldfine, C., et al. (2021). Wearable and Wireless mHealth Technologies for Substance Use Disorders. *Curr Psychiatry Rep*. https://pmc.ncbi.nlm.nih.gov/articles/PMC7963000/

[8] Ometov, A., et al. (2024). Stress and Emotion Open Access Data: A Review on Datasets and Benchmarks. *Journal of healthcare informatics research*. https://pmc.ncbi.nlm.nih.gov/articles/PMC12290141/

[9] Dahal, K., et al. (2023). Global Stress Detection Framework Combining a Reduced Set of HRV Features and Random Forest Model. *Sensors*. https://pmc.ncbi.nlm.nih.gov/articles/PMC10255919/

[10] McDermott , M. (2024). A Closer Look at AUROC and AUPRC under Class Imbalance. arXiv:2401.06091. https://arxiv.org/html/2401.06091

[11] Wyant, K., et al. (2024). Machine learning models for temporally precise lapse prediction. *OSF/PMC*. https://osf.io/preprints/psyarxiv/cgsf7_v1/

[12] Yang, J., et al. (2024). Addressing label noise for electronic health records: insights from computer vision for tabular data *BMC medical informatics and decision making*. https://pmc.ncbi.nlm.nih.gov/articles/PMC11212446/

[13] Adler, D. A., et al. (2022). Machine learning for passive mental health symptom prediction: Generalization across different longitudinal mobile sensing studies *PLOS One*. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0266516

[14] Hwang, W.-S., & Lim, J. (2019). Time-Series Aware Precision and Recall for Anomaly Detection: Considering Variety of Detection Result and Addressing Ambiguous Labeling https://doi.org/10.1145/3357384.3358118

[15] Abu-Samah, A., et al. (2025). TinyML-based stress classification on microcontrollers. *GitHub/Project*. https://github.com/eloquentarduino/micromlgen

[16] Mairittha, N., Mairittha, T., & Inoue, S. (2019). On-Device Deep Learning Inference for Efficient Activity Data Collection.*Sensors* 2019; 19(15):3434. https://doi.org/10.3390/s19153434

[17] Islam, M. R., et al. (2023). Deep Learning-Based IoT System for Remote Monitoring and Early Detection of Health Issues in Real-Time *Sensors*. https://pmc.ncbi.nlm.nih.gov/articles/PMC10255698/

[18] Wyant, K., Sant'Ana, S. J., Fronk, G. E., & Curtin, J. J. (2024). Machine learning models for temporally precise lapse prediction in alcohol use disorder. *Journal of psychopathology and clinical science*, 133(7), 527–540. https://doi.org/10.1037/abn0000901/

[19] Dhadwal, A. S. (2025). TS4H_4Hz_4Pages: Minimal 4 Hz pipeline for just-in-time relapse risk proxies from wearable time series. *GitHub repository*. https://github.com/AbhishekSinghDhadwal/TS4H_4Hz_4Pages

Appendix A: Additional analyses

A.1 Latency–performance trade-offs

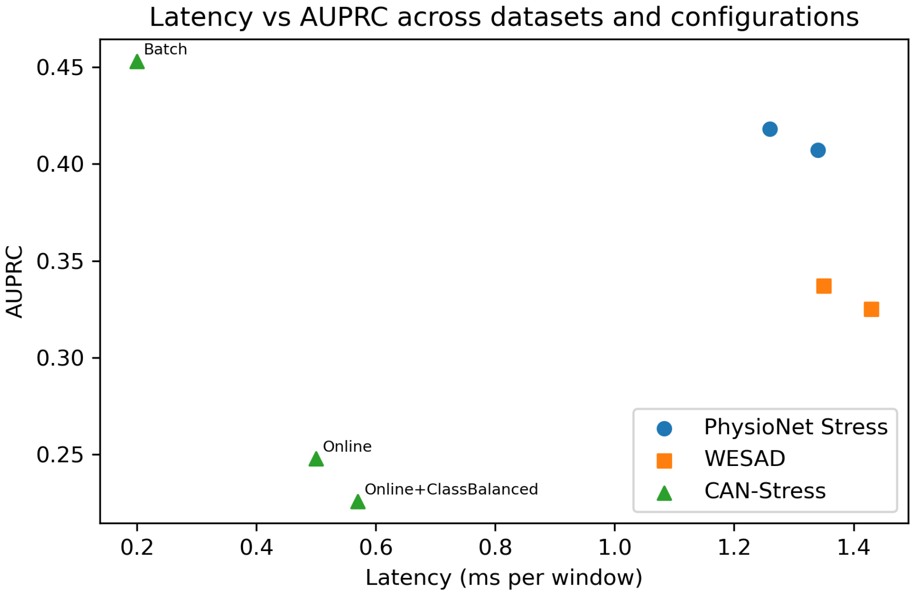

Figure A1: Latency vs AUPRC across all datasets and configurations. Each point corresponds to a configuration in Table 1. CAN-Stress configurations illustrate how the batch MiniRocket model achieves both the highest AUPRC and the lowest latency.

Figure A1 summarizes the latency–performance trade-off across all configurations in Table 1. All models operate in the 0.2–1.4 ms per-window range on the CPU-only hardware described in Section 4, and there is no configuration that sacrifices orders of magnitude in latency for marginal gains in AUPRC. In particular, the batch MiniRocket configuration on CAN-Stress achieves both the highest AUPRC and the lowest latency among all CAN-Stress models, reinforcing our choice of this configuration as the default deployment point under tight CPU and timing constraints.

A.2 Effect of stride on time-to-detection

Figure A2 shows how time-to-detection (TTD) varies with stride for batch MiniRocket configurations. On PhysioNet Stress, increasing stride from 5 to 10 seconds raises TTD from 0 to 10 seconds, while on WESAD the same change increases TTD from 5 to 70 seconds. In contrast, AUPRC and AUROC change only modestly (Table 1). This analysis confirms that evaluation protocols based solely on aggregate classification metrics may obscure large differences in temporal precision, and motivates reporting TTD alongside standard metrics for time-sensitive JITAI applications.

A.3 CAN-Stress configuration comparison

Figure A3 compares CAN-Stress performance across the three configurations evaluated: an online class-balanced compact-feature baseline, a batch MiniRocket model, and an online compact-feature model without class balancing. The batch MiniRocket model yields substantially higher AUROC and AUPRC than both online baselines, despite operating at lower per-window latency. These results suggest that simple deterministic transforms such as MiniRocket can serve as strong defaults for CPU-only JITAI pipelines, with online compact-feature models reserved for scenarios where model size or feature interpretability is more important than raw detection performance.

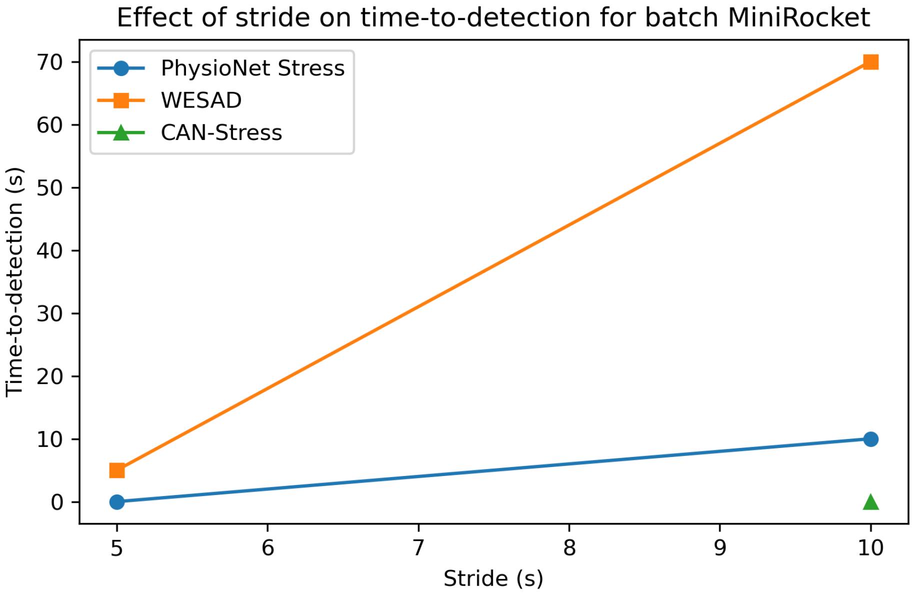

Figure A2: Effect of stride on time-to-detection (TTD) for batch MiniRocket configurations. On WESAD, increasing the stride from 5 to 10 seconds increases TTD from 5 to 70 seconds with only minor changes in AUPRC and latency (Table 1), illustrating how windowing choices can strongly influence temporal precision even when aggregate performance metrics are stable.

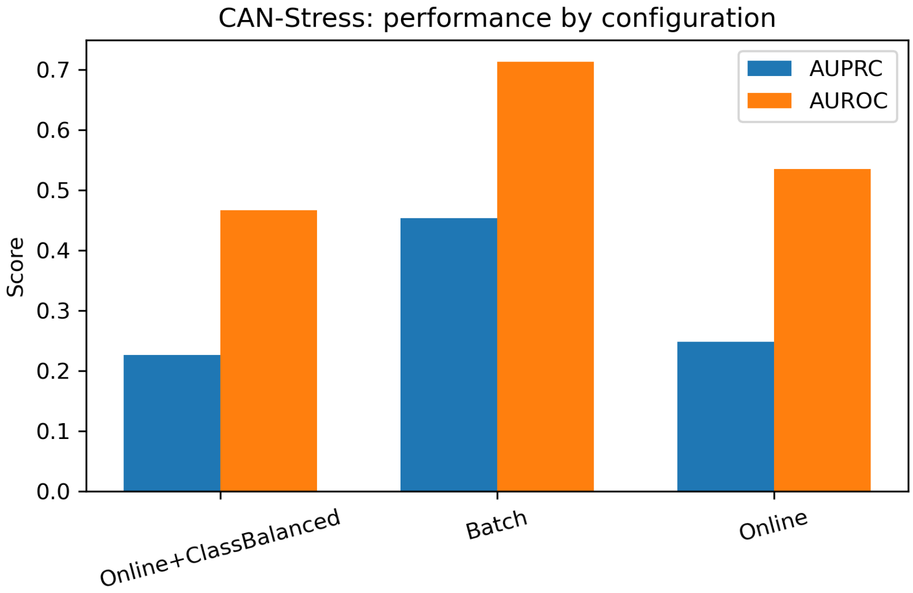

Figure A3: CAN-Stress performance by configuration. Batch MiniRocket achieves substantially higher AUROC and AUPRC than both online compact-feature baselines while retaining sub-millisecond per-window latency (Table 1). This supports our choice of MiniRocket as the default deployment configuration under CPU-only constraints.