

Received 10 April 2026, accepted 7 May 2026, date of publication 12 May 2026, date of current version 19 May 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3692752

RESEARCH ARTICLE

Machine Learning-Based Stress Detection Using Robust HRV Features and Person-Specific Normalization

MAXIMILIAN FECKE^{1,2} AND JAKOB REHOF^{2,3}

¹Department Healthcare, Fraunhofer Institute for Software and Systems Engineering ISST, 44147, Dortmund, Germany

²Chair for Software Engineering, Department of Computer Science, TU Dortmund University, 44227 Dortmund, Germany

³Lamarr Institute for Machine Learning and Artificial Intelligence, TU Dortmund University, 44227 Dortmund, Germany

Corresponding author: Maximilian Fecke (maximilian.fecke@isst.fraunhofer.de)

This work was supported in part by the German Federal Ministry of Education and Research (BMBF) through the Digitaler Engel project under Grant 02L18A080.

ABSTRACT This paper presents an artificial intelligence (AI) system for the electrocardiogram (ECG)-based detection of psychological stress, developed as part of a prototypical digital assistance system for nurses. The AI system employs the following key steps: After initial preprocessing including ECG windowing and heartbeat detection, 70 time, frequency, and time-frequency domain features are extracted from each window's heartbeats. All features are descriptors of heart rate variability (HRV), a well-known stress indicator. They are primarily calculated from sophisticated estimates of NN intervals, NN interval differences, respiration rates, and autonomic nervous system (ANS) activity. Novelty of this step include an increased noise robustness of estimated respiration rates and the derivation of ANS activity features from the integral pulse frequency modulation model of heartbeat generation. Furthermore, the entire feature extraction process is explicitly designed for robustness against irregularities like beat detection errors and ectopic beats, which could severely distort HRV analysis. All features are normalized with a novel, person-specific approach based on short baseline data sections. This lightweight personalization serves to increase classification performance while avoiding costly alternatives like person-specific modeling, and actively prevents normalization data leakage. Finally, stress detection is performed using a machine learning model of type random forest, gradient boosting machine, or multilayer perceptron. Hyperparameters of all steps were rigorously tuned using Bayesian optimization and manual search on the popular datasets WESAD and SWELL-KW as well as their union. The resulting stress detection performances under leave-one-subject-out cross-validation consistently surpassed the comparable state of the art.

INDEX TERMS Stress detection, electrocardiogram (ECG), heart rate variability (HRV), personalized feature normalization, machine learning, hyperparameter optimization, affective computing.

I. INTRODUCTION

A. CONTEXT

Psychological stress, hereafter referred to as stress, can be defined as the subjective perception that environmental demands exceed one's own adaptive capacity [1]. Frequent and chronic stress in particular, if not managed effectively, can lead to or exacerbate a variety of serious medical

issues. These include adverse mental health conditions, such as anxiety, depression, burnout, and dementia, as well as physical disorders, such as immune diseases, respiratory diseases, cardiovascular diseases, and cancer [1], [2], [3]. Due to its profound and increasing global impact, the World Health Organization has even called stress the health epidemic of the 21st century [4].

An occupational group that is particularly burdened by stress is nursing staff. In a 2023 meta-analysis covering 94 studies and 34 countries, Ge et al. concluded that

The associate editor coordinating the review of this manuscript and approving it for publication was Sangsoon Lim¹.

within the preceding ten years, the global prevalence of stress-induced burnout syndrome among nurses was about 30% and showed an overall increasing trend [5]. A major cause for this lies in the variety of physically, cognitively, and emotionally demanding tasks coupled with the high degree of responsibility that the nursing profession generally involves [6], [7]. Moreover, an acute and rising worldwide shortage in the qualified nursing workforce causes significantly increased workloads, time pressure, and overtime [8]. These factors frequently lead to nurses being overwhelmed and forced to compromise care quality, ultimately endangering the well-being of not only the nurses themselves but also their patients [9].

To address the stress burdens on nursing staff, a prototypical digital assistance system for nurses was created within the research project Digitaler Engel [10] (German for Digital Angel). The project was funded by the German Federal Ministry of Education and Research [11] and carried out by the research institutes Fraunhofer IML [12], Fraunhofer ISST [13], and IfADo [14] in collaboration with the Pius-Hospital Oldenburg [15] and the augmented reality solution provider TeamViewer Germany GmbH [16]. The assistance system, named Caring Environment (CE), is designed to support nurses in their daily work and in coping with the stress they experience. This paper is dedicated to the artificial intelligence (AI) system for the electrocardiogram (ECG)-based detection of stress that was devised, implemented, and optimized as an integral part of the CE.

ECGs were chosen as input for the AI system presented here, because they offer a particularly valuable foundation for deriving stress-related physiological indicators. Specifically, they are considered the most accurate basis for the analysis of heart rate variability (HRV), the variation in the length of time intervals between consecutive heartbeats [17], [18]. This is a widely researched indicator of autonomic nervous system (ANS) activity, which controls heartbeat generation and responds to stress level changes [19]. Moreover, in contrast to measurements of stress hormone concentrations in blood, urine, and saliva, which are considered the gold standard basis for stress detection, ECGs can be monitored continuously and unobtrusively throughout a nurse's workday using wearable sensors [20].

A nurse supported by the CE wears a chest-strapped, single-lead ECG sensor along with a pair of smart glasses. The ECG sensor continuously streams the recorded ECG to the CE app on the nurse's smartphone, which forwards it to the CE backend on a server. The backend then utilizes the developed AI system to analyze the received ECG data and infer stress labels from it, i.e., binary labels indicating the presence or absence of stress. On the one hand, whenever necessary, the backend derives individual recommendations for self-regulation and sends them to the smart glasses, where they are displayed to the nurse. On the other hand, it stores the predicted stress labels so they can later be used for departmental or individual stress analysis and optimization of

work organization. Further details on the CE's composition and its additional assistance functionalities, including the convenient entry and retrieval of patient and drug information via the smart glasses, can be found in the Digitaler Engel project's concluding white paper [21].

B. OVERVIEW

The developed AI system for ECG-based stress detection is composed of a pair of machine learning (ML) pipelines: A training pipeline and a corresponding inference pipeline. The training pipeline trains ML models based on ECGs and associated affective-state-indicating labels from the publicly available datasets Wearable Stress and Affect Detection (WESAD) [22] and Smart Reasoning for Well-being at Home and at Work - Knowledge Work (SWELL-KW) [23]. The inference pipeline, on the other hand, is part of the CE backend and continuously predicts stress labels from ECGs that are streamed to it. For this purpose, it employs the best ML model and hyperparameter (HP) assignment found by HP optimization procedures applied to the training pipeline. In this context, the term HP refers to parameters of ML pipelines as opposed to parameters of their utilized ML models, which are optimized by training [24]. The general structure of both pipelines and the applied HP optimization framework is graphically illustrated in Fig. 1. All implementations were made using Python [25].

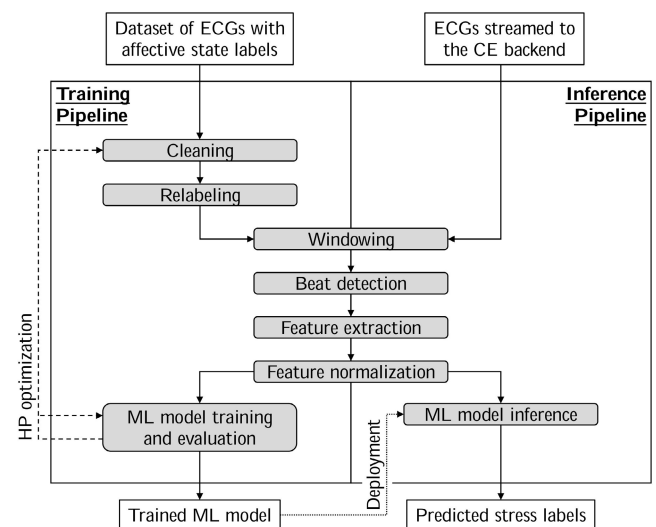


FIGURE 1. ML pipelines and HP optimization loops.

In the training pipeline, two dataset-specific preprocessing operations are initially applied to the input dataset, namely cleaning and relabeling. Their goal is to retrieve high-quality ECG sections with constant and valid stress labels. These ECG sections are then windowed to generate overlapping ECG windows of uniform duration. After that, a beat detection algorithm is applied to determine the timing of heartbeats in each of the resulting windows for subsequent feature extraction.

In total, 70 features are extracted based on each window's detected beat times and combined to create a corresponding feature vector. All of them are derived from literature-based estimates of different stress-related HRV signals, particularly the higher-frequency ANS modulation signal defined in terms of the integral pulse frequency modulation (IPFM) model [26]. However, the calculations are adjusted here to compensate the effects of irregular beats, such as beat detection errors and ectopic beats. By accounting for these anomalies, known to frequently occur and potentially cause major distortions especially in frequency and time-frequency domain HRV features [27], a robust foundation for stress detection is created.

Subsequently, all extracted features are person-specifically normalized to reduce inter-feature and inter-person variability of their distributions that might impair the training of ML models. The personalization is achieved by deriving the statistical key figures used as normalization parameters from the feature vectors of a short non-stressful baseline data segment for each person. Special care is taken to ensure that no data leakage [28] occurs as a result of this step.

Using the normalized feature vectors along with the associated stress labels, training and evaluation are performed for one of the selectable ML model types random forest (RF) [29], gradient boosting machine (GBM) [30], and multilayer perceptron (MLP) [31]. In particular, the F1-score [32] under leave-one-subject-out cross-validation (LOSO-CV) [33] is computed to assess the classification performance for data from unknown individuals. Ultimately, if desired, a corresponding final ML model is trained on all available samples and returned for possible deployment into the inference pipeline.

In the inference pipeline, windowing, beat detection, feature extraction, and feature normalization are applied to the input ECGs analogously to the training pipeline. After that, the normalized feature vectors are fed into the deployed ML model to predict their corresponding stress labels, which are finally returned.

The HP optimization framework utilized to find the best possible HP assignment and ML model for deployment is structured in two nested Bayesian optimization [34] loops applied to the training pipeline. The inner loop tunes HPs directly related to ML model training (Model HPs) while the outer one additionally tunes HPs of the preceding pipeline steps (Data HPs). This separation helps to efficiently reuse preprocessing and feature engineering results across the investigations of different HP assignments. To enable a comprehensive comparison with related work, HP optimization was performed not only for the combined dataset comprising both WESAD and SWELL-KW, but also for each of the two sub-datasets individually.

The remaining sections of this paper provide a more detailed look at all the above-mentioned aspects of the developed ML pipelines and the obtained results. First, Section II introduces the utilized datasets. Next, Section III

describes the applied dataset-dependent and -independent preprocessing operations. Sections IV and V then cover feature extraction and normalization, respectively. Following that, Section VI outlines the applied ML models, while Section VII details the evaluation of their classification performances. Section VIII then delineates the applied HP optimization procedures and Section IX presents their results. Subsequently, Section X provides a discussion including a comparison with related work, methodological considerations, and limitations. Lastly, Section XI offers a conclusion and outlook.

An overview of the central symbols and notations used throughout this paper can be found in Table 1.

II. DATA

The training pipeline's input consists of labeled ECGs originating from two of the most used datasets for the evaluation of ECG-based stress detection systems in current literature: WESAD [22] and SWELL-KW [23]. Both datasets contain single-channel ECGs and various other biosignals recorded from volunteers using wearable sensors in lab study environments. The ECGs were captured with chest-worn devices, namely a RespiBAN at 700 Hz in the WESAD study and a TMSi Mobi at 2048 Hz in the SWELL-KW study. In both cases, a standard three-point ECG electrode configuration across the chest was employed. Over the course of the two studies, the respective participants were sequentially subjected to different tasks and conditions to elicit different affective states in them as labels of the recorded data. Verification of the affective state labels was performed using questionnaires filled out by the participants.

In the WESAD study, the neutral affective state was established by reading selected neutral magazines, amusement by watching selected funny videos, and relaxation by performing guided meditations. Stress was attained by undergoing the Trier Social Stress Test, which inflicts high cognitive loads as well as social evaluative pressure and has been shown to be a reliable stressor [37]. With only three minutes of preparation time, the participants first had to give a comprehensive speech about their personal traits, strengths, and weaknesses. After that, they had to count backwards from 2023 in steps of 17 starting over every time they made a mistake. Both of these tasks were executed in front of a three-person jury that pretended to evaluate and compare the performances. The recorded dataset contains about 36 minutes of affective-state-labeled physiological data each for 15 study participants.

In the SWELL-KW study, participants performed office tasks on desktop computers, such as writing reports, preparing presentations, and responding to emails. First, neutrality was established by working on the tasks regularly. Subsequently, stress was attained by working under high time pressure or with frequent interruptions by emails containing distractions and additional assignments. In order to make the participants take their tasks seriously, they were given the false prospect of having to finish all of them in time and give

TABLE 1. Nomenclature of central mathematical symbols and notations.

Symbol	Description	Def. in Section
$d_{\text{trunc}}^{\text{WESAD}}, d_{\text{trunc}}^{\text{SWELL-KW}}$	ECG edge truncation durations for WESAD and SWELL-KW. (Data HPs)	III-A
d_{win}	ECG window duration. (Data HP)	III-C
DBT	Set of detected beat times.	III-D
stats(sig)	Statistics summary of a real-valued signal sig.	IV
θ	Maximum considered sequence length for irregularity identification. (Data HP)	IV-A
RBT, FPT, FNT, EBT, UBT	Sets of identified times of regular beats, false positives, false negatives, ectopic beats, and undefined irregular beats.	IV-A
$\widehat{\text{nn}}, \widehat{\text{nnd}}$	Estimated NN interval signal and NN interval difference signal.	IV-B, IV-C
$\widehat{\text{respr}}$	Estimator for the respiration rate signal. (Data HP)	IV-D
$\widehat{\text{respr}}^{\text{Sarkar}}, \widehat{\text{respr}}^{\text{HeartPy}}$	Estimators for the respiration rate signal based on Sarkar et al. [35] and HeartPy [36]. Potential values of $\widehat{\text{respr}}$.	IV-D
$T, \hat{T}$	IPFM model threshold and its estimate.	IV-E1, IV-E2
$\widehat{\text{mod}}, \widehat{\widehat{\text{mod}}}$	ANS modulation signal defined by the IPFM model, and its estimate.	IV-E1, IV-E2
$\widehat{\text{hmod}}, \widehat{\widehat{\text{hmod}}}$	Higher-frequency component of $\widehat{\text{mod}}$, and its estimate.	IV-E1, IV-E2
psd, spwvd	Power spectral density and smooth pseudo Wigner–Ville distribution of $\widehat{\text{hmod}}$.	IV-E3
$\text{psd_power}_{\text{band}}, \text{spwvd_power}_{\text{band}}$	Powers of $\widehat{\text{hmod}}$ in a given frequency band denoted as band, calculated from psd (scalar) and spwvd (time-varying).	IV-E3
$\text{psd_power}_{\text{band}_1}^{\text{band}_2}, \text{spwvd_power}_{\text{band}_1}^{\text{band}_2}$	Power ratios of $\widehat{\text{hmod}}$ between two given frequency bands band_1 and band_2 , calculated from psd (scalar) and spwvd (time-varying).	IV-E3
HF, LF, VLF, HLVLF	Standard high, low, and very low frequency bands for HRV analysis, and their union.	IV-E3
TOT	Total non-negative frequency band of $\widehat{\text{hmod}}$.	IV-E3
norm	Feature normalization function. (Data HP)	V-A
min_max, z_score, robust_scaler	Min-max, z-score, and robust scaler normalization functions. Potential values of norm.	V-A
d_{base}	Baseline duration for personalized feature normalization. (Data HP)	V-B

one of their prepared presentations in front of an audience to receive the full subject fee. SWELL-KW contains about 2 hours of data for each of 25 study participants.

Despite the limited scope and lack of representativeness of WESAD and SWELL-KW for the stress experienced by nursing staff, discussed further in Section X-C, they provide a valuable basis for preliminary optimization of the proposed ML pipelines. Furthermore, due to their popularity in related work, they allow a comparison with the approaches of many other researchers in terms of learning and generalization capability in Section X-A. Therefore, HP optimization and evaluation were performed individually for three input dataset configurations: WESAD, SWELL-KW, and their union, which is hereafter referred to as the combined dataset.

III. PREPROCESSING

The training pipeline's first two steps, cleaning and relabeling, are dataset-specific preprocessing operations designed to derive high-quality ECG segments with constant and valid stress labels from WESAD, SWELL-KW, or both. They are described in Sections III-A and III-B, respectively. Subsequently, Sections III-C and III-D cover windowing and heartbeat detection, the two dataset-independent preprocessing steps performed by both the training pipeline and the inference pipeline.

A. CLEANING

While the ECGs in WESAD are of consistently high quality, the usability of those in SWELL-KW varies. For 11 of the 25 participants of the SWELL-KW study, large portions of the recorded ECGs are either missing or too poor in quality to accurately detect heartbeats according to the dataset's accompanying Excel description file. The data from these participants is therefore eliminated in the cleaning step, leaving 14. Additionally, all data recorded outside the defined stages of the two studies is removed, so that each remaining ECG is uniformly associated with a corresponding affective state label.

The edges, meaning the leading and trailing sections, of the remaining ECGs from both datasets are truncated by fixed durations $d_{\text{trunc}}^{\text{WESAD}}, d_{\text{trunc}}^{\text{SWELL-KW}} \in \mathbb{R}_{\geq 0}$ in seconds, respectively. This is done to exclude sections in which the affective state targeted by a study stage may not have fully established or may be compromised due to anticipation of the stage's end. Examples of ECG edge truncation durations used in published work on the preprocessing of similarly recorded datasets for ML include 60 s [38] and 120 s [39]. Most authors, however, entirely refrained from truncating ECG edges [40], [41], [42]. Therefore, in order to optimize the cleaning step, $d_{\text{trunc}}^{\text{WESAD}}$ and $d_{\text{trunc}}^{\text{SWELL-KW}}$ were treated as HPs and initially tuned in the domain $\{0 \text{ s}, 10 \text{ s}, \dots, 120 \text{ s}\}$. An additional manual fine-tuning also

examined individual integer truncation durations that are not multiples of 10 s.

B. RELABELING

In the relabeling step, the affective state labels of the cleaned ECGs are replaced with stress labels in $\{0, 1\}$. Here, 0 represents the stress-negative class and 1 the stress-positive class. The relabeling is achieved by mapping the non-stressful affective states neutrality, amusement, and relaxation to 0 while mapping stress to 1. After this step, the proportion of ECG data associated with the label 1 is about 22% for WESAD, 63% for SWELL-KW, and 53% for the combined dataset.

C. WINDOWING

The windowing step generates overlapping ECG windows of uniform duration and regular spacing. In the training pipeline, these windows are created from cleaned, relabeled ECGs originating from WESAD or SWELL-KW, while in the inference pipeline, they are created directly from each input ECG. This step is crucial for several reasons. First, it ensures that all subsequently extracted feature vectors are based on the same amount of ECG data, providing a consistent basis for classification. Second, it reduces the amount of ECG data required for a single prediction, enabling a more temporally specific analysis. Finally, it significantly increases the volume of available training data.

A window spacing of 1 s was chosen as a compromise between high resolution serving the generation of extensive training data, and maintaining a manageable computational cost for training. In contrast, the window duration, denoted as d_{win} and measured in seconds, was optimized. Increasing d_{win} means mapping longer time periods to single stress labels, which may be less accurate due to the fluctuation of stress over time. Furthermore, larger values of d_{win} lead to fewer feature vectors for training and temporally broader influences of ECG measurement errors and irregularities, potentially hindering accurate predictions. However, longer ECG windows may also improve prediction quality by incorporating more information into each HRV feature vector. In the literature on the continuous detection of stress using HRV features, utilized window durations are often between 60 s [41], [43] and 300 s [44], [45]. Therefore, d_{win} was included in the pipeline HPs and tuned in the domain $\{60 \text{ s}, 70 \text{ s}, \dots, 300 \text{ s}\}$. Similar to the optimization of $d_{\text{trunc}}^{\text{WESAD}}$ and $d_{\text{trunc}}^{\text{SWELL-KW}}$, individually selected integer values of d_{win} that are not divisible by 10 were additionally considered during manual fine-tuning.

D. BEAT DETECTION

To detect heartbeats, or beats for short, in each ECG window as the foundation for HRV feature extraction, both pipelines employ the widely used Hamilton-Tompkins algorithm [46], as implemented in the Python biosignal processing library BioSPPy [47]. The algorithm identifies beats in its input

ECG by locating R-peaks. Each R-peak is used to represent a single beat, as it marks the initiation of ventricular contraction and typically constitutes the most prominent peak within the beat's waveform. The algorithm first applies a sequence of transformations, including band-pass filtering, differentiation, and moving-window averaging, in order to further highlight the R-peaks. Subsequently, it determines their precise timing using local maximum detection and data-adaptive thresholding. For the purposes of this paper, the resulting set of detected beat times in seconds for a given ECG window is denoted as $\text{DBT} = \{\text{dbt}_0 < \text{dbt}_1 < \dots < \text{dbt}_{|\text{DBT}|-1}\} \subset \mathbb{R}_{\geq 0}$.

IV. FEATURE EXTRACTION

In both ML pipelines, 70 features are extracted from each ECG window's detected beat times and aggregated into a corresponding feature vector. A summary of these features is given in Table 2. All features are closely related to HRV and thus potentially valuable for ANS activity analysis and stress detection. Many standard HRV measures from the literature are covered. To ultimately increase their robustness against irregularities in the detected beat times, an algorithm for the identification of regular and irregular beats is first applied. This algorithm is introduced along with the features extracted from its parameters and results in Section IV-A.

All remaining features are extracted from irregularity-corrected estimates of commonly studied HRV signals, hereafter referred to as HRV signals of interest. Sections IV-B and IV-C introduce features based on the estimated NN interval signal and NN interval difference signal, respectively. Subsequently, Section IV-D covers features related to the estimated respiration rate signal, which can be inferred from estimated NN intervals due to the physiological coupling between cardiac and respiratory cycles. Finally, Section IV-E addresses features extracted from the estimated higher-frequency ANS modulation signal, which directly relates ANS activity to beat generation and HRV according to the IPFM model. Given the high relevance of this signal for stress detection, features are extracted from its time-domain representation, as well as its frequency and time-frequency domain representations. Irregularity corrections are particularly beneficial here, as even isolated irregular beats in the time domain can cause significant distortions in the other two domains [27].

Many extracted features are basic statistics describing the HRV signals of interest. They are specified using the statistics summary function stats , defined for any real-valued signal sig as

$$\begin{aligned} \text{stats}(\text{sig}) = & (\mu(\text{sig}), \sigma(\text{sig}), \\ & \text{quart}_1(\text{sig}), \text{quart}_2(\text{sig}), \text{quart}_3(\text{sig})). \end{aligned} \quad (1)$$

Here, $\mu(\text{sig})$ and $\sigma(\text{sig})$ are sig 's mean and standard deviation, while $\text{quart}_1(\text{sig})$, $\text{quart}_2(\text{sig})$, and $\text{quart}_3(\text{sig})$ are its quartiles. These five statistics were selected for their ability to

TABLE 2. HRV feature vector extracted from the detected beat times of each ECG window.

Index	Feature	Description	Def. in Section
0	H	Maximum absolute heart rate derivative considered regular.	IV-A
1 - 5	$ RBT $, $ FPT $, $ FNT $, $ EBT $, $ UBT $	Numbers of identified regular beats, FPs, FNs, ectopic beats, and undefined irregular beats.	IV-A
6	n_{irreg}	Total number of identified irregular beats.	IV-A
7	$\max_{\text{irreg_seq}}$	Maximum length of any identified irregular beat sequence without regular beats in between.	IV-A
8 - 12	$\text{stats}(\widehat{nn})$	Statistics summary of the estimated NN interval signal.	IV-B
13 - 15	cvnn , hti , tinn	Coefficient of variation, HRV triangular index, and triangular interpolation index of the estimated NN interval signal.	IV-B
16 - 20	$\text{stats}(\widehat{nn_d})$	Statistics summary of the estimated NN interval difference signal.	IV-C
21 - 23	rmssd , pnn_{25} , pnn_{50}	Root mean square of estimated NN interval differences and relative amounts of estimated NN interval differences greater than or equal to 0.025 s and 0.05 s.	IV-C
24 - 28	$\text{stats}(\widehat{\text{respr}})$	Statistics summary of the estimated respiration rate signal.	IV-D
29 - 33	$\text{stats}(\widehat{hmod})$	Statistics summary of the estimated higher-frequency ANS modulation signal $\widehat{hmod}$.	IV-E3
34	$\text{psd_power}_{\text{TOT}}$	Total PSD power of $\widehat{hmod}$.	IV-E3
35	$\text{psd_power}_{\widehat{hmod}}^{\text{HLVLF}}$	Proportion of $\widehat{hmod}$'s PSD power in the frequency band HLVLF.	IV-E3
36 - 38	$\text{psd_power}_{\widehat{hmod}}^{\text{HF}}$, $\text{psd_power}_{\widehat{hmod}}^{\text{LF}}$, $\text{psd_power}_{\widehat{hmod}}^{\text{VLF}}$	Proportions of $\widehat{hmod}$'s HLVLF PSD power in HF, LF, and VLF.	IV-E3
39	$\text{psd_power}_{\widehat{hmod}}^{\text{LF}} / \text{psd_power}_{\widehat{hmod}}^{\text{HF}}$	Ratio between $\widehat{hmod}$'s PSD powers in LF and HF.	IV-E3
40 - 44	$\text{stats}(\text{spwvd_power}_{\text{TOT}})$	Statistics summary of $\widehat{hmod}$'s total SPWVD power over time.	IV-E3
45 - 49	$\text{stats}(\text{spwvd_power}_{\widehat{hmod}}^{\text{HLVLF}})$	Statistics summary of $\widehat{hmod}$'s SPWVD power proportion in HLVLF over time.	IV-E3
50 - 64	$\text{stats}(\text{spwvd_power}_{\widehat{hmod}}^{\text{HF}})$, $\text{stats}(\text{spwvd_power}_{\widehat{hmod}}^{\text{LF}})$, $\text{stats}(\text{spwvd_power}_{\widehat{hmod}}^{\text{VLF}})$	Statistics summaries of $\widehat{hmod}$'s HLVLF SPWVD power proportions in HF, LF, and VLF over time.	IV-E3
65 - 69	$\text{stats}(\text{spwvd_power}_{\widehat{hmod}}^{\text{LF}} / \text{spwvd_power}_{\widehat{hmod}}^{\text{HF}})$	Statistics summary of the ratio between $\widehat{hmod}$'s SPWVD powers in LF and HF over time.	IV-E3

capture key signal characteristics and their established use in biosignal-based machine learning tasks [48]. Notably, $\text{stats}(\text{sig})$ replaces the minimum and maximum of sig 's standard five-number summary with the mean and standard deviation, shifting the focus from outliers to the overall distribution for increased robustness.

While some of the utilized signals and features are more frequently defined in milliseconds than in seconds in the literature, seconds are used throughout here. This solely serves the purpose of consistency and does not affect ML model training and evaluation when feature normalization is carried out beforehand.

A. BEAT TYPE FEATURE EXTRACTION

Following Mateo and Laguna [27], three common types of irregular beats are distinguished: False positives (FPs), false negatives (FNs), and ectopic beats. FPs and FNs are beat detection errors in the form of spurious and missed beats, respectively. Ectopic beats, in contrast, are a common form of cardiac arrhythmia where the electrical impulse initiating a beat originates outside the sinoatrial node, the heart's natural pacemaker.

The particular type of ectopic beat considered here is the supraventricular ectopic beat. It is characterized by the triggering electrical impulse arising in the atria or the atrioventricular node. Due to the proximity of the impulse's origin to the sinoatrial node, the latter's periodic triggering activity is typically "reset" when this occurs, resembling the effect of a regular beat. In simplified terms, this results in all subsequent beats being slightly delayed. For a ventricular ectopic beat, on the other hand, the triggering impulse arises in the heart's ventricles and does not reset the sinoatrial node's activity. Ventricular ectopic beats are not explicitly considered here but instead treated as FPs, since they typically manifest as extra beats without significantly altering the timing of other beats.

To identify FPs, FNs, and ectopic beats, an algorithm proposed by Mateo and Laguna [27] is employed. The algorithm relies on a regularity condition that constrains the physiologically plausible acceleration and deceleration of the heart rate. Specifically, it imposes an upper threshold $H \in \mathbb{R}_{>0}$ on the absolute heart rate derivative. Let $\text{BT} = \{bt_0 < bt_1 < \dots < bt_{|\text{BT}|-1}\} \subset \mathbb{R}_{\geq 0}$ represent a set of beat times. As demonstrated by Mateo and Laguna, the derivative

of the heart rate signal induced by BT can be accurately estimated for any inner beat time bt_j using the following equation derived from the IPFM model:

$$\widehat{hr'}(bt_j) = 2 \cdot \frac{bt_{j-1} - 2bt_j + bt_{j+1}}{(bt_{j-1} - bt_j)(bt_{j-1} - bt_{j+1})(bt_j - bt_{j+1})}. \quad (2)$$

Based on the derivative estimate, the regularity condition reg_H is defined as

$$reg_H(bt_j) = \begin{cases} \text{True} & \text{if } |\widehat{hr'}(bt_j)| \leq H \\ \text{False} & \text{else.} \end{cases} \quad (3)$$

Mateo and Laguna further demonstrated empirically that optimal identification of irregular beats is achieved when H is set proportionally to the standard deviation $\sigma(|\widehat{hr'}|)$ of the estimated absolute heart rate derivatives as

$$H = \min(\{0.5, 4.3 \cdot \sigma(|\widehat{hr'}|)\}). \quad (4)$$

Given the set DBT of detected beat times for an ECG window, the algorithm proceeds as follows: The beat times are traversed in ascending order. If a beat time $bt_j \in \text{DBT}$ satisfies the condition $reg_H(bt_j)$, it is marked as regular. Otherwise, the algorithm attempts to correct the violation by iteratively considering potential sequences of FPs, FNs, and ectopic beats up to a maximum length defined by a parameter $\theta \in \mathbb{N}_{\geq 0}$. These sequences are simulated by deleting, inserting, and shifting beat times $\geq bt_j$, respectively. If a sequence of irregular beats is found that resolves the violation of $reg_H(bt_j)$, the corresponding beat times are marked accordingly. If not, bt_j is marked as undefined. This process ultimately yields the sets RBT, FPT, FNT, EBT, and UBT, containing the identified times of regular beats, FPs, FNs, ectopic beats, and undefined irregular beats, respectively. As an example, Fig. 2 illustrates the algorithm's results for the detected beat times of two short ECG window sections. Since Mateo and Laguna did not provide a recommended value for the algorithm's parameter θ , it was treated as an HP and tuned within the range $\{0, 1, \dots, 5\}$.

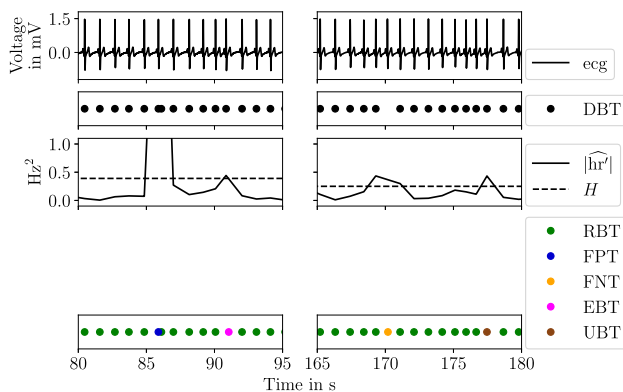


FIGURE 2. Results of Mateo and Laguna's irregularity identification algorithm for the detected beat times of two 30 s ECG window sections: Estimated absolute heart rate derivatives, corresponding thresholds, and identified times of regular beats, FPs, FNs, ectopic beats, and undefined irregular beats, respectively.

The beat type features extracted from each window include H , the maximum permissible absolute heart rate derivative used by the algorithm, along with the cardinalities $|RBT|$, $|FPT|$, $|FNT|$, $|EBT|$, and $|UBT|$ of the identified beat time sets. In addition, the total number of identified irregular beats $n_{\text{irreg}} = |FPT| + |FNT| + |EBT| + |UBT|$ and the maximum length of any identified sequence of irregular beats without regular beats in between, denoted as $\max_{\text{irreg_seq}}$, are extracted as well. Although n_{irreg} may appear redundant given its derivation from other features, it is utilized here because it noticeably improved classification performance during early experiments.

B. NN INTERVAL FEATURE EXTRACTION

The first HRV signal of interest is the NN interval signal, indicating the time period in seconds between each pair of consecutive normal beats [49].¹ It is estimated based on all pairs of identified consecutive regular beat times $rbt_{j-1}, rbt_j \in \text{RBT}$ between which there lie no identified irregular beat times in any of FPT, FNT, EBT, and UBT [27]:

$$\widehat{nn}(rbt_j) = rbt_j - rbt_{j-1}. \quad (5)$$

Features capturing general statistical information about $\widehat{nn}$ are computed in the form of its statistics summary $\text{stats}(\widehat{nn})$. Moreover, three well-established HRV features from the literature on stress and cardiovascular system analysis are calculated. The first of these is the coefficient of variation (CVNN), indicating the ratio between $\widehat{nn}$'s standard deviation and its mean [50]:

$$cvnn = \frac{\sigma(\widehat{nn})}{\mu(\widehat{nn})}. \quad (6)$$

The other two $\widehat{nn}$ -based HRV features are geometric measures derived from $\widehat{nn}$'s histogram hist , which represents the distribution of estimated NN intervals over bins of size $\frac{1}{128}$ s [51]. On the one hand, the HRV triangular index (HTI) is computed as the inverse proportion of estimated NN intervals in the most common bin, i.e.,

$$hti = \frac{|\text{dom}(\widehat{nn})|}{\max(\text{hist})}, \quad (7)$$

where $\text{dom}(\widehat{nn})$ denotes $\widehat{nn}$'s domain. On the other hand, the triangular interpolation index (TINN) is computed. For this, the histogram is first approximated by a triangular function tri . Its top corner is placed at the histogram's maximum point. Its side corners are placed on the x-axis at time points t_{left} and t_{right} to the left and right of the top corner, respectively, such that the sum of squared differences between tri and hist is minimized. Then, the TINN is computed as tri 's baseline width, i.e.,

$$tinn = t_{\text{right}} - t_{\text{left}}. \quad (8)$$

¹In mathematical terms, an NN interval is therefore not an actual interval, but an interval length. Common synonyms include normal/regular RR interval (R refers to ECG R-peak), interbeat interval, beat-to-beat interval, heart period, and cardiac cycle length.

C. NN INTERVAL DIFFERENCE FEATURE EXTRACTION

The second HRV signal of interest is the NN interval difference signal [51]. It indicates the difference between each pair of NN intervals occurring in direct succession. The signal is estimated based on all pairs of adjacent estimated NN intervals $\widehat{nn}(rbt_{j-1})$ and $\widehat{nn}(rbt_j)$ as

$$\widehat{nnd}(rbt_j) = \widehat{nn}(rbt_j) - \widehat{nn}(rbt_{j-1}). \quad (9)$$

As with $\widehat{nn}$, features are extracted from $\widehat{nnd}$ in the form of its statistics summary $\text{stats}(\widehat{nnd})$ as well as three common HRV measures from the literature. The three HRV measures first include the root mean square of successive NN interval differences (RMSSD) [52]:

$$\text{rmssd} = \sqrt{\frac{1}{|\text{dom}(\widehat{nnd})|} \cdot \sum_{rbt_j \in \text{dom}(\widehat{nnd})} \widehat{nnd}(rbt_j)^2}. \quad (10)$$

Additionally, they include the proportion of estimated NN interval differences that are at least 0.025 s and 0.05 s, respectively [44]:

$$\text{pnn}_{25} = \frac{|\{rbt_j \in \text{dom}(\widehat{nnd}) \mid \widehat{nnd}(rbt_j) \geq 0.025 \text{ s}\}|}{|\text{dom}(\widehat{nnd})|} \quad (11)$$

and

$$\text{pnn}_{50} = \frac{|\{rbt_j \in \text{dom}(\widehat{nnd}) \mid \widehat{nnd}(rbt_j) \geq 0.05 \text{ s}\}|}{|\text{dom}(\widehat{nnd})|}. \quad (12)$$

D. RESPIRATION RATE FEATURE EXTRACTION

The next HRV signal of interest is the respiration rate signal, which specifies the number of breaths taken per second. Numerous studies have shown correlations between respiration rate patterns and stress levels. Generally, both respiration rate and its variability tend to increase with stress and decrease with mental relaxation. The respiration rate signal can be estimated based on the available HRV data due to the phenomenon of respiratory sinus arrhythmia, which consists of the heart rate rising during inhalation and falling during exhalation. [53]

Two literature-based approaches for the estimation of respiration rates from HRV data were implemented and compared. Both of them first infer a continuous-time respiration signal estimate $\widehat{\text{resp}}$, whose local peak times indicate the predicted boundaries of respiration cycles. Subsequently, they directly estimate the respiration rate signal $\widehat{\text{respr}}$ in Hz from these peak times. This scheme was put forward by Sarkar et al. [35]. For the ML pipelines presented here, adjustments were made so that $\widehat{\text{resp}}$ and $\widehat{\text{respr}}$ are estimated exclusively from time periods without identified irregular beat times to improve robustness. Fig. 3 shows the results of both approaches based on the regular and irregular beat times identified for a 60 s window.

The first approach directly builds upon the methodology of Sarkar et al., who proposed estimating the respiration signal using a heart rate signal estimate after cubic spline interpolation [35]. Here, the heart rate signal is estimated as

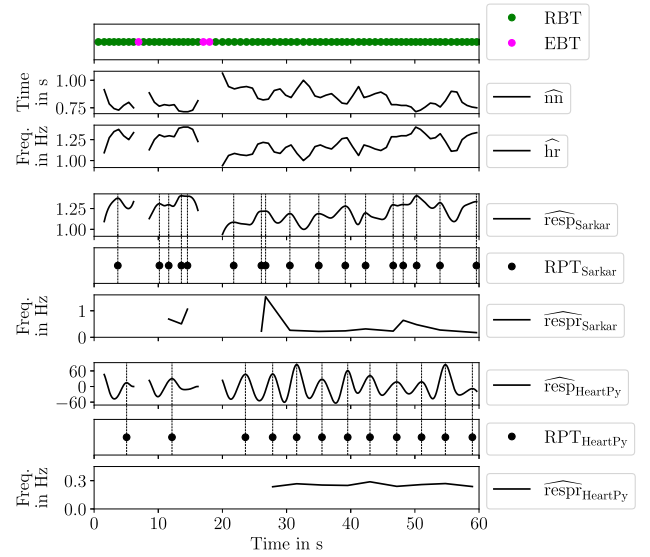


FIGURE 3. Results of the two respiration rate estimation approaches following Sarkar et al. and HeartPy, respectively, for the identified regular and irregular (ectopic) beat times of a 60 s window: Estimated NN intervals, heart rates, respirations, respiration peaks, and respiration rates.

the reciprocal of $\widehat{nn}$, i.e., for any identified regular beat time rbt_j in $\text{dom}(\widehat{nn})$,

$$\widehat{hr}(rbt_j) = \frac{1}{\widehat{nn}(rbt_j)}. \quad (13)$$

Furthermore, the cubic spline interpolation step is applied here only to sections of $\widehat{hr}$ in which there lie no identified irregular beat times in any of FPT, FNT, EBT, and UBT. The resulting piecewise continuous respiration signal estimate is denoted $\widehat{\text{resp}}_{\text{Sarkar}}$. Then, the set $\text{RPT}_{\text{Sarkar}} \subset \mathbb{R}_+$ of this signal's local peak times is determined and utilized for respiration rate estimation as follows: Based on all pairs of successive respiration peak times $\text{rpt}_{j-1}, \text{rpt}_j \in \text{RPT}_{\text{Sarkar}}$ lying in the same continuous section of $\widehat{\text{resp}}_{\text{Sarkar}}$, the respiration rate signal estimate is calculated as

$$\widehat{\text{respr}}_{\text{Sarkar}}(\text{rpt}_j) = \frac{1}{\text{rpt}_j - \text{rpt}_{j-1}}. \quad (14)$$

During experiments on the available datasets, this first approach often yielded unnaturally high or variable estimated respiration rates due to the presence of numerous local peaks in $\widehat{\text{resp}}_{\text{Sarkar}}$ resembling noise. Therefore, a modified version of the approach was investigated as well to potentially achieve greater noise robustness. The modification consists in replacing the estimation of the respiration signal with the procedure used by the established Python heart rate analysis library HeartPy within the function `calc_breathing` [36], [54]. Instead of the heart rate signal, HeartPy relies on the heart period signal, which is represented here by the more irregularity-robust $\widehat{nn}$. Similar to Sarkar et al., HeartPy then continualizes the signal using cubic spline interpolation. In addition, however, it applies a Butterworth bandpass filter

with the cutoff frequencies 0.1 Hz and 0.4 Hz to remove noise and other frequency content unrelated to respiration. Here, similar to the first approach, the interpolation and subsequent filtering are performed section-wise, to exclude the influences of identified irregular beats. The resulting alternative respiration signal estimate is denoted $\widehat{\text{resp}}_{\text{HeartPy}}$. The set of its local peak times and the induced respiration rate signal estimate, computed analogously to the first approach, are denoted $\text{RPT}_{\text{HeartPy}}$ and $\widehat{\text{respr}}_{\text{HeartPy}}$, respectively.

After respiration rate estimation, features are computed from $\widehat{\text{respr}} \in \{\widehat{\text{respr}}_{\text{Sarkar}}, \widehat{\text{respr}}_{\text{HeartPy}}\}$ in the form of its statistics summary $\text{stats}(\widehat{\text{respr}})$. In order to determine which of the two approaches yields the most useful features for stress detection, $\widehat{\text{respr}}$ was optimized as HP.

E. ANS MODULATION FEATURE EXTRACTION

The final HRV signal of interest is the higher-frequency ANS modulation signal defined within the framework of the IPFM model. This signal offers unique insights into the activity of the ANS and its responses to stress. To capture these complex dynamics, features are extracted not only from the time-domain representation but also from frequency- and time-frequency domain representations of this signal's estimate. To the best knowledge of this work's authors, there is no published work to date that uses features based on the IPFM model for stress detection or any other ML task.

Section IV-E1 first introduces the IPFM model and its associated ANS modulation signals as a theoretical foundation. Section IV-E2 subsequently outlines how the desired modulation signal is estimated based on identified regular and irregular beat times. Finally, Section IV-E3 provides the features that are extracted from the estimate.

1) IPFM MODEL AND ANS MODULATION

The IPFM model mathematically describes the generation of beats by the human heart under the regulation of the ANS. Originally proposed by Hyndman and Mohn in 1973 [55], the model remains a widely applied tool for HRV-based analysis of ANS activity and its relation to different mental and physical conditions [56], [57]. Fig. 4 shows the model as a block diagram following Bailón et al. [26].

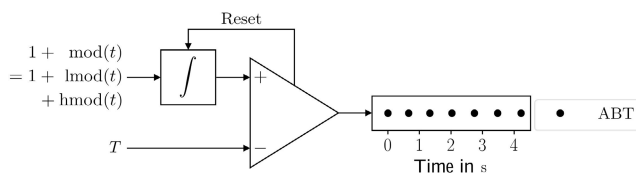


FIGURE 4. IPFM model of beat generation by the human heart, adapted from Bailón et al. [26].

Within the IPFM model, the ANS's influences on the heart are represented by a continuous modulation signal mod with a value range of $[-1, 1]$, a mean value of 0, and a frequency band limited to $[0 \text{ Hz}, 0.4 \text{ Hz}]$. The signal indicates variations in the speed at which electric charge accumulates in the

sinoatrial node, thus causing the periodic generation of beats. The first beat is triggered at 0 s. After that, a new beat is triggered whenever the integral of $1 + \text{mod}$ reaches a threshold $T \in \mathbb{R}_{>0}$, indicating the overall mean NN interval. When this occurs, the integral is reset to zero.

Formally, this behavior can be defined as follows: Given the set $\text{ABT} = \{\text{abt}_0 < \text{abt}_1 < \dots\} \subset \mathbb{R}_{\geq 0}$ of actually occurring beat times in seconds, each corresponding index $j \in \mathbb{N}_{\geq 0}$ satisfies

$$j = \int_0^{\text{abt}_j} \frac{1 + \text{mod}(t)}{T} dt. \quad (15)$$

Bailón et al. further refined the IPFM model by suggesting that mod can be decomposed into two distinct modulation signals with meaningful interpretations [26]:

$$\text{mod}(t) = \text{lmod}(t) + \text{hmod}(t). \quad (16)$$

In this, lmod and hmod are zero-mean signals that share mod 's domain and value range but are limited to non-overlapping sub-bands of $[0 \text{ Hz}, 0.4 \text{ Hz}]$. The lower-frequency modulation signal lmod represents variation of the mean NN interval over time and is strongly influenced by factors not directly related to the mental state, such as physical activity. In contrast, the higher-frequency modulation signal hmod contains accessible information on the activity and interplay of the sympathetic nervous system (SNS) and the parasympathetic nervous system (PNS), the two main branches of the ANS, which are directly linked to stress: The SNS drives the body's fight-or-flight responses to stress, whereas the PNS drives opposing rest-and-digest responses, primarily during calmer mental conditions [18], [58]. Thus, hmod provides a particularly valuable basis for the extraction of features for stress detection.

2) ANS MODULATION ESTIMATION

Given the identified regular and irregular beat times for an ECG window, mod is estimated using an elaborate approach by Mateo and Laguna [27]. In essence, the approach builds on (15) to compute the estimate $\widehat{\text{mod}}$ locally at the identified regular beat times and uses spline interpolation to obtain a continuous signal. Furthermore, it applies sophisticated corrections of the estimate for all errors presumed due to the identified irregular beat times. This is particularly important since even small local distortions of mod in the time domain often cause large broadband distortions in the frequency and time-frequency domains. Further details on the estimation are omitted here for the sake of brevity.

From mod , hmod is estimated following Bailón et al. [26]. They showed the utilized method to perform best among multiple techniques, also including simple high-pass filtering, in a simulation study. The method is built upon the continuous heart rate signal chr , which is defined as the derivative of the IPFM model's continuous beat index function induced by (15):

$$\text{chr}(t) = \frac{1 + \text{mod}(t)}{T}. \quad (17)$$

Its estimate $\widehat{\text{chr}}$ is computed by inserting $\widehat{\text{mod}}$ along with $\widehat{T}$, the estimated overall mean NN interval involved in the derivation of $\widehat{\text{mod}}$. Using $\widehat{\text{chr}}$, $\widehat{\text{hmod}}$ is estimated by

$$\widehat{\text{hmod}}(t) = \frac{\widehat{\text{chr}}(t) - \text{filt}_{<0.03}(\widehat{\text{chr}}(t))}{\text{filt}_{<0.03}(\widehat{\text{chr}}(t))}, \quad (18)$$

where $\text{filt}_{<0.03}$ is the ideal low-pass filtering function with the exclusive upper frequency bound 0.03 Hz. The calculation can be seen as an approximate high-pass filtering of $\widehat{\text{mod}}$ with 0.03 Hz as lower frequency bound. Analogous to Bailón et al. [26] and also Hernando et al. [39], the employed sampling frequency for both $\widehat{\text{mod}}$ and $\widehat{\text{hmod}}$ is 4 Hz.

3) ANS MODULATION FEATURES

Based on the estimated higher-frequency ANS modulation signal $\widehat{\text{hmod}}$, features are extracted in the time domain, the frequency domain, and the time-frequency domain. The utilized representations of $\widehat{\text{hmod}}$ in all three domains are illustrated for a 60 s window in Fig. 5.

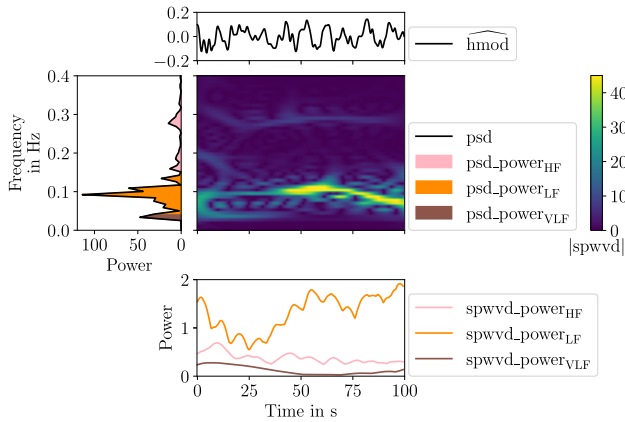


FIGURE 5. Estimated higher-frequency ANS modulation signal for a 60 s window, its PSD, its absolute SPWVD, and the corresponding powers in the HF, LF, and VLF bands.

First, time domain features are extracted from $\widehat{\text{hmod}}$ itself in the form of $\text{stats}(\widehat{\text{hmod}})$. Subsequently, frequency domain features are extracted from $\widehat{\text{hmod}}$'s power spectral density (PSD), a well-established representation for the analysis of different ANS modulation frequency components [27], [59]. The PSD is computed based on $\widehat{\text{hmod}}$'s discrete Fourier transform $\mathcal{F}$ as

$$\text{psd}(f) = |\mathcal{F}(f)|^2 \quad (19)$$

and indicates the distribution of power, i.e., variance, in $\widehat{\text{hmod}}$ over its constituting frequencies [51], [60].

Features extracted from the PSD are defined in terms of absolute and relative powers in different frequency bands. For any frequency band, denoted as band , the comprised power is computed by integrating the linearly interpolated PSD psd_{lin} :

$$\text{psd_power}_{\text{band}} = \int_{\min(\text{band})}^{\max(\text{band})} \text{psd}_{\text{lin}}(f) df. \quad (20)$$

Moreover, the power ratio between two frequency bands band_1 and band_2 is defined as

$$\text{psd_power}_{\text{band}_1}^{\text{band}_2} = \frac{\text{psd_power}_{\text{band}_1}}{\text{psd_power}_{\text{band}_2}}. \quad (21)$$

In particular, three standard HRV analysis frequency bands are considered, which have been found to provide insight into the activity of the SNS and the PNS [61]:

$$\begin{aligned} \text{HF} &= [0.15 \text{ Hz}, 0.4 \text{ Hz}], \\ \text{LF} &= [0.04 \text{ Hz}, 0.15 \text{ Hz}], \text{ and} \\ \text{VLF} &= [0.003 \text{ Hz}, 0.04 \text{ Hz}]. \end{aligned} \quad (22)$$

While HF band power primarily reflects PNS activity, LF band power is mainly influenced by the SNS, with additional contributions from the PNS and blood pressure control mechanisms. Power in the VLF band may originate from various sources including SNS, PNS, the heart's intrinsic nervous system, thermoregulation processes, and physical activity [18], [58]. Due to the approximate high-pass filtering with the cutoff-frequency 0.03 Hz involved in the computation of $\widehat{\text{hmod}}$, VLF band power is effectively mainly limited to [0.03 Hz, 0.04 Hz] and potentially has a closer relation to SNS and PNS.

The first feature extracted from the PSD is its overall power $\text{psd_power}_{\text{TOT}}$. Here, TOT denotes the signal's entire non-negative frequency band, which according to the Nyquist–Shannon sampling theorem [62] is [0 Hz, 2 Hz] due to $\widehat{\text{hmod}}$'s sampling frequency being 4 Hz. Though the frequency content of $\widehat{\text{hmod}}$ is approximately limited to [0.03 Hz, 0.4 Hz] by design, residual frequency components below and above may exist in practice. In addition, the following relative PSD band powers are computed: To capture the proportion of power that falls into $\text{HLVLF} = [0.003 \text{ Hz}, 0.4 \text{ Hz}]$, the union of all three considered standard HRV analysis bands, $\text{psd_power}_{\text{TOT}}^{\text{HLVLF}}$ is calculated. Furthermore, the proportion of that power that falls into each of the three frequency bands is captured by $\text{psd_power}_{\text{HLVLF}}^{\text{HF}}$, $\text{psd_power}_{\text{HLVLF}}^{\text{LF}}$, and $\text{psd_power}_{\text{HLVLF}}^{\text{VLF}}$. In addition, $\text{psd_power}_{\text{HF}}^{\text{LF}}$, the power ratio between the LF band and the HF band, is computed as a measure of dominance between SNS and PNS activity [18].

Lastly, time-frequency domain features are extracted from $\widehat{\text{hmod}}$'s smooth pseudo Wigner–Ville distribution (SPWVD). It is often the preferred time-frequency representation of HRV signals due to its high and flexible resolution in both dimensions as well as its smooth modeling of signal power over time [39], [63], [64]. Specifically, the SPWVD is computed directly following Hernando et al. [39]. Details on its definition and efficient computation are omitted here for brevity. However, the following intuition is adopted: For any time $t \in \text{dom}(\widehat{\text{hmod}})$, the frequency-domain signal $|\text{spwvd}(t, \cdot)|$ represents $\widehat{\text{hmod}}$'s local power spectrum, similar to the PSD but based only on a short time interval centered at t .

Consequently, the SPWVD induces the absolute and relative band powers defined in (20) and (21) as well as all frequency-domain features specified with their help as time-dependent signals. The statistics summaries of these signals, i.e., $\text{stats}(\text{spwvd_power}_{\text{TOT}})$, $\text{stats}(\text{spwvd_power}_{\text{HVLVF}}^{\text{TOT}})$, $\text{stats}(\text{spwvd_power}_{\text{HVLVF}}^{\text{HF}})$, $\text{stats}(\text{spwvd_power}_{\text{HVLVF}}^{\text{LF}})$, $\text{stats}(\text{spwvd_power}_{\text{HVLVF}}^{\text{VLF}})$, and $\text{stats}(\text{spwvd_power}_{\text{HVLVF}}^{\text{LF}})$, are calculated as the final features.

V. FEATURE NORMALIZATION

The extraction of features is followed by their normalization. Classical feature normalization techniques often seek to simplify ML tasks by reducing inter-feature differences, i.e., discrepancies between the probability distributions of different features [65]. They usually achieve this by transforming the values of each feature uniformly based on estimates of the feature's statistical properties [65]. In the pipelines presented here, however, the transformations can instead be personalized to additionally mitigate persisting inter-person differences. These are differences among the distributions of the same feature for different individuals, which, in the context of biosignals, such as ECGs, largely stem from individual general physiologies and responses to external influences [44]. They often present an obstacle for effective ML model training [44], [66]. Addressing these differences with person-specific feature normalization serves as a lightweight alternative to more complex solutions like individual ML model training for each person. The general approach is not specific to ECG analysis or stress detection and could be useful for entity-specific feature normalization in other medical and non-medical fields as well.

Section V-A introduces the three implemented feature normalization functions in general form. After that, Section V-B illustrates how they are personalized without compromising test set integrity.

A. FEATURE NORMALIZATION FUNCTIONS

The implemented feature normalization functions are min-max normalization, z-score normalization, and robust scaler normalization. All three are widely applied and aim to harmonize feature distributions by shifting and scaling them based on estimates of corresponding feature statistics [65], [67], [68]. Min-max normalization linearly maps the feature's original value range to the interval $[0, 1]$ based on its estimated minimum and maximum [65]. In contrast, z-score normalization, also known as standardization, transforms the feature's distribution to have a mean of 0 and a variance of 1 based on its estimated mean and standard deviation [67]. Finally, robust scaler normalization modifies z-score normalization by replacing mean and standard deviation with the outlier-robust measures median and interquartile range, respectively [68].

To identify the most effective feature normalization function, an HP denoted as *norm* was defined and optimized. Its domain includes all three normalization functions mentioned

above, denoted as *min_max*, *z_score*, and *robust_scaler*, respectively. Additionally, it contains the value *None*, which represents the option of leaving features unnormalized.

B. CLASSICAL AND PERSONALIZED FEATURE STATISTICS

The statistic estimates used in the above-mentioned normalization functions are usually computed per feature: Each statistic of a given feature is estimated empirically from all values of that feature in a given training set. Thus, the feature's values in both the training set and the corresponding test set are normalized with the same parameters. Though this classical type of feature normalization is not required for tree-based ML models like RFs and GBMs as these are invariant to feature-wise shifting and scaling, it is often highly beneficial for the training of neural networks, such as MLPs [65].

The personalization of feature normalization is achieved by using individually derived feature statistic estimates for each person. Each feature's statistics are empirically estimated based on its values for the windows in a fixed-duration baseline section of the person's data. The utilized baseline is a leading segment of the person's cleaned ECG associated with neutrality, since this affective state is covered by both WESAD and SWELL-KW and can be readily established in further individuals at inference time. To accomplish this, the assistance system requests each new user to initially perform the neutrality-inducing tasks of either the WESAD study or the SWELL-KW study for calibration. These tasks consist of either reading neutral magazines [22] or doing office work on desktop computers without any time-pressure or interruptions [23], respectively. The ECG recorded during this period is then used as the person's feature normalization baseline by the inference pipeline.

Regarding the training pipeline, care must be taken to prevent contamination of the test sets created in accordance with the applied evaluation techniques (see Section VII-B). Each test set simulates data from unknown individuals by including only the samples, i.e., the labeled normalized feature vectors, corresponding to a subset of the persons in the cleaned input dataset. To accurately emulate the inference pipeline's real-world conditions and prevent normalization data leakage [28], all samples derived from ECG windows that fully or partially coincide with the corresponding person's baseline must be excluded from all test sets. An intuitive justification for this is the fact that, in practice, the inference pipeline's predictions for a person only begin after their calibration step described above has been completed.

More specifically, failure to remove these samples from every test set could artificially inflate evaluation results, as they are potentially easier to classify than truly unknown data: Utilized training sets hold all the samples of the individuals selected for training, including the samples computed from windows lying inside their normalization baselines. After personalized normalization, these samples have certain fixed statistics for all features, such as the minimum 0 and the maximum 1 with min-max normalization.

This could allow ML models to learn to identify any person's baseline feature vectors and thus correctly classify them as negative. Even feature vectors whose windows only partially overlap with the respective person's baseline may hold information that could be leveraged. Excluding samples containing such feature vectors from all test sets therefore leads to a more conservative but also less biased assessment of real-world classification quality.

The duration of the utilized feature normalization baselines was tuned using the HP d_{base} in seconds. In the case of MLP as selected ML model type, the HP value 0 s was investigated, signifying the omission of baseline treatment and the estimation of feature statistics in the classical, non-personalized way. Apart from this, the minimum considered baseline duration was $2d_{\text{win}}$ to ensure that feature statistic estimates were based on multiple ECG windows.

The maximum considered baseline duration was chosen depending on the input dataset. In each case, it was set well below the respective theoretical maximum in order to prevent an excessive loss of neutrality-labeled data and thus of representativeness for the stress-negative class from the test sets. This is especially crucial regarding SWELL-KW, since for this dataset every person's neutrality-labeled ECG, spanning around $2700 \text{ s} - 2d_{\text{trunc}}^{\text{SWELL-KW}}$ after the cleaning step, constitutes the entirety of their usable stress-negative ECG data. The maximum value of d_{base} considered for SWELL-KW was thus set to $1800 \text{ s} - 2d_{\text{trunc}}^{\text{SWELL-KW}}$. For WESAD, on the other hand, where each person's cleaned neutrality-labeled ECG spans $1200 \text{ s} - 2d_{\text{trunc}}^{\text{WESAD}}$, the maximum considered value of d_{base} was set to $800 \text{ s} - 2d_{\text{trunc}}^{\text{WESAD}}$. Finally, for the combined dataset, it was set to $1200 \text{ s} - 2d_{\text{trunc}}^{\text{WESAD}}$, as this value lies in the investigated value range for SWELL-KW and is the maximum possible value for WESAD.

VI. MACHINE LEARNING MODELS

ML models of three types can be trained by the training pipeline and utilized for the detection of stress by the inference pipeline: RF, GBM, and MLP. Associated implementations were made using the Python libraries scikit-learn [69], XGBoost [70], and TensorFlow [71], respectively.

All employed ML models are designed to accept normalized 70-dimensional feature vectors as inputs and yield estimated scalar probabilities that these feature vectors are associated with stress as predictions. Binary stress labels are inferred from these probabilities by rounding to the closest integer. The ML models are trained using supervised learning, with each utilized sample consisting of an input vector and the corresponding known stress label. While the training sets used in the training and evaluation step are formed in accordance with the applied evaluation techniques (see Section VII-B), any deployable ML model is trained on the entire set of created samples. In each training run, the respective ML model's capability to apply sample weighting is leveraged to mitigate the class imbalance in its training set.

Sections VI-A, VI-B, and VI-C briefly introduce RFs, GBMs, and MLPs along with their HPs and the associated tuning domains. Each tuning domain was designed to cover a broad range of potential HP values with distinct effects and allow for sufficient maximum model complexity while still keeping the time required for each training run manageable.

A. RANDOM FOREST

An RF is an ensemble ML model that combines multiple decision trees. It utilizes a technique called bagging, where each tree is trained on a subset of the training set drawn randomly with replacement, and the individual tree predictions are aggregated by averaging. Moreover, it employs random feature selection at each node split during tree construction. Both bagging and random feature selection serve improved generalization. [29]

RF creation and training were realized using the scikit-learn class RandomForestClassifier [69]. Related HPs are listed in Table 3 along with the tuning domains selected for them.

B. GRADIENT BOOSTING MACHINE

Similar to an RF, a GBM is an ensemble ML model that combines multiple weak learners, which are typically decision trees. The decision trees, however, are created and combined by a gradient-descent-based boosting algorithm instead of a bagging algorithm. This essentially means that the trees are consecutively trained on the training set, each with the goal to predict the negative loss gradient of the sub-GBM formed by all preceding trees. Binary cross-entropy is used as the loss function here, which is the established standard for binary classification tasks. Furthermore, a GBM's overall prediction is computed as a weighted sum of its trees' predictions. [30]

GBM-related HPs subjected to optimization are given together with the selected tuning domains in Table 4. All listed HPs are parameters of the utilized XGBoost class XGBClassifier [70].

C. MULTILAYER PERCEPTRON

An MLP is a feed-forward artificial neural network comprising an input layer, a dense, i.e., fully connected output layer, and at least one hidden dense layer with a nonlinear activation function [31], [72]. MLP creation and training were realized and hyperparameterized using TensorFlow's keras.Sequential API [71], [73]. To match the nature of the given samples, each MLP's input layer has size 70 while its output layer has size 1 and applies the sigmoid activation function. Any hidden layer between these two is either a dense layer, a normalization layer, or a dropout layer [74].

To train an MLP, the utilized training set is first partitioned into an effective training set and a validation set. Here, the validation set consists of the samples corresponding to approximately 30% of the individuals in the original training set. Then, the well-known Adam algorithm [75] is applied to

TABLE 3. RF HPs and their tuning domains.

RF HP	Tuning domain	Description
n_estimators	{1, 2, ..., 300}	Number of trees. (Including the value 1 implies examining single decision trees in addition to RFs.)
max_samples	[0.5, 1.0]	Proportion of training samples to consider for each tree.
max_features	{None, 'sqrt', 'log2'}	Number of randomly selected features to consider for each node split. Listed values indicate n_{feat} , $\sqrt{n_{\text{feat}}}$, and $\log_2(n_{\text{feat}})$, respectively, where n_{feat} is the number of input features.
criterion	{'entropy', 'gini'}	Node split quality measure. Listed values indicate Shannon information gain and Gini impurity decrease, respectively.
min_samples_leaf	[0.001, 0.5]	Minimum proportion of training samples required for each leaf.
max_leaf_nodes	{2, 3, ..., 100}	Maximum number of leaves in a tree.
class_weight	{'balanced', 'balanced_subsample'}	Label class weighting. Listed values indicate class weights balancing the class frequencies among all training samples and class weights balancing class frequencies in each tree's training samples, respectively.

TABLE 4. GBM HPs and their tuning domains.

GBM HP	Tuning Domain	Description
n_estimators	{1, 2, ..., 300}	Maximum number of trees.
subsample	[0.01, 1.0]	Proportion of training samples to consider for each tree.
sampling_method	{'uniform', 'gradient_based'}	Sampling method for training samples. Listed values indicate random selection with uniform probabilities and probabilities correlated with the loss gradient, respectively. By default, the utilized loss function is the binary cross-entropy between the sigmoid-transformed model predictions and the actual labels.
colsample_bytree	[0.1, 1.0]	Proportion of features to consider for each tree.
grow_policy	{'depthwise', 'lossguide'}	Tree growing policy. Listed values indicate favoring splitting leaf nodes closest to the root and leaf nodes with highest loss improvement, respectively.
min_split_loss	[0.0, 50.0]	Minimum loss reduction required for each node split.
min_child_weight	[0.0, 50.0]	Minimum sum of weights associated with training samples required for each node split.
max_leaves	{2, 3, ..., 100}	Maximum number of leaves in a tree.
learning_rate	[0.01, 1.0]	Learning rate alias shrinkage.
reg_lambda	[0.0, 100.0]	L2 regularization coefficient.

the effective training set for 300 epochs with a batch size of 512. During the training run, model checkpoints are created and evaluated on the validation set every 3 epochs. From these checkpoints, the best-performing MLP is selected as the training run's resulting model.

Table 5 lists all MLP HPs that were defined and optimized, as well as the utilized tuning domains.

VII. EVALUATION

To assess the achieved stress detection performance and facilitate a comparison with the state of the art, the training pipeline employs the evaluation metrics and techniques described in the following. Section VII-A focuses on the evaluation metrics used to express the classification performances of trained ML models as scalar values. After that, Section VII-B presents the evaluation techniques that govern which data ML models are trained and evaluated on and how resulting metric values are ultimately aggregated.

A. EVALUATION METRICS

The central evaluation metric and target of HP optimization is the F1-score, which is calculated as the harmonic mean of precision and recall [32]. It is widely recommended and used for the evaluation of binary classification results, especially on datasets with imbalanced label distributions [76], including WESAD [22] and SWELL-KW [77].

In addition to the F1-score, the evaluation considers accuracy, which measures the overall proportion of correct classifications. Although intuitive, accuracy is often misleading in the context of binary classification on imbalanced datasets, as it ignores class proportions and thus can easily be dominated by the majority class [76]. It is included here nevertheless for comparison with other stress detection research using WESAD and SWELL-KW, in which the achieved accuracies are usually reported as well (see Section X-A).

B. EVALUATION TECHNIQUES

To evaluate the stress detection capabilities of ML models and their underlying HP assignments for data from unknown individuals, LOSO-CV [33] is applied. For each available person, LOSO-CV trains an ML model under exclusion of that person's samples, which are used as the test set for subsequent ML model inference and metric computation. It is crucial that the involved data splitting is done strictly at the subject level rather than the sample level, as this ensures that the test set consists exclusively of data from an individual unseen by the model during training. The resulting estimate for each metric is the average of its computed values across all iterations. This evaluation technique is widely employed in the context of stress detection based on both WESAD [22] and SWELL-KW [77].

TABLE 5. MLP HPs and their tuning domains.

MLP HP	Tuning domain	Description
n_dense	{2, 3, 4, 5, 6}	Number of dense layers, including the output layer.
first_dense_size	{10, 11, ..., 500}	First dense layer size.
dense_size_trend	{'const', 'decr'}	Dense layer size trend. Listed values indicate constant size for all hidden dense layers and linearly decreasing sizes between the first dense layer and the output layer, respectively.
activation	{'sigm', 'tanh', 'relu'}	Hidden dense layer activation function. Listed values indicate sigmoid, hyperbolic tangent, and rectified linear unit, respectively.
normalization	{None, 'layer', 'batch'}	Type of normalization layer to place after the input layer and after each hidden dense layer. Listed values indicate no normalization, layer normalization, and batch normalization, respectively.
dropout_rate	[0, 0.5]	Dropout rate. One dropout layer is placed after the input layer. Additionally, one dropout layer is placed after each hidden dense layer if normalization is None and after each normalization layer otherwise.
loss	{'mse', 'bce'}	Loss function. Listed values indicate mean squared error and binary cross-entropy, respectively.
gradient_clipping	{None, 'value', 'layer', 'global'}	Loss gradient clipping approach. Listed values indicate no gradient clipping, clipping of individual gradients, clipping of layer gradients, and clipping of the global gradient, respectively. Clipping is done with a maximum gradient L2 norm of 1.
learning_rate	[0.0001, 0.1]	Learning rate.

Even though the utilized datasets contain relatively few individuals, LOSO-CV is highly computationally expensive due to the involved execution of a training run for each person. This is especially true for neural networks, such as MLPs. Thus, in the MLP case, subject-wise hold-out validation (SW-HV) [78] is used as a less costly proxy evaluation technique. In this approach, the samples of two thirds of the available individuals are used as the training set and the remainder as the test set. Similar to LOSO-CV, SW-HV evaluates the generalization ability for data from unknown individuals. However, it is less robust due to its dependence on one specific split into training set and test set. Therefore, the best-performing MLP HP assignments regarding SW-HV are additionally evaluated with LOSO-CV.

After HP optimization for a given combination of ML model type and input dataset, the final HP assignment was evaluated with LOSO-CV five additional times using different random seeds. This was done to account for the stochasticity inherent in ML model training, caused for instance by random subsampling during RF and GBM training, or random weight initialization during MLP training. For each metric, the mean and standard deviation of its five values from these LOSO-CV runs form the final evaluation result reported in this work (see Section IX). The mean provides a more stable estimate of the average stress detection performance for unknown individuals than a corresponding metric value from a single LOSO-CV run. The standard deviation serves as an estimate of the extent to which this performance fluctuates specifically due to ML model training stochasticity. In contrast, authors of related work typically only report metric values from a single LOSO-CV (see Section X-A).

VIII. HYPERPARAMETER OPTIMIZATION

In the following, the procedures used to optimize the HPs of the developed ML pipelines with respect to the selected

evaluation are presented. First, Section VIII-A outlines the implemented optimization framework. Subsequently, Section VIII-B describes how this framework was applied to optimize the HPs for each available dataset.

A. HYPERPARAMETER OPTIMIZATION FRAMEWORK

To maximize the stress detection performance of the ML pipelines in terms of F1-score under LOSO-CV, Bayesian optimization with Gaussian process metamodeling is applied. This optimization strategy iteratively builds a metamodel that uses Gaussian random variables to represent the relationship between HP assignments in the Cartesian product of the selected HP tuning domains and their expected target metric values. In each iteration, the metamodel is used to strategically select the next unknown HP assignment to analyze, balancing exploration of assignments with the highest uncertainty and exploitation of assignments with the best expected target metric values. The approach has proven effective and efficient across a wide range of applications. [34]

Two nested Bayesian optimization loops applicable to the training pipeline were realized using the Python library scikit-optimize [79]. The inner loop optimizes HPs related to ML model training, referred to as model HPs. The outer loop additionally optimizes the remaining HPs, which control data preparation, i.e., preprocessing and feature engineering by the preceding pipeline steps, and are referred to as data HPs. Each of the two loops iteratively builds and uses an individual metamodel relating its concerned HPs to target metric values. This separation enables the efficient utilization of data preparation results for the investigation of multiple model HP assignments.

In each iteration of the outer loop, all training pipeline steps preceding ML model training are executed in accordance with the current data HP assignment. Then, the inner loop iteratively performs ML model training and evaluation to

find an optimized model HP assignment for the data HP assignment. If the selected ML model type is RF or GBM, LOSO-CV serves as the evaluation technique throughout all inner loop iterations. If, however, MLP is selected, SW-HV is used instead for efficiency, consistently utilizing the same training, validation, and test set split to ensure comparability across all inner loop iterations. Only the best-performing MLP HP assignment regarding SW-HV undergoes further evaluation with LOSO-CV.

After completion of the optimization loops, the combination of data and model HP assignments that performed best under LOSO-CV for the selected ML model type is identified. Using the model HP assignment, a final ML model is then trained on all samples created with the data HP assignment. The trained ML model and the data HP assignment constitute the optimization result, ready for optional deployment into the inference pipeline.

B. EXECUTION OF HYPERPARAMETER OPTIMIZATION

For each of the three investigated input datasets, namely WESAD, SWELL-KW, and the combined dataset, HP optimizations were performed individually with respect to each of the three selectable ML model types RF, GBM, and MLP. To reduce the computational effort, however, the complete optimization including both loops was executed on each dataset only for RF. The best data HP assignment thus found was then reused for GBM and MLP HP optimization. This approach was motivated by the fact that RF is often the most lightweight of the ML model types and has the fewest HPs to tune, as well as by the assumption that data preparation steps optimized for RF should also benefit GBM and MLP. In addition, the data HP optimization results for WESAD and SWELL-KW were utilized to restrict the data HP tuning domains for the combined dataset, facilitating a more targeted search. The optimized values of the two dataset-specific HPs $d_{\text{trunc}}^{\text{WESAD}}$ and $d_{\text{trunc}}^{\text{SWELL-KW}}$ were directly adopted without further tuning. The remaining numerical data HPs d_{win} , θ , and d_{base} were each limited to the interval formed by the two corresponding optimized values, while for the categorical data HPs $\widehat{\text{respr}}$ and norm , only the respective optimized values themselves were considered. As detailed in Section V-B, the upper bound for d_{base} was further restricted to $1200\text{ s} - 2d_{\text{trunc}}^{\text{WESAD}}$.

Each execution of the inner HP optimization loop comprised up to 100 iterations. Early stopping was manually applied if further model HP optimization for the current data HP assignment seemed unpromising due to non-improving results. During the HP optimization run for each dataset with RF as selected ML model type, the outer loop was executed for 20 iterations, each comprising one such execution of the inner loop. In the subsequent HP optimization run for GBM, the resulting optimized data HP assignment was used to perform the training pipeline's preprocessing and feature engineering steps, and only the inner loop was executed to tune model HPs. In the HP optimization run for MLP, two such executions of the training pipeline and the inner loop

were conducted: One using the data HP assignment optimized for RF in unchanged form and one with the alteration $d_{\text{base}} = 0$, which entails the application of classical feature normalization. This alteration was explicitly considered for MLP to account for the fact that only personalized feature normalization is ever considered for RF, as discussed in Section V-B.

The optimization loops were often supplemented with manual searches in which individually picked HP assignments, typically close to the best ones found so far, were evaluated as well. In the end, the trained ML model associated with the overall best performance on the combined dataset, which proved to be an MLP (see Section IX), was deployed into the inference pipeline along with its corresponding data HP assignment.

IX. RESULTS

Tables 6 through 9 present the best-performing data and model HP assignments for each investigated dataset, as determined by optimizing the F1-score under LOSO-CV.

TABLE 6. Best-performing data HP assignments.

Data HP	Def. in	Optimized value for dataset		
	Section	WESAD	SWELL-KW	Comb.
$d_{\text{trunc}}^{\text{WESAD}}$	III-A	2 s	—	2 s
$d_{\text{trunc}}^{\text{SWELL-KW}}$	III-A	—	95 s	95 s
d_{win}	III-C	255 s	300 s	255 s
θ	IV-A	2	2	2
$\widehat{\text{respr}}$	IV-D	$\widehat{\text{respr}}_{\text{HeartPy}}$	$\widehat{\text{respr}}_{\text{HeartPy}}$	$\widehat{\text{respr}}_{\text{HeartPy}}$
norm	V-A	robust_scaler	robust_scaler	robust_scaler
d_{base}	V-B	650 s	1595 s	1130 s
# Samples		25 507	94 716	121 759

Table 6 presents the optimized data HP assignments, which provide several key findings. The optimized ECG edge truncation duration $d_{\text{trunc}}^{\text{WESAD}}$ for WESAD was very short at 2 s, while the corresponding duration $d_{\text{trunc}}^{\text{SWELL-KW}}$ for SWELL-KW was significantly longer at 95 s. This could indicate that periods with unstable affective states at the beginning and end of each phase in the underlying study were on average significantly longer for SWELL-KW than for WESAD. With regard to the window duration d_{win} , the optimization consistently resulted in values at or near the maximum tested value of 300 s. This suggests that for the given application, the benefit of using longer windows to capture more comprehensive physiological information outweighs the drawback of having fewer windows available for training. The optimized maximum number θ of consecutive irregular beats considered as the potential source of each irregular heart rate derivative was consistently 2, likely yielding the most useful beat type features for the classification task. Regarding respiration rate estimation, the estimator $\widehat{\text{respr}}_{\text{Sarkar}}$ directly following Sarkar et al. [35] was consistently outperformed by the adapted, more noise-robust estimator $\widehat{\text{respr}}_{\text{HeartPy}}$. Finally, the optimized norm and d_{base} provide strong evidence

TABLE 7. Best-performing RF HP assignments, their implied average RF complexities in terms of total leaf count, and their achieved metric values.

RF HP	Optimized value for dataset		
	WESAD	SWELL-KW	Comb.
n_estimators	286	86	172
max_samples	1.0	0.9068	0.998553
max_features	'sqrt'	'sqrt'	'sqrt'
criterion	'entropy'	'entropy'	'entropy'
min_samples_leaf	0.0252	0.0049	0.001556
max_leaf_nodes	18	53	75
class_weight	'balanced'	'balanced'	'balanced'
Avg. Total # Leaves	3 239.96	4 249.73	12 900.0
Metric			
F1-Score	0.869 ± 0.003	0.893 ± 0.003	0.786 ± 0.006
Accuracy	0.903 ± 0.001	0.858 ± 0.003	0.788 ± 0.004

for the benefits of mitigating inter-feature and inter-person differences through personalized feature normalization. The outlier-resistant robust_scaler normalization function combined with a non-zero baseline duration proved most effective not only during the RF-based data HP optimization but also during the additional comparison with classical feature normalization for MLP.

A primary observation across all three investigated ML model types is a significantly lower performance on the combined dataset compared to WESAD and SWELL-KW individually. As shown in Tables 7, 8, and 9, the highest F1-scores on the individual datasets were 0.904 ± 0.035 with MLP on WESAD and 0.917 ± 0.002 with GBM on SWELL-KW. In contrast, the highest F1-score on the combined dataset was only 0.840 ± 0.014 , achieved with MLP. A similar decrease is apparent for accuracy, although this metric is de-emphasized in the following due to its misleading nature for imbalanced data addressed in Section VII-A. The observed performance drop is consistent with the assessment of Prajod et al. [41], who identified the different stressors used in the two underlying studies, namely social-evaluative pressure in the WESAD study versus cognitive load and interruptions in the SWELL-KW study, as the primary reason for the increased challenge posed by the combined dataset. This underlines that reliable real-world stress detection requires more diverse training data to capture the various causes and HRV manifestations of stress.

The RF HP optimization results are presented in Table 7. In terms of average F1-score, this ML model type was surpassed by at least one of the other two on each dataset, which is noteworthy given that the underlying data HPs were optimized specifically for RF. However, a core strength of RF was its high robustness to training stochasticity, achieving the overall lowest standard deviation of F1-score on WESAD and the combined dataset, and the second-lowest on SWELL-KW. The average RF complexity, indicated by the total number of leaves, generally increased with the size of the dataset. Three RF HP values were found to be consistently optimal across all three datasets. The max_features HP was always 'sqrt', which sets the number of features considered at each node split to the square root of the total feature

count. Furthermore, the node split quality measure criterion was consistently 'entropy' and class_weight was always 'balanced', making class weights inversely proportional to class frequencies across all training samples.

TABLE 8. Best-performing GBM HP assignments, their implied average GBM complexities in terms of total leaf count, and their achieved metric values.

GBM HP	Optimized value for dataset		
	WESAD	SWELL-KW	Comb.
n_estimators	167	66	107
subsample	0.636037	0.1	0.087
sampling_method	'uniform'	'gradient_based'	'gradient_based'
colsample_bytree	0.1	1.0	0.1
grow_policy	'depthwise'	'depthwise'	'lossguide'
min_split_loss	0.84747	0.0	20.0
min_child_weight	12.275	5.657	48.0
max_leaves	2	2	32
learning_rate	0.39957	0.239285	0.072578074
reg_lambda	6.34	0.0	69.0
Avg. Total # Leaves	327.93	132.0	3 418.06
Metric			
F1-Score	0.889 ± 0.020	0.917 ± 0.002	0.790 ± 0.012
Accuracy	0.932 ± 0.010	0.870 ± 0.004	0.771 ± 0.013

Table 8 shows the GBM HP optimization results. GBM consistently outperformed RF in terms of average F1-score and even achieved the overall highest value on SWELL-KW. It also demonstrated high stability with a lower F1-score standard deviation than MLP on all datasets and the lowest overall on SWELL-KW. A particularly striking result was that for both WESAD and SWELL-KW the HP max_leaves was optimized to a value of 2, effectively reducing the utilized trees to extremely weak learners known as tree stumps. As noted by Natekin and Knoll, this strong form of regularization can prevent overfitting and often performs well in practical GBM applications [30]. In contrast, the more heterogeneous combined dataset required considerably more complex GBMs with strong regularization from the HPs subsample, colsample_bytree, min_split_loss, min_child_weight, and reg_lambda.

TABLE 9. Best-performing MLP HP assignments, their implied MLP complexities in terms of trainable parameter count, and their achieved metric values.

MLP HP	Optimized value for dataset		
	WESAD	SWELL-KW	Comb.
n_dense	5	3	4
first_dense_size	446	19	85
dense_size_trend	'decr'	'const'	'const'
activation	'relu'	'tanh'	'relu'
normalization	None	None	'batch'
dropout_rate	0.1872484	0.5	0.402271
loss	'bce'	'bce'	'bce'
gradient_clipping	'global'	'global'	None
learning_rate	0.00070840904	0.000515667	0.00097
# Trainable Param.s	282 328	1 769	21 477
Metric			
F1-Score	0.904 ± 0.035	0.877 ± 0.024	0.840 ± 0.014
Accuracy	0.939 ± 0.018	0.827 ± 0.031	0.810 ± 0.015

The MLP HP optimization results presented in Table 9 show a mixed performance. While MLP achieved the overall

highest average F1-scores on both WESAD and the combined dataset, its performance on SWELL-KW was lower than that of both tree-based ML model types. Furthermore, MLP was consistently the most susceptible to training stochasticity, exhibiting the highest standard deviation of F1-score on all datasets. The model complexity in terms of trainable parameter count also revealed a surprising trend: The largest optimized MLP architecture by far emerged for WESAD, not for the much larger and more heterogeneous combined dataset, while the architecture for SWELL-KW was the smallest and the most regularized by dropout_rate. The only consistently optimal MLP HP setting was loss = 'bce', which suggests an advantage of using binary cross-entropy instead of mean squared error as a loss function.

Since the highest F1-score under LOSO-CV on the combined dataset was attained by MLP (0.840 ± 0.014), a corresponding final MLP was trained on the entirety of this dataset. This model and the combined dataset's optimized data HP assignment constitute the final optimization result. Both were deployed into the inference pipeline.

X. DISCUSSION

The following discussion analyzes and contextualizes this work's methodology and results. First, Section X-A compares the developed ML pipelines and their stress detection performances against the state of the art. Section X-B then explains the rationale behind specific design choices and their implications. Finally, Section X-C addresses this work's limitations.

A. COMPARISON WITH RELATED WORK

To facilitate a comparison of the developed ML pipelines and their classification quality with the state of the art, a comprehensive literature search was conducted for publications on systems for solely ECG-based stress detection that were evaluated on WESAD, SWELL-KW, or their combination using LOSO-CV. Table 10 lists the best results in terms of F1-score under LOSO-CV reported in each identified fitting publication and in this work. The metric values from the referenced literature are each based on a single LOSO-CV run, while those from this work are each based on five runs with varying seeds. The abbreviations LDA, CNN, FT, TF, and LSTM mentioned in the ML Model Type column stand for linear discriminant analysis, convolutional neural network, feature-tokenizer, transformer, and long short-term memory network, respectively.

As the table shows, all related work's F1-scores were exceeded on each of the three datasets with each of the three investigated ML model types. This suggests that the pipeline design and optimization proposed here facilitate the detection of stress in unknown individuals with greater reliability than the comparable state of the art, regardless of the selected ML model type. A detailed comparison of all methodologies revealed several key differences, which are discussed below.

The preprocessing steps employed in the referenced literature are generally less tailored to their input datasets.

None of the authors performed cleaning by truncation of leading and trailing ECG sections for increased label reliability. For windowing, all of them used windows with fixed and comparatively short durations, instead of seeking the best value through optimization. While the best-performing window durations in this work were 255 s and 300 s, Schmidt et al. [22], Liakopoulos et al. [80], Karan and Kaygun [81], and Prajod et al. [41] used 60 s, Behinaein et al. [82] and Henry et al. [40] used 30 s, and El Arwadi and Abu Daher [42] used 20 s. Prajod and André [43] employed window durations of 60 s and 10 s for their two approaches discussed here, i.e., for stress detection with ML models of type MLP and CNN-LSTM, respectively.

With the extraction of 70 expressive HRV features and the built-in resilience to irregular beats, this work's pipeline design creates a particularly comprehensive and robust representation of each ECG window as a basis for classification. Schmidt et al., Prajod and André in their MLP approach, Henry et al., Prajod et al., and El Arwadi and Abu Daher each utilized between 11 and 22 HRV-related features. In addition to time and frequency domain features, which were included by all five, Prajod and André, Henry et al., and Prajod et al. also included features based on Poincaré plots of RR intervals. In contrast to this work, none of them extracted any time-frequency domain features. Moreover, none of them extracted features related to respiration rate or ANS modulation in terms of a heartbeat generation model such as the IPFM model. Karan and Kaygun refrained from the use of HRV features and instead extracted topological features describing ECG windows in general rather than their HRV. Liakopoulos et al., Prajod and André in their CNN-LSTM approach, and Behinaein et al. omitted explicit scalar feature extraction altogether and fully relied on implicit feature extraction through their employed neural networks. While the former used 2D spectrograms of ECG windows as network inputs, the latter two both used the ECG windows themselves after downsampling and band-pass filtering. The only authors who incorporated corrections for irregular beats into their feature extraction process are Henry et al. However, in contrast to this work, they solely focused on the removal of FPs and did not consider FNs, ectopic beats, or other types of irregularities.

Furthermore, the compared approaches largely differ regarding the normalization of ML model inputs. While Schmidt et al. and Liakopoulos et al. did not perform such normalization, Karan and Kaygun, and Henry et al. normalized features using classical min-max normalization and z-score normalization, respectively. Behinaein et al. applied classical z-score normalization to ECGs. However, the step was conducted before windowing. Therefore, the mean and standard deviation used to normalize an ECG and thus effectively normalize all its windows were inferred from the entire ECG rather than from each window individually. The approach thereby introduced temporal data leakage [28] by providing each ECG window, except for the last one, with information from future ECG windows.

TABLE 10. Best achieved F1-scores, corresponding accuracies, and employed ML model types in related work and here.

Dataset	Year	Source	ML Model Type	F1-Score	Accuracy
WESAD	2018	Schmidt et al. [22]	LDA	0.813	0.854
	2021	Liakopoulos et al. [80]	CNN	0.794	0.824
	2021	Karan and Kaygun [81]	LDA	—	0.887
	2021	Behinaein et al. [82]	CNN-TF	0.833	0.911
	2022	Prajod and André [43]	MLP	0.859	0.895
	2023	Henry et al. [40]	FT-TF	0.722	0.706
	2024	Prajod et al. [41]	MLP	0.844	0.880
	2025	El Arwadi and Abu Daher [42]	MLP	0.760	0.908
	2026	This work	RF	0.869 ± 0.003	0.903 ± 0.001
	2026	This work	GBM	0.889 ± 0.020	0.932 ± 0.010
SWELL-KW	2021	Behinaein et al. [82]	CNN-TF	0.742	0.716
	2022	Prajod and André [43]	CNN-LSTM	0.688	0.755
	2024	Prajod et al. [41]	MLP	0.667	0.692
	2025	El Arwadi and Abu Daher [42]	MLP	0.718	0.647
	2026	This work	RF	0.893 ± 0.003	0.858 ± 0.003
	2026	This work	GBM	0.917 ± 0.002	0.870 ± 0.004
	2026	This work	MLP	0.877 ± 0.024	0.827 ± 0.031
Combined	2022	Prajod and André [43]	MLP	0.698	0.732
	2024	Prajod et al. [41]	MLP	0.683	0.725
	2026	This work	RF	0.786 ± 0.006	0.788 ± 0.004
	2026	This work	GBM	0.790 ± 0.012	0.771 ± 0.013
	2026	This work	MLP	0.840 ± 0.014	0.810 ± 0.015

Similar to the ML pipelines presented here, Prajod and André, Prajod et al., and El Arwadi and Abu Daher each incorporated a lightweight personalization into their respective ML model inputs using person-specific normalization based on short, neutral baseline data sections. However, instead of optimizing the utilized normalization function and baseline duration, the former two applied min-max normalization based on 5-minute baselines while the latter applied ratio normalization with 90-second baselines. Moreover, crucially, none of them report removing baseline samples from test sets. As discussed in Section V-B, retaining these samples introduces data leakage and thus potentially leads to overoptimistic evaluation results. None of the authors mentioned here investigated the outlier-resistant robust scaler normalization function, which turned out to consistently perform best in this work.

As far as ML model training is concerned, besides the various selected model types listed in Table 10, the distinctive solution of Behinaein et al. is particularly striking. They managed to significantly boost their classification performance through person-specific training: In each iteration of LOSO-CV, they conducted 40 regular training epochs followed by 30 fine-tuning epochs on 10% of the current test person's samples. Though highly beneficial for classification quality, this method has the downside of requiring the same type of person-specific ML model training for each new person at inference time.

Moreover, none of the mentioned researchers explicitly state their applied HP tuning algorithms, whereas the ML pipelines presented here underwent an extensive optimization procedure involving Bayesian optimization and manual search to tune HPs related to preprocessing, feature

engineering, and ML model training. Lastly, none of the authors executed LOSO-CV multiple times for their final configurations to assess the impact of ML model training stochasticity on their results.

While Table 10 focuses on purely ECG-based systems to ensure a fair comparison, significant progress has also been made in the domain of multimodal stress detection. By fusing multiple biosignals, often including ECG, such methods leverage a broader range of physiological data to potentially enhance classification performance. However, they typically necessitate more intrusive sensor setups, significantly reducing their suitability for the continuous, unobtrusive monitoring of nursing staff. Rashid et al. [83] proposed a selective, hierarchical fusion framework to dynamically process features from inputs such as ECG, electromyogram (EMG), electrodermal activity (EDA), blood volume pulse (BVP), respiration (RESP), acceleration (ACC), and temperature (TEMP). It employs a decision tree, an ensemble classifier (either RF or AdaBoost), and a Kalman filter. Using LOSO-CV, this method achieved a 2-class macro F1-score of 0.929 (and an accuracy of 0.941) on WESAD. Most notably, Bhatti et al. [84] established a new state of the art for multimodal stress detection with LOSO-CV on both WESAD and SWELL-KW. They utilized a CNN architecture with a novel attention mechanism to fuse ECG, EDA, and RESP. With an F1-score of 0.933 (and an accuracy of 0.937) on WESAD, their approach also constitutes the overall state of the art in terms of LOSO-CV on this dataset. On SWELL-KW, however, it only reached an F1-score of 0.623 (and an accuracy of 0.652), falling well below all solely ECG-based results listed in Table 10. This contrast demonstrates that sophisticated ECG-based stress detection systems, despite

relying only on a single modality, can outperform current multimodal setups, especially on noisy datasets like SWELL-KW.

Apart from the works discussed here, many authors used faster but statistically far less rigorous evaluation techniques than the gold standard LOSO-CV, preventing a direct comparison of results. A recent example of this is Abdelfattah et al. [85], who compared different ML model types and sets of modalities for feature-based stress detection on WESAD. In their first experimentation phase, they utilized SW-HV, which, in contrast to LOSO-CV, is only a point estimator for stress detection performance that is highly sensitive to a single data split. They achieved an F1-score of 0.936 with a recurrent neural network based on EDA, BVP, ACC, and TEMP. In their second experimentation phase, they utilized regular 10-fold cross-validation. Since this approach tends to distribute each person's samples across training set and test set in each iteration, it estimates the stress detection performance for known rather than unknown individuals. Furthermore, since the involved data splitting does not respect the temporal order of each person's samples, it introduces temporal data leakage [28], potentially leading to drastically inflated metric values. Consequently, this method yielded near-perfect F1-scores for multiple ML model types on multiple sets of modalities.

B. METHODOLOGICAL CONSIDERATIONS

The design of the ML pipelines presented in this work involves several methodological choices that warrant further discussion.

First, the majority of the 70 extracted features described in Section IV are naturally correlated with each other, as they were selected for their known correlation with HRV and stress. Moreover, explicit redundancy was introduced by defining the feature n_irreg as the sum of others. While such correlation can lead to instability and overfitting with some ML models, tree-based model types like the utilized RF and GBM are inherently robust to this characteristic and excel at leveraging it [86], [87]. For MLP, model instability and overfitting were mitigated through the applied regularization techniques, including dropout and batch normalization [74]. The high classification performance achieved with MLP, especially on WESAD and on the combined dataset, demonstrates the effectiveness of this approach.

Regarding the respiration rate features introduced in Section IV-D, the optimization results indicate that the adapted estimator $\widehat{respr}_{HeartPy}$ consistently yields higher classification performance than the estimator $\widehat{respr}_{Sarkar}$, which is based directly on Sarkar et al. [35]. This suggests that the additional bandpass filtering adopted from HeartPy [54] effectively suppresses low-frequency drift and high-frequency noise, thereby providing a more robust basis for stress detection. Notably, this does not necessarily imply that the estimated respiration rates are more accurate, but rather that the features derived from them have greater discriminative power.

With respect to the extraction of ANS modulation features from the PSD and SPWVD powers of $hmod$ in the standard VLF band [0.003 Hz, 0.04 Hz] (see Section IV-E3), a methodological nuance must be considered. The lower frequency bound of this band corresponds to a signal period of 333.3 s. This period cannot be fully captured with the investigated window durations, which range from 60 s to 300 s. However, the computation of $hmod$ involves an approximate high-pass filtering with the cutoff frequency 0.03 Hz (see Section IV-E2). Consequently, the signal's VLF band power is effectively dominated by the frequency content in [0.03 Hz, 0.04 Hz]. The period corresponding to the lower bound of this effective band is 33.3 s, which can be fully captured even with the shortest utilized windows.

Finally, the applied HP optimization procedures warrant discussion. As detailed in Section VIII-B, HPs were tuned and evaluated for each ML model type using non-nested LOSO-CV on the entirety of each input dataset. While a nested LOSO-CV protocol would provide more unbiased performance estimates [88], it was deemed computationally prohibitive given the complexity of the HP search spaces. The primary objective of this work was to find the single best-performing data HP assignment and ML model to deploy for stress detection into the developed assistance system. The employed methodology is specifically designed to fulfill this goal. Crucially, the resulting HP values and ML model of each conducted optimization run are conceptually identical to those that would be produced by a typical final optimization run on the entire input dataset after nested LOSO-CV. Furthermore, using regular LOSO-CV still renders the evaluation results substantially more robust against optimistic bias when compared to widespread techniques with higher variance like SW-HV or even subject-wise cross-validation with five to ten folds [89]. Another key benefit of this protocol is that it aligns with the final evaluation techniques used in the related literature, facilitating the direct comparison of results presented in Section X-A.

C. LIMITATIONS

The results presented in this work should be interpreted in light of several limitations.

The primary limitation concerns the utilized original datasets WESAD and SWELL-KW described in Section II. While they provided a valuable basis for preliminary optimization and, due to their popularity in related work, enabled a comparison of the results with those of many other researchers in Section X-A, they cannot be considered representative of the stress experienced by the target group of nursing staff. First, their amount of affective-state-labeled physiological data is limited to about 36 minutes each for 15 participants and about 2 hours each for 25 participants, respectively. Moreover, almost all participants were in their twenties with only 20% and 32% of them being female, respectively, whereas the global nursing workforce spans all adult working ages and is about 90% female [90]. Finally,

the tasks and conditions of the two studies were not designed to reflect the multifaceted work of nurses. Most notably, they did not involve any physical activities, which means that physical nursing labor can be expected to yield ECGs with fundamentally different characteristics than those used here [91], [92].

Regarding the applied HP optimization detailed in Section VIII and the results given in Section IX, it should be noted that the optimized HP assignments result from non-exhaustive tuning and are thus not necessarily globally optimal but rather the best ones found. Furthermore, as discussed in the previous subsection, the utilized non-nested optimization and evaluation carries the potential for a slight optimistic bias [89].

Therefore, while the presented results demonstrate the relative effectiveness of the proposed methodology compared to the state of the art, they remain a proxy for operational effectiveness. A definitive validation of real-world stress detection performance would require a field study where nurses use the developed assistance system during their regular work.

XI. CONCLUSION AND OUTLOOK

The proposed AI system achieved new state-of-the-art stress detection performances for unknown individuals on all three investigated input datasets with all three investigated ML model types. This can mainly be attributed to the following aspects of the utilized ML pipeline architecture and optimization procedures, which set the system apart from related work: Firstly, thorough data preprocessing is applied, including optimized ECG edge truncation for cleaning and optimized ECG windowing. Secondly, the employed feature extraction encompasses a particularly broad range of HRV features, 70 in total, related to beat types, NN intervals, NN interval differences, respiration rates, and ANS modulation influences on the heart's sinoatrial node. Especially the increased noise robustness of respiration rate features and the derivation of ANS modulation features from the IPFM model stand out as novelties. Thirdly, all non-beat-type features are extracted with methods that ensure their robustness to irregular beats, such as FPs, FNs, and ectopic beats. Fourthly, the features are normalized with a novel, person-specific approach so that inter-person differences are mitigated in addition to inter-feature differences. The approach requires only a short section of readily collectible baseline data for each person and actively avoids normalization data leakage. Its applicability is not limited to the ML task at hand but extends to entity- or group-specific feature normalization in other medical and non-medical domains. Fifthly, extensive HP tuning using Bayesian optimization and manual search was applied to find the best HP values for preprocessing, feature engineering, and ML model training.

Future work will primarily focus on the analysis of feature importances and a corresponding feature selection with the aim to increase ML model efficiency while maintaining a high stress detection performance. Before the inference

pipeline can be reliably used for stress detection in practice, it is crucial to systematically extend the training pipeline's input data to overcome WESAD's and SWELL-KW's scope limitations and to re-optimize HPs accordingly. Furthermore, a nested LOSO-CV protocol would be valuable to obtain robust performance estimates and quantify any optimistic bias introduced by the current HP optimization approach. Ultimately, however, a field study in a real-world nursing environment remains necessary to confirm the system's operational effectiveness.

ACKNOWLEDGMENT

The authors would like to thank Jasmin Henze and Tom Strube from the Fraunhofer ISST for their valuable feedback and suggestions on this article.

REFERENCES

- [1] S. Cohen, D. Janicki-Deverts, and G. E. Miller, "Psychological stress and disease," *J. Amer. Med. Assoc.*, vol. 298, no. 14, p. 1685, 2007.
- [2] N. Schneiderman, G. Ironson, and S. D. Siegel, "Stress and health: Psychological, behavioral, and biological determinants," *Annu. Rev. Clin. Psychol.*, vol. 1, no. 1, pp. 607–628, Apr. 2005.
- [3] S. Sindi, K. Håkansson, G. Hagman, J. Kulmala, H. Soininen, I. Kåreholt, A. Solomon, and M. Kivipelto, "Midlife work-related stress increases dementia risk in later life: The CAIDE 30-year study," *Alzheimer's & Dementia*, vol. 10, p. P746, Jul. 2014.
- [4] G. Fink, *Stress: Concepts, Definition and History*. Amsterdam, The Netherlands: Elsevier, 2017.
- [5] M.-W. Ge, F.-H. Hu, Y.-J. Jia, W. Tang, W.-Q. Zhang, and H.-L. Chen, "Global prevalence of nursing burnout syndrome and temporal trends for the last 10 years: A meta-analysis of 94 studies covering over 30 countries," *J. Clin. Nursing*, vol. 32, nos. 17–18, pp. 5836–5854, Sep. 2023.
- [6] A. L. West, "Associations among attachment style, burnout, and compassion fatigue in health and human service workers: A systematic review," *J. Human Behav. Social Environ.*, vol. 25, no. 6, pp. 571–590, Aug. 2015.
- [7] Annu, P. Sharma, and J. Jacob, "Burnout symptoms among nurses: Literature review," *World J. Advance Healthcare Res.*, vol. 4, no. 3, pp. 60–66, Apr. 2020.
- [8] R. Jester, "Editorial-global shortage of nurses—rebecca jester for May 2023 issue," *Int. J. Orthopaedic Trauma Nursing*, vol. 49, May 2023, Art. no. 101018.
- [9] D. P. Paul, L. Bakhamis, H. M. Smith, and A. Coustasse, "Hospital nurse burnout: A continuing problem," in *Proc. 54th Annu. MBAA Conf.*, 2018.
- [10] (2026). *Digitaler Engel—Stärkung Der Interaktionsarbeit Von Pflegekräften*. Accessed: Mar. 10, 2026. [Online]. Available: https://www.iml.fraunhofer.de/de/abteilungen/b3/health_care_logistics/forschung_hcl/forschung_digitaler_Engel.html
- [11] *Federal Ministry of Research, Technology and Space (Formerly Federal Ministry of Education and Research)*. Accessed: Mar. 10, 2026. [Online]. Available: https://www.bmbf.de/EN/Home/home_node.html
- [12] *Fraunhofer IML*. Accessed: Mar. 10, 2026. [Online]. Available: <https://www.iml.fraunhofer.de/en.html>
- [13] *Fraunhofer ISST*. Accessed: Mar. 10, 2026. [Online]. Available: <https://www.isst.fraunhofer.de/en.html>
- [14] *IfAdo*. Accessed: Mar. 10, 2026. [Online]. Available: <https://www.ifado.de/en>
- [15] *Pius-Hospital Oldenburg*. Accessed: Mar. 10, 2026. [Online]. Available: <https://www.pius-hospital.de>
- [16] *TeamViewer*. Accessed: Mar. 10, 2026. [Online]. Available: <https://www.teamviewer.com/en>
- [17] K. Georgiou, A. V. Larentzakis, N. N. Khamis, G. I. Alsuhailani, Y. A. Alaska, and E. J. Giallafos, "Can wearable devices accurately measure heart rate variability? A systematic review," *Folia Medica*, vol. 60, no. 1, pp. 7–20, Jan. 2018.
- [18] F. Shaffer and J. P. Ginsberg, "An overview of heart rate variability metrics and norms," *Frontiers Public Health*, vol. 5, p. 258, Sep. 2017.
- [19] S. Ishaque, N. Khan, and S. Krishnan, "Trends in heart-rate variability signal analysis," *Frontiers Digit. Health*, vol. 3, p. 13, Feb. 2021.

- [20] S. Gradl, M. Wirth, R. Richer, N. Rohleder, and B. M. Eskofier, "An overview of the feasibility of permanent, real-time, unobtrusive stress measurement with current wearables," in *Proc. 13th EAI Int. Conf. Pervasive Comput. Technol. Healthcare*, May 2019, pp. 360–365.
- [21] M. Hintze, S. Sobieraj, M. Fecke, C. Rauch, J. Digutsch, M. Velana, and G. Rinkenauer, "Einsatz digitaler assistenzsysteme zur stärkung von interaktionsarbeit in der pflege," Fraunhofer Inst. Mater. Flow Logistics IML, Dortmund, Germany, Tech. Rep. 36, 2023.
- [22] P. Schmidt, A. Reiss, R. Duerichen, C. Marberger, and K. Van Laerhoven, "Introducing WESAD, a multimodal dataset for wearable stress and affect detection," in *Proc. 20th ACM Int. Conf. Multimodal Interact.*, Oct. 2018, pp. 400–408.
- [23] S. Koldijk, M. Sappelli, S. Verberne, M. A. Neerincx, and W. Kraaij, "The SWELL knowledge work dataset for stress and user modeling research," in *Proc. 16th Int. Conf. Multimodal Interact.*, Nov. 2014, pp. 291–298.
- [24] M. Feurer and F. Hutter, *Hyperparameter Optimization*. Cham, Switzerland: Springer, 2019, pp. 3–33.
- [25] G. van Rossum and F. L. Drake, *Python 3 Reference Manual*. Amsterdam, The Netherlands: CreateSpace, 2009.
- [26] R. Bailón, G. Laouini, C. Grao, M. Orini, P. Laguna, and O. Meste, "The integral pulse frequency modulation model with time-varying threshold: Application to heart rate variability analysis during exercise stress testing," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 3, pp. 642–652, Mar. 2011.
- [27] J. Mateo and P. Laguna, "Analysis of heart rate variability in the presence of ectopic beats using the heart timing signal," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 3, pp. 334–343, Mar. 2003.
- [28] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, Sep. 2023, Art. no. 100804.
- [29] V. K. Ayyadevara, *Pro Machine Learning Algorithms*. Cham, Switzerland: Springer, 2018.
- [30] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers Neuroinformatics*, vol. 7, p. 21, Dec. 2013.
- [31] M.-C. Popescu, V. E. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks," *WSEAS Trans. Circuits Syst.*, vol. 8, no. 7, pp. 579–588, 2009.
- [32] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: An overview," 2020, *arXiv:2008.05756*.
- [33] D. Gholamiangonabadi, N. Kiselov, and K. Grolinger, "Deep neural networks for human activity recognition with wearable sensors: Leave-one-subject-out cross-validation for model selection," *IEEE Access*, vol. 8, pp. 133982–133994, 2020.
- [34] J. Wu, X. Y. Chen, H. Zhang, L. D. Xiong, H. Lei, and S. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *J. Electron. Sci. Technol.*, vol. 17, no. 1, pp. 26–40, 2019.
- [35] S. Sarkar, S. Bhattacharjee, and S. Pal, "Extraction of respiration signal from ECG for respiratory rate estimation," in *Proc. Michael Faraday IET Int. Summit*, Sep. 2015, pp. 336–340.
- [36] P. van Gent, H. Farah, N. van Nes, and B. van Arem, "HeartPy: A novel heart rate algorithm for the analysis of noisy signals," *Transp. Res. F, Traffic Psychol. Behav.*, vol. 66, pp. 368–378, Oct. 2019.
- [37] R. Miller and C. Kirschbaum, "Trier social stress test," in *Encyclopedia of Behavioral Medicine*. Cham, Switzerland: Springer, 2013.
- [38] K. Plarre, A. Rajj, S. M. Hossain, A. A. Ali, M. Nakajima, M. Al'absi, E. Ertin, T. Kamarck, S. Kumar, M. Scott, D. Siewiorek, A. Smailagic, and L. E. Wittmers, "Continuous inference of psychological stress from sensory measurements collected in the natural environment," in *Proc. 10th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2011, pp. 97–108.
- [39] A. Hernando, J. Lázaro, E. Gil, A. Arza, J. M. Garzón, R. López-Antón, C. de la Cámara, P. Laguna, J. Aguiló, and R. Bailón, "Inclusion of respiratory frequency information in heart rate variability analysis for stress assessment," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 4, pp. 1016–1025, Jul. 2016.
- [40] J. Henry, H. Lloyd, M. Turner, and C. Kendrick, "On the robustness of machine learning models for stress and anxiety recognition from heart activity signals," *IEEE Sensors J.*, vol. 23, no. 13, pp. 14428–14436, Jul. 2023.
- [41] P. Prajod, B. Mahesh, and E. André, "Stressor type matters!—Exploring factors influencing cross-dataset generalizability of physiological stress detection," in *Proc. Int. Conf. Multimodal Interact.*, Nov. 2024, pp. 508–517.
- [42] L. El Arwadi and L. Abu Daher, "CardioMind: Deep learning framework for stress detection using IoT devices," *IEEE Access*, vol. 13, pp. 197027–197042, 2025.
- [43] P. Prajod and E. André, "On the generalizability of ECG-based stress detection models," in *Proc. 21st IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2022, pp. 549–554.
- [44] K. Nkurikiyeyezu, A. Yokokubo, and G. Lopez, "Effect of person-specific biometrics in improving generic stress predictive models," *Sensors Mater.*, vol. 32, no. 2, pp. 703–722, Feb. 2020.
- [45] B. Szakonyi, I. Vassányi, E. Schumacher, and I. Kósa, "Efficient methods for acute stress detection using heart rate variability data from ambient assisted living sensors," *Biomed. Eng. OnLine*, vol. 20, no. 1, Jul. 2021.
- [46] P. Hamilton, "Open source ECG analysis," in *Proc. Comput. Cardiol.*, Sep. 2002, pp. 101–104.
- [47] P. Bota, R. Silva, C. Carreiras, A. Fred, and H. P. da Silva, "BioSPPy: A Python toolbox for physiological signal processing," *SoftwareX*, vol. 26, May 2024, Art. no. 101712.
- [48] C. Guo, M. Lu, and J. Chen, "An evaluation of time series summary statistics as features for clinical prediction tasks," *BMC Med. Informat. Decis. Making*, vol. 20, no. 1, p. 48, Dec. 2020.
- [49] P. Melillo, M. Bracale, and L. Pecchia, "Nonlinear heart rate variability features for real-life stress detection. Case study: Students under stress due to university examination," *Biomed. Eng. OnLine*, vol. 10, no. 1, p. 96, 2011.
- [50] D. Park, M. Lee, S. E. Park, J.-K. Seong, and I. Youn, "Determination of optimal heart rate variability features based on SVM-recursive feature elimination for cumulative stress monitoring using ECG sensor," *Sensors*, vol. 18, no. 7, p. 2387, Jul. 2018.
- [51] M. Malik, J. T. Bigger, A. J. Camm, R. E. Kleiger, A. Malliani, A. J. Moss, and P. J. Schwartz, "Heart rate variability: Standards of measurement, physiological interpretation, and clinical use," *Eur. Heart J.*, vol. 17, no. 3, pp. 354–381, Mar. 1996.
- [52] S. Boonnithi and S. Phongsuphap, "Comparison of heart rate variability measures for mental stress detection," in *Proc. Comput. Cardiol.*, Sep. 2011, pp. 85–88.
- [53] D. Widjaja, M. Orini, E. Vlemincx, and S. Van Huffel, "Cardiorespiratory dynamic response to mental stress: A multivariate time-frequency analysis," *Comput. Math. Methods Med.*, vol. 2013, pp. 1–12, Dec. 2013.
- [54] P. van Gent. (2022). *Python Heart Rate Analysis Toolkit Documentation Release 1.2.5*. [Online]. Available: <https://app.readthedocs.org/projects/python-heart-rate-analysis-toolkit/downloads/pdf/latest/>
- [55] B. Hyndman and R. Mohn, "A pulse modulator model for pacemaker activity," in *Dig. 10th Int. Conf. Med. Biol. Eng.*, 1973, p. 223.
- [56] D. Candia-Rivera, V. Catrambone, R. Barbieri, and G. Valenza, "Integral pulse frequency modulation model driven by sympathovagal dynamics: Synthetic vs. real heart rate variability," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102736.
- [57] V. Catrambone and G. Valenza, "Complex brain–heart mapping in mental and physical stress," *IEEE J. Transl. Eng. Health Med.*, vol. 11, pp. 495–504, 2023.
- [58] S. Massaro and L. Pecchia, "Heart rate variability (HRV) analysis: A methodology for organizational neuroscience," *Organizational Res. Methods*, vol. 22, no. 1, pp. 354–393, Jan. 2019.
- [59] J. Mateo and P. Laguna, "Improved heart rate variability signal analysis from the beat occurrence times according to the IPFM model," *IEEE Trans. Biomed. Eng.*, vol. 47, no. 8, pp. 985–996, Aug. 2000.
- [60] K. V. Bisina and M. A. Azeez, "Optimized estimation of power spectral density," in *Proc. Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, Jun. 2017, pp. 871–875.
- [61] A. Arza, J. M. Garzón-Rey, J. Lázaro, E. Gil, R. Lopez-Anton, C. de la Cámara, P. Laguna, R. Bailón, and J. Aguiló, "Measuring acute stress response through physiological signals: Towards a quantitative assessment of stress," *Med. Biol. Eng. Comput.*, vol. 57, no. 1, pp. 271–287, Jan. 2019.
- [62] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vol. 37, no. 1, pp. 10–21, Jan. 1949.
- [63] R. Bailón, L. Mainardi, M. Orini, L. Sörnmo, and P. Laguna, "Analysis of heart rate variability during exercise stress testing using respiratory information," *Biomed. Signal Process. Control*, vol. 5, no. 4, pp. 299–310, Oct. 2010.
- [64] M. Orini, R. Bailón, L. Mainardi, and P. Laguna, "Synthesis of HRV signals characterized by predetermined time-frequency structure by means of time-varying ARMA models," *Biomed. Signal Process. Control*, vol. 7, no. 2, pp. 141–150, Mar. 2012.

- [65] A. Zheng and A. Casari, *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. Sebastopol, CA, USA: O'Reilly Media, 2018.
- [66] J. Aigrain, S. Dubuisson, M. Detyniecki, and M. Chetouani, "Person-specific behavioural features for automatic stress detection," in *Proc. 11th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, May 2015, pp. 1–6.
- [67] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 105524.
- [68] A. Y. Lin, M. Zhang, and S. Selpi, "Using scaling methods to improve support vector regression's performance for travel time and traffic volume predictions," *Time Ser. Anal. Forecasting*, pp. 115–127, Oct. 2018.
- [69] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Nov. 2011.
- [70] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.
- [71] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," in *Proc. 12th USENIX Conf. Operating Syst. Design Implement.*, 2016, pp. 265–283.
- [72] H. Ramchoun, M. A. J. Idriissi, Y. Ghanou, and M. Ettaouil, "Multilayer perceptron: Architecture optimization and training," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 4, no. 1, pp. 26–30, 2016.
- [73] F. Chollet. (2015). *Keras*. [Online]. Available: <https://keras.io>
- [74] C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: An empirical study of their impact to deep learning," *Multimedia Tools Appl.*, vol. 79, nos. 19–20, pp. 12777–12815, May 2020.
- [75] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2014.
- [76] L. A. Jeni, J. F. Cohn, and F. De La Torre, "Facing imbalanced data-recommendations for the use of performance metrics," in *Proc. Humaine Assoc. Conf. Affect. Comput. Intell. Interact.*, Sep. 2013, pp. 245–251.
- [77] S. Koldijk, M. A. Neerinx, and W. Kraaij, "Detecting work stress in offices by combining unobtrusive sensors," *IEEE Trans. Affect. Comput.*, vol. 9, no. 2, pp. 227–239, Apr. 2018.
- [78] I. Tougui, A. Jilbab, and J. E. Mhamdi, "Impact of the choice of cross-validation techniques on the results of machine learning-based diagnostic applications," *Healthcare Informat. Res.*, vol. 27, no. 3, pp. 189–199, Jul. 2021.
- [79] T. Head, M. Kumar, H. Nahrstaedt, G. Louppe, and I. Shcherbaty, "Scikit-optimize/scikit-optimize (v0.9.0)," Tech. Rep., 2021.
- [80] L. Liakopoulos, N. Stagakis, E. I. Zacharaki, and K. Moustakas, "CNN-based stress and emotion recognition in ambulatory settings," in *Proc. 12th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2021, pp. 1–8.
- [81] A. Karan and A. Kaygun, "Time series classification via topological data analysis," *Expert Syst. Appl.*, vol. 183, Nov. 2021, Art. no. 115326.
- [82] B. Behinaein, A. Bhatti, D. Rodenburg, P. Hungler, and A. Etemad, "A transformer architecture for stress detection from ECG," in *Proc. Int. Symp. Wearable Comput.*, Sep. 2021, pp. 132–134.
- [83] N. Rashid, T. Mortlock, and M. A. A. Faruque, "Stress detection using context-aware sensor fusion from wearable devices," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14114–14127, Aug. 2023.
- [84] A. Bhatti, B. Behinaein, P. Hungler, and A. Etemad, "AttX: Attentive cross-connections for fusion of wearable signals in emotion recognition," *ACM Trans. Comput. Healthcare*, vol. 5, no. 3, pp. 1–24, Jul. 2024.
- [85] E. Abdelfattah, S. Joshi, and S. Tiwari, "Machine and deep learning models for stress detection using multimodal physiological data," *IEEE Access*, vol. 13, pp. 4597–4608, 2025.
- [86] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Cham, Switzerland: Springer, 2009.
- [87] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003. [Online]. Available: <https://www.jmlr.org/papers/v3/guyon03a.html>
- [88] S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC Bioinf.*, vol. 7, no. 1, p. 91, Dec. 2006.
- [89] G. C. Cawley and N. L. C. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *J. Mach. Learn. Res.*, vol. 11, no. 70, pp. 2079–2107, 2010.
- [90] C. McCarthy, M. Boniol, K. Daniels, G. Cometto, K. Diallo, A. D. Lawani, and J. Campbell, "State of the World's nursing 2020: Investing in education, jobs, and leadership," WHO, Geneva, Switzerland, Tech. Rep., 2020.
- [91] H. G. Van Steenis and J. H. M. Tulen, "The effects of physical activities on cardiovascular variability in ambulatory situations," in *Proc. 19th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Oct. 1997, pp. 105–108.
- [92] J.-H. Hong, J. Ramos, and A. K. Dey, "Understanding physiological responses to stressors during physical activity," in *Proc. ACM Conf. Ubiquitous Comput.*, Sep. 2012, pp. 270–279.

MAXIMILIAN FECKE received the M.Sc. degree in computer science, with specialization in artificial intelligence from TU Dortmund University, Dortmund, Germany, where he is currently pursuing a Ph.D. degree. He is also a Researcher with the Healthcare Department, Fraunhofer Institute for Software and Systems Engineering ISST, Dortmund, Germany. His research interest includes artificial intelligence and its applications in healthcare.

JAKOB REHOF received the Ph.D. degree in computer science from the University of Copenhagen, Copenhagen, Denmark. He is currently a Professor of computer science with TU Dortmund University, Dortmund, Germany, where he holds the Chair for Software Engineering and serves as a Co-Director of the Lamarr Institute for Machine Learning and Artificial Intelligence. He is also the Director of Research Strategy with the Fraunhofer Institute for Software and Systems Engineering ISST, Dortmund, Germany. His research interests include program synthesis, verification, trustworthy artificial intelligence, and algorithms in logic and type theory.

• • •