Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

AI-Driven Panic Detection system using efficient learning model

Balaram Pal

balarampal23@outlook.com

Techno Bengal Institute of Technology Kolkata

Munshi Mostafijur Rahaman

Techno Bengal Institute of Technology Kolkata

Sakshi Verma

Techno Bengal Institute of Technology Kolkata

Soumyajit Paul

Indian Institute of Science Bangalore

Research Article

Keywords:

Posted Date: November 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11578280/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

AI-Driven Panic Detection system using efficient learning model

Balaram Pal[1*], Munshi Mostafijur Rahaman[2*], Soumyajit Paul[3†] and Sakshi Verma[4†]

[1*]Computer Science and Engineering Department, Techno Bengal Institute of Technology, Kolkata, 700150, West Bengal, India.
[2*]Computer Science and Engineering Department, Techno Bengal Institute of Technology, Kolkata, 700150, West Bengal, India.
[3]Computer Science and Engineering Department, Indian Institute of Science, Bengaluru, 560012, Karnataka, India.
[4]Computer Science and Engineering Department, Techno Bengal Institute of Technology, Kolkata, 700150, West Bengal, India.

*Corresponding author(s). E-mail(s): balarampal23@outlook.com; mostafijur.rahaman@gmail.com;
Contributing authors: soumyajit651@gmail.com; vermasakshi1976@gmail.com;
†These authors contributed equally to this work.

Abstract

Wearable devices such as smartwatches are becoming useful tools for monitoring different aspects of human health. In this study, an AI-based approach is developed to identify panic events using physiological and activity-related data obtained from commercial smartwatches. The study uses six features: gender, age, weight, heart rate, step count, and physical activity type. A dataset of 16,200 observations collected from 27 participants was used, with separate training and testing sets already provided. Before training the models, duplicate records were removed, categorical data was converted into numerical form, and MinMax normalization was applied. Different machine learning algorithms, along with three deep learning models, were tested and compared using Accuracy, Precision, Recall, F1-score, ROC-AUC, Matthews Correlation Coefficient (MCC), and Cohen's Kappa. The results revealed that machine learning models performed better than deep learning models for the selected features. K-Nearest Neighbours (KNN) achieved the best accuracy of 96.05%, while Random Forest and Decision

Tree also produced good results. This work shows that basic information available through smartwatches can be explored for panic event detection and may help in developing wearable-based systems for continuous mental health monitoring.

Keywords: Artificial Intelligence; Panic Detection; Smartwatch; Wearable Healthcare; Machine Learning; Deep Learning; Mental Health Monitoring

1 Introduction

Smartwatches have recently evolved from purely fitness trackers to sophisticated healthcare monitoring devices due to rapid development in the wearable computing field. Employing various on-body sensors, such as a heart rate monitor and motion detector, they serve as a ubiquitous and convenient platform for data logging related to physiology and activity during daily life. Together with artificial intelligence (AI), these wearable systems are very promising for smart monitoring applications which can help to recognize the abnormal physiological status of the wearer in an appropriate manner. Panic attacks can occur suddenly and cause extreme changes in physiological signs, such as elevated heart rate and motion activity, making the wearer unable to operate her smart devices or seek timely assistance. Although various personal safety and wearable monitoring systems have been introduced to alarm the caregiver, most existing solutions rely on user activation to send help, and use physiological measurements that are not available on mainstream smartwatches to warrant their timely and real-time execution of panic monitoring.

In addition to the ongoing research on the use of machine learning and deep learning approaches for the diagnosis of mental health using wearable sensor data, some works leverage physiological measures such as heart rate variability (HRV), electrocardiograms (ECGs), or lab-measured physiology (like temperature, respiration rate). However, many such approaches consider sophisticated physiological measurements, for example HRV, ECG, etc., which cannot be readily acquired through the typical smart watch API. Thus, feature selection in the practical use of wearable applications should consider only those measures available across the major consumer smart watch devices.

Table 1 summarizes the differences between existing wearable-based approaches and the smartwatch-based AI approach investigated in this study.

Table 1 Comparison of Traditional Wearable-Based Approaches and the Smartwatch-Based AI Approach

Comparison Criteria	**Traditional Wearable - Based Approaches**	**Smartwatch - Based AI Approach**
Monitoring Method	May rely on specialized wearable sensors or physiological monitoring systems	Uses features accessible through commercial smartwatch data
Data Requirements	May use ECG, HRV, respiratory rate, or other physiological measurements	Uses gender, age, weight, heart rate, step count, and activity type
Data Preprocessing	Depends on the data and methods used in each study	Duplicate removal, categorical encoding, and Min-Max normalization
Classification Models	Existing studies use wearable-data-based machine learning or deep learning approaches	Evaluates five Machine Learning models and three Deep Learning models
Performance Evaluation	Evaluation metrics and reported results depend on the individual study	Accuracy, precision, recall, F1-score, ROC-AUC, MCC, Cohen's Kappa, and training/testing time
Main Objective	Mental-health monitoring, diagnosis, or panic-attack prediction, depending on the study	Panic-state classification using smartwatch-related features

Motivated by these research gaps, this study introduces a panic detection framework using a set of features acquired from smartwatch: Gender, age, weight, heart rate, step count and activity type. This framework favors deployment features over the number of available features in terms of model performance. We evaluate and compare the performances of both conventional machine learning models, such as Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors, Support Vector Machine and deep learning models, including Multilayer Perceptron (MLP), Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), for this multi-class panic recognition problem.

In addition to presenting a complete implementation of our proposed system, a feature selection strategy focused on deployment feasibility, a comprehensive empirical study, and a thorough evaluation of different machine learning and deep learning methods in this journal paper, we also make the following main contributions:

- Propose an AI-based panic state detection system that collects physical signs and activity data from smartwatches.
- Rely on smartwatch features readily available in commercial wearables.
- Investigate the usability of five machine learning algorithms and three deep learning neural network algorithms to identify panic states.
- Evaluate the classification performance across accuracy, precision, recall, F1-score, AUC (ROC-AUC), MCC, Cohen's kappa and measure computation cost in terms of training and testing time.
- Briefly discuss design issues and computational overhead associated with the deployment of AI for smartwatch-based panic detection in a healthcare application.

The rest of the paper is organized as follows. Section 2 reviews the related studies. Section 3 presents the system architecture, along with a description of the data, preprocessing, the AI techniques used, and finally the experimentation and evaluation results.

2 Related Work

The growing prevalence of wearable devices and advances in machine learning have significantly fueled the research on detecting and predicting panic attacks, anxiety and stress related disorders. Over the past years, the scientific community has proposed several approaches including physiological sensing, smartwatches-based sensing, IoT-based solutions, personalized AI models, etc. For real-time mental health assessment.

In this section, we provide a detailed survey of recent work that focuses on wearable devices for the detection of anxiety and panic attacks.

One of the very early work aiming at wearable based panic monitoring is the AI-powered panic detection framework presented in [6]. Pal et al. Designed a framework which integrates the smartwatch sensor data and LLMs to detect panic events and provide intelligent alerts to panic sufferers. The paper highlighted the integration between smart wearables and modern AI to support mental health. A substantial amount of research efforts has been put to tackle the panic attacks prediction using wearable physiological signals.

Zahed et al. [1] presented a comprehensive scoping review of wearable technology for panic attack prediction including sensing methods, ML algorithms, dataset and limitations of current methods.

It showed the need for larger dataset, continuous data stream and personalization of the predictive models. Building on their earlier work [1], Zahed et al. [2] conducted a pilot study to examine whether physiological data collected from smartwatches could help predict panic attacks before symptoms appear. Their findings showed the potential of continuous smartwatch monitoring for early intervention. However, they also pointed out that further research involving a larger group of participants is needed.

Similarly, Oh et al. [3] developed and tested a machine learning model to predict panic attacks in patients with panic disorder using wearable electrocardiography (ECG) data. Their approach used heart-rate variability and other features derived from ECG signals. The results showed promising potential for using wearable sensor data to predict panic attacks. Individual differences have long been taken into consideration, and Kairamkonda et al. [4] explored the use of personalized machine learning models based on subject specific physiology for the prediction of panic attacks.

Their work shows that personalized models perform significantly better than generalized models, reflecting individual physiology pattern of the subjects before the attack.

In addition to prediction, some efforts have focused on panic detection systems. Ajitha et al. [5] proposed Panic SyncSight: An IoT enabled framework to detect panic attacks that integrates wearables and ML. Similar to Panic SyncSight, Tampu [6] is another mobile-based application combined with wearable devices, which has been developed for university students and enables the detection of anxiety and

panic attacks, with alert functionalities. Beyond panic disorders, real-time detection of anxiety in the wider context of mental health has also been studied.

The study in [7] investigated machine learning for the continuous detection of anxiety from wearable sensing devices.

Although focused on anxiety and not panic attack, the methodologies provided useful lessons on the on-line sensing of emotional state. Anxiety detection using smartwatches have also been studied in different clinical scenarios, as in Choi et al. [8] that designed a continuous monitoring of anxiety in post-stroke patients based on smartwatch and provided information on their quality of life and well-being. Stress prediction can be considered a crucial step for panic detection.

The study by Lee et al. [9] addressed stress detection in three populations of interest (MDD, Panic Disorder, controls) using a deep learning approach and RR intervals extracted from the physiological signals. Their model can successfully distinguish different psychological stress levels using a deep model. Furthermore, the field of mental health monitoring using wearables has been very active in recent years.

A systematic review by Kapogianni et al. [10] summarized wearable-based technologies applied to stress, mental health and well-being and reported key enabling factors and limitations (energy, adhesion, privacy, model generalization).

A negative impact of the wearable monitoring is the potential to generate health anxiety. Saxena et al. [11] conducted a study that focused on health anxiety experienced by middle-aged smartwatch users. They propose an evidence-based metric anxiety hierarchy for a smartwatch design in order to prevent the undue health anxiety triggered by real time feedback. Although most of the research presented so far focuses on single user data, there are some emerging studies on population based panic detection.

Zhao et al. [12] proposed a semantic and trajectory based method to detect pedestrian panic in crowd in real time.

The authors utilized semantic information on the intention and movement patterns of pedestrians, along with real time trajectory data to infer the panicked state. Also, Lazarou et al. [13] introduced a geo-biometric data driven system to detect, assess and visualize panic events in crowded situations. In summary, great advances have been made by wearable sensing, physiological signal analysis, machine learning and IoT-based systems for mental health assessment.

However, several research challenges need to be addressed in the future, such as using large scale and real life data, more advanced and personalize prediction algorithms, utilizing multimodal data, and designing generalizable models, while addressing user privacy, adherence and usability challenges to make these technologies practically usable for real time panic attack prediction and intervention.

3 Proposed System Architecture

The panic detection system has been developed with the goal of passively monitoring the activity and physiological information based on smart watches and identifying if the user is in a normal or panicked state. Fig. 1 depicts an overview of the designed system. The proposed system comprises five subsequent stages, namely, data acquisition,

data preprocessing, AI-based classification, panic state prediction, and notification for future intervention.

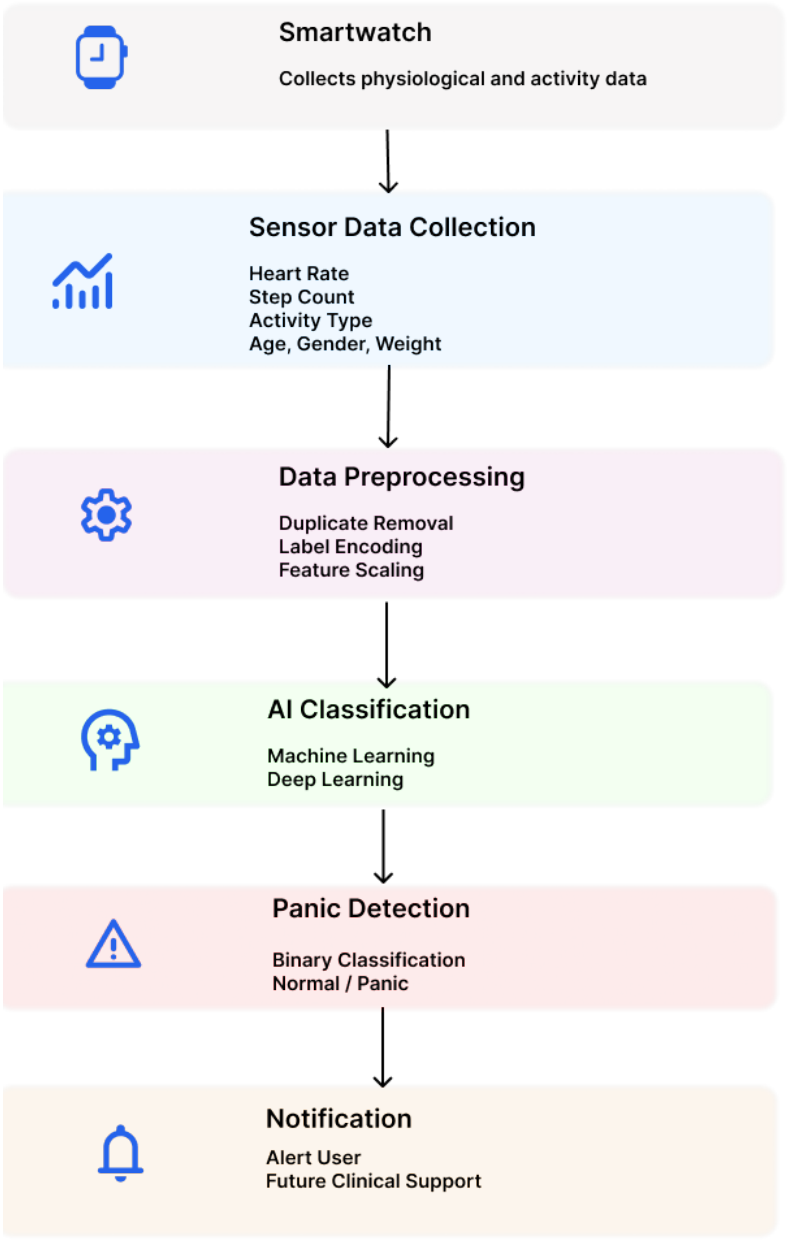


Fig. 1 Overall workflow of the proposed AI-based smartwatch panic detection framework.

The physiological and user-related information is obtained from a smartwatch. Gender, age, weight, heart rate, number of steps, and activity type were selected among the available features as these can be readily retrieved from many commercial smartwatch systems and do not demand custom medical devices.

Following data collection, the collected data are preprocessed to eliminate errors and format data for use in the learning algorithm. Duplicate records are discarded and categorical data is transformed to a numerical format. Features are normalized using the MinMaxScaler to ensure that each feature has the same range. Training and testing sets were then partitioned to facilitate the training and testing phases of our models.

The preprocessed data are fed to the AI classification stage of our design, where both conventional machine learning techniques and deep learning models have been used under similar experimental settings. Conventional models that were experimented with are Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbors, and Support Vector Machine. Deep Learning models such as Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) were also used.

Using the learned decision boundaries, the model predicts either a normal state or a panic state of the incoming smartwatch measurements. The prediction results may then be used by a smart watch application to prompt a warning or further intervention in the future. Although this paper deals only with the AI-based classification of such signals, the predicted panic state could be employed by smartwatch/smartphone applications for immediate tracking and assistance.

4 Dataset Description

To train and test the panic detection system, we use a dataset consisting of physical and behavioral signals recorded on the smartwatch from a variety of individuals in normal and panic states. Each instance within the dataset includes an individual profile and then provides physical and activity data along with the corresponding label. We utilize the training and test set divisions that we receive. Table 2 summarizes the information in the data set.

Keep in mind that other features exist in our dataset: `secureid`, `timestamp`, `HRV measures`, `HRMAD measures`, `speed`, and `linear acceleration`, which are ignored so that the proposed framework is available on smartwatches with public interfaces.

Thus, we train the model on six features readily available on most of today's commercial smartwatches: gender, age, weight, heart rate, step count and type of activity. The problem is a binary classification task in which the outcome attribute (*Outcome*) takes the form of one of two possible labels, 0 if the user is in a normal state and 1 if the user is experiencing a panic attack. In the case, our problem is to predict the outcome using data from smartwatch for any given data instance.

Table 2 Dataset attributes.

Attribute	Description
`secureid`	Individual ID for the user (do not include in model training)
`timestamp`	Datetime of acquiring the measurements (do not include in model training)
`gender`	The gender of the user
`age`	Age of the user (in years)
`weight`	The weight of the user (in kg)
`heartrate`	Heart rate of the user (beats per minute, BPM)
`stepcount`	The total number of steps made by the user
`activity`	Current physical activity of the user
`Outcome`	Label value (0=Normal, 1=Panic)

There are a total of 16,200 observations from 27 different participants collected in smartwatch.

There are 14,580 data samples designated for training and 1,620 data samples are set for testing.

There are 12,897 normal condition samples and 3,303 panic condition samples, indicating that the dataset is slightly unbalanced. Therefore, to fully analyze the result of the panic detection system, in addition to total accuracy, we also report additional evaluation measures.

Table 3 Dataset summary statistics.

Parameter	Value
Total Records	16,200
Participants	27
Training Samples	14,580
Testing Samples	1,620
Normal Samples	12,897
Panic Samples	3,303

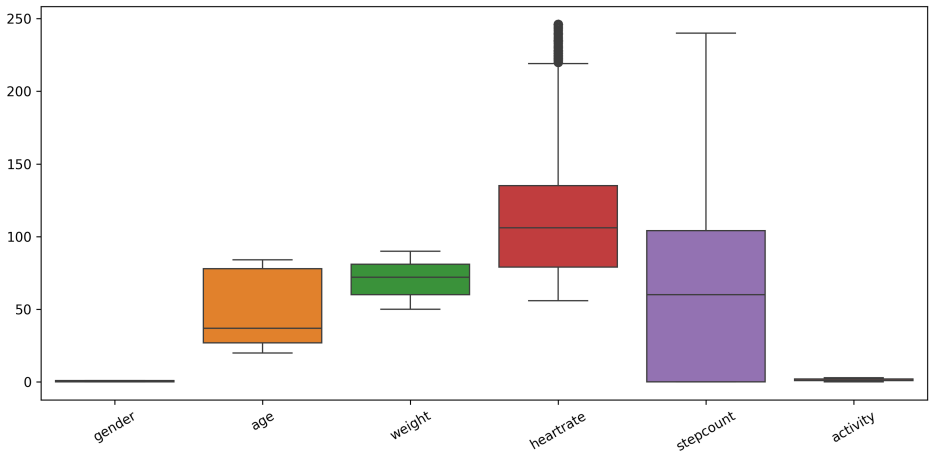


Fig. 2 Boxplot of the useful features of the dataset.

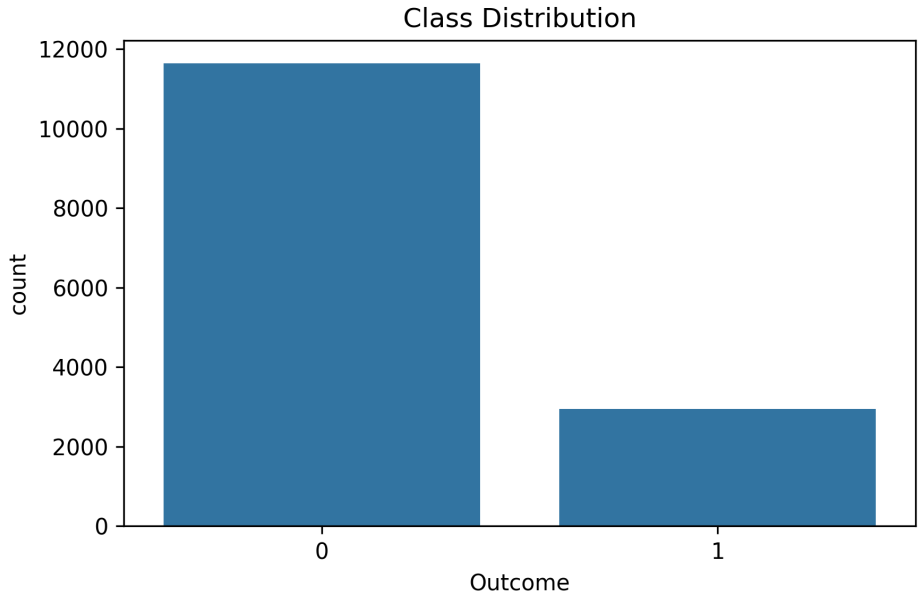


Fig. 3 Class distribution of the outcome feature in the dataset.

Subsequently, we apply machine learning and deep learning models to identify panic states based on the smartwatch data, following specific data preprocessing procedures outlined in the next section.

5 Data Preprocessing

Raw physiological data collected from wearable devices often contain inconsistencies that can affect the performance of machine learning models. Therefore, a preprocessing pipeline was applied before model training to improve data quality and ensure compatibility between all evaluated classifiers.

Initially, duplicate records were identified and removed to eliminate redundant observations that could introduce bias during model training. Removing duplicate samples helps ensure that each observation contributes uniquely to the learning process.

The dataset contains two categorical attributes, namely *gender* and *activity*. Since most machine learning and deep learning algorithms require numerical input, these categorical variables were transformed into numerical representations using label encoding. This encoding preserves the distinct category information while making the data suitable for model training.

Following categorical encoding, all input features were normalized using the MinMaxScaler technique. Because selected features have different numerical ranges, feature normalization prevents attributes with larger magnitudes from dominating the learning process. The MinMaxScaler transforms each feature into the range of $[0, 1]$ according to

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{1}$$

where x represents the original feature value, $x_{\min}$ and $x_{\max}$ denote the minimum and maximum values of that feature, respectively, and x' is the normalized value.

After preprocessing, the training and testing datasets were used to train and evaluate the models. The training data helped each model learn patterns from the data, while the separate testing data was used to check how well the models performed on unseen observations. All machine learning and deep learning models were evaluated using the same preprocessed dataset to maintain consistency and allow a fair comparison of their performance.

6 Machine Learning Models

Five supervised machine learning algorithms were selected to evaluate their performance in smartwatch-based panic detection. These algorithms use different approaches to classification, including linear models, decision trees, distance-based methods, and ensemble learning. Comparing them under the same experimental conditions helps assess their classification performance and computational efficiency for potential use in wearable devices.

6.1 Logistic Regression

Logistic Regression (LR) is a classification algorithm that estimates the probability of an observation belonging to a particular class using the logistic function. It is commonly used for binary classification and healthcare prediction because it is simple, fast, and easy to interpret.

In this study, Logistic Regression was used to examine how effectively the selected smartwatch features could distinguish between normal and panic states using a linear decision boundary. It requires relatively little computational power and offers fast training and prediction. However, it may not capture complex nonlinear relationships between physiological and activity-related features.

6.2 Decision Tree

A Decision Tree (DT) is a classification algorithm that divides data into smaller groups using a series of decision rules. These rules are arranged in a tree-like structure, making the model relatively easy to understand and interpret.

For panic-state classification, Decision Trees can identify nonlinear relationships between smartwatch measurements and the target classes. They also do not require feature scaling. However, a tree that grows too deep may overfit the training data, reducing its performance on new observations.

6.3 Random Forest

Random Forest (RF) is an ensemble learning algorithm that combines multiple decision trees to make predictions. Each tree is trained using a sample of the training data and a random selection of features. The final class is generally determined by majority voting among the trees.

By combining multiple trees, Random Forest can learn complex relationships between smartwatch features and panic states while reducing the limitations of relying on a single tree. It can also handle different types of input features without requiring feature scaling. However, it generally requires more computational resources than an individual Decision Tree.

6.4 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a classification algorithm that assigns a class to a new observation based on the labels of its nearest samples in the feature space. It uses a distance measure to identify the most similar observations and predicts the class based on their labels.

In this study, KNN was used to classify normal and panic states based on the selected smartwatch features. Its performance depends on how well similar observations form groups in the feature space. The algorithm is simple to implement and does not require a separate model-training process. However, prediction can become slower as the training dataset grows because distances must be calculated between the new observation and stored samples.

6.5 Support Vector Machine

Support Vector Machine (SVM) is a supervised classification algorithm that finds a decision boundary separating different classes while attempting to maximize the margin between them. Kernel functions can also be used to classify data when the relationship between features and classes is nonlinear.

SVM has been used in biomedical classification because it can identify useful decision boundaries in complex datasets. In this study, it was evaluated for distinguishing between normal and panic states using smartwatch features. Its performance depends on the choice of kernel and other parameters. Training can also become computationally demanding as the dataset grows, although prediction time depends on the selected model and implementation.

7 Deep Learning Models

In addition to traditional machine learning algorithms, three deep learning models were evaluated to examine their performance in classifying panic states using smartwatch data. Deep learning models can learn complex patterns from data, making them useful for different healthcare applications. In this study, three architectures were selected: Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM).

7.1 Multilayer Perceptron

A Multilayer Perceptron (MLP) is a feedforward neural network made up of an input layer, one or more hidden layers, and an output layer. It uses activation functions to learn relationships between input features and the target class.

In this study, MLP was used to classify panic states based on demographic, physiological, and activity-related features collected from smartwatches. It can learn nonlinear relationships between these features without requiring extensive manual feature engineering. However, its performance depends on the network structure and the selected training parameters. It may also require more training time than traditional machine learning models.

7.2 Convolutional Neural Network

Convolutional Neural Networks (CNNs) are commonly used for image processing, but one-dimensional CNNs can also be applied to structured and time-series data. They use convolutional filters to identify local patterns in the input data.

In this study, CNN was evaluated to determine whether these filters could help improve panic-state classification using smartwatch measurements. However, CNN performance depends on factors such as the size and diversity of the training dataset. When working with relatively small structured datasets, CNNs may not always perform better than traditional machine learning algorithms.

7.3 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a type of recurrent neural network designed to process sequential data. It uses memory cells and gating mechanisms to retain important information from earlier observations and manage how that information is used during learning.

Wearable devices can collect physiological measurements that change over time, making LSTM a potential choice for healthcare monitoring and activity recognition.

In this study, LSTM was evaluated to examine its ability to identify patterns associated with panic states. However, its effectiveness depends on the availability of suitable sequential data. LSTM models can also require longer training times and more computational resources than traditional machine learning models, which may be important when considering their use in real-time smartwatch applications.

8 Experimental Setup

This section explains the experimental setup and training process used to evaluate the proposed panic detection framework. To make the comparison fair, all machine learning and deep learning models were trained and tested using the same preprocessed dataset and predefined training and testing sets.

8.1 Experimental Environment

The experiments were carried out using Python 3.12.13. NumPy and Pandas were used for data handling and preprocessing, while Scikit-learn was used to develop the traditional machine learning models. The deep learning models were built using TensorFlow and Keras. Matplotlib was used to present the experimental results in graphical form. The software and their respective versions are listed in Table 4.

All experiments were performed on a CPU-based system with a dual-core x86_64 processor and 12.67 GB of system memory. No dedicated GPU was used for training or making predictions.

Table 4 Software environment used for model development and evaluation.

Software	**Version**
Python	3.12.13
NumPy	2.0.2
Pandas	2.2.2
Scikit-learn	1.6.1
TensorFlow	2.20.0
Keras	3.13.2
Matplotlib	3.10.0

8.2 Training Configuration

The machine learning models were trained on the predefined training set and evaluated using the independent test set. For the deep learning models, 20% of the training data was used as a validation set. This set helped monitor the models' performance during training and identify possible overfitting.

All deep learning models used the Adam optimizer with a learning rate of 0.001 and Binary Crossentropy as the loss function. The ReLU activation function was used in the hidden layers, while the output layer used Sigmoid for binary classification. Training was performed with a batch size of 32 for a maximum of 50 epochs. Early stopping was also applied to stop training when the validation performance no longer improved, helping avoid unnecessary training and reduce overfitting.

Table 5 summarizes the training parameters used for the deep learning models.

Table 5 Training configuration for the deep learning models.

Parameter	**Value**
Task	Binary Classification
Epochs	50 (Early Stopping Enabled)
Batch Size	32
Optimizer	Adam
Learning Rate	0.001
Loss Function	Binary Crossentropy
Hidden Activation	ReLU
Output Activation	Sigmoid
Validation Split	20%
Early Stopping	Yes

9 Performance Evaluation Metrics

Multiple evaluation measures were used to compare the predictive performance of the selected machine learning and deep learning models. Because the dataset contains moderate imbalance between the classes, accuracy alone may not fully represent how effectively the models classify normal and panic states. Therefore, additional metrics, including Precision, Recall, F1-score, Receiver Operating Characteristic Area Under the Curve (ROC-AUC), Matthews Correlation Coefficient (MCC), Cohen's Kappa, training time, and prediction time, were also considered.

Evaluation metrics are evaluated based on the confusion matrix, where True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) represent correctly and incorrectly classified panic and normal samples.

9.1 Accuracy

Accuracy represents the proportion of correctly classified samples among all observations and provides an overall measure of classification performance.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Although accuracy provides a general indication of model performance, it can be misleading when class distributions are imbalanced.

9.2 Precision

Precision measures the proportion of predicted panic cases that are actually panic cases.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

A high precision value indicates a low false positive rate, which is important to minimize unnecessary panic alerts.

9.3 Recall

Recall, also known as sensitivity, measures the ability of a model to correctly identify panic cases.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

In panic detection applications, high recall is desirable because failing to detect an actual panic event can delay timely intervention.

9.4 F1-score

The F1-score is the harmonic mean of precision and recall, providing a balanced measure when both false positives and false negatives are important.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

9.5 ROC-AUC

The Receiver Operating Characteristic Area Under the Curve (ROC-AUC) evaluates the ability of a classifier to distinguish between normal and panic states across different classification thresholds.

$$\text{ROC-AUC} = \int_0^1 TPR(FPR)\, d(FPR) \tag{6}$$

A higher ROC-AUC value indicates better discriminative capacity irrespective of the selected decision threshold.

9.6 Matthews Correlation Coefficient

The Matthews Correlation Coefficient (MCC) is a balanced evaluation metric that considers all elements of the confusion matrix and is particularly suitable for unbalanced datasets.

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{7}$$

An MCC value of 1 indicates perfect classification, while 0 represents random prediction.

9.7 Cohen's Kappa

Cohen's Kappa measures the agreement between predicted and actual class labels while accounting for agreement occurring by chance.

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{8}$$

where P_o is the observed agreement and P_e is the expected agreement by chance.

9.8 Training Time

Training time represents the total computational time required to train each classification model.

$$T_{\text{train}} = t_{\text{end}} - t_{\text{start}} \tag{9}$$

The shorter training time indicates greater computational efficiency during model development.

9.9 Prediction Time

Prediction time measures the time required by a trained model to classify the testing dataset.

$$T_{\text{predict}} = t_{\text{prediction_end}} - t_{\text{prediction_start}} \tag{10}$$

Prediction time is an important consideration for wearable healthcare applications, where rapid inference is required to support real-time panic detection.

10 Experimental Results

This section compares the performance of the machine learning and deep learning models used for smartwatch-based panic detection. All models were trained on the same preprocessed training dataset and tested on the same independent test dataset to ensure a fair comparison. Their performance was evaluated using the metrics described in the previous section.

10.1 Overall Performance Comparison

Table 6 presents the classification results and computational efficiency of all the evaluated models. It includes performance metrics, along with training and prediction times, to compare how well the models classify panic states and how suitable they may be for practical use.

Table 6 Performance comparison of the evaluated machine learning and deep learning models.

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC	MCC	Kappa
K-Nearest Neighbors	0.9605	0.9058	0.9160	0.9109	0.9950	0.8855	0.8855
Random Forest	0.9562	0.9040	0.8964	0.9001	0.9958	0.8721	0.8721
Decision Tree	0.9549	0.9128	0.8796	0.8959	0.9563	0.8674	0.8671
Support Vector Machine	0.9457	0.9439	0.8011	0.8667	0.9861	0.8373	0.8328
Logistic Regression	0.9136	0.8945	0.6891	0.7785	0.9440	0.7355	0.7259
MLP	0.8673	0.7076	0.6779	0.6924	0.9216	0.6081	0.6079
LSTM	0.7398	0.4570	0.2575	0.3294	0.6917	0.1949	0.1833
CNN	0.6843	0.3166	0.2351	0.2698	0.5312	0.0753	0.0740

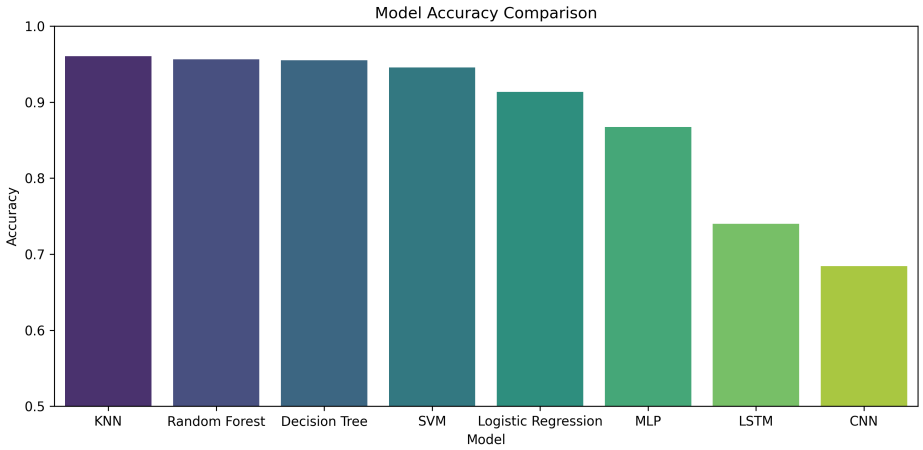


Fig. 4 ROC curves of the evaluated machine learning and deep learning models.

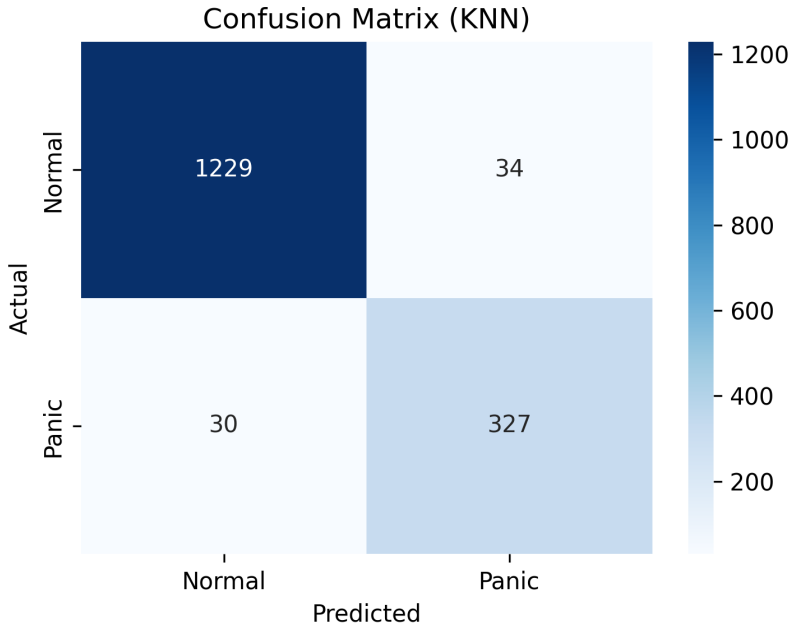


Fig. 5 ROC curves of the evaluated machine learning and deep learning models.

As shown in Table 6, the machine learning models generally performed better than the deep learning models across most evaluation metrics. Among all the classifiers, K-Nearest Neighbors (KNN) achieved the highest accuracy of 96.05%, recall of 91.60%, F1-score of 91.09%, and MCC of 0.8855. These results show that KNN provided the best overall classification performance for distinguishing between normal and panic states using the selected smartwatch features.

Random Forest also performed well, achieving an accuracy of 95.62% and the highest ROC-AUC score of 0.9958. This indicates that it was highly effective at distinguishing between the two classes across different classification thresholds. Although its training time was longer than that of KNN and Decision Tree, the model maintained strong predictive performance while remaining suitable for offline model development.

The Decision Tree classifier also achieved competitive results, obtaining an accuracy of 95.49%. Furthermore, it required only 0.0201 seconds for training and less than one millisecond for prediction, making it one of the fastest models evaluated in this study.

Support Vector Machine exhibited the highest precision (94.39%), indicating a relatively low false positive rate. However, its recall was lower than that of KNN and Random Forest, suggesting that a greater number of panic cases remained undetected. In addition, SVM required substantially longer training time compared with the tree-based models.

Among the deep learning approaches, the MLP produced the best overall performance, whereas the CNN and LSTM models achieved considerably lower classification

accuracy. The relatively modest performance of the deep learning models may be attributed to the structured nature of the dataset and its moderate size, where conventional machine learning algorithms are often capable of learning effective decision boundaries without requiring the large training datasets typically needed by deep neural networks.

In addition to predictive performance, the computational efficiency of each model was evaluated in terms of training time and prediction time. These metrics provide insight into the practical feasibility of deploying the models in real-time wearable healthcare applications. The computational performance of all evaluated models is summarized in Table 7.

Table 7 Training and prediction time of the evaluated models.

Model	**Training Time (s)**	**Prediction Time (s)**
K-Nearest Neighbors	0.0165	0.0114
Random Forest	1.2815	0.0231
Decision Tree	0.0201	0.0007
Support Vector Machine	11.1216	0.1839
Logistic Regression	0.0384	0.0005
MLP	–	0.2009
LSTM	45.4905	0.6493
CNN	19.3603	0.7139

As shown in Table 7, the conventional machine learning models generally required less computational time than the deep learning models. K-Nearest Neighbors and Decision Tree completed training within a fraction of a second, whereas Random Forest required a comparatively longer training time due to the construction of multiple decision trees. Support Vector Machine exhibited the highest training time among the evaluated machine learning models because of the computational cost associated with determining the optimal separating hyperplane.

Among the deep learning models, LSTM took the longest time to train, followed by CNN. This may be because these models require repeated calculations over several training epochs. The training time for MLP was not recorded, but its prediction time suggests that its computational requirements were similar to those of the other deep learning models.

For real-time panic detection, prediction time is more important than training time because the model needs to analyse data after it has been deployed. Logistic Regression and Decision Tree had the shortest prediction times, while K-Nearest Neighbors (KNN) also made predictions quickly despite relying on nearby training samples for classification. The deep learning models took relatively longer to make predictions. However, all the evaluated models completed prediction in less than one second. This suggests that they may be suitable for real-time smartwatch monitoring, although their actual performance will depend on the processing capabilities of the target device.

10.2 ROC Curve Analysis

The Receiver Operating Characteristic (ROC) curve is used to evaluate how well a model can distinguish between normal and panic states at different classification thresholds. It shows the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR). Unlike accuracy, which measures the proportion of correct predictions at a particular threshold, the ROC curve shows how the model performs across different thresholds. This provides a broader view of its ability to distinguish between the two classes, especially when the dataset has an uneven class distribution.

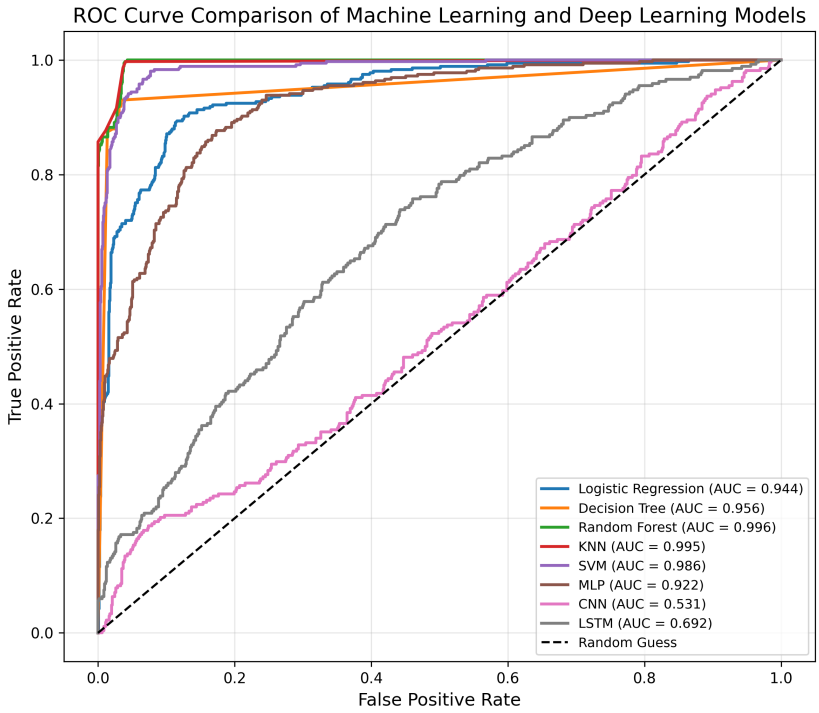


Fig. 6 ROC curves of the evaluated machine learning and deep learning models.

Figure 6 shows the ROC curves for all the classifiers evaluated in this study. According to Table 6, Random Forest achieved the highest ROC-AUC score of 0.9958, followed closely by K-Nearest Neighbors (KNN) with 0.9950. These scores indicate that both models can distinguish well between normal and panic states across different classification thresholds.

Support Vector Machine (SVM) also showed good performance, with a ROC-AUC score of 0.9861. However, its lower recall suggests that it missed more actual panic cases than KNN and Random Forest at the selected classification threshold. Decision Tree and Logistic Regression also achieved reasonable ROC-AUC scores, although their overall classification results were comparatively lower.

Among the deep learning models, MLP achieved the highest ROC-AUC score, while LSTM and CNN obtained lower scores. These results suggest that, for the dataset and features used in this study, conventional machine learning models generally distinguished between the two classes more effectively than the evaluated deep learning models.

11 Discussion

The comparison of the models shows that traditional machine learning algorithms performed better than the tested deep learning models for panic detection using the selected smartwatch features. K-Nearest Neighbors (KNN) achieved the best overall results across multiple evaluation metrics, while Random Forest and Decision Tree also performed well.

One possible reason for KNN's performance is that normal and panic observations may have similar patterns within their respective groups in the feature space. KNN classifies a new observation based on the labels of nearby samples, so it can work well when similar observations belong to the same class. Random Forest also performed well because it combines multiple decision trees to learn complex relationships in the data and reduce the risk of overfitting. Decision Tree provided competitive results with a relatively simple structure, making its predictions easier to interpret.

Support Vector Machine (SVM) achieved the highest precision, but its lower recall suggests that it missed more actual panic observations than a model with higher recall. In wearable healthcare applications, missing a potential panic episode may be a concern because it could delay a timely response. However, false positive predictions can also lead to unnecessary alerts and reduce user confidence. Therefore, the choice of model should consider the balance between recall and precision, depending on the needs of the application.

The deep learning models performed less well than the traditional machine learning algorithms in this study. One possible reason is that the dataset contains structured data with only six input features. This may limit the advantage of deep learning models, which often benefit from larger datasets and more complex patterns. The available data may therefore be better suited to traditional machine learning methods.

Another important part of this study is the selection of features for practical use. Although the original dataset included additional physiological measurements, the proposed framework uses only features commonly available through smartwatch interfaces, including gender, age, weight, heart rate, step count, and activity type. This choice may make the framework easier to adapt to different commercial wearable devices without requiring specialised sensors.

Overall, the results suggest that AI-based panic-state classification using commonly available smartwatch data is worth further investigation. The trained model could potentially be integrated into a smartwatch or smartphone application to analyse incoming data and notify users or designated caregivers when patterns associated with a possible panic episode are identified. However, such a system should be treated as a support tool and not as a replacement for professional medical assessment. Further real-world testing is necessary before it can be relied upon for timely intervention.

12 Limitations

Although the proposed framework shows promising results for smartwatch-based panic detection, some limitations need to be considered.

First, the model uses only physiological and activity-related features available from commercial smartwatches. This makes the framework easier to implement, but additional measurements, such as heart rate variability (HRV), electrocardiogram (ECG), and blood oxygen saturation (SpO_2), could be explored to determine whether they improve classification performance.

Second, the model classifies the data into two categories: normal and panic states. However, psychological stress and anxiety can occur at different levels. Future work could explore methods for identifying these different levels instead of using only two classes.

Third, the dataset used in this study includes observations from 27 participants. Although it contains many individual observations, data from a larger and more diverse group would help determine whether the models perform consistently across people with different demographic backgrounds and physiological characteristics.

Finally, the study evaluates the models using offline smartwatch data. The framework has not yet been tested through real-time implementation on commercial smartwatches or validated in clinical settings. Further testing is therefore needed to examine its performance in everyday use, including its reliability, user acceptance, battery consumption, and long-term operation.

13 Future Work

There are several ways in which the proposed smartwatch-based panic detection framework can be improved in future work. One possibility is to develop personalised models that learn an individual's normal physiological patterns. Since these patterns differ from person to person, a personalised model may help improve prediction performance compared with a single model used for everyone.

Future research could also include additional data, such as heart rate variability, blood oxygen levels, sleep patterns, and environmental conditions, depending on the sensors available on the device. Using these features along with the existing smartwatch data may help the model perform better in different real-world situations.

Another direction is to implement the framework for continuous monitoring on commercial smartwatches. Connecting it with a smartphone application or cloud service could support data synchronisation and timely alerts when a potential panic episode is identified. Such alerts could be made available to users or designated caregivers, depending on their preferences and consent.

Future studies could also test other machine learning techniques, including transformer-based models and hybrid architectures, to compare their performance with the models evaluated in this study. Explainable Artificial Intelligence (XAI) could also be explored to help users and healthcare professionals understand which features influence a model's predictions.

Finally, federated learning could be investigated to improve privacy when training models across multiple devices. This approach allows devices to contribute to model

training without directly sharing their raw physiological data with a central server. It may therefore support collaborative model development while reducing the transfer of sensitive personal information.

14 Conclusion

This study presents an AI-based framework for detecting panic states using physiological and activity data collected from commercial smartwatches. The framework uses features that are commonly available on these devices, including gender, age, weight, heart rate, step count, and activity type. By relying on these features, the approach avoids the need for specialised physiological sensors and may be easier to implement on different smartwatch platforms.

The study compares five machine learning algorithms and three deep learning models using a smartwatch-based dataset. The results show that the machine learning models performed better overall than the deep learning models tested in this study. K-Nearest Neighbors (KNN) achieved the best overall performance, while Random Forest and Decision Tree also gave competitive results. The computational analysis further suggests that traditional machine learning models can provide a useful balance between prediction performance and computational cost, which is important for wearable applications.

Overall, the findings suggest that smartwatch data can be useful for developing AI-based systems for panic state monitoring. In the future, such systems could help identify possible panic episodes and provide timely alerts to users, caregivers, or healthcare professionals. However, further testing with larger and more diverse groups of participants is needed before these systems can be used reliably in real-world settings. This study provides a starting point for further research into smartwatch-based panic monitoring and its potential role in digital healthcare.

References

[1] Zahed, K., Hallak, Y., Azad, K., Haidar, J., Ghomrawi, H.: Utility of wearable technology in predicting panic attacks: A scoping review. Digital Health **11**, 20552076251390475 (2025)

[2] Zahed, K., Saker, I., Hallak, Y., Aal, S.A., Talih, F., Ghomrawi, H.: Towards anticipating panic attacks using a smartwatch: A pilot study. In: Frontiers in Biomedical Devices, vol. 89435, pp. 001–03001 (2026). American Society of Mechanical Engineers

[3] Oh, H., Do, H., Maeng, C., Park, J., Yoon, T., Kim, J., Hwang, H., Choi, S., Huilin, P.: Panic attack prediction for patients with panic disorder via machine learning and wearable electrocardiography monitoring: Model development and validation study. Journal of medical Internet research **27**, 69045 (2025)

[4] Kairamkonda, D.D., McGinnis, E.W., Cherian, J., Mascia, G., Price, M., Hidalgo, J.E., Kim, J.N., Llorin, J., Stanton, K., Ha, J., et al.: Personalized machine

learning models for predicting panic attacks (2026)

[5] Ajitha, N., Ashwin, G., Divakar, S., Dhanushree, M., *et al.*: Machine learning-based panic syncsight: An iot-integrated panic attack detection system for enhanced therapeutic insights. In: 2025 5th International Conference on Soft Computing for Security Applications (ICSCSA), pp. 290–297 (2025). IEEE

[6] Curi, I.C., Quispe, E.Y., López, N.B.: Tampu: Mobile application using wearable technology for real-time detection and alert of anxiety and panic attacks in university students. In: 2025 IEEE 5th International Conference on Advanced Learning Technologies on Education & Research (ICALTER), pp. 1–5 (2025). IEEE

[7] Attacks, P.: Real-time detection of anxiety. Advances in Data Science and Management: Proceedings of ICDSM 2024, Volume 1 **1**, 269 (2025)

[8] Choi, S., Seo, W.-K., Jung, J.-M., Park, S., Seo, H.-C.: Smartwatch-based unobtrusive continuous anxiety tracker for evaluating post-stroke patients' quality of life. IEEE Journal of Biomedical and Health Informatics (2025)

[9] Lee, K.H., Cho, C.-H., Kim, A.Y., Jeon, H.J., Byun, S.: Deep learning-based stress detection from rr intervals in major depressive disorder, panic disorder, and healthy individuals. Frontiers in Psychiatry **16**, 1672260 (2025)

[10] Kapogianni, N.-A., Sideraki, A., Anagnostopoulos, C.-N.: Using smartwatches in stress management, mental health, and well-being: A systematic review. Algorithms **18**(7), 419 (2025)

[11] Saxena, A., Sinha, P., Kochuveetil, P.C.: Smartwatch health metrics and health anxiety among middle-aged urban professionals in india: An evidence-based metric anxiety hierarchy for dashboard design (2026)

[12] Zhao, R., Han, L., Cai, Y., Wei, B., Rahman, A., Li, C., Ma, Y.: Harnessing semantic and trajectory analysis for real-time pedestrian panic detection in crowded micro-road networks. Applied Sciences **15**(10), 5394 (2025)

[13] Lazarou, I., Kesidis, A.L., Tsatsaris, A.: Real-time detection, evaluation, and mapping of crowd panic emergencies based on geo-biometrical data and machine learning. Digital **5**(1), 2 (2025)