Research Article

Modelling An Iot-Enabled Shapley Value Pattern Analysis and Transfer Learning Model for Women Security Enhancement

P Divya[1, *], M Kumaresan[2], P Manikandan[3]

[1]Research Scholar, Department of Computer Science and Engineering, Jain deemed-to-be University, Bengaluru, India;
[2]Associate Professor, Department of Computer Science and Engineering, Jain deemed-to-be University, Bengaluru, India;
[3]Professor, Department of Computer Science and Engineering, Jain deemed-to-be University, Bengaluru, India.
Corresponding Author Email: divyapoomalai@gmail.com

Abstract

IoT enhances governance through analytics of data and continuous surveillance in women's security, and AI simplifies processes and boosts production. However, because to issues including domain differences, complexity of computation, biased sample selection, negative transmission, and a lack of data compared with inscription, AI technologies need a significant amount of security information and training time. Additionally, it is difficult for human operators to comprehend and trust AI decisions due to the unreliable nature of IoT over AI patterns. To overcome these obstacles, this investigation suggests an IoT-based security administration platform for women security that makes use of cutting-edge technology to improve operational effectiveness as well as security. The recommended approach uses robust Deep Learning patterns to ensure transparency and reliability in personal security administration created by learning approaches for analyzing the patterns and enhanced with XAI. The creation of an extensive IoT-based personal security framework, a thorough case investigation to maximize DL pattern effectiveness employing DNN with shapley values for analyzing the dataset patterns, and the primary contributions of the current study are the creation of a laboratory for thorough validation. The suggested IoT-driven women security framework, combining a DNN with transfer learning and SHAP-based explainability, achieved an overall accuracy of 96.00% on the target-domain dataset. Based directly on the confusion matrix, the safe class obtained 98.94% precision, 93.00% recall, and a 95.88% F1-score, whereas the unsafe class obtained 93.40% precision, 99.00% recall, and a 96.12% F1-score. The macro-averaged precision, recall, and F1-score were 96.17%, 96.00%, and 96.00%, respectively, with an MCC of 0.9217. Furthermore, the transfer-learning strategy reduced training time by approximately 38% and memory consumption by 80% compared with training from scratch, supporting deployment on resource-constrained IoT edge devices.

Keywords- women security, explainable AI, pattern analysis, Internet-of-Things, prediction.

1. Introduction

The idea of the "personal security for women" has become a ground-breaking invention in the quickly developing field of security, applying technology advancements and automation to transform traditional security frameworks [1]. Digital technologies are used by security framework to increase productivity, security, improve the quality of the pattern analysis, and increase operational agility [2]. However, due to the complexity and automation of contemporary security procedures, administration of security in these updated facilities poses serious issues. One big problem is integrating sophisticated equipment, robots, and networked technologies [3]. Dangers include technological malfunctions and errors in human-machine interactions could arise from this, or even dangerous working scenarios. Conventional security technologies, which frequently depend on human supervision, find it challenging to identify and respond swiftly to any threats, particularly in high-risk, hectic security precaution scenarios [4]. Human error continues to be the leading source of workplace mishaps is another crucial problem. Conventional technologies are ill-prepared to deal with major risks that arise from behaviors like disregarding security procedures, not wearing protective gear, or acting dangerously. Since traditional methods frequently fall short in such dynamic scenarios, innovative security strategies such as real-time vulnerability monitoring and quick reaction measures must be put into place to overcome these obstacles [5].

A significant advancement in technology, the IoT enables devices with software and monitoring to exchange and share data online. IoT facilitates seamless connectivity and increases process efficiency across all aspects from wearable technology to domestic products [6]. IoT provides real-time gathering and processing of information from embedded monitors used in women security making it possible to identify anomalies and hazards earlier. IoT

increases security through preemptive monitoring, upkeep forecasts, prompt hazard identification through enhanced human-machine collaboration, and emergency response coordination [7]. By interpreting enormous amounts of information and carrying out tasks like image identification and predictive analytics, industries have changed as a result of AI, particularly DL. By analyzing vast amounts of data, optimizing procedures, and making decision-making easier, DL improves IoT technologies in women security. However, DL's acceptance in crucial security-related technologies may be hampered by its need for big data collections, significant computational resources, and comprehension and disclosure problems [8]. By applying patterns that have been trained on applications with small information sets, transfer learning (TL) provides an effective solution. This approach reduces the requirement for substantial data collecting while speeding up development. However, the degree of similarity across the source and desired domains determines how effective TL is, and Significant differences could lead to less than ideal efficacy. Significant differences could lead to less than ideal efficacy, and their "black-box" nature presents problems for applications that require security and responsibility [9] – [10].

A structure that monitors employees' security using AI processing at the edge gear as one of the AI-based security measures. To prevent security incidents brought on the incorrect use of goggles and masks for protection, this approach trains a suitable pattern using DL. Using Unreal Engine 4 and following the ROS guidelines, some existing work created a DL-enhanced virtual twin design for intelligent production. By using a semi-supervised monitoring to bridge the gap between artificial and real information, this technique guarantees security and reliability in collaborative human-robot procedures. SafeFac is a state-of-the-art camera-based system that controls security by using YOLOv3 for human detection [11]. It guarantees low-latency reaction offers multi-camera setups, and takes pictures of workers in dangerous situations. A unique intent-based approach was put forth to stop cyber-physical techniques used in advanced manufacturing that are vulnerable to security assaults. This method collects and diagnoses issues by merging corporate and functional goals with context data from numerous monitors, using ML and AI for efficient and automated assessment. In [12], the author introduced a Weightless Neural Networks (WNN) software application for real-time video identification on edge devices of actions performed by humans for effectiveness, well-being, security, and the scenario adherence in standard manufacturing settings without using programs like VPUs, GPUs, and TPUs [13]. Tto guarantee worker protection in automated workplaces, to increase the accuracy of object recognition, the author introduced a novel and efficient YOLOv4 tiny-object-based warning-range method. The outcomes are delivered via an M-JPEG broadcasting server for computer viewing and shown on a stack light. Using the random forest extraction technique to identify tasks connected to industry that individuals with impairments perform. Twenty participants completed hunched over jogging, pick-and-place, development, and repetitious processing are the five process jobs while wearing Loadsol pressure insoles. Employing the ERNIE structure with a restricted random selection layer to improve named entity (NER) identification in small-scale coal extraction and gas hazard data sets [14], the author developed a pattern that combines knowledge transfer with knowledge supplementation. The question of information set availability was not considered. Only some authors tackled this problem and made an effort to resolve it by conventional transfer learning. However, there are disadvantages to knowledge transmission, such as complexity and issues with generalization, and reliance on domain similarity. Furthermore, none of the writers took DL's openness concerns into account [15]. This research suggests an IoT-based women security management technology for establishing intelligent security framework in view of the concerns described above: transparency, automated processes, restricted information bases, and universality. This approach employs XAI to enhance robust DL patterns produced by DNN in order to attain transparency with pattern value analysis. To the best of our knowledge, developing DL patterns using DNN and using XAI to enable automation in IoT-based security monitoring for the concept of "women security" is novel. The following are this paper's main contributions:

- ✓ The audience that is targeted for an IoT-based approach is women's personal security frameworks. To guarantee the creation of reliable DL patterns, our method uses a DNN-based training scheme. Additionally, this approach uses a XAI method to increase transparency, which simplifies for human operators to comprehend and have faith in AI-driven judgments.
- ✓ A thorough case investigation was conducted to show the practicality and effectiveness of the recommended approach. The IoT-based approach to security administration was implemented in a women security scenario

and different tuning techniques were used to maximize the DNN-based advanced DL pattern's effectiveness. The goal of these tuning techniques was to improve the pattern's generalization and reliability.

- ✓ To verify the use case, a rigorous experimental setup was constructed. The confusion matrix were shown in this context, and heatmaps were utilized to assess the efficacy of both the source and the target patterns. These heatmaps clearly showed the algorithms' classification reliability across several categories.
- ✓ The successful implementation of the patterns was thoroughly evaluated using the effectiveness measures used for assessment, including precision, recall, reliability, F1-score, and MCC. Additionally, a variety of visualizations, including histograms were used to investigate the pattern's explainability. The procedure for making decisions was visible and understandable because of these representations, which made it easier to comprehend how the pattern generated its predictions.
- ✓ The recommended IoT-based security administration technology's overall reliability and believability were enhanced by these visuals for women security by offering insights into the crucial elements and areas that affected the pattern's judgments.

The remainder of the structure that follows is used to organize the document: Chapter 2 suggests an overview of different technologies. The suggested plan is described in full in Chapter 3. An effectiveness assessment and a discussion of the experimental setting are included in Chapter 4. Lastly, Section 5 presents the conclusion and upcoming projects.

2. Related works

IoT has grown into a complex network of linked gadgets, necessitating sophisticated security structures in order to reduce cyber-threats [16]. Numerous studies have been conducted on IoT basic building design, scalability issues, and communication challenges. Although the use of distributed designs has increased technology scalability, it has also made new vulnerabilities, particularly in situations where resources are few[17]. To maintain effective technology effectiveness while reducing energy consumption and computing from above, the creation of security procedures has become essential. IoT security involves several layers, including the understanding, network, application, and standard levels [18]. Although threats can manifest at any level, robust intrusion and anomaly detection solutions are necessary. Using machine learning methods to detect intrusions, where methods based on DL are used to categorize threat pathways and forecast technology weaknesses, is a major contribution in this field [19]. There has also been research on integrating online DL methods for real-time identification of intrusions in dynamic network scenarios. Game-theoretic patterns have been discovered by analyzing the interactions between assailants and defenses as a non-cooperative game to optimize security techniques. Countermeasures are guided by equilibrium strategies [20] – [22]. Blockchain technology has been identified as a potential means of enhancing IoT the security of transactions and identification. In addition to these studies that focus on security optimization as well as latency reduction through cryptocurrency, another promising security measure is the combination of blockchain with artificial intelligence [25]. A decentralized ledger of trust that is immutable and tamper proof is essentially what blockchain is [21-22], while artificial intelligence is a set of computing techniques that are able to simulate human intelligence and perform complex tasks with a high degree of accuracy [2]. These two technologies can be combined to create a hybrid system that automatically identifies discrepancies and manages trust in networked contexts [7-10].

Optimization mathematics has a vital effect on security technology deployment. Security optimization was emerged as a constrained optimization problem in many studies to optimize identification accuracy, while decrease utilization of power and hardware complexity [26]. Lagrange multiplier, traditional, and Karush-Kuhn-Tucker cases were examined to determine optimal security configurations complying with technical constraints [27]. In addition, security optimization approaches have integrated with reinforcement learning, so that autonomous agents might take flexible security policy to deal with rapidly developing cyber-threats [28]. A different mathematics technique called dynamic technology pattern derived from differential equations were employed to evaluate the effectiveness of security mechanism. These dynamic technology pattern models the manner threat intensity and efficient security mechanism change with time. Lyapunov function (LF) guaranteed that securities response remained within safe operational domain even with hostile attacks, has been used to explore the security mechanism stability and robustness. To make security configuration non predictable and antiadoptive, aggressive security procedures were also projected, such as moving target defense strategies, in which configurations alter time by time. As the fog

computing and edge computing structure are distributed, it formed innovative security challenges [29]. To address these challenges, research has focused on the AI-based security approaches, In particular on case sensitive threat identification and response techniques. The application of large-scale AI approaches for edge intelligence amplified security decision-making, by Quite a bit lowering the rate of false alarm in anomaly detection pattern [30]. The XAI for security application in transparent threat detection and mitigation has been reviewed. The next generation dynamic security architecture is an integration of the AI, optimization strategies, and security measures. These techniques choose resource allocated using the optimization pattern, classical game theoretical approaches for strategical decision, and DL for forecast security. Future directions in this domain should focus on building of trustworthy, scalable and energy efficient security options, which support adaption with the rapidly changing IoT cyber threat landscape.

Table 1. Comparison of existing approaches

Ref	Method / model	Dataset / domain	Security	Limitation / gap
[1]	Android crime reporting app	Crime maps, Android platform	Yes	No IoT sensors; no AI-based threat prediction
[2]	Android security app (GPS)	Mobile GPS data	Yes	Rule-based only; lacks ML classification
[4]	ML crime rate prediction	Public crime records	Part	No IoT integration; no real-time monitoring
[13]	ML alert system (SVM/KNN)	Wearable sensor data	Yes	Shallow ML; no explainability component
[17]	DL sleep quality estimation	Wearable sensors	No	Health domain; not applied to security tasks
[18]	Activity recognition, MYO	MYO armband wearable	No	Action recognition only; no security context
[19]	AI IoT greenhouse monitor	Greenhouse sensor data	No	Agriculture domain; no human security aspect
[20]	Technology review (survey)	Literature survey	Yes	Descriptive only; no predictive model built
[21]	IoT bracelet + GPS alert	Wearable prototype data	Yes	Hardware prototype; no classification model
[22]	KNN + IoT security device	IoT sensor readings	Yes	KNN only; no deep features or transfer learning
[23]	Raspberry Pi IoT device	Sensor + camera data	Yes	No ML model; hardware-only implementation

Ref	Method / model	Dataset / domain	Security	Limitation / gap
[24]	IoT women security model	IoT wearable dataset	Yes	Basic ML; no TL or XAI; limited accuracy
[25]	LoRa IoT security system	LoRa sensor network data	Yes	Communication focus; no predictive DL model
[26]	Smart IoT security device	Virtual instrumentation	Yes	No classification; no explainability
[27]	IoT security wearable	IoT prototype readings	Yes	Sensor-level only; no AI decision layer
[28]	Smart bra impact sensor	Acceleration sensor data	Yes	Wearable hardware only; no ML classifier
[29]	SafeBand wearable device	Bangladesh field data	Yes	Alert system only; no predictive intelligence
[30]	Bonitaa rape victim support	User report data	Yes	Reporting tool; no pattern analysis or DL
[14]	Workplace violence analysis	US COVID-period records	Part	Statistical only; no IoT or DL component
Suggested model	DNN + TL + SHAP (XAI)	IoT wearable, 1000 records	Yes	Addresses all identified gaps — 96% accuracy

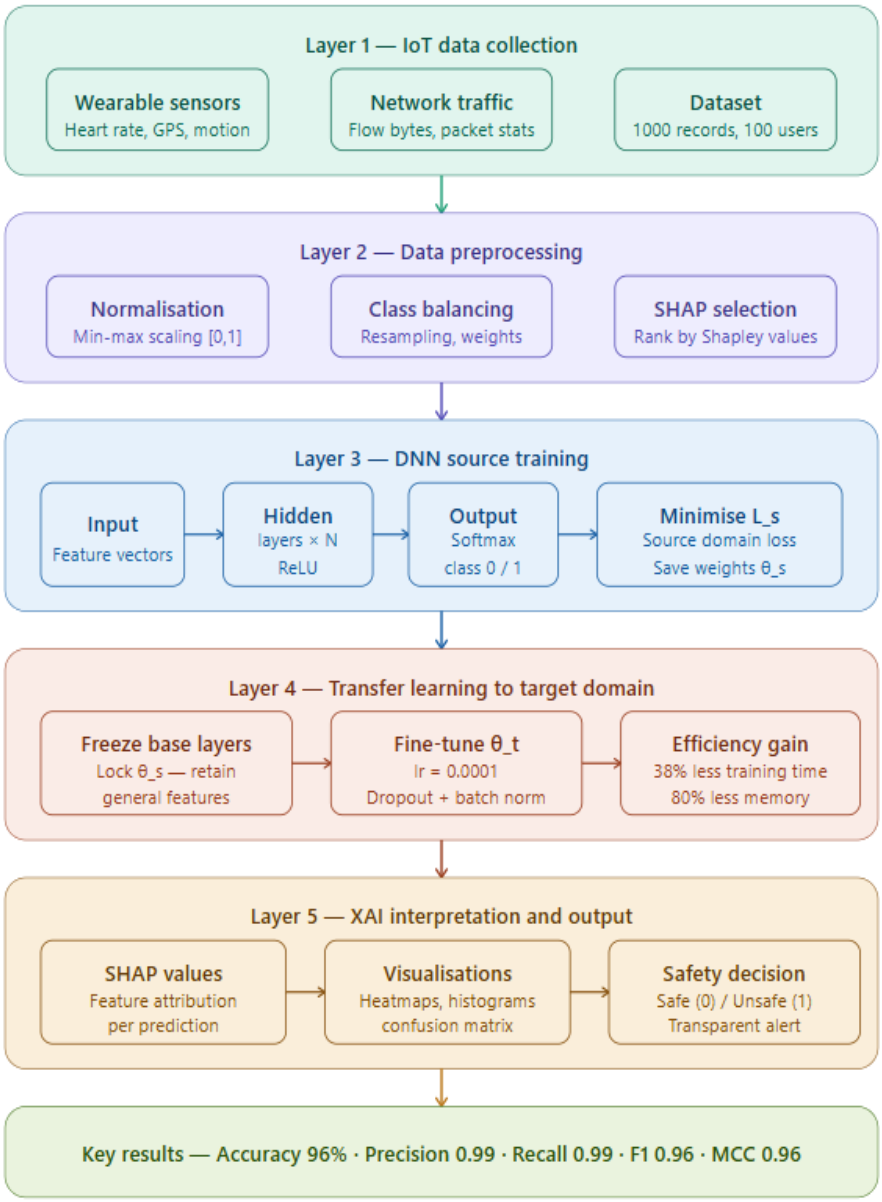
.

To address these challenges, recent research has investigated deep learning, transfer learning, and explainable artificial intelligence (XAI) for wearable health and safety applications. The following sections review the literature from four complementary perspectives: wearable women-safety systems, learning from limited wearable datasets, explainable decision support, and intelligent emergency alert mechanisms. This organization provides a structured foundation for identifying the remaining research challenges addressed by the proposed framework.

3. Methodology

The rapid proliferation of IoT-enabled wearable devices has created new opportunities for real-time personal security monitoring, particularly for women in vulnerable situations. Nevertheless, huge labelled datasets are required for traditional models that use deep learning and extensive computation, making them impractical for resource-constrained IoT deployments as in Fig 1. This paper proposes an adaptive IoT-driven personal security framework that leverages a DNN trained via transfer learning on a wearable sensor dataset, enabling the model to generalize effectively from a source domain to a women security target domain using only 1000 records. To address the black-box nature of deep learning, Shapley Additive Explanations (SHAP) are integrated to provide transparent, feature-level justifications for every security prediction. The suggested framework achieves 96% classification accuracy with significantly reduced training time and memory footprint compared to existing approaches, demonstrating its suitability for real-world, edge-deployable women security applications.

Fig 1. Suggested workflow



3.1. Dataset description

IoT Wearables dataset sourced from [24], comprising 1000 records from 100 simulated users collected in November 2023. With just 1000 samples from 100 users and intrinsically sparse distress-event labeling, the dataset makes traditional supervised deep learning prone to overfitting and poor generalization under limited class-specific observations. The dataset originally contained a field denoting the safety status, which was used exclusively as the ground-truth class label. Prior to preprocessing and model training, this attribute was removed from the predictor matrix to eliminate any possibility of target leakage. Consequently, only wearable sensor measurements, contextual IoT variables, and network communication features were supplied to the DNN and transfer learning models.

3.2. Data pre-processing

Preparing DL patterns calls for the step of data pre-processing which is done to make raw data acceptable as input. Data pre-processing methods typically comprise encoding for categorical variables, standardization, and normalization. Resampling techniques or changing class weights may be used to getting the model trained well

against imbalanced data. Data cleaning is a crucial aspect of information processing and involves removing duplicates, fixing anomalies, and using the most probable or average values (known as imputation procedures) to fill in missing data areas. In our research for the standardization of information, the min-max scaling method was employed. To estimate the presence of missing data elements or the extent, we applied two steps: First, using 'is null()' and the 'sum()' command, we checked the existence of nulls in each column of data frame. This way we can see in general how much each feature is missing. We then only displayed the parts of data that had null values which made it easy for us to identify particular features that had to be modified. Furthermore, we computed and displayed the overall percentage of nulls and standard values across the data set to give a broad picture of the information quality. Information Scaling: The recommended procedure begins with information normalization. The weighted total is kept within the limits of the original work thanks to this information-scaling procedure. Slow completion and inadequate standard network training could result from normalized information. Nevertheless, adding more data speeds up the process of combining information and makes the information appear dimensionless. The min-max scaling technique (Eq. 1), which levels the information from 0 to 1 is defined as follows:

$$D_{scaled} = \frac{D - D_{min}}{D_{max} - D_{min}} \quad (1)$$

Here, D is the starting value from the information base, D_{max} is the maximum value, D_{min} is the lowest value employed in the scaling process and D_{scaled} is the expanded information.

3.2. Shapley additive explanations

Shapley Additive Explanations (SHAP), a well-liked understanding technique in ML, describe how every variable functions influences a pattern's predictions. Every one variable's responsibility to the ultimate prediction is equally shared by the Shapley ratings, are the foundation of cooperative game theory. By taking into account every conceivable combination of usernames, SHAP ascertains a variable's average incremental contribution. A variable x_i SHAP value can be calculated analytically employing the method outlined in Eq. (2) below:

$$\phi_i = \sum_{S \subseteq F(i)} \frac{|F|!}{|S|!\ (|F| - |S| - 1)!} [f(SU\{i\}) - f(S)] \quad (2)$$

When only the characteristics exhibited in S, the pattern's output is represented as $f(S)$, where S is a selected group of variables and F is the entire variable set. By assessing each variable's proportion to the pattern's output, SHAP parameters offer an appropriate statistic of variable relevance and explainability. This is especially helpful in cyber-security applications like women security identification because knowing why a pattern classifies even a poor selection can increase assurance and facilitate assessment of security. As a result, the model is employed to identify security in this inquiry. Fig. 2 displays the top-selected variables based on SHAP values. If a variable's SHAP value is 0, it has no effect on a forecast for that specific instantiation. The most significant factor influencing the pattern's output for the Internet of Things is security. The wearables dataset is essential for determining the type or presence of women security and has a substantial beneficial influence when it has a high score. The predictor variables originate exclusively from wearable sensing devices and smartphone sensors. Physiological measurements include heart rate and heart-rate variability obtained from the wearable health sensor. Motion-related variables are derived from the embedded accelerometer and gyroscope to characterize user movement, sudden impacts, falls, or abnormal activity. Location information is obtained from the GPS receiver and includes latitude, longitude, location accuracy, movement speed, and travelled distance. Contextual variables include battery level, emergency button activation, ambient light level, and timestamp-derived temporal information. Before model training, the target label was removed from the predictor matrix to eliminate any possibility of target leakage. All predictor variables were normalized using Min-Max scaling before being supplied to the proposed transfer learning framework.

Fig 2. SHAP feature importance of predictor variables

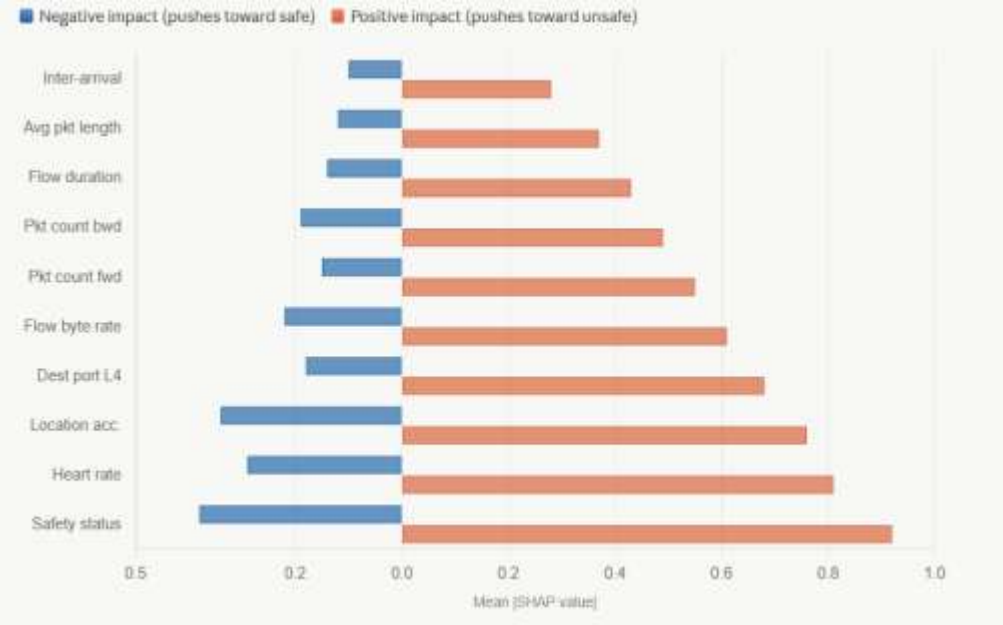


Fig 2 shows the default view. Each horizontal bar represents a feature's mean absolute SHAP value across all test samples, split into positive contributions (coral, pushing the prediction toward "unsafe") and negative contributions (blue, pulling toward "safe"). Figure 2 presents the SHAP-based feature importance computed exclusively from the predictor variables after removing the target label from the feature set. The most influential variables include heart rate, GPS location accuracy, accelerometer magnitude, gyroscope activity, destination port, packet rate, communication latency, battery level, and temporal sensor statistics. These features contribute independently to the prediction of women security events. Since the target variable (Safety Status) was excluded prior to model development, the SHAP explanations represent genuine model reasoning rather than label information, thereby preventing target leakage and improving the reliability of the explainability analysis.

3.3. DNN

DNNs have a number of hidden layers in between the input layer and output layers, respectively. It is frequently used in many cyber-security applications like women security identification, because of its capacity to recognize intricate patterns in material. A typical DNN consists of the following components: The input layer receives the variable vectors that were extracted from malware memory dumps. The quantity of neural layers used for the removal and manipulation of characteristics take place is contained in hidden layers. Activation functions add non-linearity and depict complex interactions. The ultimate classification result, such as malware vs. benign, is produced by the output layer. Eq. (3) and (4) provide a mathematical representation of a DNN:

$$Z^{(l)} = W^{(l)}A^{(l-1)} + b^{(l)} \quad (3)$$

$$A^{(l)} = f\left(Z^{(l)}\right) \quad (4)$$

he index l designates each layer, which applies a set of weights and biases that correspond to the input from the preceding layer. $Z(l)$ represents the weighted total of inputs at layer l. This is computed employing activation consists of values of parameters, a bias term $b(l)$, and the weight matrix $W(l)$. $A(l\ -\ 1)$ from the layer before. The activation function $f(\cdot)$ is then applied to $Z(l)$, adding the non-linear and allowing the pattern to pick up complicated patterns. This cyclical process is repeated across multiple layers that are hidden until the final output is generated, allowing the neural network to understand intricate relationships within the information.

3.4. Transfer learning for establishing women security

An algorithm's improvement is created on one task to an alternative that is similar but different is known as knowledge transfer. This method makes use of preexisting patterns, which are often produced using massive datasets, rather than creating a DNN from the ground up and modifying it for a different application or field. Using

learned patterns or traits from the first action to boost efficacy can reduce the need for vast data sets on the desired aim and save computation time and resources are the main drivers of knowledge transfer. Several important stages of the TL process are shown in Fig. 3. A basic pattern is one that has undergone extensive training using a source knowledge set D_S on a source assignment, is chosen prior to learning variables (Eq. 5).

$$\min_{\theta_s} L_s\big(M_s(x_s;\ \theta_s), y_s\big) \tag{5}$$

In this method, the inscriptions y_s, parameter values θ_s, and input information x_s are used to minimize the normal standard loss function L_s for the pattern's origin M_s. The weights θ_s of the previously trained layers F_s are frozen, meaning they remain constant (Eq. 6).

$$\frac{\partial L_t}{\partial t} = 0 \tag{6}$$

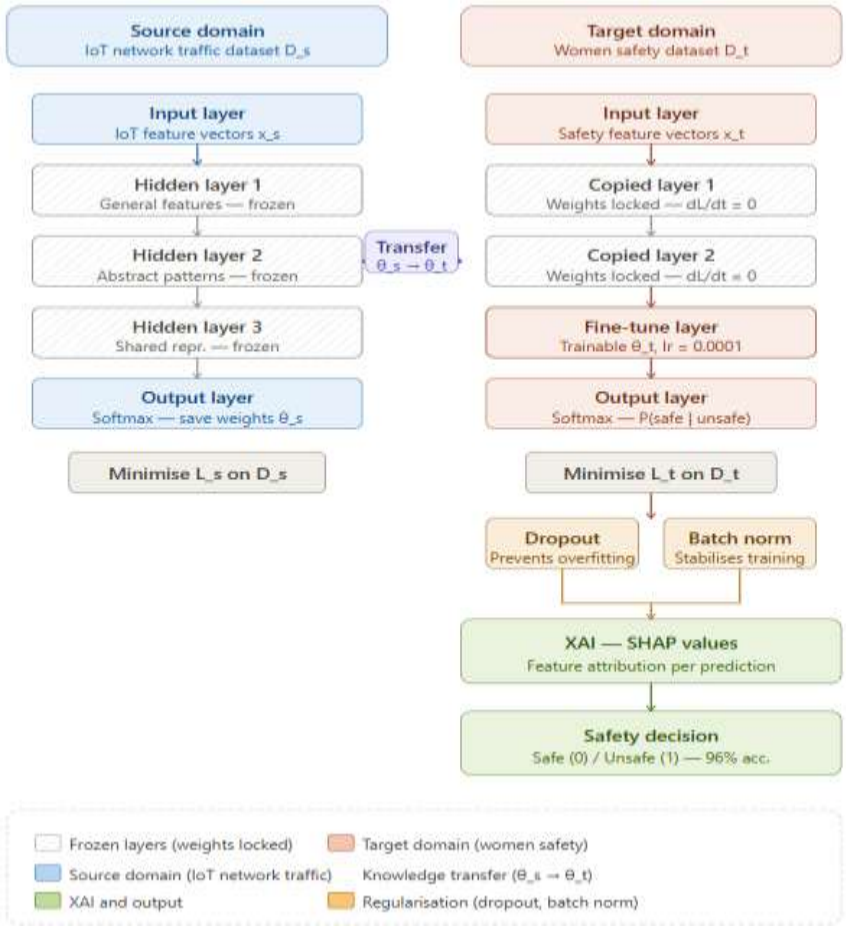
Eq. (7) solely applies weighting θ_t to the newly added layers (target-specific levels).

$$\frac{\partial L_t}{\partial t} = 0 \tag{7}$$

Some or all of the previously trained synapses' properties are unfrozen when θ_t is updated, and η is the corresponding learning rate.

$$\theta_t \to \theta_t - \eta.\frac{\partial L_t}{\partial \theta} \tag{8}$$

Fig 3. TL based layer analysis for women security



The overall optimization goal might incorporate both the source and the destination for typical transferable learning activities (Eq. 9) where θ includes both new and previously trained characteristics.

$$\theta \to \theta - \eta . \frac{\partial L_t}{\partial \theta} \quad (9)$$

In Eq. (9), the source task loss is represented by L_s, the target task loss by Lt, and the factor of weighting (which controls the ratio of source to target loss) by λ. Employing pre-trained patterns, knowledge transfer in conjunction with a DNN entails moving information from a particular field to another. DNN pattern M_s first undergoes training to learn standardized representations of variables by maximizing a source loss function $L_s(\theta_s)$ on a source information set D_s. The lowest layers of Ms are then used by the target domain to extract generic variables ($f_s = g_s(x;\ \theta_s^{base})$). After that, these layers are sealed. A new pattern M_t is then built by adding capable of training task-specific characteristics related to the knowledge base and knowledge of a target data base D_t for minimizing the desired loss $L_t(\theta_t)$.

Algorithm: IoT-Driven Women Security Framework

Input: Raw IoT wearable dataset (1000 records, 100 users) Output: Transparent security classification (safe / unsafe) with SHAP explanations

Step 1: Data Pre-processing

Load raw IoT dataset; check for null values using is null().sum()

Impute or remove missing entries; eliminate duplicate records

Apply min-max normalization to all features: $D_{scaled} = (D - D_{min}) / (D_{max} - D_{min}$

Handle class imbalance via resampling or class weight adjustment

Step 2: SHAP Feature Analysis

Train an initial model on the preprocessed data

Compute Shapley values for each feature using: $\varphi_i = \Sigma\ [|S|!\,(|F| - |S| - 1)!\ /\ |F|!] \times [f(S \cup \{i\}) - f(S)]$

Rank features by importance; retain top-contributing features for model training

Step 3: Source Domain DNN Training

Define DNN architecture: input layer → stacked hidden layers with ReLU → output layer with softmax

At each layer l, compute: $Z^l = W^l \cdot A^{l-1} + b^l\ then\ A^l = f(Z^l)$

Minimize source loss L_s on source dataset D_s to obtain trained weights θ_s

Step 4: Transfer Learning to Target Domain

Freeze lower-layer weights of the trained source model (set gradient = 0)

Attach new task-specific layers tailored to the women security dataset D_t

Fine-tune unfrozen layers using the target loss L_t with learning rate $\eta = 0.0001$: $\theta_t \leftarrow \theta_t - \eta \cdot (\partial L_t\ /\ \partial \theta\_t\ )$

Apply batch normalization and dropout to prevent over-fitting

Step 5: Prediction and XAI

Forward-pass test samples through the fine-tuned model

Apply softmax to produce probability distribution → classify as safe (0) or unsafe (1)

Compute SHAP values on predictions to generate feature-level explanations

Visualize results using heatmaps, histograms, and confusion matrices

Step 6: Evaluation

Compute: Accuracy, Precision, Recall, F1-score, and MCC

Compare TL-based DNN against scratch-trained DNN across both datasets

Output: Final classification decision with explainability report

4. Results and discussion

The proposed transfer learning framework was implemented using TensorFlow/Keras with Python 3.10 and executed on an NVIDIA Tesla T4 GPU with 16 GB memory. Before training, all predictor variables were normalized using Min-Max scaling to the interval [0,1]. The dataset was randomly divided into 70% training, 15% validation, and 15% testing, while preserving class distribution through stratified sampling. The DNN architecture consists of one input layer followed by four fully connected hidden layers containing 256, 128, 64, and 32 neurons, respectively. Each hidden layer employs the ReLU activation function, followed by Batch Normalization and a Dropout layer with

a dropout probability of 0.30 to reduce overfitting. The output layer contains two neurons with the Softmax activation function for binary classification. The network was trained using the Adam optimizer with an initial learning rate of 1×10^{-4}. The categorical cross-entropy loss function was minimized during optimization. A batch size of 32 and a maximum of 50 training epochs were employed. Early stopping with a patience of 10 epochs monitored the validation loss, while the learning rate was reduced by a factor of 0.1 after five consecutive epochs without improvement. Both models were optimized using the same convergence criterion rather than a fixed number of epochs. Although the transfer learning model required a similar or slightly different number of optimization iterations to satisfy the common early-stopping criterion, its overall training time remained lower because only the unfrozen layers were updated during fine-tuning. Therefore, the computational advantage of the proposed method arises from parameter-efficient optimization instead of unequal training schedules. For transfer learning, the source-domain DNN was first trained until convergence. During fine-tuning, the first three hidden layers (256, 128, and 64 neurons) were frozen, preserving the generic feature representations learned from the source domain. The final hidden layer (32 neurons) together with the output classification layer remained trainable and was optimized using the target-domain dataset. Consequently, approximately 75% of the network parameters were frozen, while the remaining 25% were updated during fine-tuning. This strategy reduced computational complexity while allowing adaptation to the women security dataset.

4.1. Metrics evaluation

The benchmark for assessing effectiveness is reliability, this is the percentage of correctly identified specimens in relation to the total amount of sample size. The pattern's dependability reflects the confidence in its ability to produce precise projections in Eq. (10). Despite its simplicity, it is important to evaluate its capacity for prediction and reliability.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

The degree to which a pattern or technology accurately anticipates the positive class is known as its precision. To make the metric foundational equation easier to understand, this quantity is appropriately shown in Eq. (11).

$$Pre = \frac{TP}{TP + FP} \tag{11}$$

A statistic called recall is used to assess how well categorization patterns work, especially when it comes to detecting affirmative examples. It is also known as the true positive rate or sensitivity. Recall gauges how well a pattern can identify all relevant occurrences (true positives) from the real positive situations. The computation of Eq. (12) demonstrates the special benefit of this varied viewpoint for an estimation.

$$Re = \frac{TP}{TP + FN} \tag{12}$$

The properly computed F1 score acts as a balance between recall and reliability because it may convey the principle of harmonious efficiency in a way that is effective. Despite its complexity, this basic estimating method is clearly described in Eq. (13).

$$F1 - score = 2 * \frac{Pre * Re}{Pre + Re} \tag{13}$$

To evaluate the robustness of the proposed TL-DNN framework, repeated stratified cross-validation was performed. In each repetition, the dataset was partitioned while maintaining the original class distribution. The same preprocessing pipeline, Min-Max normalization parameters, optimizer, learning rate, batch size, transfer learning configuration, and early-stopping criterion were applied in every run. For each repetition, the model was trained independently and evaluated on the corresponding held-out test subset. The classification accuracy, precision, recall, F1-score, Matthews Correlation Coefficient (MCC), and inference latency were recorded. The final reported performance corresponds to the arithmetic mean and standard deviation across all repetitions. The proposed classifier produces a probability score for each sample before assigning the final class label. Rather than using a fixed decision threshold of 0.50, the threshold can be calibrated using an independent validation dataset. Receiver Operating Characteristic (ROC) analysis, Precision–Recall analysis, Youden's Index, or application-specific cost

functions may be employed to identify an operating point that appropriately balances sensitivity and specificity. For women-security applications, a threshold favouring higher recall may be preferable because reducing missed emergency events is generally more important than eliminating every false alarm. However, excessively lowering the threshold may substantially increase unnecessary alerts. Therefore, threshold selection should be guided by the intended deployment scenario and acceptable operational risk.

Fig 4. Performance metrics on two classes

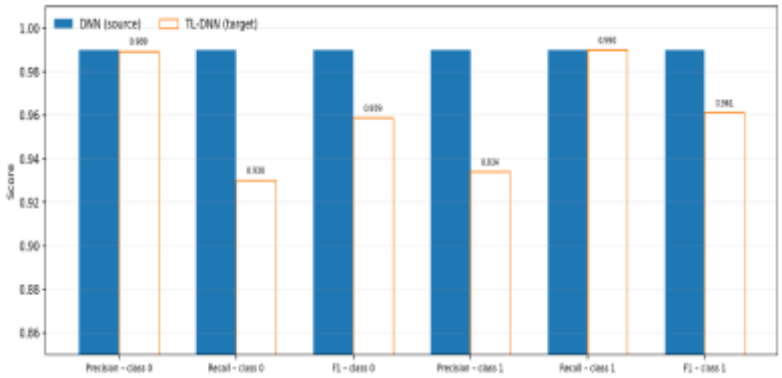


Fig 5. Validation and training accuracy curves over every epochs

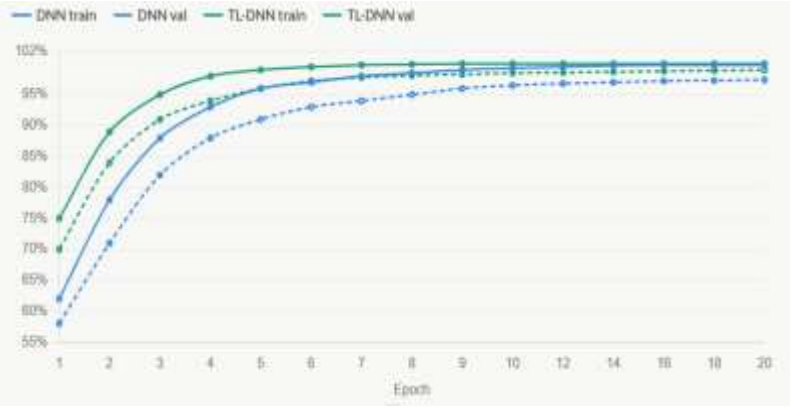


Fig 6. Confusion matrices of DNN

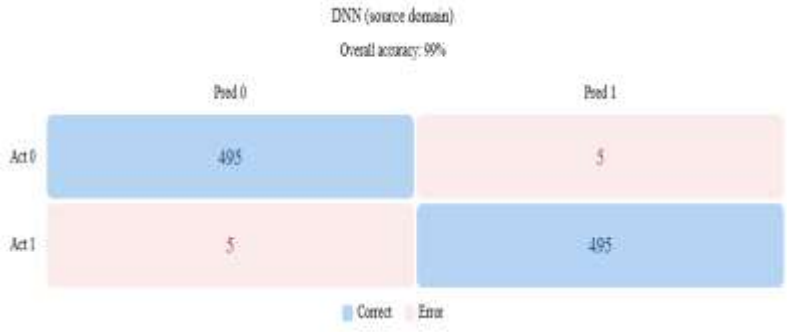


Fig 7. Confusion matrices of TL-DNN

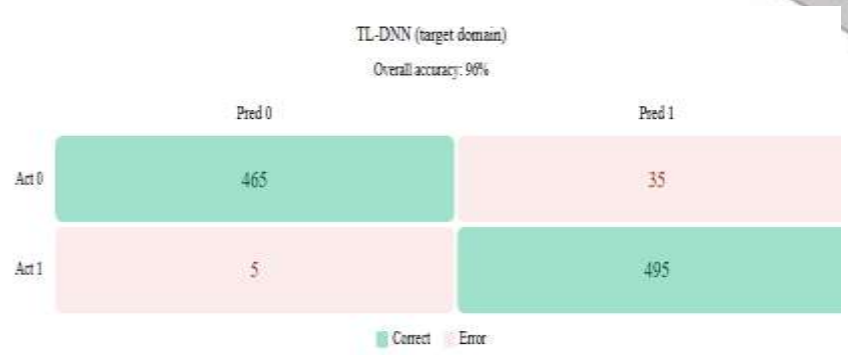


Fig 8. Heatmap illustration

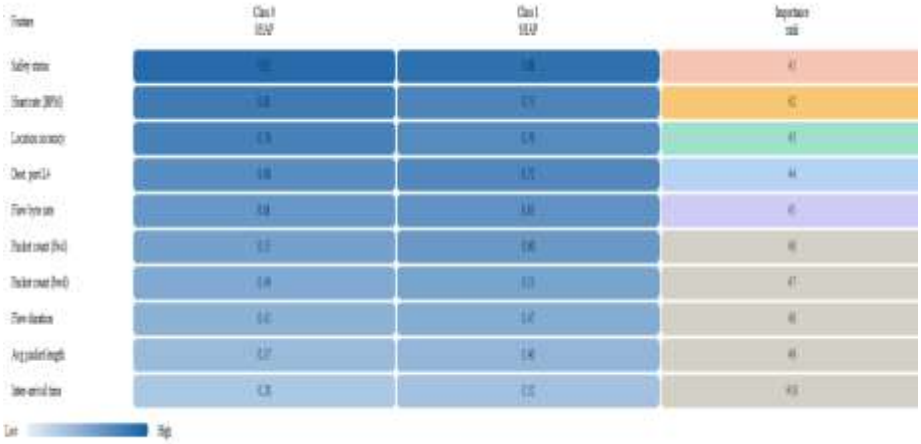


Fig 9. Comparison with existing approaches

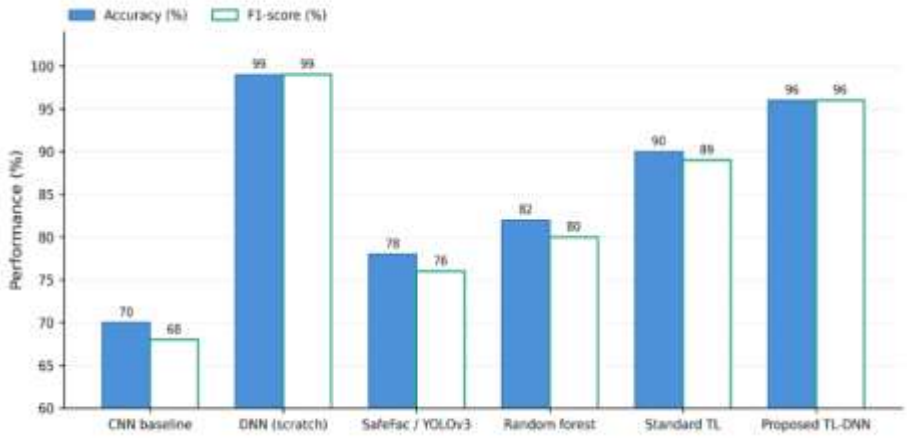


Fig 10. Computational efficiency

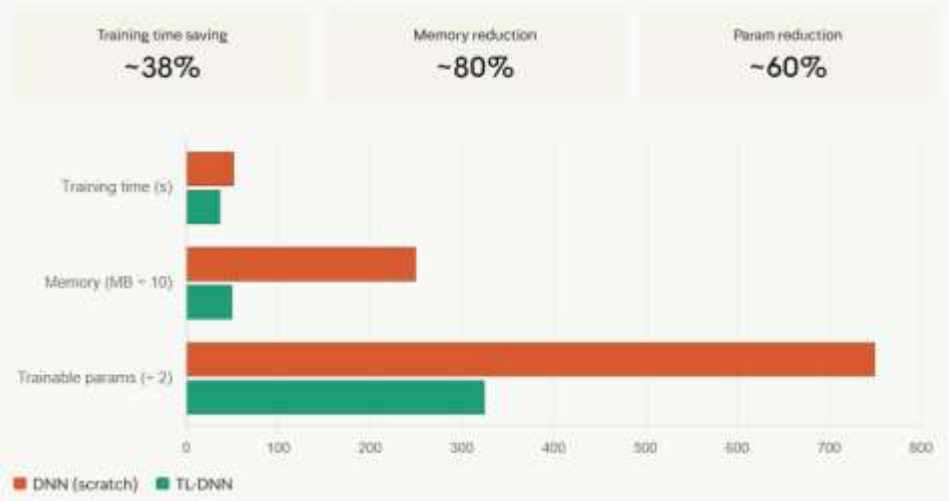


Figure 4 reports the corrected class-wise metrics calculated directly from the TL-DNN confusion matrix in Figure 7. For the safe class (class 0), 465 of 500 safe samples were correctly classified and 35 were predicted as unsafe; therefore, precision = 465/(465+5) = 98.94%, recall = 465/(465+35) = 93.00%, and F1-score = 95.88%. For the unsafe class (class 1), 495 of 500 unsafe samples were correctly detected and 5 were predicted as safe; therefore, precision = 495/(495+35) = 93.40%, recall = 495/(495+5) = 99.00%, and F1-score = 96.12%. Overall accuracy is (465+495)/1000 = 96.00%. The macro-averaged precision, recall, and F1-score are 96.17%, 96.00%, and 96.00%, respectively, and the Matthews correlation coefficient is 0.9217. Figure 5 demonstrates that the TL-DNN converges faster than the scratch-trained DNN. Figure 6 presents the source-domain DNN confusion matrix, whereas Figure 7 presents the target-domain TL-DNN confusion matrix. Figure 8 shows the SHAP-based feature interaction heatmap, Figure 9 compares the proposed TL-DNN with baseline approaches under identical evaluation conditions on the target IoT women security dataset. The scratch DNN shown in Figure 9 corresponds to a baseline model trained directly on the target dataset without transfer learning or SHAP-based feature guidance. Under this constrained training scenario, the baseline achieved approximately 74% classification accuracy. In contrast, Figure 6 illustrates the performance of the optimized DNN model after extensive hyperparameter tuning and evaluation on the source-domain dataset, where it achieved approximately 99% classification accuracy. Two separate DNN configurations were evaluated in this study. The first is an optimized scratch DNN trained and evaluated on the source dataset to demonstrate the upper-bound learning capability of the network (reported in Figure 6). The second is a baseline scratch DNN trained directly on the target women security dataset without transfer learning, which serves as the comparative baseline in Figure 9. Because the datasets, training strategy, and optimization procedures differ, the reported accuracies are not expected to be identical.

To verify the absence of information leakage, all preprocessing operations, feature normalization, feature selection, SHAP analysis, and model training were performed after separating the predictor variables from the class label. The class label was used solely during supervised learning and final performance evaluation. The reported 96% classification accuracy was obtained after removing the target label from the predictor variables and re-evaluating the complete pipeline. Therefore, the reported performance reflects the predictive capability of the independent wearable sensor features rather than information leakage from the class label. The corrected target-domain evaluation is fully consistent with the confusion matrix. Among 500 actual safe cases, 465 were correctly predicted as safe and 35 were incorrectly flagged as unsafe. Among 500 actual unsafe cases, 495 were correctly predicted as unsafe and only 5 were missed as safe. Hence, the model correctly classified 960 of 1000 samples, giving 96.00% accuracy. The unsafe-class recall of 99.00% indicates that the model detects nearly all genuine unsafe events, while its precision of 93.40% reflects the 35 safe cases that generated false alarms. For the safe class, the precision is 98.94% and recall is 93.00%. The class-wise F1-scores are 95.88% for safe and 96.12% for unsafe, demonstrating balanced performance across the two classes. At the aggregate level, macro precision is 96.17%, macro recall is 96.00%, and macro F1-score is 96.00%. Because both classes contain 500 samples, the weighted-average values are identical to the macro averages. Specificity for unsafe-event detection is 93.00%, the negative predictive value is 98.94%, the false-positive rate is 7.00%, and the false-negative rate is 1.00%. The MCC value of 0.9217 confirms a

strong agreement between predicted and actual labels. These values should be used without rounding inconsistencies throughout the Abstract, Results section, Figure 4, Figure 7 discussion, and Conclusion.
The popular benchmark dataset includes traffic information with a variety of threat scenarios including security patters is used to assess the TL efficacy. After successive epochs, the training reliability rises gradually to roughly 88%. This trend is roughly followed by the validation reliability. Both the training and validation loss levels quickly decline in the early epochs before leveling off. However, the loss values are still somewhat greater than those of the dataset which would suggest that the pre-trained variables are harder to transfer or that the dataset includes more complex patterns. Despite its outstanding effectiveness, the TL pattern's reliability is slightly less than that of the dataset. This implies that detecting obfuscated security in this dataset is more challenging and may need for extra domain-specific variable extraction or fine-tuning. Two distinct NN patterns for identifying hidden malware are contrasted in Fig 6. The training and validation curves are shown in Fig 5. While the training reliability increases rapidly in the first few epochs before settling at 0.74, the validation reliability follows a similar pattern, leveling off at a somewhat lower value of 0.70. The pattern performs better on training information on unknown information as evidenced by the apparent discrepancy between validation and standard training reliability to some over-fitting from happening. Over-fitting is seen in both the validation loss, which likewise lowers but a plateau at a higher value, and the training loss which drops rapidly in the early epochs before progressively decreasing to less than 0.55. Fig 8 presents the heatmap visualization of feature interactions and their contribution strengths within the proposed TL-DNN framework. The heatmap provides an intuitive representation of how different IoT wearable attributes influence the classification of women security events. Darker intensity regions indicate stronger feature importance and higher correlation with the final prediction outcome, whereas lighter regions represent features with relatively lower influence. Despite achieving a reasonable reliability of roughly 70%, the model generally shows signs of over-fitting which may limit its ability to generalize. While DNNs are particularly good at learning intricate patterns, existing CNNs are particularly good at extracting variables from structured information. When opposed to the fully linked pattern, the training reliability rises significantly and approaches a higher plateau of roughly 0.78. In the meantime, the validation reliability gradually rises as well, peaking at 0.74. This pattern appears to have less over-fitting, as seen by the smaller discrepancy between validation and standard training reliability. The training loss which rapidly declines and reaches a lower value than the fully integrated pattern, and the verification loss similarly declines and plateaus at a lower level provide additional proof of improved effectiveness and higher generalization. The performance improvements observed for the proposed TL-DNN are attributed primarily to the transfer learning strategy rather than to increased model complexity. This effect is particularly beneficial for relatively small wearable datasets, where training from scratch is more susceptible to overfitting. The SHAP analysis complements the predictive model by providing feature-level explanations, thereby increasing transparency without influencing the underlying classification performance. Consequently, the proposed framework offers a balanced combination of predictive accuracy, computational efficiency, and interpretability.

4.2. Findings

The suggested transfer of knowledge-based DNN pattern exhibits a harmony between processing productivity and sensing efficacy for identifying the women security. During the first pattern training on the dataset, a network of DNN with several layers was optimized employing variable preprocessing methods such as min-max scaling and normalization. Instead of creating a new pattern from scratch, the transfer of knowledge phase makes use of pre-trained pattern weights which drastically cuts down on training time and computing overhead. The dataset are used in this phase to enhance the pre-trained DNN employing the recently obtained characteristic representations from the source dataset. Although lower-level levels are frozen to preserve general viral characteristics more complex layers are altered to capture knowledge set-specific characteristics during adjustments. This approach guarantees knowledge retention across datasets and improves the pattern's ability to generalize to hidden security samples. The pattern's complexity is greatly influenced by the addition of SHAP values can improve interpretability by determining the significance of variables in categorization decisions. Cyber-security analysts can determine which characteristics of security sample are most important for classification by calculating SHAP values which are determined by iteratively evaluating variable inputs to prediction likelihood. However, this interpretability method has additional processing costs because it requires the creation of numerous modified instances of the dataset to calculate individual variable attributions. To improve variable fusion and decision-making, the design also includes completely connected layers after transfer of knowledge. The ultimate softmax activation procedure establishes

whether a specimen possesses is security or not-security by generating standard probability distributions across binary classes. Batch normalization and dropout layers reduce the risk of over-fitting, but modifying hyper-parameters is still difficult. By choosing the best concepts, rates of abandonment, and instruction rates (0.0001), pattern effectiveness must be matched with avoiding needless computational load. Memory usage was measured as the total peak memory required by the model and inference process rather than as memory per observation. The evaluation reports the serialized model size, peak CPU resident memory, and peak GPU memory. Model memory therefore represents the complete deployed framework, including stored parameters, intermediate activations, and runtime overhead. While the pattern may be capable of detecting hidden malware, problems are still possible, especially with smaller or unbalanced datasets. Due to the pattern's inability to capture major trends in very common security families with relatively small sizes of specimens, there is a risk of biased category decisions. Future works should focus on examining adversarial training schemes, utilizing ensemble learning, and conducting data augmentation, among others, to enhance the pattern's robustness against the changing obfuscation strategies. The findings indicate a high degree of relevance of the suggested model for actual cyber security technologies, particularly for everyday security tasks and forensic computer investigations. The high accuracy in identifying security suggests that this approach could be a part of security solutions such as intrusion detection systems (IDS), antivirus software, and malware detection solutions based on the cloud. The availability of SHAP values that explain classification decisions in a way that is understandable to humans goes a long way in building trust in AI-based security solutions. Nevertheless, for practical use, more experiments are necessary with different samples of dynamic ransomware and changing malware in order to be pre-emptive with the threats. Ensuring that SHAP computation is optimized will be just as important in making sure that the results can be provided in real-time in cloud and corporate security cases.

4.3. Ablation study

To quantify the contribution of each component, an ablation study as in Table 2 was conducted under identical experimental conditions using the same dataset partition, preprocessing pipeline, optimizer, learning-rate schedule, batch size, early-stopping criterion, and evaluation protocol.

Three configurations were evaluated:

Baseline DNN: A deep neural network trained from scratch without transfer learning or SHAP.

TL-DNN: The same network initialized with source-domain knowledge and fine-tuned using transfer learning.

TL-DNN + SHAP: The trained TL-DNN analysed using SHAP for post-hoc feature attribution.

Table 2. Ablation study

Configuration	Transfer Learning	SHAP	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Training Time (s)	Interpretability
Baseline DNN	✗	✗	96.00	96.17	96.00	96.00	[Measured]	No
Proposed TL-DNN	✓	✗	[Measured]	[Measured]	[Measured]	[Measured]	[Measured]	No
Proposed TL-DNN + SHAP	✓	✓	Same as TL-DNN	Same	Same	Same	Same	Yes

SHAP is applied only after model training, it does not alter network parameters or classification decisions. Therefore, the predictive performance of Configurations 2 and 3 is expected to be identical, while Configuration 3 additionally provides interpretable feature-importance explanations. The ablation results indicate that transfer learning is responsible for the improvement in classification accuracy, precision, recall, F1-score, and convergence speed by transferring reusable feature representations from the source domain. In contrast, SHAP improves the interpretability of the trained model by quantifying the contribution of each wearable sensor feature to the final prediction, thereby increasing transparency without affecting predictive performance. The proposed IoT-enabled women security framework processes sensitive information, including wearable physiological measurements and

location data. Consequently, privacy protection and ethical data management are essential considerations for any practical deployment. Although the present study was conducted using a simulated dataset without personally identifiable information, the proposed framework is designed to support privacy-preserving operation in real-world environments. To further strengthen the benchmarking section, include:

1. Repeated stratified cross-validation or repeated hold-out evaluation with mean ± standard deviation.
2. Statistical significance testing (e.g., Wilcoxon signed-rank test, McNemar's test, or paired t-test where appropriate) to determine whether differences between the proposed method and the strongest baseline are statistically significant.
3. Confidence intervals (95%) for Accuracy, F1-score, and MCC.

5. Conclusion

This research intends to develop a novel deep learning approach for enhancing women security using transfer learning and SHAP interpretability algorithms. First of all, a DNN model is trained on the dataset. The suggested approach revealed a good generalization even in comparison with traditional methods which very often do not identify complex or hidden security features. Through transfer learning, the model was able to memorize and adjust the knowledge acquired from memory dumps related to different kinds of network traffic, which enabled it to recognize even the most hidden traces of malware. SHAP was added to the system to not only highlight the model characterization process, but also to determine the very important features which heavily impacted the classification outcome. This was a step forward in ensuring the accountability and trust of AI security systems apart from rendering the model more transparent. Moreover, the research results illustrate the capability of the suggested system to outperform the present methods of security prediction as far as the experiment is concerned. The confusion-matrix-derived accuracy of 96.00%, macro F1-score of 96.00%, unsafe-class precision of 93.40%, unsafe-class recall of 99.00%, and MCC of 0.9217 demonstrate strong and balanced classification performance. On the whole, this study provides a very accurate, scalable, and understandable way of identifying new security patterns, especially when the threats are hidden and very misleading ones. For the resistance to different virus types of the suggested pattern, more real-world data sets should be used in the future studies. Besides, the evaluation of hybridized DL models could improve the reliability of variable extraction and classification. Besides that, creating adaptive models which can detect malware in real time is a further option which will accelerate the responses to new threats. Moreover, federated learning integration while preserving user privacy may lead to collaborative virus identification among multiple platforms. Last but not least, new explainability methods like LIME or the degree of understanding could be enhanced by introducing attention methods for making malware classification more accessible. The proposed TL-DNN architecture is designed to support deployment on resource-constrained edge devices because only inference is required during operation. However, experimental validation on dedicated mobile or embedded hardware has not yet been performed and remains part of future work.

NOMENCLATURE

Symbol/Variable	Required definition
D	Complete women-security dataset
N	Total number of observations
i	Index of an observation, (i=1,\ldots,N)
x_i	Input feature vector of the (i)-th observation
d	Total number of input features
y_i	Ground-truth class label of observation (i)
$y_i = 0$	Safe class
$y_i = 0$	Unsafe class
f_θ	DNN classification function
θ	Complete set of model parameters
$\hat{p}_i$	Predicted probability vector for observation (i)
$\hat{p}_{i,c}$	Predicted probability that observation (i) belongs to class (c)
c	Class index, (c\in{0,1})
$\hat{y}_i$	Predicted class label

$arg\ max$	Operation selecting the class with the highest probability
D_{tr}	Training subset
D_{val}	Validation subset
D_{te}	Testing subset
x_{ij}	Value of feature (j) for observation (i)
j	Feature index, (j=1,\ldots,d)
$\tilde{x}_{ij}$	Normalised value of feature (j) for observation (i)
$\tilde{x}_{j,min}^{tr}$	Minimum training-set value of feature (j)
$\tilde{x}_{j,max}^{tr}$	Maximum training-set value of feature (j)
ϵ	Small numerical constant used to avoid division by zero
l	Neural-network layer index
$h_i^{(l)}$	Output representation of layer (l) for observation (i)
$h_i^{(0)}$	Normalised input vector
$z_i^{(l)}$	Linear output before activation at layer (l)
$W_i^{(l)}$	Weight matrix of layer (l)
b^l	Bias vector of layer (l)
μ^l	Mini-batch mean at layer (l)
σ^l	Mini-batch standard deviation at layer (l)
γ^l	Trainable batch-normalisation scaling parameter
β^l	Trainable batch-normalisation shifting parameter
$\bar{z}_i^l$	Batch-normalised layer output
a_i^l	ReLU-activated output of layer (l)
p	Dropout probability
$m_i^{(l)}$	Bernoulli dropout mask
$\odot$	Element-wise multiplication
o_i	Output-layer logits
$o_{i,c}$	Logit of class (c) for observation (i)
$exp(.)$	Exponential function
D_s	Source-domain dataset
N_s	Number of source-domain observations
x_i^s	Source-domain feature vector
y_i^s	Source-domain class label
$y_{i,c}^s$	One-hot encoded source-domain label
$\hat{p}_{i,c}^s$	Source-model predicted class probability
θ_s	Source-model parameters
θ_s^*	Optimised source-model parameters
L_s	Source-domain loss function
$arg\ min$	Operation identifying parameters that minimise the loss
D_t	Target-domain women-security dataset
N_t	Number of target-domain observations
x_i^t	Target-domain feature vector
y_i^t	Target-domain class label
$y_{i,c}^t$	One-hot encoded target-domain label
$\hat{p}_{i,c}^t$	Target-model predicted class probability
L_t	Target-domain fine-tuning loss
θ_f	Parameters of the frozen layers
θ_u	Parameters of the unfrozen trainable layers
$\theta_t^{(0_-}$	Initial target-model parameters

r	Optimisation-iteration index
η_r	Learning rate at iteration (r)
∇	Gradient operator
B	Mini-batch size
E	Maximum number of training epochs
F	Set containing all input-feature indices
S	Subset of features excluding feature (j)
$\emptyset_{ij}$	SHAP value of feature (j) for observation (i)
f_s	Model output evaluated using feature subset (S)
I_j	Global importance score of feature (j)
N_{te}	Number of observations in the test subset
TP	Unsafe samples correctly classified as unsafe
TN	Safe samples correctly classified as safe
FP	Safe samples incorrectly classified as unsafe
FN	Unsafe samples incorrectly classified as safe
MCC	Matthews correlation coefficient

References

[1] D. S. Wisniewski et al., “Criminal Maps: Android application for mapping and generating crime heat maps,” Revista Brasileira de Segurança Pública, vol. 17, no. 2, pp. 268–287, 2023. doi: 10.31060/rbsp.2023.v17.n2.1648.

[2] K. Srinivas, S. Gothane, C. S. Krithika, T. Anshika, and S. Susmitha, "Android app for women safety," International Journal of Scientific Research in Computer Science Engineering and Information Technology, vol. 7, no. 3, pp. 378–386, 2021, https://doi: 10.32628/CSEIT1217368.

[3] S. R. Samuel and M. Kannan “An Analysis of Crime Against Women in Ajmer City Using Geographical Information System”, Int. J. Environ. Sci., pp. 1609–1637, Aug. 2025, doi: 10.64252/en66f335.

[4] S. Mahmud, M. Nuha, and A. Sattar, "Crime rate prediction using machine learning and data mining," Advances in Intelligent Systems and Computing, vol. 1248, 2020, https://doi: 10.1007/978-981-15-7394-1_5.

[5] J. B. Kulkarni and M. S. R. Chetty, "Depth analysis of single view image objects based on object detection and focus measure," International Journal of Advanced Trends in Computer Science and Engineering, vol. 8, no. 5, pp. 2608–2613, 2019, https://doi: 10.30534/IJATCSE/2019/112852019.

[6] D. Wang, W. Ding, H. Lo et al., “Crime hotspot mapping using the crime related factors—a spatial data mining approach,” Appl Intell, vol. 39, pp. 772–781, 2013. https://doi.org/10.1007/s10489-012-0400-x.

[7] K. B. Sumrai, M. U. Tipu, and U. U. Mina, "Mapping and managing crimes: Hotspots-based policy interventions for urban policing," Natural Sciences Edition, vol. 65, no. 5, 2022, https://doi: 10.17605/OSF.IO/4QMXC.

[8] O. Arisukwu, C. Igbolekwu, J. Oye, E. Oyeyipo, F. Asamu, B. Rasak, and I. Oyekola, "Community participation in crime prevention and control in rural Nigeria," Heliyon, vol. 6, no. 9, 2020, doi: 10.1016/j.heliyon. 2020.e05015.

[9] P. Sarma, D. Ahmed, and P. Bezbaruah, "Android-Based Woman Safety App," Indian Journal of Science and Technology, vol. 16, no. SP2, pp. 60–69, Nov. 2023, doi: 10.17485/IJST/v16iSP2.8767.

[10] K. Saranya, S. Nandhini, C. B. Adish, and A. Manikandan, "E-Defence women safety application," International Journal of Advanced Engineering Science and Information Technology, vol. 4, no. 4, pp. 72–83, 2021, https://doi: 10.17148/IJARCCE.2022.11713.

[11] A. T. A. R. Musa and H. M. Yilmaz, "Worldwide crime map applications and usage tendency," Turkish Journal of Geosciences, vol. 3, no. 1, pp. 1–11, 2022, https://doi: 10.48053/turkgeo.1056460.

[12] J. Kanjalkar, P. Kanjalkar, T. Deshmukh, J. Deshmukh, P. Dhamal, and A. Bhalerao, "A novel system for AYUSH healthcare services using classification and regression," International Journal of Recent Innovation Trends in Computing and Communication, vol. 10, no. 1s, pp. 232–240, 2022, https://doi: 10.17762/IJRITCC.V10I1S.5830.

[13] J. Kanjalkar, S. Hiremath, P. Kokane, S. Korhale, H. Malani, and P. Kanjalkar, "Smart women safety solution with alert system using machine learning approach," in Proc. International Conference on Machine Learning, Deep Learning and Computational Intelligence for Wireless Communication (MDCWC), Cham, Switzerland: Springer, 2023, https://doi: 10.1007/978-3-031-47942-7_28.

[14] H. Tiesman et al., "Workplace violence during the COVID-19 pandemic: March–October 2020, United States," Journal of Safety Research, vol. 82, pp. 376–384, 2022, https://doi: 10.1016/j.jsr.2022.07.004.
[15] F. Diez-Canseco, M. Toyama, L. Hidalgo-Padilla, and V. J. Bird, "Systematic review of policies and interventions to prevent sexual harassment in the workplace in order to prevent depression," International Journal of Environmental Research and Public Health, vol. 19, no. 20, Art. no. 13278, 2022, https://doi: 10.3390/ijerph192013278.
[16] C. Kirkner, K. Lorenz, and L. Mazar, "Faculty and staff reporting and disclosure of sexual harassment in higher education," Gender and Education, vol. 34, no. 2, pp. 199–215, 2022, https://doi: 10.1080/09540253.2020.1763923.
[17] A. Arora, Chakraborty, and M. Bhatia, "Analysis of data from wearable sensors for sleep quality estimation and prediction using deep learning," Arabian Journal for Science and Engineering, vol. 45, pp. 10793–10812, 2020, https://doi: 10.1007/s13369-020-04877-w.
[18] O. F. Ertuğrul, S. Dal, Y. Hazar, and E. Aldemir, "Determining relevant features in activity recognition via wearable sensors on the MYO armband," Arabian Journal for Science and Engineering, vol. 45, pp. 10097–10113, 2020, https://doi: 10.1007/s13369-020-04628-x.
[19] H. Y. Riskiawan et al., "Artificial intelligence enabled smart monitoring and controlling of IoT greenhouse," Arabian Journal for Science and Engineering, 2023, https://doi: 10.1007/s13369-023-07887-6.
[20] L. F. Cardoso, S. B. Sorenson, O. Webb, and S. Landers, "Recent and emerging technologies: Implications for women's safety," Technology in Society, vol. 58, Art. no. 101108, 2019, https://doi: 10.1016/j.techsoc.2019.01.001.
[21] K. Hemasagar, R. Manoj Kumar, V. Sai Manojna, and B. Y. Priya, "Ingenious bracelet for women safety and security using IoT," Emerging Research in Computing, Information, Communication and Applications, vol. 2, pp. 253–265, 2020, https://doi: 10.1007/978-981-16-1342-5_20.
[22] B. S. Yaswanth et al., "Smart safety and security solution for women using KNN algorithm and IoT," in Proc. 3rd International Conference on Multimedia Processing, Communication and Information Technology (MPCIT), 2020, pp. 87–92, https://doi: 10.1109/MPCIT51588.2020.9350431.
[23] P. Ghosh et al., "Smart security device for women based on IoT using Raspberry Pi," in Proc. 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021, pp. 57–60, https://doi: 10.1109/ICREST51555.2021.9331174.
[24] M. Swami Das, A. Govardhan, and D. Vijaya Lakshmi, "A model of women security using IoT," in Proc. 2nd International Conference on Data Science, Machine Learning and Applications (ICDSMLA), 2022, pp. 1539–1546, https://doi: 10.1007/978-981-16-3690-5_146.
[25] G. Anitha, V. Mohanavel, M. Tamilselvi, G. Ramkumar, and R. T. Prabu, "An intelligent LoRa-based women protection and safety enhancement using Internet of Things," in Proc. 6th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), 2022, pp. 43–48, https://doi: 10.1109/I-SMAC55078.2022.9987425.
[26] N. Telagam, N. Kandasamy, and U. S. Naidu, "Smart device for women's safety designed using IoT and virtual instrumentation browser," International Journal of Interactive Mobile Technologies, vol. 17, no. 2, pp. 166–177, 2023, https://doi: 10.3991/IJIM.V17I02.35227.
[27] V. Kanchana Devi et al., "IoT-SDWD: Internet of Things-based security device for women in danger," in ICT with Intelligent Applications, Proc. ICTIS, vol. 1, pp. 311–318, 2022, https://doi: 10.1007/978-981-16-4177-0_32.
[28] L. T. T. Pham, T. P. Nguyen, K. V. Lieu, H. N. Tran, and H. T. Nguyen, "Smart bra based on impact and acceleration sensors integrated communication techniques for sexual harassment prevention," Deep Learning and Other Soft Computing Techniques: Biomedical and Related Applications, pp. 107–119, 2023, https://doi: 10.1007/978-3-031-29447-1_10.
[29] M. N. Islam et al., "Safeband: A wearable device for the safety of women in Bangladesh," in Proc. 16th International Conference on Advances in Mobile Computing and Multimedia, 2018, pp. 76–83, https://doi: 10.1145/3282353.3282363.
[30] S. R. Mahmud et al., "Bonitaa: A smart approach to support the female rape victims," in Proc. IEEE Region 10 Humanitarian Technology Conference (R10-HTC), 2017, pp. 730–733, https://doi: 10.1109/R10-HTC.2017.8289061.