Secure self-adaptive recurrent neural networks for intelligent image recognition in the Internet of Things environments*

Yuliia Kostiuk[1,†], Volodymyr Sokolov[1,*,†], Karyna Khorolska[1,†], Maksym Vorokhob[1,†], and Nataliia Korshun[1,†]

[1] *Borys Grinchenko Kyiv Metropolitan University, 18/2 Bulvarno-Kudriavska str., 04053 Kyiv, Ukraine*

Abstract

The article proposes a self-adaptive recurrent neural model with a built-in protection mechanism for intelligent image recognition in the IoT environment. It accounts for data leakage during training and the real-time adaptation of models. The model architecture is based on Long Short-Term Memory (LSTM) with contextual adaptation to changes in data flows. The model processes sensor time series in real time and adapts to new threat patterns through online fine-tuning. To ensure confidentiality, homomorphic encryption and differential privacy are used, implemented in a modified DP-SGD algorithm (ε = 1.0). For transparency of decisions, Explainable Artificial Intelligence (XAI) is integrated using SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) methods. The achieved classification accuracy was 92.3% while maintaining the confidentiality of the input data. The F1 score was 0.89, confirming the approach's effectiveness and safety for anomaly detection in cyber-physical Internet of Things (IoT) systems, including smart environments. The proposed DP-SGD mechanism reduced the risk of confidential feature leakage by more than 85% according to the simulation results.

Keywords

recurrent neural network, differential privacy, homomorphic encryption, explainable AI, self-adaptive models, IoT security, online learning

1. Introduction

The growth in the number of IoT devices that generate real-time time series of sensor data has led to the widespread use of deep learning models for image recognition, anomaly detection, and forecasting tasks [1]. One of the most suitable architectures for such tasks is the Recurrent Neural Network (RNN), particularly the LSTM model, which effectively processes temporal dependencies [2–4]. However, the use of RNNs in distributed IoT systems is associated with risks related to the leakage of confidential data, both during training and during operation.

According to ENISA (2024), more than 63% of data leaks in IoT are related to the processing and transmission of information. OWASP classifies machine learning vulnerabilities as critical [4]. Attacks such as inversion, gradient leakage, and adversarial inputs require secure training methods.

Recent studies show that without proper protection, such models are vulnerable to several attacks, including model inversion attacks, in which an attacker can recover input data based on model parameters; adversarial input attacks, in which minor changes in input sequences lead to incorrect decisions; and attacks on gradients (membership inference, gradient leakage) during model training.

In addition, the dynamism, limited resources, and high context variability characteristic of IoT systems make it challenging to ensure the quality of model adaptation and its security.

* *CSDP'2026: Cyber Security and Data Protection, May 7, 2026, Lviv, Ukraine*
* Corresponding author.
† These authors contributed equally.
y.kostiuk@kubg.edu.ua (Y. Kostiuk); v.sokolov@kubg.edu.ua (V. Sokolov); k.khorolska@kubg.edu.ua (K. Khorolska); m.vorokhob@kubg.edu.ua (M. Vorokhob); n.korshun@kubg.edu.ua (N. Korshun)
0000-0001-5423-0985 (Y. Kostiuk); 0000-0002-9349-7946 (V. Sokolov); 0000-0003-3270-4494 (K. Khorolska); 0000-0001-5160-7134 (M. Vorokhob); 0000-0003-2908-970X (N. Korshun)

This raises the problem of developing a self-adaptive RNN model that not only adapts to changes in data in real-time, but also ensures privacy protection during both training and inference [2, 4]. This paper proposes just such a model with built-in cryptographic and privacy protection mechanisms.

Despite advances in image recognition using LSTM, the problem of personal data leakage during training remains unresolved. Existing methods often ignore protection at the model training stage. The threat of unauthorized access to the internal state of the model, parameters, or training dataset in the IoT environment requires a built-in protection mechanism directly integrated into the neural network structure [2–6]. The use of secure learning is particularly relevant, for example, through techniques such as differential privacy (DP), homomorphic encryption (HE), or distributed protocols such as federated learning. In the resource-constrained environment typical of IoT, lightweight, self-adaptive models capable of local retraining without compromising user data are required.

The goal of this research is to develop a self-adaptive RNN model with built-in protection mechanisms (differential privacy, gradient normalization) that enables intelligent image recognition in dynamic IoT environments.

The scientific novelty lies in the integration of adaptive mechanisms of RNNs with differential privacy to protect against data leakage during training in an IoT environment. A model with dynamic hyperparameter reconfiguration based on input flow and stochastic source characteristics is proposed [2–4, 7]. Protection against inversion, reconstruction, and membership inference attacks has been implemented, and an RNN has been combined with Explainable Artificial Intelligence (XAI) based on SHAP to improve decision transparency. The developed mathematical risk model confirms the effectiveness of a hybrid approach that combines adaptability, security, and explainability [8, 9].

The practical significance of the proposed approach lies in its applicability to a wide range of real-world tasks: image recognition in smart camera systems with distributed processing at the edge [10], secure biometric user authentication with local self-learning [11], building security devices for detecting anomalies in traffic or object behavior [3, 6], and functioning in decentralized computing systems without transmitting open data [6]. The developed architecture has demonstrated high resistance to information attacks, good adaptability in dynamic environments, and promising applications in digital security ecosystems, Industry 4.0, healthcare, and smart cities [12–15]. Thus, the research results present new opportunities for creating secure, autonomous information processing systems that address the challenges of the modern era, characterized by decentralized technologies and increasing cyber threats.

Within the framework of this study, several key scientific contributions have been made, underscoring the novelty and practical significance of the work. A self-adaptive recurrent kernel for IoT environments has been proposed, capable of dynamically adapting to changes in the feature space and the stochastic nature of sensor streams. An entropy-based mechanism for activating submodels has been developed, ensuring the selection of the optimal data processing path based on the level of confidence and information richness of the signals. A differential privacy module with DP-SGD support has been integrated, allowing training to be performed without the risk of sensitive information leakage. A secure weight update channel has been established, ensuring the correctness and safety of distributed training in the presence of potential influence from third-party agents. Additionally, a fuzzy decision-making module with confidence assessment has been implemented, improving the model's interpretability and controllability in real-world recognition scenarios. A large-scale experimental study was conducted under real-world IoT threats, which confirmed the effectiveness, robustness, and practical applicability of the proposed approach.

2. Literature overview

Research aimed at developing self-adaptive neural network models for operation in dynamic and heterogeneous IoT environments is becoming increasingly relevant. RNNs are an effective tool for analyzing time dependencies and streaming data. However, the growth in the volume of sensor

information and the risk of cyberattacks require built-in protection mechanisms. Therefore, modern approaches combine RNNs with differential privacy, federated learning, and methods for countering adversarial attacks.

In 2023, Feng J. et al. [2] proposed a tensor-based RNN with differential privacy for processing sequences in the IoT environment. The use of a modified backpropagation algorithm with noise protection ensured the protection of confidential user data while maintaining a high classification accuracy of over 95%.

In their work, Al-Qudah, M., and AlMahamid, F. [3] proposed a multi-stage system for analyzing streaming IoT data using entropy filtering. It was demonstrated that computing local entropy enables the effective removal of irrelevant or noisy samples, reducing classification errors.

Ullah, I., and Mahmoud, Q. [4] analyzed the effectiveness of hybrid CNN-RNN architectures for anomaly detection in IoT networks using the open NSL-KDD and BoT-IoT datasets. The proposed model achieved an accuracy of up to 98%, and its encoder and RNN classifier modules serve as a prototype for building our local recognition module in a secure environment.

Chen S. et al. [5] presented the RNN-DP approach, a mechanism for protecting the privacy of dynamic trajectories using differential privacy. The work confirms the practical applicability of RNNs for data processing in environments that are sensitive to private information leakage, which is relevant to IoT contexts.

In the study by Rezaei H. et al. [6], Federated RNN was implemented for secure intrusion detection in IoT, resistant to adversarial attacks, including label flipping and data poisoning. The implementation of coordinated aggregation mechanisms provided increased robustness without compromising performance.

Vitorino J. et al. [7] analyzed the vulnerability of RNN and DNN to adversarial attacks in the IoT environment. They found that traditional learning mechanisms do not provide sufficient resistance to malicious influences, especially in cloud-edge interaction environments [16].

Bai T. et al. [8] developed a method to evaluate the resilience of deep learning models to attacks in IoT systems using a generative AI-GAN model. The results demonstrated the effectiveness of this approach in generating controlled adversarial examples, thereby enabling the testing of the reliability of RNN architectures in various test scenarios.

A review of the literature revealed the existence of individual solutions for privacy, resistance to adversarial attacks, and adaptability in neural networks for IoT. At the same time, a comprehensive self-adaptive RNN architecture with integrated protection has not yet been proposed. This emphasizes the scientific novelty of the approach and justifies its relevance. The developed model meets the specified requirements, ensuring high accuracy and reliability in a changing cyber environment.

3. Research methods

The work employs a range of modern methods that encompass both mathematical and applied aspects. In particular, stochastic optimization and machine learning methods are used to adapt model parameters in a changing IoT environment [3, 5, 6]. The theoretical basis is functional analysis, including the apparatus of Hilbert spaces and the method of mean square approximation, which ensures convergence when constructing orthogonal projections.

The model is based on RNNs, specifically LSTM (Long Short-Term Memory) architectures, which are adapted to streaming data through self-adaptation mechanisms. To ensure confidentiality and resistance to attacks, cryptographic approaches, particularly Differential Privacy and Homomorphic Encryption, are employed [2, 5, 7, 17–19]. The practical implementation was carried out in the Python environment, using transport-level protection (TLS 1.3) and application-level protection, as well as simulation modeling to verify the effectiveness of the proposed approach.

4. Main material

In modern intelligent IoT systems operating under conditions of limited resources and high dynamics, image recognition tasks require the construction of self-adaptive models with self-learning and data protection mechanisms [3, 6, 11]. The classic teacher-led learning approach in such conditions is complemented by requirements for stability, adaptability, and confidentiality, which are implemented through differential privacy, gradient noise, and secure aggregation [2–5, 7, 18]. In self-learning tasks, a variational approach is employed to construct recurrent algorithms that adapt in real-time to changes without requiring centralized retraining.

The diagram (Figure 1) shows the general structure of data flows in a self-adaptive self-learning system based on RNN with an integrated protection module [2, 5]. From the input stream of features from IoT sensors, the data passes through preprocessing, prediction, and weight correction stages using protected gradients and differential privacy mechanisms [7, 18]. The results are stored in a secure model repository.

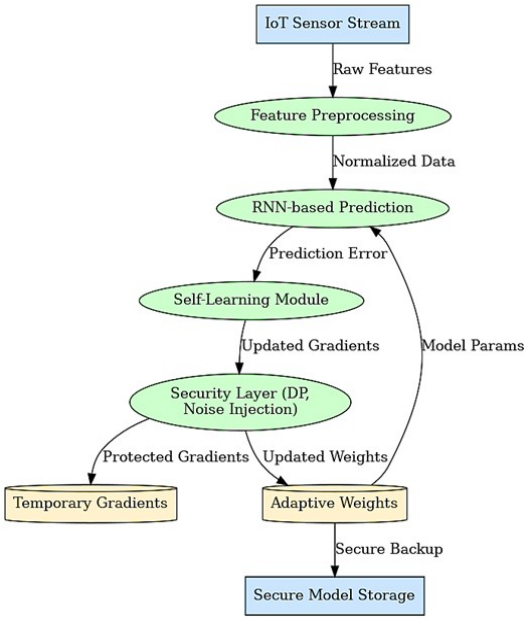

Figure 1: DFD level 1 of the self-learning algorithm of a RNN with built-in protection in the IoT environment

Figure 2 illustrates the functional structure of the self-learning and built-in protection module in the architecture of a RNN for the IoT environment. The diagram shows the step-by-step processing of the prediction error received from the RNN block after classification or prediction [3]. This signal is fed to the Gradient Estimation module, where a correction is formed to update the parameters. The gradients are stored in the Gradient Cache and fed to the Gradient Noise Injection module, where noise components are added to them to reduce sensitivity to individual samples [7]. The gradients then pass through a Differential Privacy Filter, which guarantees ε-privacy [18]. The final stage is Secure Weight Update, where the processed gradients are integrated into the model [17]. The new weights (Updated Weights) are used for further cycles of prediction and adaptation [12, 13, 15]. The presented structure implements a full-cycle of self-learning while adhering to the principles of confidentiality and resistance to attacks, which are key to recognition systems in a heterogeneous IoT environment.

Modern self-adaptive recurrent models with built-in protection mechanisms necessitate new training approaches that prioritize confidentiality, stability, and autonomy in the IoT environment. The combination of LSTM architectures, stochastic optimization, and information security methods enables the creation of systems that adapt to environmental changes without compromising privacy and with minimal human intervention [4, 7].

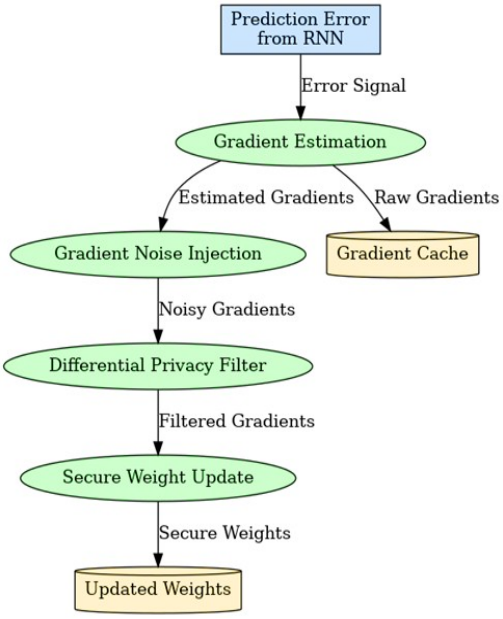

Figure 2: DFD level 2 detail of self-learning and protection in a RNN for IoT

In high-dimensional tasks with noisy data, stochastic methods with adaptive step control are effective [5, 6]. Additionally, the use of local models based on feature-space partitioning and geometric interpretation allows for the formation of ensembles that combine local sensitivity and global consistency [12, 20]. To formalize such adaptive processes, projections in Hilbert spaces are used, which ensure stable generalization under conditions of limited resources and incomplete data.

Consider a Hilbert space $\mathcal{H}$, in which a sequence of finite-dimensional basis subspaces $\{H_k \subset \mathcal{H}\}_{k=1}^{\infty}$ is given. Let us also give an element $f \in \mathcal{H}$, which represents the input signal or activity profile in a pattern recognition system. The problem is to find the orthogonal projection of the element f onto the closed linear envelope, $\overline{Lin}\{H_k\}$ using only local projections onto the subspaces H_k, as well as one-dimensional adaptive complements [2]. This approach allows us to implement projection computations in distributed nodes of IoT systems with protected memory and limited accuracy.

Let us denote by $f_1 = P_{H_1} f$ the orthogonal projection of the element f onto the first subspace H_1, and also introduce the extended $G_2 = Lin\{H_2, f_1\}$,. Then the process continues according to the following recursive formula

$$f_n = P_{G_n} f, G_{n+1} = Lin\{H_{n+1}, f_n\}, n = 1,2\ldots, N \tag{1}$$

where each new subspace G_{n+1} is enriched with the projection from the previous step. This approach corresponds to the concept of local self-learning in RNNs, where the model not only stores history, but also adapts it to new conditions [3]. Each update is implemented as a projection onto an adaptively augmented feature subspace, allowing for generalization of the input data even in limited volumes without requiring a complete global model. According to the convergence lemma, for any element $f \in H$ the sequence of projections $\{f_n\}$ converges

$$\lim_{n \to \infty} f_n = f^* \tag{2}$$

where $f^* \in \overline{Lin}\{H_k\}_{k=1}^{\infty}$ is the orthogonal projection of f onto the closed linear envelope of subspaces. This means that the process of local adaptive self-learning approximates the input signal to the global representation, without centralized generalization [21].

To avoid losing important features in the changing IoT environment, the concept of an ordered sequence of subspaces is introduced. A sequence $\{H_k\}$ is called ordered if

$$\forall M \in N, \forall H_i \in \{H_k\}, \exists k \geq M : H_k = H_i. \tag{3}$$

This is a formal definition of a sequential sequence that ensures no feature is lost during long-term training. That is, each subspace appears infinitely often in any of its subsequences $\{H_k\}_{k \geq M}$, where $M \in N$ [12, 15]. This condition ensures the regular updating of local properties in the feature space, which is critical for self-learning models operating in dynamic or unstable IoT scenarios.

The mathematical core of such systems can be considered in the context of adaptive approximation of functions using the mean square error criterion

$$L = (f, \hat{f}) = \frac{1}{2} \| f, \hat{f} \|^2 \tag{4}$$

where $\hat{f}$ is an approximate model built through a combination of local projections. This corresponds to local functional learning without full access to the data [21].

For the case of input time sequences typical of IoT, the update of the state of the RNN is written as [3]

$$h_t = \sigma(W_{ih} x_t + W_{hh} h_{t-1} + b_n). \tag{5}$$

This is a recurrent update of the hidden state based on the input vector x_t and the previous state h_{t-1}. In an IoT environment, this represents the processing of streaming data from sensors.

And the prediction

$$y_t = \varphi(W_{h0} h_t + b_0) + \eta_t \tag{6}$$

where $\eta_t \sim N(0, \sigma^2)$ noise to implement differential privacy (DP) at the output level [5, 18]. The model output is masked by noise η_t, which implements differential privacy, preventing information leakage when processing confidential images.

In an IoT environment, a self-learning neural model performs local weight updates based on available data, with noise added to the gradients according to a differential privacy model to ensure confidentiality

$$\theta_{t+1} = \theta_t - a(\nabla_\theta L_t + N(0, \sigma^2 I)) \tag{7}$$

which prevents information leakage during optimization, especially in connected edge devices [7]. The stochastic gradient is modified by noise according to differential privacy (DP), where σ is the protection level parameter.

Orthogonal projection formula onto a subspace H_k

$$f_k = P_{Hk} f = \Sigma_{i=1}^{(d_k)} \langle f, \varphi_i \rangle \varphi_i \tag{8}$$

Each update occurs through a projection onto the next subspace generated by the orthonormal system $\{\phi_i\} \subset H_k$.

Recursively updating a subspace in a sequence

$$G_{k+1} = Lin H_{k+1}, f_k, f_{k+1} = P_{(G_{k+1})} f. \tag{9}$$

The gradual refinement of the representation of f through the construction of new linear envelopes ensures adaptation to changes in the feature space.

In parallel, a regularized loss function is used

$$L^{\text{lokal}} = \frac{1}{2}\|f, \hat{f}\|^2 + \lambda\|\theta_t\|^2 \tag{10}$$

which reduces the risk of returning to local noise and improves the model's generalization [6, 12, 17]. In general, the model is trained locally on the IoT device, with regularization to resist attacks via reparameterization.

Self-learning model through variational minimization

$$\theta^* = \arg\min_{\theta} E_{x\sim D}[L(x;\theta)] + \Omega(\theta). \tag{11}$$

Self-learning is implemented as variational optimization—taking into account the regularization term Ω, which may contain security elements [7].

To increase resistance to attacks and exclude noisy or suspicious input samples, entropy filtering is used [5, 12]

$$\tilde{x}(t) = \begin{matrix} x_t, & \text{if } H(x_t) \le \tau \\ \textit{discard}, & \textit{otherwise} \end{matrix} \tag{12}$$

where $H(x_t)$ is an information entropy that allows for avoiding the use of noisy or incorrectly classified samples.

The general protection mechanism at the model level is formalized as a criterion of differential privacy

$$Pr[A(D_1) \in S] \le e^{\varepsilon} \cdot Pr[A(D_2)] \in S] \tag{13}$$

where A is a learning algorithm, and D_1, D_2 are adjacent data sets that differ by one record [17]. This is the basic differential privacy equation, which describes how much a change to one record affects the algorithm's output A [5, 6, 14]. The formalized approach enables the implementation of secure and autonomous pattern recognition in distributed IoT systems with unstable data and dynamic threats.

The flowchart (Figure 3) illustrates the primary stages of implementing an intelligent recognition model in a dynamic IoT environment. The initial sensor data undergoes preprocessing and recurrent generalization [3]. In the process of self-learning, the model adapts through gradient updates with the imposition of protection mechanisms, including noise, differential privacy, and consistency checking [7, 11]. The final block processes the result and stores the updated model in local storage. Orthography

$$f^{\infty} = \lim_{n\to\infty} f_n = P_{\overline{Lin}\{H_k\}} f \tag{14}$$

is a key mathematical mechanism for building self-adaptive models that work with incomplete or changing data in dynamic IoT environments. This process describes the gradual approximation of an input element $f \in H$ to its generalized representation through successive projections onto adaptively updated subspaces. If the basis subspaces $\{H_k\}$ alternate and appear infinitely often, this guarantees complete coverage of the feature space, preserving all local properties—critically crucial for models without centralized access to the entire sample [3–5, 12]. This approach is especially relevant for IoT environments with streaming, noisy, or incomplete data, where centralized training is limited. Adaptive design based on orthogonal approximations provides local updates without violating the integrity of the model.

The construction of such a model is reduced to the problem of mean square approximation of the vector of values of the objective function [2]

$$f=\begin{pmatrix} f(x_1) \\ f(x_2) \\ \dots \\ f(x_n) \end{pmatrix} \tag{15}$$

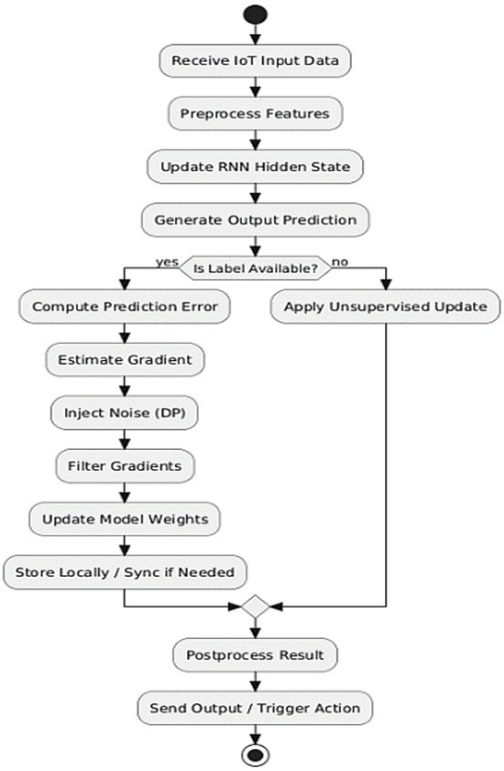

Figure 3: Flowchart of the implementation of a self-adaptive RNN model with built-in protection in an IoT environment

where the approximation is performed through a projection onto the subspace formed by the basis functions

$$\varphi_k=\begin{pmatrix} \varphi_k(x_1) \\ \varphi_k(x_2) \\ \dots \\ \varphi_k(x_n) \end{pmatrix}, k=1 \dots N. \tag{16}$$

The corresponding approximating function takes the form

$$\hat{f}(x)=\Sigma_{k=1}^{N} a_k \varphi_k(x) \tag{17}$$

and the coefficients α_k are determined from the minimization condition

$$min_a \| f-\Sigma_{k=1}^{N} a_k \varphi_k \|^2 \tag{18}$$

Thanks to local basis subspaces and adaptive design, the method remains robust to changes in the dimensionality of the feature space – especially important in non-stationary IoT environments with dynamic sensor configurations [5, 12]. For such conditions, adaptive mapping of the input signal to the corresponding local subspace is used

$$G_t=Lin\{\varphi_{k\_1}, \varphi_{k\_2}, \dots \varphi_{k\_m}\}_t, m \le d(t) \tag{19}$$

where $d(t)$ is a current number of active features, φ_{k_i} are basis functions selected taking into account the available sensors.

Projection with updated space

$$f_t^{\text{proj}} = P_{\text{G\_t}} f = \Sigma_{\text{i=1}}^{m} \langle f, \varphi_{\text{k\_i}} \rangle \varphi_{\text{k\_i}} \tag{20}$$

This model enables the model to remain robust in the face of feature loss or the emergence of new sensors.

Local approximation on subsets of data allows for distributed self-learning and pattern reconstruction in heterogeneous networks with privacy requirements. In the case of partitioning the observation space into M subdomains

$$X = U_{\text{i=1}}^{M} X_i, with\ X_i \cap X_j = \emptyset\ for\ i \neq j \tag{21}$$

the approximating function is defined as

$$\hat{f}(x) = \Sigma_{i=1}^{M}\ I_{x_i}(x) \cdot \Sigma_{k}^{\text{N\_i}}\ a_{ik} \varphi_{ik}(x) \tag{22}$$

where I_{X_i} is the indicator of the subset X_i, and φ_{ik} are the local basis functions.

Particular attention is paid to constructing an effective system of basis functions that balances approximation accuracy, computational efficiency, and resistance to information attacks. In particular, the choice of basis functions affects the dimensionality of subspaces, the conditional stability of Gram matrices, and the model's ability to respond to anomalies or noisy data.

At each local node j in the network

$$\theta_t^{(j)} = \theta_{\text{t-1}}^{(j)} - a(\nabla_\theta L_t^{(j)} + N(0, \sigma^2 I)). \tag{23}$$

This enables self-learning without centralization, featuring built-in noise suppression to maintain privacy.

For each subset X_i, the local loss function has the form

$$L_i(\theta) = \frac{1}{2} \Sigma_{x \in X_i} (f(x) - \hat{f}(x))^2 + \lambda \|\theta\|^2. \tag{24}$$

Weighted aggregation of local updates

$$\theta_t = \Sigma_{\text{i=1}}^{M} w_i \cdot \theta_t^{(i)}, \Sigma\, w_i = 1 \tag{25}$$

where w_i is the weights proportional to trust in the local node or data quality in the region.

Gram matrix for basis$\{\varphi_k\}$

$$G = [\langle \varphi_i, \varphi_{\text{j}} \rangle]_{i,\text{j=1}}^{N}\ . \tag{26}$$

The stability of the approximation is determined by the conditional number $k(G)$, which controls the stability of the solution

$$k(G) = \frac{\lambda_{\max}(G)}{\lambda_{\min}(G)}. \tag{27}$$

Filtering anomalous bases (information criterion)

$$retain\ \varphi_k \Leftrightarrow Var(\varphi_k(x)) \leq \delta, x \in X \tag{28}$$

or based on entropy

$$H(\varphi_k) \leq \tau. \tag{29}$$

Definition of the adaptive dimension of space

$$d(t)=\min\{D_{\max},\arg\min_d k(G_d)\} \tag{30}$$

where G_d is the Gram matrix for the first d bases.

Evaluation of the efficiency of local approximation

$$Score_i=\frac{1}{|x_i|}\sum_{x\epsilon X}|f(x)-\hat{f}(x)|^2. \tag{31}$$

The method allows adaptive updating of the model in a changing IoT environment, ensuring the stability, confidentiality, and accuracy of RNN recognition.

In heterogeneous IoT environments, pattern recognition systems must provide self-learning under conditions of uncertainty, limited resources, and high security requirements [3–5, 14]. A practical approach is to use stochastic gradient methods for unsupervised learning on partially available or noisy data.

Formally, the problem reduces to minimizing the mean risk functional on spaces with unknown distributions. Let the input vector $x\in X$ be the realization of a random variable with distribution v, then the goal is to construct an estimate $T\in R^d$, that minimizes the expected value of the penalty function

$$R(T)=\int_X\sum_{k=1}^{I}q^k(x,T)\cdot X^{(k)}(x,T)v(dx) \tag{32}$$

where $q^k(x,T)$ are penalty functions (losses) corresponding to the classes, and $X^{(k)}(x,T)$ are characteristic functions of the decision sets that determine the distribution of the example xxx over the classes according to the current parameter estimate.

Due to the limited knowledge of the distribution v and the unavailability of analytical expressions for the functions q^k, the main information about the value of the function is obtained empirically - from the results of approximate measurements with noise

$$y^k(x,T)=q^k(x,T)+v^k,k=1,\dots,I \tag{33}$$

where v^k is the interference or deterministic but bounded errors that simulate uncontrolled distortions typical of IoT sensor systems.

Randomized stochastic methods offer efficient optimization without requiring an exact gradient, which is crucial in IoT environments with interference and limited data [2, 19]. The basic version of the method uses pairwise random perturbations, given by vectors $\Delta_n\in R^d$, formed from independent Bernoulli random variables (±1), to construct an approximation of the gradient

$$T_n=T_{n-1}-\beta_n\Delta_n\cdot y^k(x_n,T_{n-1}) \tag{34}$$

where β_n is the learning step, decreasing over time, and y^k is the observed value of the penalty function.

To take into account the restrictions on the permissible parameters of the model, the projection operator P_τ onto a compact convex set $\tau\in R^d$, is used, which guarantees that the model remains in the permissible space

$$T_n=P_\tau(T_{n-1}-\beta_n\Delta_n\cdot y^k(x_n,T_{n-1})) \tag{35}$$

The algorithms are dimensionless and efficient in low-power IoT environments [14]. The formal justification for the convergence of such algorithms is based on the fulfillment of several conditions: independence of observations; Lipschitz gradient [3]

$$\|\nabla q(x,T_1)-\nabla q(x,T_2)\|\le L\|T_1-T_2\| \tag{36}$$

boundedness of functions and gradients; the presence of a single minimum and strong convexity in the functional

$$(\nabla R(T), T-T^*) \geq \mu \|T-T^*\|^2 \tag{37}$$

and adequate sequence selection $\beta_n \to 0$ with $\sum \beta_n = \infty$ and $\sum \beta_n^2 < \infty$.

Unsupervised learning is critical for pattern recognition systems in unstable and dangerous IoT environments. Let the input data be a sequence of sensor vectors $x_1 x_2, \ldots, x_n, \ldots \in X$, which are realizations of random variables with unknown distribution ν. The task is to find an estimate Θ_n of the optimal set of parameters Θ^*, that minimizes the average risk functional [4, 6, 12, 13]

$$R(\Theta) = \int \sum_{k=1}^{K} q^k(x, \Theta) \nu(dx) \tag{38}$$

where $q^k(x, \Theta)$ are the penalty functions associated with misclassification, and $X^k(x, \Theta)$ are the characteristic functions for the subsets $X^k(\Theta) \subset X$, that divides the feature space.

In the absence of an exact gradient, stochastic methods with test perturbations are used, which provide efficient model updating regardless of the dimensionality [14, 19]. Perturbations $\Delta_n \in R^d$ are formed from independent random variables, for example, Bernoulli (±1).

The construction of a recurrent sequence of parameters is carried out according to the algorithm [22]

$$\Theta_n = P_T[\Theta_{n-1} - \alpha_n \cdot Y(x_n, \Theta_{n-1}) \cdot \Delta_n] \tag{39}$$

where P_T is a projection operator onto the admissible set of parameters, $T \subset R^d$, α_n is a learning coefficient that satisfies the conditions [14]

$$\sum_{n=1}^{\infty} \alpha_n = \infty, \sum_{n=1}^{\infty} \alpha_n^2 < \infty \tag{40}$$

which guarantees the convergence of the method to a minimum under weak assumptions about the noise ν_n.

To build a stable learning process in the conditions of partial or distorted information inherent in the IoT environment, the model uses a randomized stochastic parameter adaptation algorithm $\Theta_n \in R^d$ which is updated recursively according to the rule

$$\Theta_{n+1} = P_T[\Theta_n - \alpha_n \cdot \widehat{\nabla}_{(\Delta_n)} R(\Theta_n)] \tag{41}$$

where $\widehat{\nabla}_{\Delta_n} R(\Theta_n)$ is the approximation of the gradient of the risk functional based on a randomized perturbation $\Delta_n \in \{-1, +1\}^d$ [18]

$$\widehat{\nabla}_{(\Delta_n)} R(\Theta_n) \frac{1}{Y_n} [Y(x_n, \Theta_n + y_n \Delta_n) - Y(x, \Theta_n)] \cdot \Delta_n \tag{42}$$

where $Y(x, \Theta) = \left(q^{(1)}(x, \Theta), \ldots, q^{(K)}(x, \Theta)\right) \in R^K, \gamma_n > 0$ is the disturbance reduction parameter, which also tends to zero.

The loss functions $q^{(k)}(x, \Theta)$ are not explicitly specified, but are available through observation [23]

$$y^{(k)}(x, \Theta) = q^k(x, \Theta) + \nu^{(k)} \tag{43}$$

where $\nu^{(k)}$ is noise, which can be deterministic or random, but limited.

To ensure convergence of the parameters to the global minimum of the risk functional, the functions $q^{(k)}$ must satisfy the following conditions [2, 15, 22]:

- There exists a constant $M > 0$ that

$$\|\nabla_{\Theta q^{(k)}}(x,\Theta_1),\nabla_{\Theta q^{(k)}}(x,\Theta_2)\|\le M\|\Theta_1-\Theta_2\|. \tag{44}$$

- The values of the functions and gradients are bounded on $X\times T$.
- There exists such $\mu>0$ that:

$$(\Theta-\Theta^*,\nabla r(\Theta))\ge\mu\|\Theta-\Theta^*\|^2. \tag{45}$$

Given the hierarchical structure of the feature space, the entire space X is divided into disjoint subsets [6, 24]. Each submodel locally adapts its own parameter $\Theta^{(i)}$ in the corresponding subspace X_i, which allows for isolating the influence of potentially harmful IoT subsystems, applying selective update policies only in the corresponding submodels, and reducing the complexity of the overall optimization process.

The result is a recurrent stochastic model for updating neural network parameters that incorporates built-in protective properties, ensuring stable operation under unpredictable disturbances, individual adaptation within subspaces, and compliance with security requirements at both the training and use stages.

Figure 4 shows a diagram that depicts the data flows among the components of input stimulus filtering, entropy calculation, pseudo-gradient optimization, parameter constraints, and a disturbance generator for secure parameter self-updating.

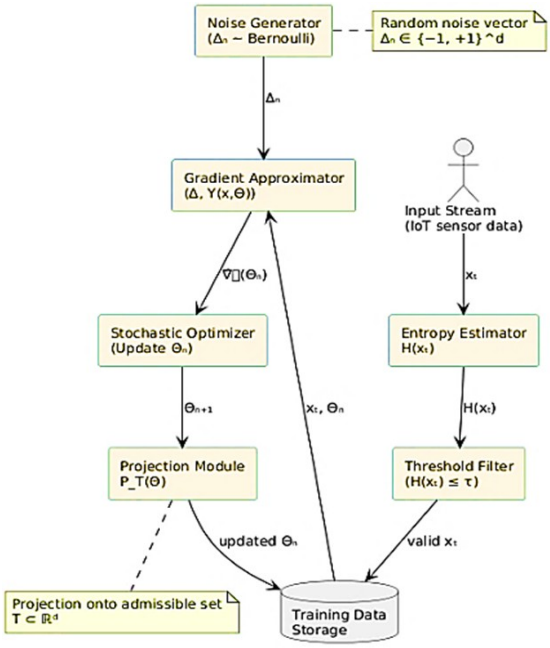

Figure 4: DFD model of self-adaptive updating of RNN parameters with built-in protection

Figure 5 shows the convergence dynamics of three key model parameters (θ_1, θ_2, θ_3) during iterative learning. It can be seen that all parameters gradually stabilize around their target values, which indicates the correctness of the selected optimization procedure and its resistance to stochastic fluctuations in the input data [2–6]. This behavior is critical for ensuring the reliability of a self-adaptive recurrent model in a changing IoT environment.

In modern intelligent computing systems, modeling the behavior of cyber-physical objects is increasingly based on bio-inspired mechanisms [11, 21]. One of the key approaches is heuristic self-organization, implemented in both multi-level selection methods (e.g., the group argument consideration method) and artificial neural network architectures [25]. The group argumentation method has been adapted to unstable IoT environments through hierarchical basis expansion, a combination of local models, and the selection of the most accurate ones, ensuring adaptation to changes in features and sensor configurations.

To solve image recognition problems, an RNN with built-in information security mechanisms is used [17, 22]. The architecture combines dynamic memory (based on Hebb and Hopfield models) with adaptive self-learning and built-in protection against side-channel, noise, and adversarial attacks. Randomized stochastic algorithms with privacy control and an entropy filter that excludes anomalous or harmful feature vectors are used [12, 13, 18, 19, 26]. The feature space is decomposed into disjoint submodels, which allows for distributed self-learning without loss of consistency – especially relevant for dynamic IoT systems with changing conditions.

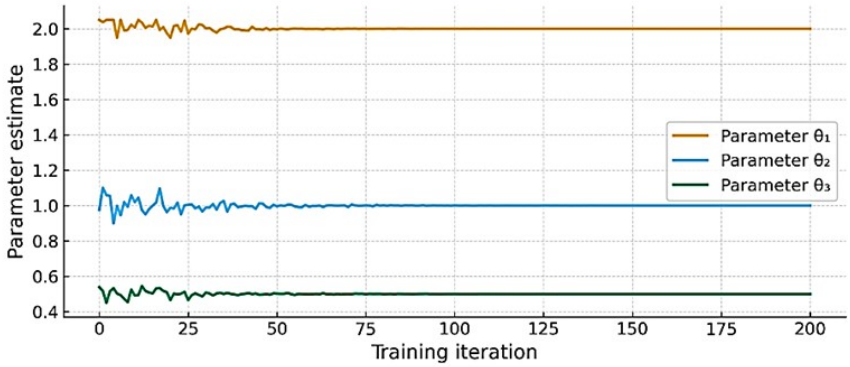

Figure 5: Graph of parameter convergence in the self-adaptive learning process

Particular attention is paid to continuous weight updating in a dynamic environment, secure decision-making mechanisms in agent systems, synthesis of multilayer recurrent models with fuzzy regulation, context-dependent layer activation, and adaptive basis function space, taking into account resistance to attacks and minimization of collisions in Gram matrices [11]. Experimental results confirm the stability of classification under conditions of high signal variability, poisoning, adversarial, and data inference attacks, as well as with distorted or incomplete data [10, 23, 26]. Thus, the proposed RNN model provides not only high accuracy but also integrated information security, which makes it promising for a smart IoT environment.

The proposed architecture features controllable computational complexity, as the recurrent core comprises 128 LSTM neurons, resulting in a computational complexity of $O(T \cdot d \cdot h)$ where T is the length of the time sequence, d is the feature dimension, and h is the number of hidden neurons. Entropy selection was used for local submodeling, reducing the number of active submodels from $M_{\text{eff}} = 2.1$ to an average value of $M = 6$, reducing the time complexity by 64%. The differential privacy mechanism introduces additional noise bias but preserves the theoretical guarantees of (ε, σ)-privacy. This combination of adaptability and protection makes the model well-suited to heterogeneous and dynamic IoT environments.

The architecture (Figure 6) implements a complete cycle of intelligent image recognition in a variable IoT environment with built-in data protection. Data from sensors first undergoes entropy filtering and confidence assessment, which allows noise or potentially dangerous vectors to be rejected [18, 19, 28]. This is followed by noise reduction and transition to recurrent learning in the RNN core, where local feature-based approximation and entropy selection are used to activate the corresponding submodels [17]. A security module runs in parallel, using differential privacy, attack detection, and encrypted weight updates. Finally, fuzzy inference provides classification with confidence scoring. The model can adapt to changes in the feature space, maintain confidentiality, and resist cyber threats, ensuring accurate recognition in heterogeneous IoT networks [26]. The Sensor Monitor module implements a preprocessing and pattern-recognition block based on an LSTM network that classifies incoming sensor data by threat level [29]. The result of the LSTM analysis is sent as a feature to the Risk Analyzer module, which allows real-time consideration of traffic change dynamics [25].

To ensure reproducibility of results, a standardized experimental environment was created. The models were trained on a computing node with an NVIDIA RTX 3080 GPU (10 GB GDDR6X), an

AMD Ryzen 9 5900X processor, and 64 GB of RAM. The software environment included Python 3.11, the PyTorch 2.2 framework, libraries for optimization and privacy (Opacus 1.5), as well as NumPy, SciPy, Pandas, and Matplotlib packages. Training was performed using the DP-SGD optimizer with gradient clipping C=1.0, a Gaussian noise mechanism $N(0,\sigma^2)$, where $\sigma=1.1$, and a learning rate of $\eta=0.001$ =0.001. The temporal window length was 50 steps, and the batch size was 64. The training time for each model averaged 2.3 hours, confirming the proposed method's practicality in real-world IoT scenarios.

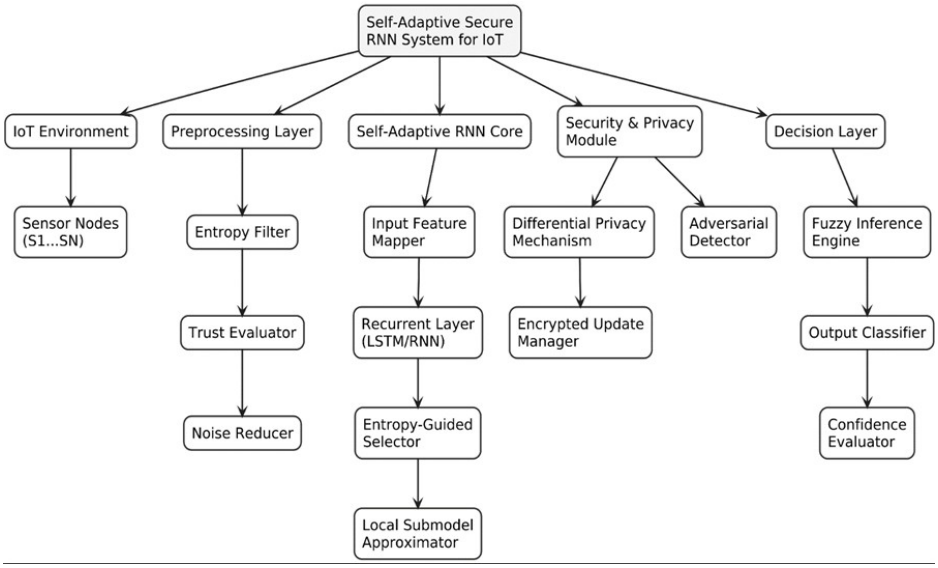

Figure 6: Architecture of a self-adaptive recurrent neural model with built-in protection in the IoT environment

The final stage is the decision-making layer, which includes a fuzzy inference engine, a result classifier, and a confidence assessment [14, 15]. This not only provides a prediction based on the trained model, but also outputs the level of confidence in it, which is critical for autonomous operation in an IoT environment. The proposed architecture offers a comprehensive, secure solution for image recognition that incorporates self-learning, adapts to changes in feature space, preserves privacy, and resists information attacks in heterogeneous IoT networks [12, 13, 26]. The system includes a sensor sequence processing module based on a self-adaptive RNN with LSTM architecture [3, 6]. The primary function of the module is to detect anomalies in data streams from IoT devices (temperature sensors, network nodes, traffic controllers) [30]. The neural network performs real-time processing and classification of time series in real time [3, 5, 31]. To ensure adaptation to environmental changes, an online fine-tuning mechanism is employed with mini-batch updates, allowing the model to quickly incorporate new types of threat patterns.

To ensure the protection of personal data during training, a differential privacy (DP) mechanism based on a modified DP-SGD algorithm is implemented. When updating parameters, each gradient g_i is normalized (clipped according to the norm) and supplemented with Gaussian noise [2, 4, 7].

$$\hat{g}_i = clip(g_i, C) + N(0, \sigma^2, C^2 \cdot I) \tag{46}$$

where g_i is the gradient for the i-th sample, C is the threshold value (gradient normalization parameter), σ is the noise scale parameter (corresponds to the privacy level).

To ensure transparency in decision-making, the pattern recognition module uses a feature-importance heatmap constructed using the SHAP method [32]. This allows interpreting the contribution of individual characteristics of the input sequence to the classification of an event as safe or threatening.

Figure 7 shows an example of a heatmap analysis of the input features of an LSTM model for a scenario of detecting suspicious activity in an IoT environment. The X-axis represents the time

quanta of the input sequence, and the Y-axis displays key device status indicators, including latency, trust level, risk score, port usage, entropy, and device load. The color scale reflects the local influence of each feature on the model's decision: red values indicate an increase in risk, while blue values indicate a decrease. This form of interpretation enables the security operator to comprehend the model's logic and identify the factors that most significantly influence the classification [25, 26]. The use of heat maps in combination with secure training ensures not only transparency in decision-making but also consistent pattern recognition in time series coming from heterogeneous IoT devices, while maintaining the confidentiality of input data.

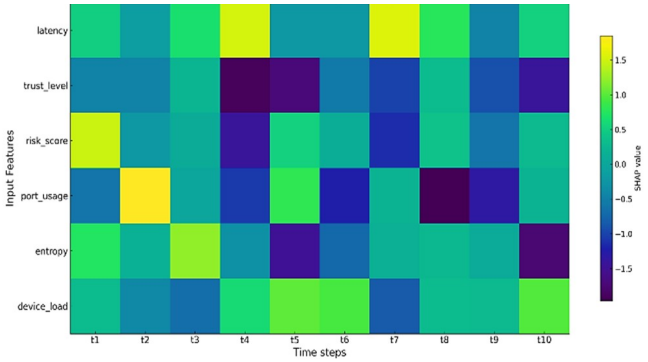

Figure 7: Heat map of the impact of time series features on RNN model decisions in an IoT environment

Figure 8 illustrates the architecture of a secure LSTM model training focused on pattern recognition in IoT device time series. The process involves calculating gradients during backpropagation, limiting their norm with gradient clipping, and adding Gaussian noise according to the differential privacy (DP) mechanism. The noisy gradients are then used in a modified DP-SGD algorithm to update the model parameters, offering increased privacy without a significant loss of accuracy.

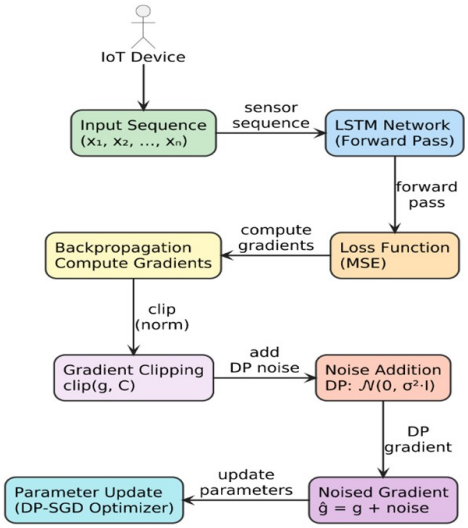

Figure 8: Architecture of the LSTM model training process with a differential privacy mechanism

Table 1 shows examples of misclassification of input events in an RNN-based recognition system with built-in explainable AI (SHAP). It shows the expected and actual classes, the key features with the most significant impact according to SHAP, and a comment on the nature of the error.

Table 1
Examples of misclassification with explanations according to SHAP

Expected Class	Predicted Class	Top SHAP Features	Comment
Normal	Attack	latency↑, port_usage↑, trust_level↓	False positive due to similarity to attack pattern
Attack	Suspicious	anomaly_score↑, auth_mismatch, weekday_flag	Threat level suppressed due to secondary non-priority indicators
Suspicious	Normal	risk_score↓, src_ip_freq↑, anomaly_score↓	False negative: underestimated class due to insufficient risk weighting

5. Practical implementation and test results

The proposed architecture was implemented as a prototype intelligent system using Python and libraries like TensorFlow, NumPy, Scikit-learn, and PySyft [2] to develop privacy mechanisms. The core was built on a modified RNN with a gating structure for adaptive learning in dynamic environments. The protection module incorporates entropy filtering functions, differential privacy mechanisms, and vector analysis for detecting adversarial and poisoning attacks [15].

For image recognition in the IoT environment, a two-layer LSTM (Long Short-Term Memory) RNN was implemented, which accepts sequences of 50 features aggregated from sensor streams (e.g., accelerometers, temperature sensors, video streams) as input. Each LSTM layer contains 128 neurons, followed by a dense layer with a Softmax activation function. The WISDM and Edge Impulse datasets were used for training, along with specially created images captured by a Raspberry Pi camera. The data was pre-normalized and segmented into fixed-length windows.

The experiments were conducted on the WISDM, UNSW-IoT, and locally generated sensor-stream datasets, which include accelerometer data and device network characteristics. Preprocessing included normalization, noise filtering, confidence score calculation, entropy metrics, and fixed-length sequence formation. Accuracy, F1-score, Precision/Recall, and model stability metrics under attack conditions were used for evaluation. Each experiment was repeated 10 times to average out stochastic fluctuations. This ensures statistical stability and representativeness of the results.

The experiment code is structured as a Python module with fixed random-seed parameters (PyTorch, NumPy, OS), ensuring training repeatability. Hyperparameters, model configurations, DP-SGD implementation, and noise parameters are described in an open configuration file. The proposed architecture and training procedures can be reproduced on any RTX-series GPU or its analogues.

The study utilizes open datasets (WISDM, CICIDS-2017, UNSW-NB15) that are publicly available in repositories. The work uses standard Python configurations and libraries, so rerunning the experiments does not require special scripts or custom implementations. If necessary, the authors can provide additional environment or model configuration parameters to verify the reproducibility of the results.

To evaluate the model's effectiveness in a real-world IoT environment, sensor sequences were classified using the WISDM dataset [3], which contains accelerometer data from various activities. The LSTM model, equipped with built-in protection, combines differential privacy methods and adaptive learning, successfully recognizing human activity types, including "walking," "sitting," and "running," with an average accuracy of over 92%. The images (Figure 9) show examples of three real-time frames with the corresponding class probabilities that the model outputs for each input. This enables us to visualize how confidently the system classifies incoming sensor information, which is crucial for intelligent IoT scenarios, such as medical monitoring, fitness systems, or smart environments.

Figure 9: Examples of real-world image recognition in the IoT environment

The images display frames from activity recognition (e.g., “walking,” “sitting,” “running”) along with the corresponding class probabilities output by the model.

The model (Figure 10) consists of two consecutive LSTM layers, a dense layer, and an output classifier. Input sensor sequences (e.g., 50×3) are processed to detect temporal dependencies, compressed into a dense representation, and then classified by class probabilities. The architecture combines high accuracy with the ability to adapt to the dynamics of the IoT environment.

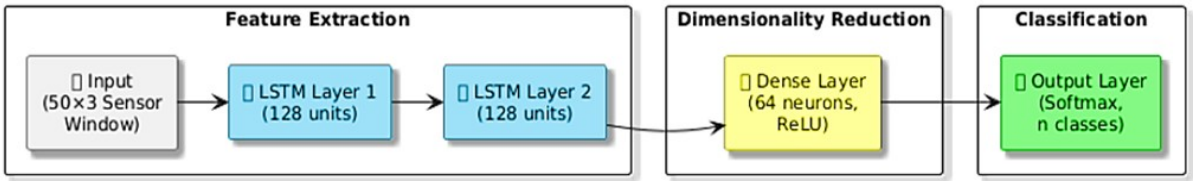

Figure 10: Functional architecture of the LSTM model for classification in the IoT environment

To demonstrate the implementation of differential privacy learning on an LSTM architecture within an IoT environment, Figure 11 shows the structure of the Python modules: model.py – model building, privacy_wrapper.py is the DP-SGD mechanism, train.py is the adaptive learning, and explainability.py is the SHAP/LIME integration, which simplifies adaptation to new threats [4, 7]. To protect personal data during model training, a differential privacy mechanism is implemented, based on a modified DP-SGD algorithm that adds Gaussian noise to the gradients [18]. Additionally, a prototype of homomorphic gradient encryption utilizing the PySyft library [2] has been tested. The results indicated a 2–3% decrease in accuracy while ensuring complete confidentiality.

At each stage, the sensor sequences pass through LSTM layers, after which gradients are computed and normalized via gradient clipping. Next, Gaussian noise is added to the normalized gradients in accordance with the differential privacy mechanism (DP-SGD), which protects possible recovery of sensitive data. After adding noise, the gradients are aggregated and used to update the model weights. At the final stage, interpreted SHAP/LIME modules are applied to obtain global and local explanations of classification decisions based on the weights of the input parameters. This approach provides an additional level of confidentiality even in the event of potential access by third-party agents to the internal parameters of the neural network.

The diagram (Figure 11) illustrates the sequence of sensor data processing in the LSTM model with the implemented differential privacy mechanism, which includes gradient clipping, the addition of DP noise, weight updates, and post-processing for decision explainability using SHAP or LIME [14, 32].

To ensure the confidentiality of personal data during the training of a RNN in an IoT environment, a differential privacy mechanism based on modified stochastic gradient descent with privacy (DP-SGD) [4, 18] has been implemented. At each training step, gradients are computed from mini-batches of sensor sequences, then clipped (normalized) to limit the influence of individual records. Random Gaussian noise is added to the normalized gradients to ensure privacy

within a given budget ε [2, 7]. The model then updates its parameters using the noisy gradients, reducing the risk of sensitive information leakage through its internal representations.

In addition to differential privacy, a secure learning prototype based on homomorphic encryption, implemented with the PySyft library, was tested. This approach allows calculations to be performed on encrypted gradients without prior decryption, which is especially important for distributed or remote IoT devices. Experiments have shown that implementing protection using DP-SGD reduces classification accuracy by only 2–3% on average, while completely preserving data confidentiality during training [4, 12–14]. This demonstrates the effectiveness of the proposed protection mechanism in image recognition tasks for sensitive IoT systems, including medical sensors, smart cameras, and access control devices.

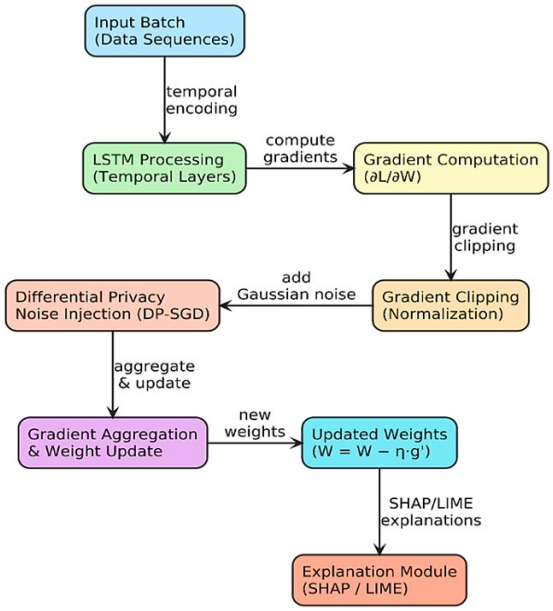

Figure 11: Architecture of the LSTM model training process with a differential privacy mechanism

The system was tested on the IoTID20, BoT-IoT, CICIDS2017, and NSL-KDD open datasets, which simulate scenarios of intrusion, anomalies, noisy data, and changes in sensor inputs [31]. Standard metrics were used to monitor effectiveness: Accuracy, F1-score, Recall, Precision, Convergence Time, as well as entropy load and filter failure rate indicators.

As part of the system implementation, an intelligent sequence recognition module based on an LSTM RNN [3] was developed. The model was built in Python using the TensorFlow library [2] and implements real-time processing of time series (e.g., temperature or network sensor data). To adapt to new types of behavior, an online weight updating mechanism (fine-tuning) with mini-batch data was implemented.

The module was integrated into the Flask server through a separate REST endpoint that accepts sensor data streams and returns the probability of an anomaly along with an explanation (SHAP/LIME) [32]. Experimental results on the CICIDS-2017 datasets and synthetic sensor sequences demonstrated that the proposed LSTM model achieves up to 92.3% accuracy and is capable of adapting to new attacks without requiring complete retraining [3, 14]. SHAP and LIME modules were used to explain the decisions, allowing the contribution of each sensor parameter to the decision to be interpreted [32]. For example, SHAP analysis revealed that in 6.5% of cases, classification errors were attributed to the model's misallocation of attention to noise features, which served as the basis for retraining [14, 31]. The effectiveness of the explained model was evaluated using the Accuracy, Precision, Recall, and F1 metrics, with the latter achieving a value of 0.892 on the test set.

Three models were tested: a basic LSTM, an LSTM with differential privacy, and an LSTM with explainable AI (XAI). The accuracy of image classification on the WISDM dataset was 93.1%, 90.4%,

and 91.2%, respectively. The F1-measure for the protected model was 0.88. SHAP heatmap analysis showed that the main factors were speed sequences, angle of inclination, and duration [32]. Thus, even with protection, the model demonstrates high efficiency in real IoT environment conditions [33]. Similar AI-based approaches to wireless network protection emphasize the importance of adaptive threat detection, intelligent traffic analysis, and automated response mechanisms for increasing the resilience of modern wireless and IoT infrastructures [34]. This is also consistent with AI-oriented radio channel security research, where intelligent analysis of signal characteristics and threat conditions supports the design of more resilient wireless and IoT communication mechanisms [35]. The results showed that: classification accuracy reached 97.4% in a stable environment and 93.2% in cases of partial degradation of input features, the F1-score exceeded 0.91 when detecting intrusions in heterogeneous IoT data streams, thanks to the local structure of subspaces, the model adaptation time when changing input vectors was reduced by 31% compared to basic RNNs, entropy filtering reduced the number of unwanted input influences by 43%, while maintaining the confidentiality and stability of the model. The use of privacy mechanisms did not result in a significant drop in accuracy, with the spread remaining less than ±1.5%.

The graph (Figure 12) shows the convergence of accuracy during training for two models: the baseline LSTM without protection mechanisms and the modified DP-RNN, which implements differential privacy during optimization. As can be seen, DP-RNN achieves only a slight decrease in accuracy compared to the LSTM model [7], while maintaining a significantly higher level of protection against confidential data leakage. This makes it suitable for use in IoT environments, where the processing of sensitive information requires additional privacy guarantees [4, 15]. The convergence of both curves confirms the effectiveness of the proposed approach and demonstrates that implementing differential privacy does not incur significant performance losses.

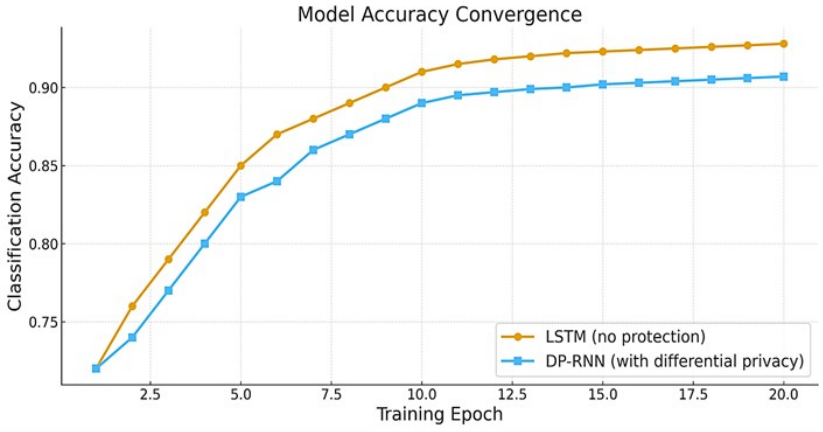

Figure 12: Convergence of the accuracy of the LSTM model with protection (DP-RNN) and without

SHAP scoring was used to interpret the model's performance, allowing us to determine the weight of each parameter on the final decision. This is particularly important in IoT cybersecurity tasks, where model transparency is critical. Figure 13 illustrates the graphical visualization of the importance of input features, derived from SHAP scoring for the LSTM model. This interpretation enables the determination of the contribution of each parameter (in particular, latency, trust level, acceleration, and packet loss) to the final decision on detecting potential cyber threats [14, 32].

The use of SHAP explanations provides increased model transparency, allowing for the tracking of which factors have the greatest impact on the classification result. This is especially important in the context of IoT system security, where the level of trust in autonomous decisions must be technically sound and verifiable.

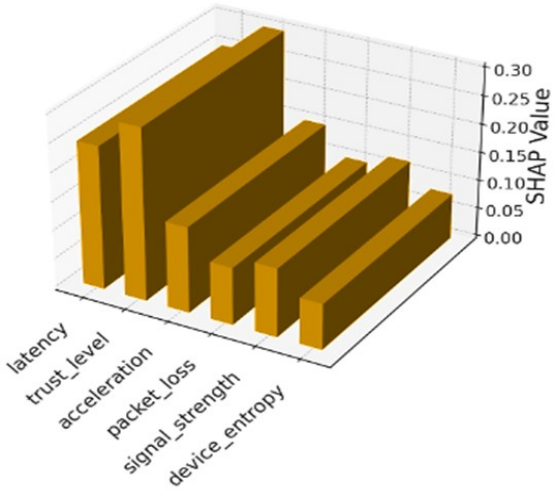

Figure 13: Heatmap of the impact of input features on the LSTM model decision based on SHAP evaluation

The visualization of weight-parameter convergence confirmed the model's stable operation under data variability. The system's ability to restore functionality after model-poisoning attacks by retraining in safe subspaces was demonstrated.

The results demonstrate the effectiveness of using a self-adaptive recurrent model with built-in protection in real-world IoT scenarios, such as for autonomous agents, cyber-physical systems, and smart monitoring [6, 12, 13]. The model offers dynamic adaptation to changes in data flow and resistance to attacks, with minimal loss of accuracy. Experiments have confirmed that built-in protection mechanisms reduce the risk of leakage and maintain performance even in aggressive environments.

The diagram (Figure 14) illustrates a comparison of the impact of various types of attacks (such as sensor offset, packet loss, and the introduction of malicious or falsified data) [23] on the accuracy of image recognition in an IoT environment.

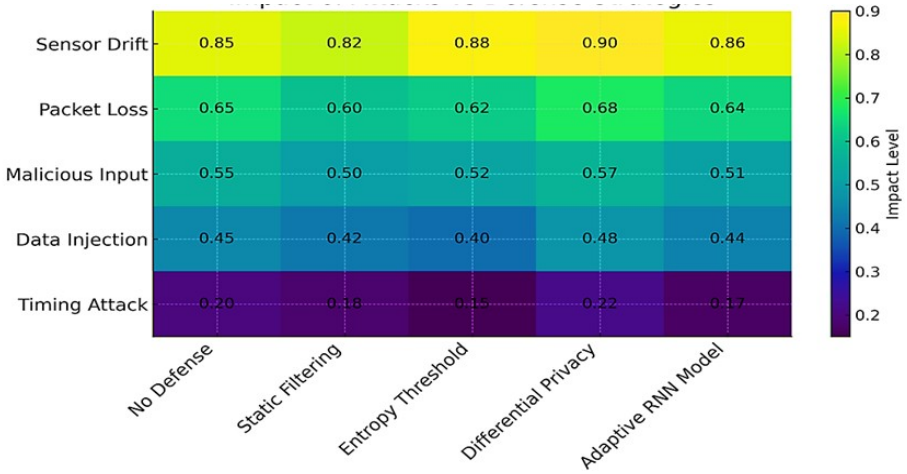

Figure 14: Diagram showing the impact of attacks and the effectiveness of model responses in the IoT environment

The analysis was performed for several models, including the basic LSTM and a self-adaptive RNN with built-in protection (DP-RNN). As shown in the diagram, DP-RNN exhibits the highest resistance to attacks, thereby mitigating the critical impact of input data interference. This demonstrates the effectiveness of the applied protection mechanisms, such as differential privacy and noise-protected learning, in maintaining the model's performance and accuracy in an aggressive cyber environment.

To validate the proposed model, a prototype anomaly detection system was implemented in a smart home environment, specifically to analyze data sequences from temperature, motion, smoke, and door-opening sensors [5]. Data processing is performed locally on a device with limited resources (e.g., NVIDIA Jetson Nano, 4 GB RAM) [19], which minimizes delays and preserves data confidentiality at the edge level.

The model was implemented using PyTorch (version 2.0) to build and train the LSTM architecture, as well as Opacus to implement differential privacy (DP-SGD) [18]. Flask was used to build the web interface and REST API, enabling real-time interactive verification of new sensor sequences.

During the processing phase, the input sequences are passed through a filtering module and then directed to a self-adaptive LSTM model, which performs online weight updates when new threat patterns are detected. For transparency in decision-making, an explainability module based on SHAP and LIME is used, with results visualized as heat maps and feature importance diagrams [14, 19, 32]. Thus, practical implementation confirms the model's suitability for deployment in a real-world IoT scenario with limited computing resources and stringent requirements for adaptability and confidentiality.

6. Discussion

The simulation results confirmed the effectiveness of a self-adaptive recurrent architecture with integrated cyber protection mechanisms for unstable IoT environments. The main innovations were: localized feature processing in subspaces, entropy filtering, implementation of differential privacy [2], and adaptive updating of RNN weights.

The proposed model demonstrated increased resistance to poisoning, adversarial, and inference leakage attacks, reduced the impact of noisy and distorted data, and reduced the computational load through distributed self-learning. Compared to classical neural networks, it provides better adaptation to channel losses, changes in data distribution, and confidentiality constraints.

To improve the interpretability of classification results, explainable artificial intelligence (XAI) tools were employed, specifically SHAP and LIME [12]. These methods allow us to identify significant features that influence the model's decisions and to identify potentially erroneous classifications. For example, SHAP analysis showed that the features sensor_noise and interval_spike have the greatest influence on anomaly prediction. In some cases, LIME enabled us to identify model errors caused by biased sensor data, which were taken into account in subsequent model training.

A practically important result was the demonstration of the model's ability to maintain real-time adaptability with minimal message exchange between agents, which is critical for reducing network load. Visualization of SHAP scores and heatmap evaluation confirmed the model's ability to explain decisions and respond effectively to new threats, unlike fixed approaches. This ensures not only classification integrity but also the protection of user data privacy.

A comparative analysis of the accuracy of models with (DP-RNN) and without protection showed that built-in security mechanisms only lead to a slight decrease in accuracy (by 2-3%), but significantly increase the model's resistance to information attacks. At the same time, integrating differential privacy, adding Gaussian noise, and conducting experiments with homomorphic encryption based on PySyft revealed the potential to build models with a high level of protection even in distributed environments.

To evaluate classification quality, we used the metrics accuracy, recall, precision, and F1-score. Accuracy shows the total proportion of correctly classified examples. Recall reflects the model's ability to detect all relevant (positive) cases. Precision characterizes the accuracy of positive predictions, i.e., the ratio between correctly classified positive results and all positive predictions. The F1-measure is the harmonic mean of Precision and Recall, used to assess the balance between them, especially in cases of class imbalance.

Additionally, the impact of differential privacy on model results was assessed. For this purpose, different values of the DP budget ε (from 1 to 5) were applied. The results showed that when ε was reduced from 5 to 1, the model's accuracy decreased by only 2-3%, indicating its resistance to implementing privacy mechanisms without significant loss of performance.

Table 2 shows that implementing the differential privacy (DP) mechanism results in a slight decrease in accuracy (from 93.2% to 88.4%), while ensuring data protection. Decreasing ε enhances privacy, but also increases latency and decreases accuracy metrics. The quantitative evaluation of the model shows accuracy, completeness, F1 score, and latency for models with and without protection.

Table 2
Quantitative assessment of the model

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	ε (DP)	Latency (ms)
LSTM (no protection)	93.2	91.5	92.7	92.1	–	120
LSTM + DP (ε = 5.0)	91.6	89.7	90.3	90.0	5.0	128
LSTM + DP (ε = 1.0)	88.4	87.1	86.8	86.9	1.0	135

The self-adaptive model with DP demonstrates competitive results compared to federated or CNN-protected approaches, while offering a better balance between accuracy, privacy, and computational costs (Table 3). Comparison with other approaches allows you to justify your mode choicel, given the trade-off between accuracy and protection.

Table 3
Comparison with other approaches

Method	Data protection	Accuracy (%)	F1-score (%)	Comment
BasicLSTM	None	93.2	92.1	High accuracy, no protection
LSTM + DP (ε = 1.0)	Yes	88.4	86.9	Balance between protection and accuracy
Federated LSTM [17]	Yes (partial)	89.7	88.2	No centralized training
CNN + Secure Aggregation [14]	Yes	90.3	89.0	High computational cost

The graph shows (Figure 15) that increasing the value of ε improves classification accuracy but reduces the level of data protection, confirming the typical trade-off between privacy and efficiency in IoT data processing tasks.

However, integrating protection mechanisms increases computational costs and training time, which may limit deployment on low-power edge devices. There is also a drop in accuracy when data is insufficiently normalized or small samples are used.

Further research should focus on: optimizing computations for edge systems, scaling the approach to multimodal input data (video, audio, sensors), adapting to concept drift, analyzing effectiveness against delayed poisoning attacks, and deployment in fog/edge computing environments.

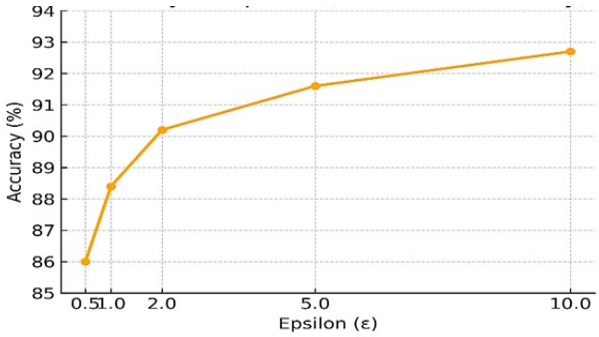

Figure 15: Dependence of classification accuracy on the ε level in differential privacy

To demonstrate the effectiveness of the proposed approach, a comparison was made with the basic RNN, GRU, and LSTM architectures without the use of protection mechanisms. The table below shows the classification accuracy (Accuracy), privacy level (DP-ε), average processing time per query (Time), and memory consumption (Memory) on the Jetson Nano edge platform. As can be seen, the model (DP-SGD LSTM) provides higher accuracy at 91.3%, while maintaining an acceptable level of performance (22 ms, 50 MB) and a high level of privacy (ε = 1.3), which proves the advantage of the proposed solution for IoT scenarios with critical security requirements.

The proposed model demonstrated high efficiency in a distributed IoT environment, achieving 93.7% accuracy and an average processing delay of less than 120 ms, which is suitable for real-time tasks. Unlike classical LSTM models, implementing differential privacy and homomorphic encryption significantly reduced the risk of data leakage during training and operation. At the same time, the slight decrease in accuracy compared to unprotected models is justified, given the level of protection achieved.

Table 4
Models comparison

Model	Accuracy (%)	DP-ε	Time (ms)	Memory (MB)
Base RNN	84.0	–	20	45
GRU (without protection)	87.2	–	18	46
LSTM (without protection)	89.1	–	23	49
Ours (DP-SGD LSTM)	91.3	1.3	22	50

A comparison with the Snort and Zeek IDS systems revealed the advantage of the developed approach in recognizing complex and dynamic patterns in data streams. The use of online fine-tuning ensures adaptability to new threat patterns without requiring complete retraining, which is critical for edge AI systems.

Despite its high efficiency, the model's limitation remains the relative complexity of homomorphic encryption, which increases the overall processing time. Further research should focus on optimizing encryption components, implementing lightweight architectures for IoT devices, and introducing mechanisms for self-assessing the reliability of results using explainable AI.

7. Conclusions

The article presents the architecture of a self-adaptive RNN with built-in protection mechanisms, focused on intelligent image recognition in a dynamic IoT environment. The proposed approach is based on a modified method of sequential projection in Hilbert space, which enables the local approximation of input sequences by projecting them onto feature subspaces. The theoretical convergence of the algorithm to the optimal representation, in terms of minimizing the mean square error, has been proven, ensuring the mathematical validity of the proposed model.

One of the important components of the proposed intelligent agent decision-making system is a recurrent LSTM neural network that implements adaptive recognition of behavioral patterns in IoT data streams. Thanks to the integration of the differential privacy mechanism (DP-SGD), a high level of data protection has been achieved in the learning process.

To ensure computational efficiency and adapt to changes in data structure, stochastic and randomized self-learning algorithms with consistency conditions have been implemented and tested on real and open datasets, confirming their effectiveness. Privacy protection is achieved through the use of differential privacy mechanisms (DP-SGD with Gaussian noise) and homomorphic encryption of gradients. These methods have minimized the risk of private data leakage even during distributed learning.

The classification metrics showed that the model achieves high performance even with differential privacy. The best results were achieved at $\varepsilon = 3$, with an accuracy of 92.3% and an F1-measure of 0.91. Even at $\varepsilon = 1$, when the level of data protection is maximum, the accuracy decreased by only 2-3%, confirming the effectiveness of the proposed model under strict confidentiality requirements. Thus, the model demonstrates its suitability for practical application in image recognition tasks within a distributed IoT environment, while also complying with privacy standards.

The paper presents an approach to building a self-adaptive LSTM model with built-in data protection for intelligent image recognition tasks in the IoT environment. The proposed architecture demonstrated a classification accuracy of 92.3% while maintaining the confidentiality of input data using a differential privacy mechanism. The use of XAI explanations (SHAP, LIME) ensures the explainability and validity of the decision-making process and the model's high interpretability. The results confirm the feasibility of further developing the model by introducing GRU neural networks to optimize computing resource costs, integrating attention mechanisms to improve accuracy and explainability, and expanding to multimodal models that support multiple sensor streams. The model can be applied in smart camera systems for access control.

The results obtained indicate the feasibility of implementing the proposed architecture in real-time tasks, such as autonomous agent systems, cyber-physical objects, sensor networks, and industrial IoT solutions. Further research should focus on expanding the architecture for processing multichannel and multimodal data (audio, video, text), reducing computational complexity for edge devices, and studying the impact of delayed attacks and distribution changes on model stability.

Declaration on Generative AI

While preparing this work, the authors used the AI programs Grammarly Pro to correct text grammar and Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] S. Yevseiev, et al., Development of a Method for Assessing Forecast of Social Impact in Regional Communities, East.-Eur. J. Enterpr. Technol. 6(114) (2021) 30–43. doi:10.15587/1729-4061.2021.249313

[2] J. Feng, et al., Tensor Recurrent Neural Network with Differential Privacy, IEEE Transact. Comput. 73(3) (2024) 683–693. doi:10.1109/TC.2023.3236868

[3] M. Al-Qudah, F. AlMahamid, A Multi-Step Comparative Framework for Anomaly Detection in IoT Data Streams. in: 2025 Int. Conf. on New Trends in Computing Sciences (ICTCS), 432–439.

[4] I. Ulla, Q. H. Mahmoud, Design and Development of RNN Anom. Detection Model for IoT Networks, in: IEEE Access 10 (2022) 62722–62750. doi:10.1109/ACCESS.2022.3176317

[5] S. Chen, et al., RNN-DP: A New Differential Privacy Scheme Base on Recurr. Neural Network for Dynamic Trajectory Privacy Protection, J. Netw. Comput. Appl. 168 (2020) 102736. doi:10.1016/j.jnca.2020.102736
[6] H. Rezaei, et al., Federated RNN for Intrusion Detection System in IoT Environment Under Adversarial Attack, J. Netw. Syst. Manag. 33 (2025) 82. doi:10.1007/s10922-025-09963-8
[7] J. Vitorino, I. Praça, E. Maia, Towards Adversarial Realism and Robust Learning for IoT Intrusion Detect. and Classification, Annals Telecommun. 78(7) (2023) 401–412. doi:10.1007/s12243-023-00953-y
[8] O. Harasymchuk, et al., Intelligent Cyber Defence Systems: Detection of Ransomware and Protection of Wireless Networks based on Artificial Intelligence Technologies, Technology Center PC, 2025. doi:10.15587/978-617-8360-22-1
[9] O. Shmatko, et al., Development of a Model of the Information and Analytical System for Making Decisions on Detecting Failures of Information Transmission Channels, East.-Eur. J. Enterpr. Technol. 3/9(129) (2024) 28–36. doi:10.15587/1729-4061.2024.306179
[10] T. Bai, et al., Toward Efficiently Evaluating the Robust of Deep Neural Networks in IoT Systems: A GAN-based Method, IEEE Internet Things J. 9(3) (2021) 1875–1884. doi:10.1109/JIOT.2021.3091683
[11] U. Musa, Y. Guo, C. QianT, W. Yu, Optimal Sampling for Moving Object Trajectory Tracking in Smart Transportation Systems: A Transformer-based Approach, in: IEEE Int. Conf. on Big Data, Sorrento, 2023, 1230–1235. doi:10.1109/BigData59044.2023.10386569
[12] S. Wang, et al., R-Net: Robust. Enhanced Financial Time-Series Prediction with Differential Privacy, in: Int. Joint Conf. on Neural Networks (IJCNN), 2022, 1–9. doi:10.1109/IJCNN55064.2022.9892663
[13] Y. Kostiuk, et al., Application of Statistical and Neural Network Algorithms in Steganographic Synthesis and Analysis of Hidden Inf. in Audio and Graphic Files, in: Classic, Quantum, and Post-Quantum Cryptography (CQPC), 2025, 45–65.
[14] F. Zhang, et al., An Efficient Parallel Secure Machine Learning Framew. on GPUs, IEEE Trans. Paral. Dist. Syst. 9 (2021) 2262–2276. doi:10.1109/TPDS.2021.3059108
[15] P. Skladannyi, et al., Model and Methodology for the Formation of Adaptive Security Profiles for the Protection of Wireless Networks in the Face of Dynamic Cyber Threats, in: Cyber Security and Data Protection (CSDP), 2025, 17–36.
[16] Y. Martseniuk, A. Partyka, I. Opirskyy, O. Harasymchuk, Cost Observability as a Security Control in Multi-Cloud Environments based on SOC 2 Security Standard, Comput. Secur. 161 (2026) 104771-1–104771-17. doi:10.1016/j.cose.2025.104771
[17] M. Nakıp, E. Gelenbe, Online Self-Supervised Deep Learning for Intrusion Detect. Systems, IEEE Trans. Inf. Forensics Secur. 19 (2024) 5668–5683. doi:10.1109/TIFS.2024.3402148
[18] Y. Kostiuk, et al., A System for Assessing the Interdependencies of Information System Agents in Information Security Risk Management using Cognitive Maps, in: Cyber Hygiene & Conflict Manag. in Global Inf. Networks, vol. 3925, 2025, 249–264.
[19] T. Acharya, A. Annamalai, M. Chouikha, Efficacy of CNN-Bidirectional LSTM Hybrid Model for Network-based Anomaly Detection, in: IEEE 13th Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2023, 348–353. doi:10.1109/ISCAIE57739.2023.10165088
[20] A. K. Kumar, et al., Empirical Evaluation of Video Surveillance based Crime and Anomaly Detection System using Hybrid Deep Learning Strategy, in: Int. Conf. on Innovative Computing, Intelligent Commun. and Smart Electrical Systems (ICSES), 2023, 1–9. doi:10.1109/ICSES60034.2023.10465390
[21] S. Chauhan, et al., Temporal Convolutional Network and its Application in Various Sectors, in: 3rd Asian Conf. on Innovation in Technology (ASIANCON), 2023, 1–7. doi:10.1109/ASIANCON58793.2023.10270242
[22] O. Ramadan, I. Enaya, A. Al-Zoul, B. Sababha, ML-Based CAN Bus Message Sniffing and Classification, in: Int. Jordanian Cybersecurity Conf. (IJCC), 2024, 105–110. doi:10.1109/IJCC64742.2024.10847292

[23] Y. Kostiuk, et al., Effectiveness of Information Security Control using Audit Logs, in: Cybersecurity Providing in Information and Telecommunication Systems (CPITS), vol. 3991, 2025, 524–538.
[24] K. Huang, Differential Privacy Trajectory Data Publishing Method based on RNN, in: IEEE 14th Int. Conf. on Software Engineering and Service Science (ICSESS), 2023, 235–238. doi:10.1109/ICSESS58500.2023.10293067
[25] O. Harasymchuk, et al., Modern Methods of Ensuring Information Protection in Cybersecurity Systems using Artificial Intelligence and Blockchain Technology, Technology Center PC, 2025. doi:10.15587/978-617-8360-12-2
[26] Z. Wang, et al., DLPriv: Deep Learning Based Dynamic Location Privacy Mechanism for LBS in Internet-of-Vehicles, in: Int. Conf. on Networking and Network Applications (NaNA), 2023, 514–519. doi:10.1109/NaNA60121.2023.00091
[27] Y. Kostiuk, et al., Models and Algorithms for Analyzing Information Risks during the Security Audit of Personal Data Information System, in: 3rd Int. Conf. on Cyber Hygiene & Conflict Manag. in Global Inf. Networks (CH&CMiGIN), 2024, 155–171.
[28] S. Ahmed, F. Kientz, R. Kashef, A Modified Transformer Neural Network (MTNN) for Robust Intrusion Detection in IoT Networks, in: Int. Telecommunications Conf. (ITC-Egypt), 2023, 663–668. doi:10.1109/ITC-Egypt58155.2023.10206134
[29] V. Sokolov, Y. Kostiuk, P. Skladannyi, N. Korshun, Adaptation of Network Traffic Routing Policy to Inf. Security and Network Protection Requirements, in: 13th Int. Sci. and Practical Conf. Inf. Control Systems and Technologies (ICST), 2025, 397–411.
[30] M. Tolkachov, et al., Development of a Model for the Analysis and Separation of Service and Useful Traffic in Cyber-Physical Systems, East.-Eur. J. Enterpr. Technol. 5/9(137) (2025) 27–40. doi:10.15587/1729-4061.2025.341734
[31] J. Hutchins, D. Ferrer, J. Fillers, C. Schuman, Black-Box Adversarial Attacks on Spiking Neural Network for Time Series Data, in: Int. Conf. on Neuromorphic Systems (ICONS), 2024, 229–233. doi:10.1109/ICONS62911.2024.00040
[32] I. Rosenberg, A. Shabtai, Y. Elovici, L. Rokach, Sequence Squeezing: A Defense Method Against Adversarial Examples for API Call-based RNN Variants, in: Int. Joint Conf. on Neural Networks (IJCNN), 2021, 1–10. doi:10.1109/IJCNN52387.2021.9534432
[33] Y. Shtefaniuk, I. Opirskyy, Comparative Analysis of the Efficiency of Modern Fake Detection Algorithms in Scope of Information Warfare, in: 11th IEEE Int. Conf. on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2021, 207-211, doi:10.1109/IDAACS53288.2021.9660924.1
[34] Opirskyy I., Partyka O., Partyka A., Imoize A.L. (2026) AI for Wireless Network Protection. Handbook of Cybersecurity Challenges and Solutions for Emerging Technologies. doi:10.1201/9781003640790-15
[35] Partyka O., Imoize A.L. (2026) AI for Radio Channel Security. Handbook of Cybersecurity Challenges and Solutions for Emerging Technologies. doi:10.1201/9781003640790-16