KMANPUB | AI and Tech in Behavioral and Social Sciences Vol. 5 No. 1 (2027): 1-11

Article history:
Received 10 April 2026
Revised 15 July 2026
Accepted 24 July 2026
Initial Publication 31 July 2026
Final Publication 01 January 2027

Deep and Ensemble Learning for IoT Network Intrusion Detection: Comparative Performance and Deployment Considerations for AI-Enabled Connected Environments

Sadiq. Sahep Majeed[1], Mahmood. Deypir[1*], Razieh. Farazkish[1], Ali. Broumandnia[1]

[1] Department of Computer Engineering, ST.C., Islamic Azad University, Tehran, Iran

*** Corresponding author email address**: mdeypir@iau.ac.ir

A r t i c l e I n f o

Article type:
Original Research

How to cite this article:
Sahep Majeed, S., Deypir, M., Farazkish, R., & Broumandnia, A. (2027). Deep and Ensemble Learning for IoT Network Intrusion Detection: Comparative Performance and Deployment Considerations for AI-Enabled Connected Environments. *AI and Tech in Behavioral and Social Sciences*, *5*(1), 1-11.
https://doi.org/10.61838/kman.aitech.5855

A B S T R A C T

Internet of Things (IoT) networks connect devices used in smart homes, education, healthcare, transportation, and critical infrastructure, but their heterogeneity and limited device resources create a broad and rapidly changing attack surface. This study compared four standalone deep-learning architectures—deep neural network (DNN), one-dimensional convolutional neural network (CNN), long short-term memory (LSTM), and gated recurrent unit (GRU)—with all six pairwise probability ensembles formed from these models for binary intrusion detection. The analysis used 461,043 observations and 45 features from the TON-IoT network-traffic data. Data were divided into 322,730 training observations and 138,313 test observations using a 70/30 class-preserving record-level split. Preprocessing included missing-value treatment, categorical encoding, removal of non-informative features, and standardization estimated from the training data and applied to the test data. Models were trained under common optimization settings and evaluated using accuracy, F1 score, area under the receiver operating characteristic curve (AUC-ROC), and confusion-matrix errors. Every pairwise ensemble ranked above the four standalone models by F1 score in this single-split analysis. The DNN + GRU ensemble achieved the highest point estimates, with accuracy of 0.9774, F1 score of 0.9680, and AUC-ROC of 0.9963. It produced 87,926 true negatives, 47,260 true positives, 2,074 false positives, and 1,053 false negatives. The DNN + LSTM ensemble ranked second, with accuracy of 0.9769, F1 score of 0.9673, and AUC-ROC of 0.9957. Because the evaluation used one split and single point estimates, the small differences between leading models should be interpreted descriptively rather than as evidence of statistically significant superiority. For deployment, external validation, calibrated thresholds, explainable alerts, drift monitoring, and analyst oversight remain necessary.

Keywords: *Internet of Things; intrusion detection system; deep learning; ensemble learning; DNN; CNN; LSTM; GRU; TON-IoT; trustworthy artificial intelligence*

1. Introduction

The Internet of Things (IoT) has changed networked computing from a set of relatively stable endpoints into a large ecosystem of sensors, actuators, household devices, industrial controllers, vehicles, medical systems, educational technologies, and cloud-connected services. These devices continuously exchange information and support automation, real-time monitoring, and data-driven decisions. The same connectivity also expands the number

of entry points available to attackers. Compromised IoT devices may be used for distributed denial-of-service attacks, unauthorized surveillance, ransomware, data exfiltration, or manipulation of services on which people increasingly depend. Cybersecurity in connected environments is therefore not solely a technical matter: it affects privacy, perceived safety, institutional trust, and willingness to adopt AI-enabled services.

Conventional defenses based on fixed signatures and manually specified rules remain useful for recognized threats, but they are less effective when attacks change rapidly or when normal activity varies across devices and contexts. Intrusion detection systems (IDSs) address this limitation by analyzing traffic and system behavior to identify activity that departs from expected patterns. In IoT settings, IDS design is complicated by heterogeneous protocols, class imbalance, high traffic volume, limited computing resources, and the temporal structure of coordinated attacks. The TON-IoT benchmark was developed from heterogeneous IoT, industrial IoT, operating-system, and network sources to support evaluation of AI-based cybersecurity applications under more realistic conditions (Alsaedi et al., 2020).

Deep learning is attractive for intrusion detection because it can learn nonlinear and hierarchical representations without relying exclusively on hand-crafted rules (Goodfellow et al., 2016). Different architectures, however, model different properties of network data. A fully connected DNN learns nonlinear interactions among tabular features. A one-dimensional CNN can detect local combinations and short-range patterns. LSTM networks retain long-range dependencies through gated memory (Hochreiter & Schmidhuber, 1997), whereas GRU networks provide a more compact recurrent mechanism that often achieves comparable temporal modeling with fewer parameters (Cho et al., 2014). No single representation can be assumed to capture all properties of normal and malicious traffic.

The rationale for combining models extends beyond network security. Reviews of deep learning in complex sequential domains indicate that hybrid architectures can improve stability and generalization when different models capture complementary structure (Karimi Dastgerdi & Zamani Boroujeni, 2020). Sensor-based human-activity recognition likewise illustrates the value of deep feature extraction from multidimensional signals (Mohammadi & Anisheh, 2024). In IoT-enabled educational environments, Bayesian modeling has been used to identify hidden relationships in learner activity (Moradi, 2023), while reinforcement learning has supported adaptive, load-balanced decisions in vehicular IoT systems (Ravaei et al., 2022). These studies address different tasks, but they demonstrate the broader capacity of AI models to analyze heterogeneous, dynamic, and behaviorally meaningful IoT data.

Ensemble learning provides a formal way to exploit complementary representations. When models make partially uncorrelated errors, averaging or weighting their predicted probabilities may reduce variance and improve generalization (Breiman, 1996; Sagi & Rokach, 2018). Stacked generalization similarly combines the outputs of multiple learners through a higher-level decision process (Wolpert, 1992). The same principle has improved prediction in social-media data: Rahmani Seryasat et al. (2021) combined heterogeneous regression models through stacked generalization to predict Facebook comment volume (Rahmani Seryasat et al., 2021). For intrusion detection, an ensemble may combine the broad nonlinear interactions captured by a DNN, local feature patterns learned by a CNN, and ordered dependencies modeled by LSTM or GRU networks.

This study comparatively evaluated four standalone deep-learning models and all six pairwise ensembles under the same dataset and evaluation framework. The primary objective was to describe whether pairwise combination improved intrusion-discrimination metrics within the reported split and to identify the pairing with the highest observed point estimates. A secondary objective was to discuss deployment considerations. Because no repeated runs, confidence intervals, external validation, calibration analysis, or human-factors evaluation were available, the study does not establish statistically significant model superiority or empirically validate trustworthy-AI performance.

2. Related Work

Deep-learning-based intrusion detection has developed along three main directions: representation learning from traffic features, temporal modeling of network behavior, and the integration of multiple models. Early deep architectures demonstrated that nonlinear representation learning could improve network anomaly detection compared with shallow or manually engineered pipelines (Diro & Chilamkurti, 2018; Shone et al., 2018; Vinayakumar et al., 2019). Comparative reviews have

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

likewise identified dataset quality, class imbalance, and cross-environment generalization as recurring challenges for deep-learning IDS research (Ferrag et al., 2020). Online systems such as Kitsune further showed that ensembles of compact autoencoders could detect anomalous traffic with limited computational resources, an important requirement for network gateways and edge devices (Mirsky et al., 2018).

IoT-specific datasets have been central to progress because conventional enterprise-network benchmarks do not fully reflect device heterogeneity or modern attack behavior. TON-IoT combines telemetry, operating-system, and network data collected in a realistic IoT/IIoT testbed and includes attacks such as denial of service, distributed denial of service, scanning, backdoors, injection, and ransomware (Alsaedi et al., 2020). Evaluations using TON-IoT Linux data have emphasized the importance of realistic preprocessing and data-analytics pipelines for intrusion detection (Moustafa et al., 2020).

Recent hybrid models combine convolutional and recurrent components to capture local and temporal information. Afraji et al. (2025) evaluated CNN-LSTM-GRU configurations for IoT and IIoT intrusion detection (Afraji et al., 2025), while Ishtiaq et al. (2025) combined multi-scale convolution, bidirectional recurrence, and attention (Ishtiaq et al., 2025). Other work has addressed minority-class prediction and imbalance through staged deep-learning frameworks (Maoudj & Belghiat, 2025). These studies support the premise that architecture diversity is valuable, but many focus on one proposed hybrid rather than a controlled comparison of several standalone models and all pairwise combinations.

Interpretability and operational trust are increasingly recognized as necessary components of IDS deployment (Ahmad et al., 2025). A detector may achieve high aggregate accuracy while still generating errors that are costly in a specific setting. A false negative in a medical or industrial IoT system can allow an attack to persist, whereas repeated false positives may produce alert fatigue and encourage operators to ignore warnings. Model explanations, calibrated confidence, and audit trails help analysts understand why an alert was produced and whether it is consistent with the operational context (Lundberg & Lee, 2017). The present study therefore treats ensemble performance as a contribution to decision support, not as a justification for fully autonomous blocking or sanctioning.

The research gap addressed here is a controlled descriptive comparison under common conditions. Four base architectures were trained on the same processed TON-IoT data, and every possible two-model ensemble was evaluated using the same reported metrics. This design reduces variation caused by different datasets and preprocessing pipelines, but it does not by itself establish generalization beyond the single split or demonstrate that observed differences are robust to alternative seeds, devices, sessions, or time periods.

3. Methods and Materials

3.1. Study Design

This computational experimental study compared binary classifiers for identifying normal and malicious IoT network traffic. The analysis included four standalone architectures—DNN, CNN, LSTM, and GRU—and the six pairwise ensembles that can be formed from those four models. All models were evaluated on a common held-out test set created through a record-level 70/30 split. The unit of analysis was a network-traffic observation. Because records originating from the same device, session, connection, or attack episode may be correlated, future validation should use group-aware or chronological partitioning to assess possible split-related leakage.

3.2. Dataset and Preprocessing

The final analytical dataset contained 461,043 observations and 45 predictor features. The target was binary, distinguishing normal traffic from attack traffic. Numeric attributes included variables such as source and destination ports, connection duration, and transmitted or received byte counts. Categorical attributes included network protocol, service type, and address-related fields. The data were divided into 322,730 training observations (70%) and 138,313 test observations (30%) while preserving the original class distribution. The available analysis record does not identify the exact source file, collection period, duplicate-removal procedure, attack-family counts, or whether records from the same device, connection, or session were separated across partitions.

Preprocessing removed non-informative and zero-variance fields, treated missing values using summary-value imputation, encoded categorical variables numerically, and standardized numeric predictors. Parameters for imputation, encoding, and scaling were estimated from the training data and then applied to the test data. The exact feature-by-feature imputation rules,

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

category-handling procedure, and complete list of excluded or retained identifiers were not available in the source record. For recurrent and convolutional models, the processed feature vector was reshaped as a one-dimensional ordered sequence in which each feature occupied one position. This representation permits architectural comparison but does not constitute a genuine timestamp-ordered packet or flow sequence; consequently, recurrent performance should not be interpreted as direct evidence of temporal attack modeling.

Table 1

Dataset and preprocessing characteristics

Element	Specification	Purpose
Dataset	TON-IoT network-traffic subset	Realistic IoT/IIoT intrusion benchmark
Observations	461,043	Binary normal-versus-attack classification
Predictor features	45	Numeric and categorical network attributes
Training set	322,730 observations (70%)	Model fitting and preprocessing estimation
Test set	138,313 observations (30%)	Held-out comparative evaluation
Preprocessing	Missing-value treatment, encoding, feature screening, standardization	Comparable numeric inputs and leakage control

3.3. Standalone Deep-Learning Architectures

The DNN received the processed feature vector directly. It used hidden layers with 128, 64, and 32 units, rectified linear activation, and dropout rates of 0.30 and 0.20 in the earlier hidden stages. A single sigmoid output represented the probability of attack traffic. This model served as the principal tabular nonlinear baseline.

The one-dimensional CNN used two convolutional layers with 32 and 16 filters, respectively, followed by flattening, a 32-unit dense layer, dropout of 0.20, and sigmoid output. The convolutional kernel size was one. Therefore, the model applied pointwise transformations at individual feature positions rather than learning wider neighboring-feature patterns in the conventional convolutional sense. This architectural limitation should be considered when interpreting the comparatively weaker CNN performance.

The LSTM model used two recurrent layers with 32 and 16 units. The first returned the full sequence to the second layer, and dropout of 0.20 was applied between recurrent stages. A 16-unit dense layer preceded the sigmoid classifier. The GRU model used a parallel 32-unit and 16-unit structure, enabling direct comparison between recurrent cell types while reducing gate complexity relative to LSTM.

Table 2

Configuration of the standalone models

Model	Principal layers	Regularization	Intended representation
DNN	Dense 128 → 64 → 32 → 1	Dropout 0.30 and 0.20	Nonlinear interactions among tabular features
CNN	Conv1D 32 → Conv1D 16 → Dense 32 → 1	Dropout 0.20	Feature-position and local structural patterns
LSTM	LSTM 32 → LSTM 16 → Dense 16 → 1	Dropout 0.20	Longer-range ordered dependencies
GRU	GRU 32 → GRU 16 → Dense 16 → 1	Dropout 0.20	Efficient gated dependency modeling

3.4. Training and Pairwise Ensembles

All standalone models were trained with the Adam optimizer, an initial learning rate of 0.001, binary cross-entropy loss, a maximum of eight epochs, and a batch size of 64. Early stopping restored the weights associated with the best monitored validation state. However, the source record did not specify the validation fraction, monitored metric, patience, random seed, hardware, software version, or hyperparameter-selection procedure. These omissions limit exact reproducibility and should be addressed in future replication.

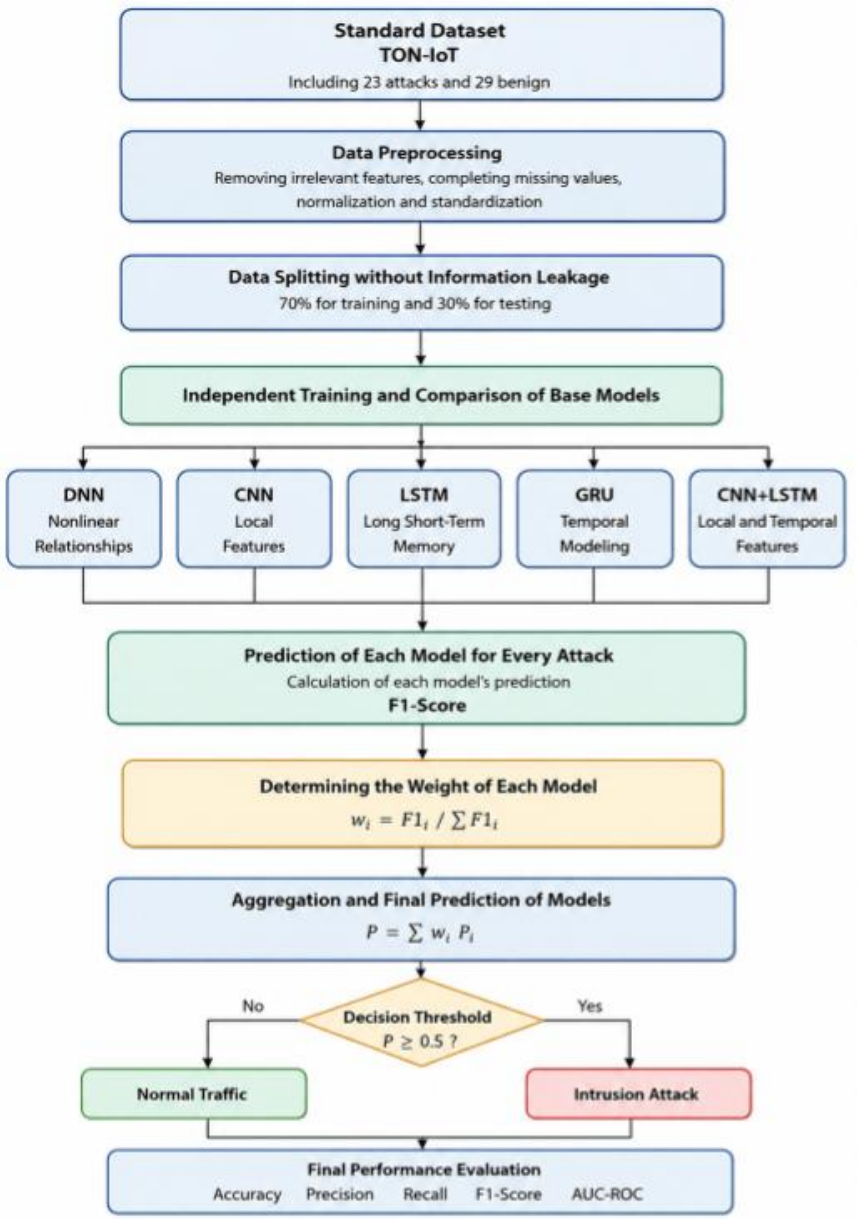
After fitting the four base models, predicted attack probabilities were retained for the held-out observations. The available workflow indicates an F1-weighted probability ensemble, with normalized weights calculated as $w_i = F1_i / \Sigma F1$ and the ensemble probability calculated as $P = \Sigma w_i P_i$; a threshold of 0.50 was then applied. The

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

evaluated pairs were DNN + CNN, DNN + LSTM, DNN + GRU, CNN + LSTM, CNN + GRU, and LSTM + GRU. The source documentation does not identify whether the F1-based weights were estimated from training, validation, or test predictions. If test-set F1 values were used, this would constitute information leakage; therefore, the ensemble results require cautious interpretation until the weighting dataset is documented.

Figure 1

Original analytical workflow for TON-IoT preprocessing, base-model training, ensemble construction, classification, and evaluation.



Note. The reported comparative analysis is restricted to the four standalone models and six pairwise ensembles for which complete Accuracy, F1, and AUC-ROC values were available.

3.5. Evaluation Measures

Performance was evaluated using accuracy, precision, recall, F1 score, AUC-ROC, and confusion matrices. Accuracy summarizes total correctness but may be optimistic under unequal class prevalence. Precision quantifies the proportion of attack alerts that are correct, whereas recall measures the proportion of attacks detected. The F1 score summarizes their balance, and AUC-ROC evaluates ranking discrimination across thresholds. For a

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

more complete operational assessment, future analyses should also report class prevalence, specificity, balanced accuracy, Matthews correlation coefficient, area under the precision–recall curve, calibration error, and threshold-specific false-positive rates.

The source analysis reported single point estimates from one record-level train–test split rather than means from repeated random seeds, cross-validation, group-aware splitting, or external validation. Accordingly, all ranking differences are interpreted descriptively. No inferential claim is made that the small differences between leading ensembles would replicate under alternative initializations, devices, sessions, time periods, or independent IoT datasets.

4. Findings and Results

4.1. Overall Model Comparison

The pairwise ensembles occupied all six leading positions in the observed F1 ranking, followed by the four standalone models. The highest point estimates were obtained by Ensemble (DNN + GRU), with accuracy of 0.9774, F1 score of 0.9680, and AUC-ROC of 0.9963. Ensemble (DNN + LSTM) was a close second, with corresponding values of 0.9769, 0.9673, and 0.9957. The F1 difference between these models was 0.0007 and cannot be interpreted as statistically meaningful without repeated runs, paired error analysis, or confidence intervals.

The CNN + GRU and CNN + LSTM ensembles ranked third and fourth by F1 score, achieving 0.9635 and 0.9601, respectively. Although CNN was the weakest individual model, its inclusion in higher-ranked ensembles suggests possible prediction diversity. However, probability correlations, disagreement rates, and error-overlap measures were not reported, so complementary errors remain a plausible interpretation rather than an empirically demonstrated mechanism.

Table 3

Performance of standalone and pairwise ensemble models

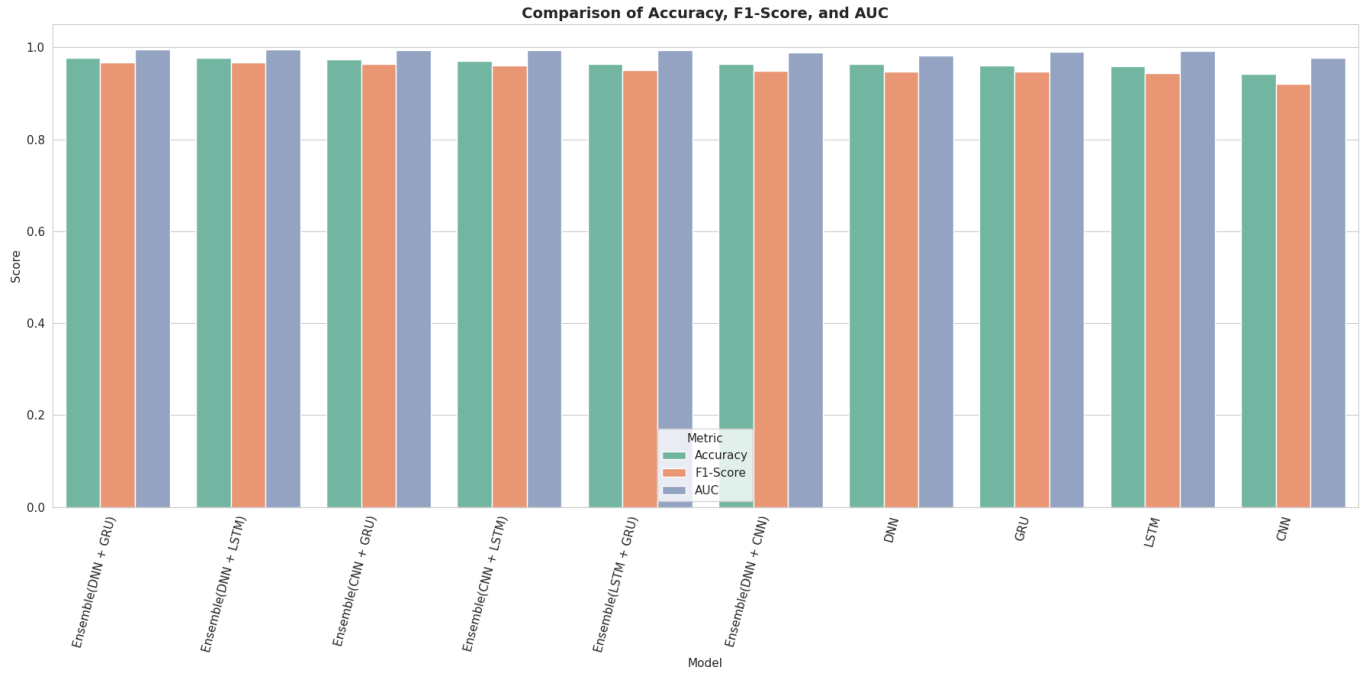
Rank	Model	Accuracy	F1 score	AUC-ROC
1	Ensemble (DNN + GRU)	0.9774	0.9680	0.9963
2	Ensemble (DNN + LSTM)	0.9769	0.9673	0.9957
3	Ensemble (CNN + GRU)	0.9737	0.9635	0.9942
4	Ensemble (CNN + LSTM)	0.9713	0.9601	0.9932
5	Ensemble (LSTM + GRU)	0.9648	0.9509	0.9941
6	Ensemble (DNN + CNN)	0.9646	0.9493	0.9890
7	DNN	0.9639	0.9482	0.9823
8	GRU	0.9614	0.9467	0.9912
9	LSTM	0.9594	0.9435	0.9928
10	CNN	0.9429	0.9211	0.9777

Across the complete set, accuracy ranged from 0.9429 to 0.9774, F1 score from 0.9211 to 0.9680, and AUC-ROC from 0.9777 to 0.9963. These values indicate strong within-split discrimination. They should not be interpreted as evidence of deployment-level generalization because no group-aware, temporal, cross-dataset, or repeated-seed validation was reported.

Figure 2

Original comparison of Accuracy, F1 score, and AUC-ROC across the evaluated standalone and ensemble models.

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433



Note. Models are ordered by F1 score. Higher values indicate better performance.

4.2. Standalone Models

Among standalone models, DNN produced the highest observed accuracy (0.9639) and F1 score (0.9482). GRU followed with accuracy of 0.9614 and F1 score of 0.9467, while LSTM achieved 0.9594 and 0.9435. GRU and LSTM recorded AUC-ROC values of 0.9912 and 0.9928, respectively. Because the input represented ordered feature positions rather than genuine time sequences, these recurrent results should be interpreted as performance on an imposed feature ordering, not direct evidence of temporal dependency modeling.

CNN produced the lowest standalone point estimates: accuracy of 0.9429, F1 score of 0.9211, and AUC-ROC of 0.9777. The kernel size of one limited the network to pointwise feature-position transformations rather than wider local windows. The performance of CNN-containing ensembles may indicate useful prediction diversity, but this explanation requires direct analysis of model disagreement and error overlap.

4.3. Error Profiles Across Models

Figure 3 presents confusion matrices for the standalone and pairwise ensemble models. For Ensemble (DNN + GRU), the test set contained 87,926 true negatives and 47,260 true positives, with 2,074 false positives and 1,053 false negatives. These counts imply 90,000 normal observations and 48,313 attack observations in the test set. The false-negative count is operationally relevant, but the 2,074 false-positive alerts also demonstrate that high aggregate metrics do not eliminate alert burden. Exact confusion-matrix counts and derived precision, recall, specificity, and balanced accuracy should be tabulated for all models in a fully reproducible report.

Figure 3

Original confusion matrices for the reported standalone and pairwise ensemble models.

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

Note. Normal traffic was coded 0 and attack traffic was coded 1. The upper-left panel shows the best-performing DNN + GRU ensemble; N = 138,313 test observations.

5. Discussion

5.1. *Principal Findings*

The central descriptive finding was that all pairwise ensembles achieved higher observed F1 scores than the standalone architectures in this single split, and the DNN + GRU pair produced the highest point estimates for the three reported metrics. This pattern is consistent with a potential benefit from combining heterogeneous representations, but the study did not measure model-error diversity, repeat training across seeds, or perform inferential comparisons. Therefore, ensemble superiority should be described as an observed within-split result rather than a general conclusion.

The second-ranked DNN + LSTM model showed a nearly identical pattern, with an F1 score only 0.0007 below DNN + GRU. This difference is too small to support a general claim that GRU is superior to LSTM without repeated runs and paired statistical testing. The defensible conclusion is that both recurrent outputs complemented the DNN under the reported configuration.

CNN alone ranked last, whereas CNN + GRU and CNN + LSTM ranked third and fourth. This pattern is compatible with ensemble theory, but complementary errors were not directly quantified. Future work should report probability correlations, pairwise disagreement, double-fault measures, and overlap among false positives and false negatives before attributing gains to diversity.

The results can be compared with recent IoT IDS studies that integrate convolutional, recurrent, and attention-based representations (Afraji et al., 2025; Ishtiaq et al., 2025). Cross-domain references may illustrate general machine-learning principles, but they do not directly validate the present intrusion detector. The literature review and interpretation should therefore prioritize TON-IoT studies, leakage-resistant partitioning, cross-dataset generalization, calibration, and explainable cybersecurity evaluation.

5.2. *Implications for Trustworthy AI and Human-Centered IoT*

Intrusion detection affects users through the systems it protects, but the present study evaluated network-traffic

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

classification rather than human trust, fairness, privacy perceptions, analyst behavior, or social outcomes. The human-centered discussion should therefore be understood as a set of deployment considerations rather than empirical evidence of trustworthy-AI performance.

The best observed ensemble generated 2,074 false positives on the test set. In operational environments, alert volume may create analyst fatigue or interrupt legitimate activity. Deployment should therefore evaluate calibrated thresholds, precision–recall trade-offs, Brier score or other calibration measures, alert aggregation, and explanation quality on independent operational data rather than assuming that a threshold of 0.50 is optimal.

The framework should be treated as an AI-assisted security layer rather than an autonomous authority. High-risk alerts may justify immediate containment in safety-critical systems, but routine decisions should incorporate context, asset criticality, prior device behavior, and human oversight. Logs should record model version, threshold, input-processing steps, and analyst action. Users and organizations also need clear policies describing what network data are collected, how long they are retained, and how automated decisions may affect access to services.

IoT traffic changes when new devices, applications, user routines, and attack strategies are introduced. Continuous drift monitoring and scheduled re-evaluation are therefore necessary. However, drift robustness was not tested in this study, and no conclusion about adaptive performance can be drawn from the reported static benchmark.

5.3. Limitations

The study has several limitations. First, the evaluation used one processed dataset, one record-level 70/30 split, and single point estimates. Repeated seeds, group-aware or chronological splitting, confidence intervals, paired error tests, and external validation are required to estimate uncertainty and generalization. Second, the exact TON-IoT subset, duplicate-removal procedure, class composition, complete feature list, identifier handling, validation partition, and source of ensemble weights were not fully documented. These omissions limit reproducibility and leave unresolved risks of session-, device-, or test-set leakage.

Third, CNN, LSTM, and GRU received an imposed ordering of processed tabular features rather than genuine timestamp-ordered packet or flow sequences. Feature order is not equivalent to time, and recurrent gains should not be interpreted as temporal attack modeling. Sensitivity to alternative feature orders and evaluation on true sequences are needed. Fourth, the F1-weighted ensemble procedure requires documentation of the data used to estimate weights; test-derived weighting would invalidate independent evaluation.

Fifth, the study evaluated binary normal-versus-attack classification and did not report attack-family performance, AUC-PR, calibration, model complexity, training time, memory use, or edge-device feasibility. Sixth, the trustworthy-AI discussion was not accompanied by empirical explainability, robustness, privacy, fairness, drift, or human-factors analyses. Finally, the study did not include conventional baselines such as logistic regression, random forest, support-vector machines, or gradient boosting.

6. Conclusion

This study compared four standalone deep-learning models and six pairwise ensembles for binary intrusion detection using 461,043 TON-IoT observations. The DNN + GRU ensemble produced the highest observed point estimates, followed closely by DNN + LSTM, and all six ensembles ranked above the four individual models by F1 score within the reported split. These findings describe performance under one preprocessing pipeline and one record-level partition; they do not establish statistically significant superiority, temporal modeling, leakage-free generalization, or deployment readiness. Stronger conclusions require transparent ensemble-weight estimation, group-aware or chronological splitting, repeated seeds, external validation, calibration analysis, complete reproducibility materials, and operational evaluation.

Authors' Contributions

Sadiq Sahep Majeed: Conceptualization, methodology, software, investigation, data curation, visualization, and writing—original draft. Mahmood Deypir: Conceptualization, methodology, supervision, project administration, and writing—review and editing. Razieh Farazkish: Validation, formal analysis, interpretation, and writing—review and editing. Ali Broumandnia: Supervision, methodological review, interpretation, and critical revision. All authors reviewed and approved the final manuscript and accept responsibility for its content.

Declaration

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433

Generative artificial intelligence was used only to assist English translation, language polishing, and structural editing of the manuscript. It was not used to generate the dataset, fabricate results, execute the reported experiments, select the final metrics, or make authorship decisions. The authors reviewed and revised all AI-assisted text and remain fully responsible for the accuracy, integrity, interpretation, and final approval of the manuscript.

Transparency Statement

The TON-IoT dataset is publicly available through the University of New South Wales TON-IoT project repository. To enable independent verification, the exact source files, preprocessing code, feature list, split indices, random seeds, ensemble-weight calculations, model configurations, trained weights, and raw prediction outputs should be deposited in a permanent public repository. These materials were not included with the present manuscript.

Acknowledgments

The authors acknowledge the developers and maintainers of the TON-IoT dataset and the open-source software communities whose tools supported the computational analysis.

Declaration of Interest

The authors report no conflict of interest.

Funding

According to the authors, this article has no financial support.

Ethics Considerations

This study used the publicly available TON-IoT cybersecurity dataset and did not involve recruitment of human participants, animal experimentation, clinical intervention, or collection of personally identifiable information by the authors. The analysis was limited to secondary network-traffic records. Formal institutional ethics approval and informed consent were therefore not required. The authors affirm that the dataset was used for legitimate academic cybersecurity research and that the proposed system is intended for defensive intrusion detection, not unauthorized surveillance or offensive activity.

References

Afraji, D. M. A. A., Lloret, J., & Peñalver, L. (2025). An integrated hybrid deep learning framework for intrusion detection in IoT and IIoT networks using CNN-LSTM-GRU architecture. *Computation*, *13*(9), 222. https://doi.org/10.3390/computation13090222

Ahmad, J., Latif, S., Khan, I. U., Alshehri, M. S., Khan, M. S., Alasbali, N., & Jiang, W. (2025). An interpretable deep learning framework for intrusion detection in industrial Internet of Things. *Internet of Things*, *33*, 101681. https://doi.org/10.1016/j.iot.2025.101681

Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., & Anwar, A. (2020). TON-IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems. *IEEE Access*, *8*, 165130-165150. https://doi.org/10.1109/ACCESS.2020.3022862

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*, 123-140. https://doi.org/10.1007/BF00058655

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing,

Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Generation Computer Systems*, *82*, 761-768. https://doi.org/10.1016/j.future.2017.08.043

Ferrag, M. A., Maglaras, L., Moschoyiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, *50*, 102419. https://doi.org/10.1016/j.jisa.2019.102419

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. https://synapse.koreamed.org/pdf/10.4258/hir.2016.22.4.351

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Ishtiaq, W., Zannat, A., Parvez, A. H. M. S., Hossain, M. A., Kanchan, M. H., & Tarek, M. M. (2025). CST-AFNet: A dual attention-based deep learning framework for intrusion detection in IoT networks. *Array*, *27*, 100501. https://doi.org/10.1016/j.array.2025.100501

Karimi Dastgerdi, A., & Zamani Boroujeni, F. (2020). A review of deep learning methods for financial market prediction. *Transactions on Data Analysis in Social Science*, *2*(3), 164-172. https://doi.org/10.47176/TDASS.2020.164

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems 30,

Maoudj, S. E., & Belghiat, A. (2025). A deep learning-based approach with two-step minority classes prediction for intrusion detection in Internet of Things networks. *Knowledge-Based Systems*, *312*, 113143. https://doi.org/10.1016/j.knosys.2025.113143

Mirsky, Y., Doitshman, T., Elovici, Y., & Shabtai, A. (2018). Kitsune: An ensemble of autoencoders for online network intrusion detection. Proceedings of the Network and Distributed System Security Symposium,

Mohammadi, S., & Anisheh, S. M. (2024). Human activity recognition based on deep learning using sensor data. *Transactions on Data Analysis in Social Science*, *6*(4), 222-230. https://doi.org/10.47176/TDASS.2024.222

Moradi, M. (2023). A Bayesian Model and Bayesian Classification on the Data Obtained from Children's Educational Activity in the IoT Environment. *Transactions on Machine Intelligence*, *6*(3), 126-136. https://doi.org/10.47176/TMI.2023.126

Moustafa, N., Ahmed, M., & Ahmed, S. (2020). Data analytics-enabled intrusion detection: Evaluations of ToN-IoT Linux datasets. 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications,

Rahmani Seryasat, O., Kor, I., Ghayoumi Zadeh, H., & Shams Taleghani, A. (2021). Predicting the number of comments on Facebook posts using an ensemble regression model. *International Journal of Nonlinear Analysis and Applications*, *12*(Special Issue), 49-62. https://ijnaa.semnan.ac.ir/article_4796.html

Ravaei, B., Ravaei, S., Moshrefzadeh, S., & Rahmani Seryasat, O. (2022). An efficient and load-balanced task offloading in vehicular Internet of Things. *Transactions on Machine Intelligence*, *5*(1), 46-56. https://doi.org/10.47176/TMI.2022.46

Sagi, O., & Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(4), e1249. https://doi.org/10.1002/widm.1249

Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *2*(1), 41-50. https://doi.org/10.1109/TETCI.2017.2772792

Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep learning approach for intelligent intrusion detection system. *IEEE Access*, *7*, 41525-41550. https://doi.org/10.1109/ACCESS.2019.2895334

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*(2), 241-259. https://doi.org/10.1016/S0893-6080(05)80023-1

AITBSS
AI and Tech in Behavioral and Social Sciences
E-ISSN: 3041-9433