

Research Methodology in Computer Science

Course Code: AI800

ASSIGNMENT 1

From Traffic Classification to Early Intervention: A Comparative Review of Machine Learning Approaches for IoT Botnet Detection

Student Name: Ibrahim Mohammed Alsidrani

Student ID: 262000987

Lecturer: Dr. Mohammed Anbar

Submission Date: 04 July 2026

Assignment Weight: 20%

Table of Contents

1. Introduction	1
2. Review Scope and Paper Selection	1
2.1 Review Questions	1
2.2 Quality Assessment Criteria	2
2.3 Analytical Framework: Classification-to-Intervention Continuum	2
3. Summary of the Selected Research Papers	2
3.1 Paper 1: Intrusion Detection System for IoT Botnet Attacks Using Deep Learning	2
3.2 Paper 2: SMOTE-DRNN: A Deep Learning Algorithm for Botnet Detection in IoT Networks	2
3.3 Paper 3: Toward a Deep Learning-Based Intrusion Detection System for IoT Against Botnet Attacks	3
3.4 Paper 4: A Collaborative Approach to Early Detection of IoT Botnet	3
3.5 Paper 5: ELBA-IoT: An Ensemble Learning Model for Botnet Attack Detection in IoT Networks	5
3.6 Paper 6: Using Ensemble Learning Technique for Detecting Botnet on IoT	5
4. Comparative Analysis Across the Classification-to-Intervention Continuum	5
4.1 Detection Reliability	6
4.2 Evidence Quality and Class Imbalance	6
4.3 Operational Cost and Gateway Suitability	6
4.4 Transferability to New Networks and Botnets	6
4.5 Response Timing: Classification versus Early Intervention	7
5. Connections Between the Studies	7
6. Conflicts and Differences Between the Findings	7
7. Synthesis Matrix	9
8. Synthesis of Findings and Research Gaps	10
8.1 Research Position Derived from the Review	10
8.2 Identified Research Gaps	10
8.3 Proposed Direction for Future Research	10
8.4 Methodological Quality Assessment	11
8.5 Direct Cross-Paper Comparison	11
8.6 Threats to Validity	11
9. Conclusion	12
References	12

1. Introduction

The Internet of Things (IoT) connects everyday and industrial devices, including cameras, sensors, smart home equipment, industrial controllers, and wearable devices. These devices are useful, but they also increase the number of possible entry points for attackers. Many IoT devices have limited processing power, weak default passwords, and infrequent security updates. Attackers can exploit these weaknesses, add the devices to a botnet, and use them for DDoS attacks, network scanning, credential theft, keylogging, or data exfiltration.

Traditional signature-based tools work best when the attack pattern is already known. For that reason, they may miss a new or modified botnet. Machine learning (ML) and deep learning (DL) provide another option because they can learn how normal and malicious behavior differ. Even so, a high accuracy score does not automatically mean that a model is ready for real use. Class imbalance, processing cost, prediction time, the type of data available, and how early the attack is detected are also important.

This review discusses six papers published in 2021 and 2022. The papers cover deep neural networks, recurrent networks combined with SMOTE, convolutional neural networks, collaborative early detection, and ensemble learning. They are not treated as if they solve the same problem. Instead, they are placed on a path that starts with classifying network traffic and moves toward earlier intervention. The comparison focuses on their aims, methods, datasets, results, strengths, and limitations, before identifying what is still missing from a practical IoT botnet detection system.

2. Review Scope and Paper Selection

The six papers were chosen because each one is directly related to machine learning-based IoT botnet detection, while still taking a different approach. Most study network traffic or device behavior. One paper focuses on early detection by combining several types of behavior, and another compares many machine learning algorithms before building a sequential ensemble. This mix gives the review enough variety for a useful comparison.

- Direct relevance to IoT botnet or IoT cyberattack detection.
- A clearly described machine learning, deep learning, ensemble, or collaborative learning method.
- Experimental evaluation using an established or explicitly constructed cybersecurity dataset.
- Reported quantitative results that permit comparison of predictive performance or computational efficiency.
- A publication date within the recent 2021-2022 period represented by the provided papers.

The review first explains each paper on its own, then compares the studies using the same set of points: data source, preprocessing, model type, evaluation measures, efficiency, and generalization. Attention is paid to whether a high accuracy score is supported by other metrics and whether the test setting is close to a real IoT environment.

2.1 Review Questions

- RQ1: How do the reviewed machine learning approaches progress from classifying observed IoT traffic toward supporting earlier intervention?
- RQ2: How do dataset choice, class imbalance, preprocessing, and evaluation metrics affect the reliability of the reported results?
- RQ3: Which approaches are most suitable for real-time or resource-constrained IoT deployment?
- RQ4: What limitations remain regarding unseen botnets, generalization, and real-world implementation?

2.2 Quality Assessment Criteria

All six papers were reviewed using the same criteria. I considered how closely each paper matched the topic, how clearly it stated its objective, whether the dataset and preprocessing steps were suitable, whether the chosen model made sense, and whether the evaluation was complete. I also looked at comparisons with baseline methods, computational cost, reproducibility, and practical use. Applying the same criteria made the comparison fairer and easier to follow.

2.3 Analytical Framework: Classification-to-Intervention Continuum

To keep the analysis connected to the title, the papers are placed on a continuum from traffic classification to early intervention. I compare them using four points: detection reliability, operational cost, response timing, and transferability. Detection reliability looks beyond overall accuracy and considers class imbalance and per-class results. Operational cost covers inference time, preprocessing effort, and whether the model could run on a constrained gateway. Response timing asks whether the system reacts only after malicious traffic is visible or can act during partial malware execution. Transferability considers whether the same result is likely to hold for new devices, datasets, and botnet families. The same framework is used again in Sections 4 and 8.

3. Summary of the Selected Research Papers

3.1 Paper 1: Intrusion Detection System for IoT Botnet Attacks Using Deep Learning

Jithu et al. (2021) aimed to develop an anomaly-based intrusion detection system for identifying IoT botnet attacks, with a focus on DDoS behavior. The authors explained that conventional protection mechanisms are difficult to apply to heterogeneous and resource-constrained IoT networks. Their solution employed a deep neural network (DNN) to learn patterns in network traffic and classify malicious activity.

The researchers used the Bot-IoT dataset, which was generated in a realistic network environment at the UNSW Canberra Cyber Range Lab and contains both normal and attack traffic. The authors selected predictive traffic features and evaluated the model using accuracy, precision, F-score, Cohen's kappa, and the area under the receiver operating characteristic curve. The model achieved an accuracy of about 94% and an AUC of 0.98, which indicated strong discrimination between normal and malicious traffic.

One strength of this paper is its use of an IoT-specific dataset containing realistic network traffic and its evaluation with several measures rather than accuracy alone. The proposed DNN also shows that

relatively straightforward deep architecture can provide strong anomaly detection performance. At the same time, the paper gives limited detail about deployment on actual low-power IoT hardware, and the reported 94% accuracy is lower than the results achieved by several more specialized models reviewed later. The study also mainly shows performance on known Bot-IoT patterns, so its ability to recognize unseen botnet families is not established.

3.2 Paper 2: SMOTE-DRNN: A Deep Learning Algorithm for Botnet Detection in IoT Networks

Popoola et al. (2021) addressed one of the most important methodological problems in intrusion detection: severe class imbalance. In multiclass botnet datasets, common attack classes may contain millions of examples while rare attacks contain only a small number. A model trained directly on such data can achieve almost perfect overall accuracy while failing to recognize minority attacks.

The authors combined the Synthetic Minority Oversampling Technique (SMOTE) with a deep recurrent neural network (DRNN). SMOTE generated synthetic observations for minority classes, while the recurrent network learned hierarchical representations from the balanced Bot-IoT traffic. The model performed multiclass classification over normal traffic and ten botnet attack classes and was tested using accuracy, precision, recall, F1 score, false-positive rate, negative predictive value, AUC, geometric mean, and Matthews correlation coefficient.

The SMOTE-DRNN achieved 99.50% precision, 99.75% recall, a 99.62% F1 score, 99.87% AUC, 99.74% geometric mean, and 99.62% MCC, with an overall accuracy reported as 100%. The most valuable finding was not merely the high accuracy, but the demonstration that the unbalanced DRNN produced misleadingly high accuracy while performing poorly on rare classes. For example, the basic DRNN failed to classify the data-exfiltration minority sample, whereas SMOTE-DRNN correctly recognized it.

A key contribution of the paper is its rigorous treatment of imbalance and its use of metrics that are sensitive to minority-class performance. It offers a more trustworthy evaluation than studies that rely primarily on accuracy. Its main limitation is computational cost: oversampling enlarges the training data and the hybrid model required a longer training time than the basic DRNN. Synthetic examples may also reproduce patterns from the existing dataset rather than guarantee performance against genuinely new attacks.

3.3 Paper 3: Toward a Deep Learning-Based Intrusion Detection System for IoT Against Botnet Attacks

Idrissi et al. (2021) proposed BotIDS, a convolutional neural network (CNN)-based intrusion detection system designed for multiclass recognition of IoT botnet attacks. The objective was to construct a single model that could identify normal traffic and multiple attack categories, rather than requiring a separate binary detector for each type of attack.

The researchers used the Bot-IoT dataset, originally containing about 72 million traffic records. A 5% full-feature subset of about 3.6 million records and a reduced ten-feature version were used. Categorical and IP-address values were numerically encoded, the attack category and subcategory were merged into eleven output classes, one-hot encoding was applied, and feature values were normalized. The

transformed feature vectors were reshaped into image-like inputs for the CNN. The data was divided into 75% training and 25% testing subsets.

BotIDS was compared with simple RNN, LSTM, and GRU models. It achieved 99.94% validation accuracy, a validation loss of 0.58%, and prediction execution time below 0.34 milliseconds. These results suggest that a CNN can efficiently learn local relationships in transformed network features and provide fast multiclass prediction.

The paper gives a clear preprocessing pipeline and a useful comparison among several deep learning architectures. Its very low prediction time is attractive for near-real-time detection. Even so, reshaping tabular traffic data into an image does not necessarily preserve a natural spatial relationship among features, and the authors do not provide a detailed analysis of how class imbalance influenced each attack class. The random train-test split may also allow highly similar traffic records to appear in both subsets, potentially producing optimistic generalization results.

3.4 Paper 4: A Collaborative Approach to Early Detection of IoT Botnet

Nguyen et al. (2022) shifted the focus from detecting a botnet after its malicious behavior is fully expressed to detecting it earlier in the malware lifecycle. The authors noted that many existing dynamic-analysis methods require complete time-series data collected throughout malware execution. By that stage, a bot may already have scanned targets, communicated with command-and-control infrastructure, or participated in an attack.

The framework collected three categories of behavior through a V-Sandbox environment: network behavior, device performance or resource-use information, and system calls. Separate classifiers, including support vector machines, k-nearest neighbors, decision trees, and random forests, were evaluated for each data type. The best individual models were combined through a voting-based collaborative learning approach. The dataset contained 5,023 IoT botnet malware samples and 3,888 benign samples.

The final combination used KNN for network data, random forest for performance data, and KNN for system-call data. It achieved 99.37% accuracy, an AUC of 0.9896, and a false-positive rate of 0.0194. For malware samples, the collaborative model obtained an F1 score of 0.9957. For benign samples, the F1 score was 0.9884. The prediction stage required about six seconds and could operate on partial execution behavior, enabling earlier detection.

The main strength of the study is the integration of heterogeneous behavioral evidence. It reduces dependence on a single source and directly addresses the timing of detection. It also tested previously unseen samples. At the same time, the sandbox must start virtual machines and repeatedly collect behavior, causing total processing time of about three minutes per sample. This limits high-throughput use and makes the framework more suitable for gateway or laboratory analysis than direct execution on constrained devices.

3.5 Paper 5: ELBA-IoT: An Ensemble Learning Model for Botnet Attack Detection in IoT Networks

Abu Al-Haija and Al-Dala'ien (2022) proposed ELBA-IoT, an anomaly-based and lightweight ensemble learning model for identifying botnet traffic generated by compromised IoT devices. The model profiles behavioral traffic features and evaluates three decision-tree ensemble techniques: AdaBoosted trees, RUSBoosted trees, and bagged trees.

The researchers used the N-BaIoT-2021 dataset, which contains normal traffic and Mirai and Bashlite botnet traffic captured from infected IoT devices. The models were evaluated as ten-class classifiers, allowing the system to distinguish multiple attack behaviors rather than only benign and malicious traffic. The best ensemble produced 99.6% detection accuracy and an inference overhead of about 40 microseconds.

An important feature of ELBA-IoT is the combination of high multiclass accuracy with extremely low inference cost. The approach is therefore more directly aligned with the constraints of time-sensitive IoT gateways than computationally heavy deep networks. The comparison with earlier IDS models also suggests that the model performs competitively across multiple output classes.

One limitation is that the work was evaluated offline on a single dataset and was not deployed on a physical IoT gateway. The authors themselves identify real-time gateway implementation as future work. In addition, the model's strong performance on Mirai and Bashlite does not demonstrate equivalent performance on unseen botnet families or under changing network conditions.

3.6 Paper 6: Using Ensemble Learning Technique for Detecting Botnet on IoT

Rezaei (2021) aimed to improve botnet detection accuracy while reducing the number of required traffic features. The study evaluated supervised, unsupervised, and regression-based machine learning algorithms and then combined the strongest candidates through a sequential ensemble. Seventeen algorithms were initially examined. Algorithms with accuracy above 90% and execution time below 20 seconds were shortlisted.

The final ensemble combined an artificial neural network (ANN) and a decision tree (DT). Traffic first passed through the ANN, which removed detected bot or botnet addresses, and the remaining records were then evaluated by the DT. The dataset contained 990,517 domain records, including 199,772 malicious domains and 790,745 normal domains. The authors standardized the features and reduced the input to 20 selected features. Several train-test proportions and five-fold cross-validation were used.

The proposed ensemble reported 100% accuracy using 20 selected features, with no reported false-positive or false-negative errors in the authors' experimental results. The authors also stated that reducing the feature set below 20 lowered performance, suggesting that the selected subset represented a practical minimum for their experimental setting.

The study is relevant because it directly compares many learning paradigms and shows how a sequential ensemble can combine complementary classifiers. Feature reduction and repeated data splits are additional strengths. At the same time, a perfect result on a highly imbalanced dataset requires cautious interpretation. The paper does not establish performance on an independent external dataset, a different

botnet family, or live IoT devices. The sequential filtering design may also propagate errors from the first classifier, and the paper provides limited analysis of concept drift and the operational cost of inspecting high-volume traffic.

4. Comparative Analysis Across the Classification-to-Intervention Continuum

Before comparing the models in detail, it is useful to place the six studies on the same continuum. Five studies mainly classify network traffic after enough benign or malicious evidence has appeared: the basic DNN, SMOTE-DRNN, BotIDS, ELBA-IoT, and Rezaei models. Nguyen et al. move closer to early intervention because their framework uses partial execution behavior, system calls, network activity, and device performance before malware completes its full activity. This shows that current research is stronger in accurate traffic classification than in experimentally validated early intervention.

The six studies show that machine learning can achieve strong results in IoT botnet detection. At the same time, the meaning of high performance depends on the dataset, the way the experiment was designed, and the main goal of the system.

4.1 Detection Reliability

Three studies used Bot-IoT, which supports direct comparison but also creates a common dependency. Bot-IoT was generated in a realistic cyber-range environment and contains many attack categories, yet it remains a controlled dataset. Repeated use of the same dataset can lead researchers to optimize methods for its specific feature distributions rather than for the diversity of traffic found in production networks. ELBA-IoT used N-BaIoT-2021 with Mirai and Bashlite traffic collected from infected devices, while the early-detection study constructed a behavioral dataset from sandbox executions. Rezaei used a substantially larger domain-name dataset than the other reviewed studies, but the model was not tested on an independent external dataset. Therefore, dataset size alone does not establish cross-network generalization.

4.2 Evidence Quality and Class Imbalance

Reported accuracy ranged from about 94% to 100%. These values cannot be compared without considering class distribution and task complexity. The SMOTE-DRNN paper provides the strongest warning: a model can report nearly perfect accuracy while completely missing a rare class. Its use of recall, F1 score, geometric mean, and MCC is methodologically stronger than relying on accuracy alone. BotIDS and ELBA-IoT report exceptional accuracy and speed, but more detailed per-class performance and tests under distribution shift would improve confidence.

4.3 Operational Cost and Gateway Suitability

Deep models such as DNN, DRNN, and CNN can automatically learn complex representations, but training and deployment costs differ substantially. SMOTE-DRNN has excellent minority-class performance but requires additional computation and training time. BotIDS provides very fast inference, although its reported execution time was obtained in an experimental computing environment rather than on a constrained IoT processor. ELBA-IoT offers the clearest lightweight result, with 40-microsecond

inference using decision-tree ensembles. The early-detection framework is more naturally deployed at gateways, edge servers, or laboratory infrastructure rather than directly on endpoint devices.

4.4 Transferability to New Networks and Botnets

Most papers split a known dataset into training and testing subsets. This measures generalization to held-out samples from the same distribution, not necessarily to a new botnet family or a different organization. The early-detection study strengthened its evaluation by testing previously unseen samples. Rezaei used several train-test proportions and five-fold cross-validation, but did not perform cross-dataset testing. Future studies should use temporal splits, cross-dataset testing, and leave-one-attack-family-out experiments to determine whether models can recognize genuinely novel threats.

4.5 Response Timing: Classification versus Early Intervention

The reviewed methods solve different operational problems. The DNN, SMOTE-DRNN, BotIDS, and ELBA-IoT studies classify network traffic. The early-detection framework uses partial malware execution behavior to intervene before the complete attack lifecycle. Rezaei's framework focuses on reducing the number of required features while combining ANN and DT sequentially. For this reason, the best method depends on the deployment objective: real-time packet classification, rare-attack recognition, pre-attack malware analysis, or feature-efficient sequential detection.

5. Connections Between the Studies

Anomaly and behavior-based detection: All six papers move beyond fixed signatures and learn patterns from network traffic or system behavior. This makes them potentially more adaptable to modified attack variants.

IoT resource constraints: Each study recognizes that IoT endpoints have limited memory, computation, or energy. The proposed responses differ, but lightweight operation is a recurring requirement.

Need for multiclass detection: SMOTE-DRNN, BotIDS, and ELBA-IoT distinguish multiple botnet activities rather than only predicting benign or malicious traffic. This provides more actionable security information.

Importance of preprocessing: Feature selection, encoding, normalization, oversampling, and data transformation are central to performance. The studies show that model architecture alone is not sufficient.

Combination of models: SMOTE-DRNN combines oversampling with recurrent learning; the early-detection paper combines feature-specific classifiers; ELBA-IoT combines boosted or bagged decision trees; and Rezaei combines ANN and DT sequentially. These studies support the view that complementary components can outperform a single model.

Experimental performance above 99% in specialized settings: Most specialized models report very high accuracy. This indicates the discriminative potential of IoT behavioral data but also raises the need for stronger cross-dataset and real-world validation.

6. Conflicts and Differences Between the Findings

Deep learning versus ensemble learning: The DNN, DRNN, and CNN papers emphasize automatic representation learning from large datasets. ELBA-IoT shows that decision-tree ensembles can achieve comparable or better accuracy with much lower inference overhead.

Raw imbalance versus synthetic balancing: BotIDS and the basic DNN study do not place class imbalance at the center of their methods, whereas SMOTE-DRNN shows that imbalance can make accuracy misleading and argues for broader metrics and explicit balancing.

Traffic-based detection versus execution-based detection: Most studies analyze network traffic. Nguyen et al. combine network, system-call, and performance data, enabling earlier and richer malware detection but increasing data-collection time and complexity.

Parallel versus sequential ensemble design: ELBA-IoT aggregates decision-tree ensembles, whereas Rezaei applies ANN and DT sequentially, removing detected bot records between stages. The approaches differ in error propagation, inference organization, and interpretability.

Prediction speed versus preparation time: BotIDS and ELBA-IoT report sub-millisecond inference. The early-detection framework takes seconds to predict and about three minutes to collect and process a sample, illustrating a trade-off between richer evidence and operational speed.

Specialization versus generalization: Bot-IoT and N-BaIoT-based models achieve outstanding results on known attack distributions. Rezaei's 100% result is especially specialized and should be confirmed through cross-dataset and time-separated testing before it can be interpreted as general botnet detection capability.

7. Synthesis Matrix

Study	Year	Method / Design	Dataset	Key Findings	Strengths	Limitations	Primary deployment role
Jithu et al.	2021	Anomaly-based DNN	Bot-IoT	accuracy; AUC 0.98 94%≈	IoT-specific data; multiple metrics; relatively simple model	Limited physical deployment evidence; unknown-attack performance not established	Baseline network IDS
Popoola et al.	2021	SMOTE + DRNN multiclass model	Bot-IoT	F1 score = 99.62%; AUC = 99.87%; strong minority-class detection	Directly addresses imbalance; comprehensive metrics	Higher computational cost and training time; synthetic samples may not reflect new attacks	Imbalance-aware multiclass detector
Idrissi et al.	2021	CNN-based BotIDS; compared with RNN, LSTM, GRU	Bot-IoT, 5% subset (~3.6M records)	validation accuracy; prediction time < 0.34 ms 99.94%	Clear preprocessing; fast multiclass inference	Tabular-to-image transformation is debatable; limited imbalance and cross-dataset analysis	Fast deep gateway classifier
Nguyen et al.	2022	Voting-based collaborative ML over network, performance, and system-call models	5,023 malware and 3,888 benign samples	accuracy; AUC = 0.9896; early prediction from partial behavior 99.37%	Multimodal evidence; early detection; unseen-sample testing	Sandbox processing takes ~3 minutes per sample; deployment complexity	Early malware-analysis layer
Abu Al-Haija & Al-Dala'ien	2022	AdaBoosted, RUSBoosted, and bagged decision-tree ensembles	N-BaIoT-2021	accuracy; inference overhead = 40 μs 99.6%	Lightweight and fast; ten-class detection	Offline single-dataset evaluation; not yet deployed on physical gateways	Lightweight real-time gateway detector
Rezaei	2021	Sequential ANN-DT ensemble after screening 17 ML algorithms	990,517 domain records: 199,772 malicious and 790,745 normal	reported accuracy; 20 selected features; 0% reported FPR and FNR 100%	Direct IoT-botnet relevance; algorithm screening; feature reduction; multiple splits/cross-validation	No external dataset or live deployment; perfect result may be dataset-specific; sequential error propagation	Feature-reduced sequential detector

8. Synthesis of Findings and Research Gaps

The reviewed studies show a clear development in IoT botnet detection methods. The basic DNN study shows that deep learning can detect abnormal traffic. BotIDS improves multiclass accuracy and prediction speed using CNN. SMOTE-DRNN shows that minority attacks must be considered and that accuracy should not be used alone. ELBA-IoT shows that an ensemble model can provide high accuracy with low prediction cost. The early-detection study adds system calls and device behavior to detect malware before it completes its activity. Rezaei's ANN-DT model also reduces the number of features, but its perfect result still needs to be confirmed using other datasets.

None of the reviewed methods satisfies all practical requirements at the same time. The models with the highest accuracy were mainly tested on controlled datasets. The method that handled class imbalance most carefully required more training time. The fastest ensemble was not tested on a real gateway. The early-detection method used richer data, but its data collection process was slower. The ANN-DT ensemble reported perfect results, but it was not tested on another dataset or in a live IoT environment.

8.1 Research Position Derived from the Review

The main conclusion of this review is that moving from traffic classification to early intervention cannot be judged by accuracy alone. A stronger practical model should still detect minority attacks, fit within gateway limits, produce a decision before major damage occurs, and remain useful when traffic patterns change. From this perspective, SMOTE-DRNN gives the strongest evidence for reliable minority-class detection. ELBA-IoT is the clearest lightweight option, while BotIDS provides a strong high-speed deep-learning benchmark. The collaborative sandbox model is the only paper that clearly moves beyond classification toward early intervention. Rezaei's perfect in-dataset result is useful when discussing feature reduction, but it needs external validation before its transferability can be trusted.

8.2 Identified Research Gaps

- Cross-dataset and temporal validation: Most models are evaluated on random splits of one dataset rather than on traffic collected at a different time or organization.
- Unknown botnet-family detection: Performance against attacks excluded from training is rarely measured systematically.
- End-to-end resource evaluation: Inference time is sometimes reported, but memory use, energy consumption, preprocessing time, and gateway throughput are not consistently evaluated together.
- Class imbalance under real traffic: SMOTE-DRNN addresses imbalance effectively, but oversampling methods should be compared with cost-sensitive learning, focal loss, and generative approaches.
- Robustness against changing and adversarial traffic: The reviewed models should be tested under concept drift, evasion attempts, poisoned data, and changes in device behavior to determine whether their performance remains stable over time.
- Early and lightweight detection in one framework: Existing studies separately address early detection and lightweight inference. An integrated approach remains needed.

- Explainability and operational response: The papers focus on classification performance but provide limited explanations that would help analysts understand why a flow or device was marked malicious.

8.3 Proposed Direction for Future Research

Based on the reviewed studies, a useful future system could use a lightweight ensemble model at the IoT gateway as the first detection stage. More complex traffic could then be sent to a deeper model for additional analysis. The evaluation should use class-sensitive measures and should include unseen botnet families, time-separated data, and tests on real gateway hardware. This layered design could combine the speed of ELBA-IoT, the minority-class performance of SMOTE-DRNN, the early evidence used by the collaborative detection model, and the feature reduction used in Rezaei's ensemble.

8.4 Methodological Quality Assessment

According to the quality criteria in Section 2.2, SMOTE-DRNN gives the strongest analysis of class imbalance because it uses several suitable measures and shows how an unbalanced model can fail on a rare attack. BotIDS gives the clearest explanation of preprocessing and model comparison. The early-detection study gives the strongest evidence about detection timing and testing on unseen samples. ELBA-IoT gives the clearest result for prediction cost. Rezaei compares the largest number of algorithms and reduces the features to 20, but the reported 100% accuracy needs independent confirmation. The basic DNN paper provides a useful starting point, although it gives less evidence about practical deployment.

8.5 Direct Cross-Paper Comparison

- BotIDS versus SMOTE-DRNN: BotIDS reports slightly higher headline accuracy, but SMOTE-DRNN provides more reliable evidence for rare classes. Their results are not directly comparable because preprocessing, model objectives, and class-balance treatment differ.
- ELBA-IoT versus deep models: ELBA-IoT reaches 99.6% accuracy with about 40 microseconds of inference, suggesting that deep learning is not automatically the best deployment choice when gateway latency and power are central constraints.
- Rezaei versus ELBA-IoT: Both use ensemble ideas, but Rezaei combines ANN and DT sequentially and reports 100% accuracy using 20 features, whereas ELBA-IoT evaluates decision-tree ensemble families and reports 99.6% multiclass accuracy with explicit inference overhead. ELBA-IoT therefore provides stronger evidence for lightweight deployment, while Rezaei provides stronger feature-reduction evidence.
- Early detection versus traffic classification: The early-detection framework is operationally proactive because it uses partial execution behavior, but its sandbox collection process is much slower than packet-level models. The difference reflects a trade-off between early contextual evidence and high-throughput inference.
- Shared-dataset limitation: Three papers rely on Bot-IoT. Agreement among their high results strengthens evidence that Bot-IoT attacks are learnable, but it does not independently demonstrate performance on different networks.

8.6 Threats to Validity

There are several limitations that affect the overall conclusions. First, the studies used different datasets, class definitions, hardware, and training-test splits. For this reason, their accuracy values should not be treated as a direct ranking. Second, random splitting may place very similar records in both the training and testing sets. Third, most experiments were offline and did not test issues such as packet loss, encrypted traffic, concept drift, evasion attacks, or gateway workload. Finally, some limitations were identified from the experimental design even when the original authors did not test them directly.

Taken together, the six papers satisfy the review questions by showing that predictive performance depends not only on model architecture, but also on class balance, feature design, validation strategy, and deployment constraints. The strongest overall evidence favors a layered IoT defense: a lightweight ensemble for rapid gateway screening, an imbalance-aware deep model for uncertain multiclass cases, and behavioral analysis for suspicious executables. This synthesis is supported by the comparative findings rather than by any single reported accuracy value.

9. Conclusion

This review compared six machine learning approaches for IoT botnet detection. Overall, the studies show that both network traffic and behavioral data can support accurate detection. Still, the model with the highest accuracy is not necessarily the best practical choice. A reliable solution also needs balanced evaluation, fast prediction, early warning, and testing on more than one dataset.

SMOTE-DRNN provides the most convincing treatment of imbalanced multiclass traffic, BotIDS and ELBA-IoT provide the strongest inference-speed results, and the collaborative early-detection method offers the clearest attempt to intervene before complete malicious execution. Rezaei's ensemble offers the most aggressive feature reduction and reported perfect classification, although its generalization remains uncertain. A practical future solution should combine these strengths rather than depend on one architecture.

Overall, the papers suggest a gradual move from accurate traffic classification toward earlier security action. Classification models are useful for fast screening, while the collaborative behavioral approach shows that partial execution evidence may allow action before the attack is complete. At the same time, laboratory accuracy is not enough to show that a model is ready for deployment. Future work should measure generalization, processing cost, memory and energy use, response time, and whether earlier detection actually reduces damage.

The main contribution of this review is the distinction between good prediction results and actual deployment readiness. This helps explain why the model with the highest numerical score may still not be the most suitable option for a real IoT environment.

References

- Abu Al-Haija, Q., & Al-Dala'ien, M. (2022). ELBA-IoT: An ensemble learning model for botnet attack detection in IoT networks. *Journal of Sensor and Actuator Networks*, 11(1), 18. <https://doi.org/10.3390/jsan11010018>
- Idrissi, I., Boukabous, M., Azizi, M., Moussaoui, O., & El Fadili, H. (2021). Toward a deep learning-based intrusion detection system for IoT against botnet attacks. *IAES International Journal of Artificial Intelligence*, 10(1), 110-120. <https://doi.org/10.11591/ijai.v10.i1.pp110-120>
- Jithu, P., Shareena, J., Ramdas, A., & Haripriya, A. P. (2021). Intrusion detection system for IoT botnet attacks using deep learning. *SN Computer Science*, 2, 205. <https://doi.org/10.1007/s42979-021-00516-9>
- Rezaei, A. (2021). Using ensemble learning technique for detecting botnet on IoT. *SN Computer Science*, 2, 148. <https://doi.org/10.1007/s42979-021-00585-w>
- Nguyen, G. L., Dumba, B., Ngo, Q.-D., Le, H.-V., & Nguyen, T. N. (2022). A collaborative approach to early detection of IoT botnet. *Computers & Electrical Engineering*, 97, 107525. <https://doi.org/10.1016/j.compeleceng.2021.107525>
- Popoola, S. I., Adebisi, B., Ande, R., Hammoudeh, M., Anoh, K., & Atayero, A. A. (2021). SMOTE-DRNN: A deep learning algorithm for botnet detection in the Internet-of-Things networks. *Sensors*, 21(9), 2985. <https://doi.org/10.3390/s21092985>