A Bayesian-optimized Ensemble Framework for Intrusion Detection Using Hybrid Feature Selection

S Phani Praveen [1], Sreedhar Bhukya [2], N. S. Koti Mani Kumar Tirumanadham [3], Deshinta Arrova Dewi [4], and Tri Basuki Kurniawan [5,*]

[1] Department of Computer Science and Engineering,
Prasad V Potluri Siddhartha Institute of Technology, Vijayawada, India
[2] Department of Computer Science and Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India
[3] School of Computer Science and Engineering, VIT-AP University, Amaravathi, India
[4] Center for Data Science and Sustainable Technologies, INTI International University, Nilai, Malaysia
[5] Post Graduate Program, Universitas Bina Darma, Palembang, Indonesia
Email: sppraveen@pvpsiddhartha.ac.in (S.P.P.); sreedhar.b@sreenidhi.edu.in (S.B.);
tirumanadham.koti@vitap.ac.in (N.S.K.M.K.T.); deshinta.ad@newinti.edu.my (D.A.D.);
tribasukikurniawan@binadarma.ac.id (T.B.K.)
*Corresponding author

Abstract—**The rapid growth of network connectivity has increased the frequency and complexity of cyberattacks, placing heavy demands on Intrusion Detection Systems (IDS) to detect evolving threats accurately and efficiently. Traditional IDS models often suffer from high false-positive rates, poor generalization, and limited ability to process high-dimensional and imbalanced network traffic. To address these deficiencies, recent studies have explored machine learning and ensemble-based IDS solutions; however, they commonly lack robust hybrid feature selection, effective handling of class imbalance, and advanced hyperparameter optimization. These gaps limit detection accuracy and scalability in real-world environments. This paper proposes Boruta-Logistic-Extra Trees-Xensemble (BLEX), a modular IDS framework integrating a hybrid feature selection method Boruta + Mutual Information (BOMI) and a Bayesian-optimized stacking ensemble (BATO-tuned ExtraTrees, CatBoost, and Logistic Regression). The framework incorporates Interquartile Range (IQR)-based outlier removal, Borderline-SMOTE for class balancing, and probabilistic hyperparameter tuning using Tree-structured Parzen Estimators. Simulation results on a large Kaggle network-traffic dataset demonstrate the effectiveness of the proposed framework, achieving 97.28% accuracy, 93.49% precision, 91.88% recall, 94.23% F1-Score, and a low Root Mean Square Error (RMSE) of 0.2592. These results highlight the improved generalization, scalability, and robustness of the proposed BLEX model for modern, high-dimensional intrusion detection.**

Keywords—**Intrusion Detection System (IDS), Boruta-mutual information, Bayesian optimization with Tree-structured Parzen Estimator (TPE), social protection, vulnerable, process innovation**

I. Introduction

The ever-growing digital interconnectedness of the modern world has seen the frequency, complexity, and sophistication of cyberattacks increase to severe levels and threaten to compromise the integrity, confidentiality, and availability of sensitive information on networked systems. Intrusion Detection Systems (IDS) are very important within a second layer of defence since they check on the network traffic with an aim of detecting malicious activity and evaluating policy violations. Nevertheless, conventional IDS methods have a reputation for high false positive rates, poor performance, and detection of newly developed attacks and scalability issues when they have to process multidimensional traffic data [1]. Such issues require the creation of intelligent, adaptive, data-driven detection that is able to consume large amounts of changing network information in a more precise and more effective way.

This paper suggests a new machine learning-based intrusion detection framework called Boruta-Logistic-Extra Trees-Xensemble (BLEX) and implements two major innovations into the framework: a hybrid feature selection method (Boruta + Mutual Information, BOMI) and an ensemble classification method with an improved strategy and hyperparameter tuning using streams of Bayesian Optimization Parzen Estimator with Tree-structured (BATO) [2]. BOMI is the integration of Boruta and Mutual Information that takes advantage of both the interpretability of Boruta and the statistical dependency power of Mutual Information to discard redundant and irrelevant features and reserve the predictive features. This

Manuscript received October 9, 2025; revised November 18, 2025; accepted December 19, 2025; published August 26, 2026.

not only saves computation overhead but also decreases the chance of overfitting. At the same time, the ensemble learning model combines Extra Trees and CatBoost classifiers; the hyperparameters will be fine-tuned in BATO to approximate the decision boundaries, thus guaranteeing the reliable detection of different types of intrusions [3].

When fitted against a large real-world network traffic dataset, the proposed BLEX framework yielded a high level of accuracy, precision, recall, F1-Score, and Root Mean Square Error (RMSE) compared to several of the proposed baseline models. This is evidenced by the development of the methodological contributions, namely, hybrid feature selection and intelligent ensemble optimization, which makes the framework suitable towards future deployment in open environments like dynamic and high-dimensional (e.g. cloud computing, IoT, and SDN-based structures). Thus, the study presents a scalable and effective IDS solution that aligns with the evolving cybersecurity demands of modern digital ecosystems. Thus, the study presents a scalable and effective IDS solution that aligns with the evolving cybersecurity demands of modern digital ecosystems.

As much as the research has laid bare the developments in IDS, there still stand other research gaps that will be directly filled in, courtesy of the proposed BLEX framework. Past studies have concentrated on single classifiers (e.g. K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Deep Neural Networks (DNN)), or simple ensembles with little to no feature engineering or hyperparameter optimization. Kumar *et al.* [4] have used dimension reduction through PCA without using built-in feature importance, and Sarkar *et al.* [5] focused on general tuning of the ensemble, but did not include extensive hybrid feature selection. In addition, the application of behind-time or simpler datasets, such as KDD-99 and NSL-KDD, results in poor generalizability to contemporary and high-dimensional network settings. Srinvas *et al.* [6] investigated deep learning, but they had not introduced hybrid selection or sophisticated tuning ideas, which afforded flexibility. Moreover, the strategy of Bose *et al.* [7] was SDN-oriented and focused on deep learning technology and neglected the lightweight capability of the IDS execution in general. Alternatively, the given methodology proposes a new hybrid technique of feature selection (BOMI) based on Boruta and Mutual Information and an enhanced variety (BLEX) based on Bayesian hyperparameter tuning (BATO) to handle both significance of features and generalization of models, thus bridging the essential extents of scalability, generalization and detection accuracy in dynamic intrusion environment.

To address the identified research gaps, this paper provides an in-depth, well-founded framework, BLEX—that promotes intrusion detection with a synergetic combination of hybrid feature selection and hyperparameter-adjusted ensemble modeling. As an addition to the above methodology, I will be proposing a new approach to hybrid feature selection termed BOMI, built on the ideas of both Boruta and Mutual Information features selection, to allow me to extract highly relevant and non-redundant features (of the highest quality) on the data in my high-dimensional network traffic data. This is in response to the weaknesses of the earlier research that either dimension reduction methods were/are semantically irrelevant or that they did not incorporate a stringent feature selection process. In addition, the paper proposes BATO, which is a Bayesian Optimization approach based on Tree-structured Parzen Estimator, and it optimizes the parameters of the Extra Trees and CatBoost classifiers integrated. This makes sure that the ensemble called BLEX has the best performance, that is achieved with adaptive learning and generalized decision boundaries. In combination, BOMI and BATO present a very precise, low-cost and scalable IDS with a strong base in absurd breadth and depth that can process applications in the context of real networks. The contributions are confirmed through the experimental results showing impressive gains in levels of the type of classification performances compared to other existing models and proving that the framework can be realized in a contemporary environment of cybersecurity spectrum.

II. Literature Review

Kumar *et al.* [4] suggested an IDS in 2020, which uses supervised machine learning algorithms, able to recognize normal and malicious data packets to enhance the security of a network. Since interchange of data on a network in this contemporary world is a requirement, security during transfer of such data between these networks becomes of extreme importance in view of the possibility of internal and external intruders. Kumar *et al.* [4] have used the KDD-99 dataset which is considered to be one of the benchmarks in the field of research in IDS; this dataset has 32,640 samples, 12,440 of which are normal whereas 20,200 as the attack samples. It possesses balanced training set as compared with the test set. In supervised learning, the use of SVM and KNN models as classifiers, PCA as dimension reduction to remove the data and to maximize the efficiency of the classifier was conducted. The approach PCA-KNN produced the accuracy of 90.07% on a distance metric of cosine and principal component (pc) value of 5 on both models tested. The findings provided give an illustration of how the dimensionality reduction methods can enhance the performance of the supervised machine learning models in the context of IDS: the examples of PCA and KNN complementing each other then suggest a remarkably precise solution to the problem of normal and attacking data packets identification in network traffic.

In 2022, Sarkar *et al.* [5] presented a superior machine learning ensemble approach to enhance the effectiveness of network IDS besides the detection accuracy. The work, it appeared, was dedicated to the fact that tuning hyperparameters and preprocessing of data can contribute to the enhancement of the abilities of the detection, particularly, the abilities of the detection of rare kinds of attacks, which are not easy to detect usually root-to-local (R2L) and Root (U2R) attacks. The rebalancing procedure was carried out on two commonly used datasets KDD Cup99 and NSL-KDD since a common issue of intrusion

detection is related to class imbalance which was addressed through the data augmentation. The cascade, meta-specialized classes of these classifiers consisting of MLPs will offer varying layers of training to categorize select attack classes, improvement in the classification accuracy besides figuring out the false positives. On one of the datasets, they got a detection accuracy of 89.32% and FPR as 1.95 and same analysis when applied on the NSL-KDD dataset gave an accuracy of 87.63% and the FPR as resulting to 1.68. This ensemble method increased performance significantly in detection by assigning larger weights to those algorithms whose performance was superior and introduced the possibility of future application of hyperparameter-optimized ensemble methods to construct more consistent and able to work IDS than conventional models.

Srinivas *et al.* [6] suggested a new machine learning-driven algorithm that can be applied to real-time network-based intrusion detection tasks. New types of attacks exist constantly, so generally, traditional IDS is not able to identify complex attacks in real-time primarily because of their non-generalization to varied forms of attacks. The known list of machine learning algorithms is long, including Decision Trees, Random Forests, SVM, KNN and, even the DNN, which have generated many expectations regarding classification and network intrusion prediction. In their research study, Srinivas *et al.* [6] also thought about applying a variety of ML models that comprised such models as Logistic Regression, Naive les, Decision Trees, and DNN with high-dimensional data sets. Their testing performance was more than 92% with these kinds of models, such as Random Forest and Decision Trees, with high values of precision and recall. As in the case of deep learning models, the DNN with multilayered results demonstrated an extremely strong result- a five-layer DNN achieved a 95.5% F1-Score with an accuracy of 93.2%. The significance of the study was rather high on the aspects of the application of integration of various machine learning methods in recognizing known and unknown attacks in an online, real-time manner. The implications of the findings as regard to this study area are that combining conventional ML with deep learning could be effectively used to boost the performance of IDS and improve its efficiency in dealing with complex, dynamic network traffic.

In Bose *et al.* [7], the IDS is to target an adaptable and scalable form of security by 2024 by proposing a robust multi-layered security structure in an SDN setting. IDS contains conventional models that are problematic in terms of accuracy and scalability because of the challenging element of SDN traffic and its rapidly evolving nature. As a rule, NSL-KDD datasets were tested in general. To address such issues, authors proposed a model called BAT-MC that combines Bi-directional Long Short-Term Memory networks, attention mechanism, and several convolutional layers to process the context in both directions without a single feature engineering. Consequently, this strategy has streamlined the traffic examination of the SDN infrastructures to detect the minor anomalies in real time. In this situation, it was shown, on the In-SDN dataset by Bose *et al*. [7], tailored deliberately to attend to all of such different intrusion forms and traffic shapes within an SDN-that has an extensively impressive performance advantage. Combined with ensemble techniques, BAT-MC delivered an accuracy of 86%, indicating a great potential for high-definition anomaly detection and high scalability. This paper gives an understanding of how deep learning methods, in the present case BiLSTM along with attention tiers, can be applied in IDSs, and thus, in what sense, this approach offers a new solution that suits to address the challenging security issues of dynamic SDNs.

III. Proposed Methodology

The study provided under the proposed methodology shown in Fig. 1 proposes a modular end-to-end machine learning pipeline capable of constructing an efficient and scalable IDS to suit contemporary network setups. This will start with a process of acquiring a relatively varied and complete network traffic data set from Kaggle with a variety of normal and abnormal behavioral patterns. Data is pre-processed so that the quality of the data is maintained, and data contains no missing values, and outliers are determined in multivariate data through the Interquartile Range (IQR) [8]. This method defines an outlier as a value, which is beyond the range given by 1.5 times the IQR by calculating the first and third quartile (Q1 and Q3) and removing the outlying values beyond these boundaries, so it is a strong, non-parametric statistical approach to eliminate outlying values, and it does not require an underlying distribution. To cope with the problem of class imbalance, the Borderline-SMOTE [9] technique is used, which creates pseudo-instances of the minority classes close to the decision border, and, in that way, increases the sensitivity of the model and minimizes the bias of classification. It is used in the process of selecting features called BOMI, a hybrid of the wrapper-based Boruta algorithm [10] and Mutual Information [11] filter-based criterion that allows both the deletion of features with insignificant statistical measures and those irrelevant to the model learning. To optimize hyperparameters, BATO, a probabilistic search algorithm, is used within the framework, and it is efficient in exploring the hyperparameter space to find high-performing configurations. The last classification model, referred to as BLEX, is based on a stacked ensemble structure that uses ExtraTrees [12], CatBoost, and Logistic Regression models and applies stacked generalization to complex data patterns with robustness and interpretable properties. Comprehensively, the intended pipeline encompasses statistical preprocessing, refined balancing, hybrid feature selection as well as ensemble learning to present a tremendously powerful solution to the intrusion detection issue in high-dimensional and dynamic networks.

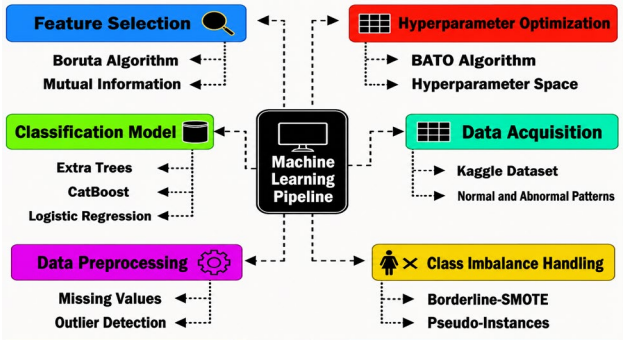

Fig. 1. Proposed methodology.

A. Data Collection

This analysis draws its data from a very thorough cybersecurity dataset that was published on Kaggle and records 75 different parameters of network and session-level characteristics. The values recorded of each of these characteristics are the session durations (dur) the type of protocol used (proto), the type of service (service) and packet level values of sent and received packets (spkts, dpkts), volumes of data in sbytes and dbytes. The metrics such as session load (sload, dload), packet rates (rate) and TCP-related fields are likely to assist in a better identification and detection of the anomalous behaviour of the network. One more significant area is the information concerning the session states (state), Time-to-Live values (sttl, dttl) and other session characteristics that are rather important in the process of separating the normal and suspicious network traffic. Known intrusion instances are labeled with classes of attacks (attack_cat) and binary labels (label), hence making the dataset suitable in the development of supervised learning models. The instances of attack labels and detail in the dataset form a great training and validation base of IDS models, which allows high classification of an intruding type of network.

B. Data Cleaning

This step of cleaning the data involved inspection of the data set and preparation. Records: 175341 Attributes: 45. Every detail is discussed in order to be complete and correct. Some initial tests were set to see whether there exist any missing values in any column of the data set, and as a result it was proven that none is to be found and no data imputation or row deletion will be needed prior to analysis. This data contains integer, floating point and categorical objects. In this way, all the features, including those of the packet counts (spkts, dpkts) and the session metrics (dur, rate) are compared with the desired value. This also helps that the integrity and concurrence of the data type remains favorable with the algorithms being handled in machine learning. Such object types as the proto and attack_cat were maintained so as to allow their easy encoding at a later date. The verified and clean dataset will guarantee that the further analysis and training of models will be received on the basis of valid data with high qualities, i.e., this step will introduce the further possibility of correct understanding of the results of intrusion detection tasks.

1) Dataset limitations and mitigation strategies

Although the Kaggle network traffic dataset used in this study provides diverse session-level and flow-level attributes, it exhibits several inherent limitations such as restricted dataset size, high variance across attack categories, elevated error rates in the raw test split, and inconsistencies in data quality due to noisy or unstable traffic behaviors. To ensure robustness and generalizability of the proposed BLEX framework, multiple mitigation strategies were systematically integrated. First, an Interquartile Range (IQR)-based outlier removal process was applied to eliminate extreme or inconsistent values, reducing variance and preventing model overfitting on noisy samples. Second, to address the limited dataset size and imbalance between normal and minority attack categories, Borderline-SMOTE was used to generate synthetic minority samples near decision boundaries, thereby enhancing classifier sensitivity to rare intrusions. Third, to reduce dimensional noise and redundancy, the BOMI hybrid feature selection method (Boruta + Mutual Information) was employed to retain only statistically significant and model-relevant attributes, resulting in a cleaner and more stable learning space. Fourth, to prevent error propagation during model training, Bayesian Optimization with Tree-Structured Parzen Estimator (BATO) was adopted to identify optimal hyperparameters while accounting for fluctuations caused by dataset imperfections. Finally, the stacked ensemble architecture (ExtraTrees, CatBoost, and Logistic Regression) further enhanced generalization capability by combining diverse learning behaviors and minimizing dependency on dataset-specific artifacts. Collectively, these measures ensure that the proposed IDS model remains reliable, scalable, and resilient despite the intrinsic limitations of the Kaggle dataset.

2) Key parameters used in data preprocessing

The following key parameters were applied during data preprocessing to ensure consistency, quality, and reproducibility:

a) Outlier removal (IQR method)

(1) Lower threshold: $Q1 - 1.5 \times IQR$;
(2) Upper threshold: $Q3 + 1.5 \times IQR$;
(3) Applied to all continuous numerical features.

b) Class balancing (borderline-SMOTE)

(1) k_neighbors = 5;
(2) m_neighbors = 10;
(3) Borderline type: Type-1;
(4) Oversampling applied until minority classes reached balanced representation.

c) Categorical encoding

(1) One-Hot Encoding for low-cardinality categorical features;
(2) CatBoost's built-in encoder used for high-cardinality attributes.

d) Normalization

(1) Method: Min–Max scaling;
(2) Scaled output range: [0,1];
(3) Applied only to numerical attributes.

e) Feature selection (BOMI)

(1) Boruta parameters:

- n_estimators = 500;
- alpha = 0.05.

(2) Mutual Information:

- Top 20 features selected based on Mutual Information (MI) scores.

These parameters collectively ensure clean, balanced, and well-structured data for the BLEX intrusion detection framework.

C. Handling Imbalanced Dataset

Imbalanced dataset is a typical problem of machine learning. Models bias towards the majority race since they do not work well with the minority race. One of the popular methods to solve this problem is Synthetic Minority Oversampling Technique also known as Borderline-SMOTE shown in Fig. 2. Borderline-SMOTE [13, 14] creates artificial samples of the minority class rather than copying the existing ones. It randomly chooses a sample of the minority class, and determines its k-nearest neighbors, and generates new synthetic samples by interpolation in the space between the sample and k-nearest neighbors of the sample. The fact that it has greater minorities-class representation enables a model to learn stronger patterns. Duplicating overfitting samples is unavoidable because of synthetic data points produced. Although a good method, Borderline-Borderline-SMOTE can add noise in the case of overlapping classes or in sparse regions as depicted in Fig. 2. Nevertheless, when Borderline-SMOTE is used accurately, the general performance of the model will be improved, particularly in the imbalanced classification issue and the model will guarantee that both the majority and minority classes, have a balanced number of representations.

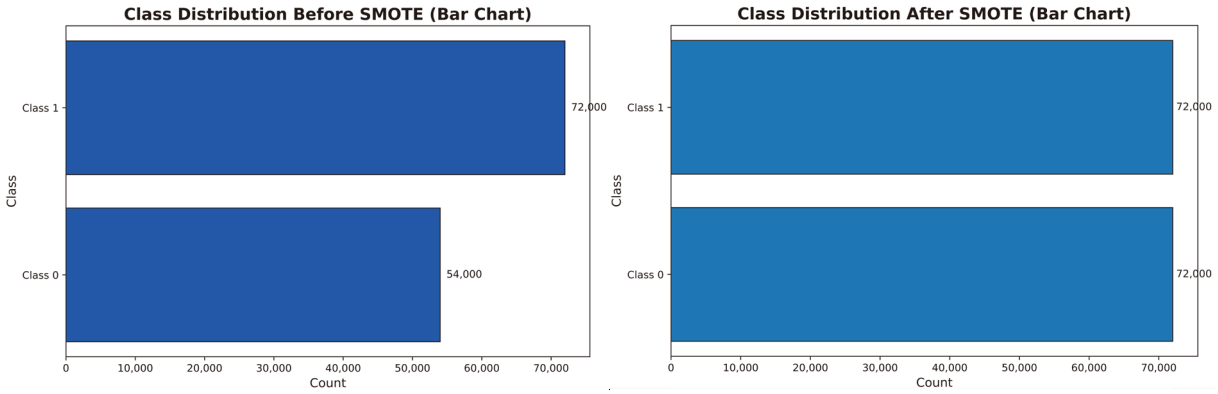

Fig. 2. Before and after applying SMOTE.

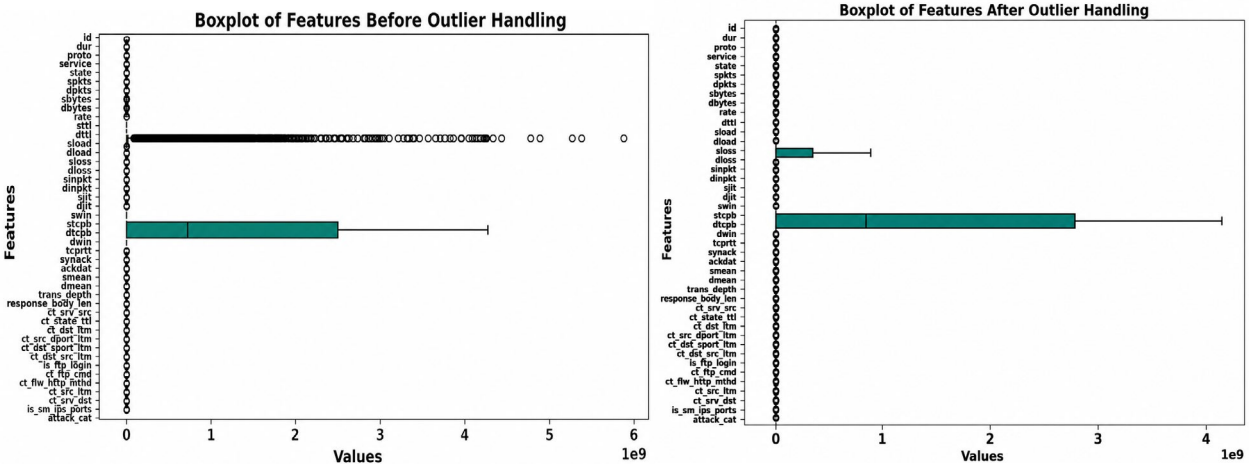

Fig. 3. Handling outlier using before and after applying IQR.

Outliers refer to those data points that differ a lot with most observations and have a negative effect on statistical analyses and machine learning performances. A common powerful approach to identify outliers would be the Interquartile Range (IQR) technique [14] that works especially well with skewed or data not following a normal distribution. The IQR is described to be the spread or the difference between the third and first quartile (Q3–Q1) giving the mid 50% of the information. Usually, the outliers are determined by determining counts under $Q1 - 1.5 \times IQR$ or over $Q3 + 1.5 \times IQR$. These levels assist in the isolation of extreme values without any presumptions concerning the distribution of data shown in Fig. 3. IQR method has a number of strengths because it is an easy method and not affected by outliers themselves. Once identified, outliers may be deleted, transformed, or treated as a function of the aspects of an analysis and the information that they may convey. Although getting rid of

outliers can improve the accuracy of the model, there are chances that they might be eliminating events that could be meaningful or could be few in number. one should be very cautious when dealing with such data points so that the integrity and richness of the data set is maintained (Fig. 3). Thus, IQR method is a statistically-based and pragmatic way of dealing with outliers especially in the exploratory data analysis and preprocessing phases of machine learning pipelines.

D. Feature Selection Using BOMI

The feature selection process is critical in boosting the overall output and generalizability of an intrusion detection system to realize enriched performance in a network scenario with large dimensionality in network traffic. Overfitting, increasing computational complexity, and decreasing model interpretability may occur due to the presence of irrelevant features, redundant features or noise. So as to solve these issues, a hybrid feature selection technique that refers to as BOMI has been purposed in this paper and combines the characteristics of wraper-type feature selection and filter-type feature selection strategies. Boruta algorithm is a wrapper technique that is constructed on the Random Forest, repeatedly assessing importance of features in comparison to the randomized version of the same. Boruta algorithm eliminates all useless features, and just the significant features prevail. In complement, Mutual Information is also used as a filter measure, which senses the statistical dependency between every feature and the target variable, thereby helping in showing features with high predictive possibilities. The BOMI strategy combines the advantages of Boruta, its model-based selection, with such the statistical analysis-oriented mechanism of the feature set selection as the Mutual Information that allows delivering a more robust and balanced feature set selection, enhancing classification accuracy, minimizing overfitting, and speeding up the training process. Such two-layered solution has a specific advantage in the in high dimensional, dynamic intrusion detection settings, where the relevance as well as statistical strength is essential to measure features.

E. Boruta Algorithm

Boruta algorithm is a very powerful wrapper-type of feature selection algorithm whose task is to select all of relevant features in relation to a certain variable. Constructed as an extension of the Random Forest classifier, Boruta follows a shadow-features approach, or more precisely, the original features are shadowed by duplicate representations of the original features which are randomly shuffled and then a Random Forest model is trained as a baseline using both original and shadow features. The algorithm determines the scores of feature importance and comparing the importance of each of the original features to the maximum importance of the corresponding shadows. When some of the most desirable shadow features are all the time found more crucial than the key features, these are identified as being of importance, otherwise, not. This is re-iterative and Boruta [15] is capable of covering the interactions and non-linear relationships amongst features. Boruta is not a greedy feature selector like some other greedy feature selectors, it is instead an all-relevant method, i.e. would be interested in identifying all features that hold any quantity of information, rather than a minimal optimal subset. This is especially convenient in areas such as network intrusion detection, where, even weak predictors can be helpful in combination with other predictors.

F. Mutual Information

Mutual Information (MI) is a filter based algorithm of feature selection and measures the information a variable in a set of variables holds about another. With respect to feature selection, MI quantifies the relationship between every input feature and the target class, irrespective of the linearity of such dependency. The greater the level of MI, the more dependent and, therefore, relevant to the predictive task. Unlike correlation-based techniques, whose limits are in assuming the linear relationship between the variables, Mutual Information can capture non-linear and multidimensional relation between the variables. The MI has the great advantage of being a computationally efficient method operating in a nonassumptive manner in a high-dimensional dataset, which makes it particularly advantageous. Mutual Information is employed in this research to ascertain the ranking of features under which features are ranked according to their relevance and this forms a complement to the Boruta algorithm. Boruta concentrates on model-based importance in the framework of feature interactions whereas Mutual Information goes to the extent of making sure that statistically significant features are selected making the selection process very robust.

G. Model Building using BLEX

The intrusion detection framework suggested mostly revolves around the model-building stage of building a strong, arbitrary fusion model of classification described as Boruta-Logistic-ExtraTrees-CatBoost Ensemble (BLEX) as illustrated in Algorithm 1. Such an ensemble is created to utilise the specific advantages of three parallel models of machine learning: specifically, Logistic Regression, ExtraTrees Classifier, and CatBoost. Logistic Regression has good interpretability and has applicable use on linearly separable data, making a stable baseline model. Another ensemble is Extremely Randomized Trees (ExtraTrees), which is similar to Random Forest in that it has many decision trees yet provides high variance reduction using randomized feature splits and bagging, and thus obtains complex patterns at the expense of few overestimates. CatBoost is a gradient boosting optimization that is optimized to categorical features yet specified that can be used in ordered boosting; CatBoost is useful in providing good results on non-linear data and is efficient in processing high-cardinality variables. These models get combined using a stacked generalization method, in which base learners (ExtraTrees and CatBoost) provide their initial predictions, which are then returned to a meta-learner (Logistic Regression) which is trained to combine the outputs provided by the base learners in order to give an additional prediction. The arrangement of the hierarchy of such an ensemble increasingly enables BLEX

to apply linear and non-linear relationships, increase generalization and be less vulnerable to noise and overfitting. Each model has hyperparameters that are optimized through Bayesian Optimization with Tree-structured Parzen Estimator (BATO) which is an automatic method to adjust the hyperparameter to the optimal point without much manual exploration of the navigational space. BLEX ensemble results in better classification results in network intrusion detection tasks in high and dynamic dimensions when trained on the optimized and the BOMI-selected feature.

Algorithm 1. BLEX building pseudocode

Input:

- X_{selected}: Feature-selected dataset from BOMI
- y: Corresponding class labels
- Key parameters:
  - Number of cross-validation folds K
  - BATO iterations N
  - Search spaces S_{ET}, S_{CB}, S_{LR} for ExtraTrees, CatBoost, and Logistic Regression

Output:

- Final trained BLEX ensemble model: $\{ET_{\text{best}}, CB_{\text{best}}, LR_{\text{meta}}\}$

```
Step 1: Dataset Splitting
  1.1 Perform stratified split on (X_selected, y)
    → (X_train, X_test, y_train, y_test)
Step 2: Initialize Base Models
  2.1 ET ← ExtraTreesClassifier()
  2.2 CB ← CatBoostClassifier()
Step 3: Hyperparameter Optimization using BATO
  For each model M ∈ {ET, CB}:
    3.1 Define search space S_M
    3.2 For iteration i = 1 to N:
        h_i ← TPE_Suggest(S_M)
        score_i ← CrossValidate(M, h_i, X_train, y_train,
K)
    3.3 Select best parameters:
        h_best ← argmax(score_i)
    3.4 Train optimized base model:
        M_best ← Train(M, h_best, X_train, y_train)

  Output: ET_best and CB_best

Step 4: Generate Out-of-Fold (OOF) Predictions for Stacking
  4.1 Initialize Z_train as empty
  4.2 For each fold k = 1 to K:
      - Train ET_best on training folds: ET_k
      - Train CB_best on training folds: CB_k
      - Predict on validation fold:
          p_ET ← ET_k.predict(X_val)
          p_CB ← CB_k.predict(X_val)
      - Append predictions to Z_train
Step 5: Train Meta-Learner
  5.1 LR_meta ← LogisticRegression()
  5.2 LR_meta ← Train(LR_meta, Z_train, y_train)
Step 6: Final Test Prediction
  6.1 Generate test predictions:
      Z_test ← Concatenate(ET_best.predict(X_test),
                CB_best.predict(X_test))
  6.2 Compute final outputs:
      y_pred ← LR_meta.predict(Z_test)
Step 7: Return Final BLEX Model
  Return {ET_best, CB_best, LR_meta}
```

1) Logistic regression

One of the most popular binary and multiclass linear classification algorithms is the Logistic Regression [16]. It approximates the likelihood that a specified input point falls into a certain one of the classes by the logistic (sigmoid) function of a linear combination of the input features. On mathematical terms, the model predicts probabilities in the formula:

$$P(y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots \beta_n x_n)}}$$

where β_0 is the bias term and $\beta_1, \ldots, \beta_{\text{n}}$ are the model coefficients. Logistic Regression is interpretable and computationally efficient, making it suitable as a meta-learner in stacked ensemble architectures. In the BLEX model, Logistic Regression is utilized to combine predictions from more complex base learners (ExtraTrees and CatBoost), learning optimal weights to minimize classification error across aggregated outputs. Its simplicity and robustness to overfitting under regularization make it an effective decision-level integrator in ensemble systems.

2) ExtraTrees classifier

The ExtraTrees Classifier is an ensemble decision tree algorithm, although it has a similar spirit to Random Forests. Nevertheless, it is more randomized in that it uses random selection of cutoffs on each feature when building the tree instead of picking the most discriminative split. This leads to decorrelated trees and typically achieves a reduction of variance than Random Forests. ExtraTrees uses a mechanism of bagging (bootstrap aggregating), in that a plurality of trees is fitted on different random subsets of the data and the estimates are then averaged (The average is usually through majority vote) to obtain the final projection. Its speed of computation is graceful, and it can work well in high-dimensional spaces and has a good performance on structured information. ExtraTrees [17] in the BLEX ensemble is a strong base learner that can capture complex patterns in the features of the network traffic and match the number of sweeps to the complexity of the problem it is tackling, and increase the generalization ability of the model to previously unseen threats.

3) CatBoost

CatBoost is an efficient gradient boosting algorithm with support for both numerical and categorical features created by the Russian big data company Yandex. Because of this, CatBoost uses an original ordered boosting strategy that blocks target leakage since every data point is taught using previous data only. It also brings about a strong system of representation of a categorical variable in a way that it does not require a lot of preprocessing. CatBoost trains an ensemble of decision trees sequentially, with one tree correcting the error made by the trees before it, and makes use of gradient descent to optimize a preselected loss function. Theoretical features, such as the possibility of capturing even highly complex, non-linear relationships with a minimum of parameter tuning, and resistance to

overfitting, make it in particular suitable to use in intrusion detection. CatBoost [18] as a powerful base learner is used in the BLEX model, as a complement to ExtraTrees, having a gradient-based to the learning process and extracting deep feature interactions in the feature space selected by the BOMI.

4) Hyperparameter tuning

Hyperparameter tuning is a crucial phase in the development of high-performing machine learning models, as it directly influences learning behavior, model complexity, convergence rate, and generalization performance. In this study, we adopt a probabilistic optimization framework known as Bayesian Optimization with Tree-structured Parzen Estimator (TPE) [19], termed BATO, for fine-tuning the hyperparameters of the ensemble constituents in the proposed BLEX model. Unlike traditional grid or random search methods, which are computationally exhaustive and often inefficient in high-dimensional parameter spaces, Bayesian optimization builds a probabilistic model of the objective function and uses it to select the most promising hyperparameter configurations to evaluate next.

The TPE variant models the conditional probability $P(x \mid y)$ rather than $P(y \mid x)$ where x represents a configuration of hyperparameters and y represents the corresponding performance score (e.g., cross-validated accuracy). TPE partitions observations into two groups: one for configurations with the best scores and one for the rest, and then models them using separate kernel density estimators. This allows the optimizer to focus on regions of the search space with high expected improvement. The BATO process iteratively explores and exploits the parameter space for each model—CatBoost, ExtraTrees, and Logistic Regression—ensuring an optimal configuration tailored to the feature space derived via BOMI selection.

By using BATO, the proposed framework not only reduces manual effort and computation time but also improves the ensemble's predictive accuracy, robustness, and stability in diverse and dynamic network environments.

IV. Results and Discussion

A. Feature Selection Using BOMI

Further analysis of the selected features using the BOMI framework revealed that attack_cat, id, and sttl were the most influential variables, with importance scores of 0.654, 0.359, and 0.234, respectively. The prominence of attack_cat confirms its role as a categorical discriminator that encapsulates attack-specific behaviors, while id likely captures session-level uniqueness across traffic flows, and sttl (source time-to-live) reflects network-level packet dynamics crucial for detecting anomalies. Features such as sbytes and dbytes, although included, exhibited lower influence with scores of 0.049 and 0.028, respectively, suggesting limited contribution to the overall classification performance shown in Fig. 4. This insight underscores the strength of the BOMI selection mechanism in isolating a minimal yet high-impact feature subset, reducing model complexity and training time while preserving accuracy. The alignment between feature importance and predictive outcomes further validates the reliability and interpretability of the BLEX ensemble, offering a streamlined and robust approach for real-time intrusion detection in complex networked environments.

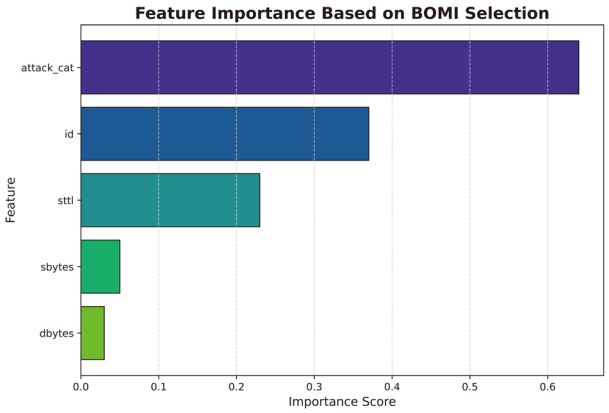

Fig. 4. Visualization for selected features using BOMI.

B. Model Building and Hyperparameter Optimization

The construction of the BLEX ensemble model was strategically designed to optimize performance while maintaining generalization in dynamic and high-dimensional intrusion detection environments. Extra Trees (ET) and CatBoost (CB) were chosen as the base learners based on the complementary properties attributed to them in learning high-variance patterns via randomized splits in ET and strong support of learning categorical and imbalanced data via gradient boosting all over oblivious trees in CB. Further optimization of these base models can be achieved by means of BATO, Bayesian Optimization with Tree-structured Parzen Estimator (TPE) [20, 21], which was used to traverse hyperparameter search space automatically to find configurations that result in best validation performances [22–25]. Compared to the traditional grid or random search, BATO deploys a probabilistic model that estimates the hyperparameter region with the highest potential, and thus minimizes calculations by a huge mechanism and enhances convergence. Extra Trees was tuned in n_estimators, max_features, min_samples_split whereas CatBoost was optimized in depth, learning_rate and l2_leaf_reg. Regularization Tuning (C) was done on the meta-learner (Logistic Regression) to have bias-variance trade- off on the final predictions. The stacking ensemble learning algorithm was used through which the results of tuned base models were taken as input features and the meta-learner was trained. Using this architecture, the model would learn high-level interactions and rectify personal model weaknesses. The performance difference due to tuning and stacking was also substantiated empirically as shown in Table I and Fig. 5 the following comparison of the results of the performances of the tuning and stacking.

TABLE I. IMPACT OF HYPERPARAMETER TUNING AND STACKING ON MODEL PERFORMANCE

Model Variant	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	RMSE
Extra Trees	94.21	89.45	87.12	88.27	0.3721
CatBoost	95.08	90.17	89.63	89.90	0.3449
ET + CB (Without Tuning)	95.61	91.02	90.18	90.60	0.3124
BLEX (with BATO Tuning)	97.28	93.49	91.88	94.23	0.2592

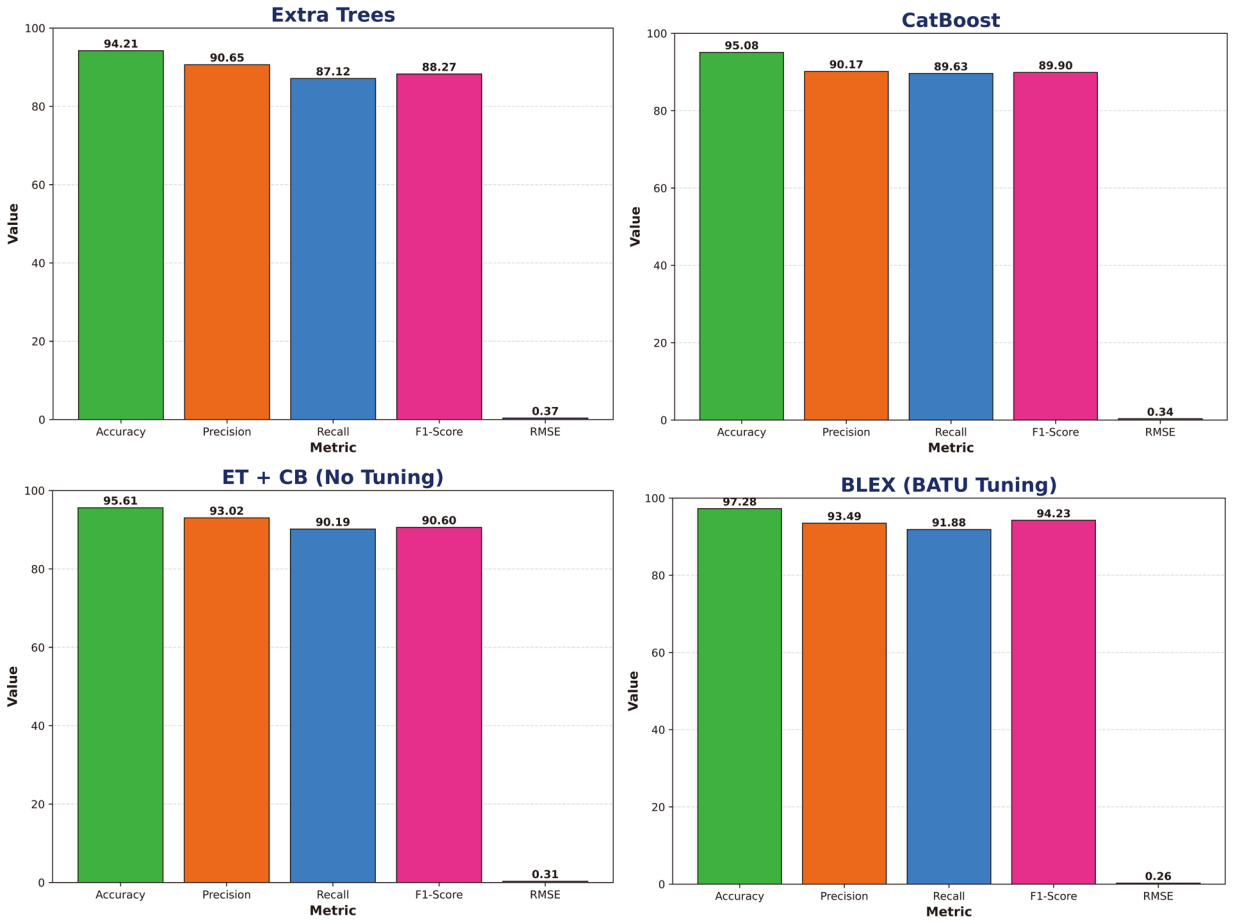

Fig. 5. Performance metric of Individual models.

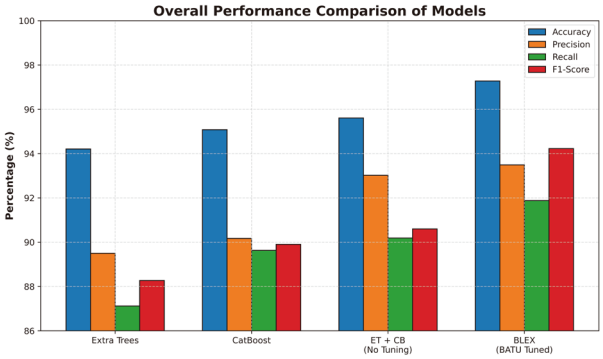

Fig. 6. Performance comparison of overall models.

The tuned BLEX model performed better on every significant measure of evaluation when compared to individual and untuned ensemble in all cases, as indicated in the table. The increase draws attention to the importance of automated hyperparameter optimization on the way to the state-of-the-art performance. Moreover, the architecture of the ensemble is solid and is adaptive and has the capacity to learn complicated relationships involving various traffic incidences and intrusion patterns shown in Fig. 6.

This study indicates that the proposed BLEX framework is effective. This intrusion detection model follows an ensemble-based scheme and combines the use of a hybrid feature selection method (BOMI) and a Bayesian two-stage model, which tunes hyperparameters in a classification ensemble (BATO). This model has been tested with a benchmark dataset of network traffic, and compared to a number of baselines, among which were Extra Trees, CatBoost, and also an untuned ensemble of both. One of the most interesting findings of the research is that the performance of the BLEX model exceeded all the standard measures of evaluation. BLEX has attained an arduous accuracy rate of 97.28%, precision of 93.49%, recall of 91.88%, F1-Score of 94.23%, and low RMSE rate of 0.2592, which is much better than its competitors. These advantages are the outcome of the synergy of the feature selection and model tuning strategies. BOMI which merges Boruta and Mutual Information in feature selection, played a critical role in selecting the most appropriate and discriminative features in the

high-dimensional data. Not only did this impede the computational demands, but it alleviated the complications of overfitting since irrelevant or redundant features were not considered. In addition, an ablation study shows that certain features have higher values of importance, and based on this fact, features like attack_cat, id, and sttl have the highest values of importance. Such characteristics were a good indicator in differentiating classes and could explain why there was an increase in the two parameters, recording the recall and precision scores of classifying various types of attacks.

Altogether, a combination of intelligent feature selection (BOMI), ensemble optimized learning (BLEX), as well as, probabilistic hyperparameter optimization (BATO) would provide a highly effective, scalable and interpretable framework to deliver intrusion detection. This design can be touted to be recommended to be deployed in a real life and high throughput network security applications.

C. Comparison with Existing Models

Existing intrusion detection studies demonstrate notable strengths but also suffer from several critical limitations that restrict their applicability in modern, high-dimensional network environments. For example, Kumar *et al.* [4] focused on PCA-based dimensionality reduction with KNN, achieving reasonable accuracy but lacking any mechanism for feature relevance evaluation, leading to weak generalization in complex traffic scenarios. Sarkar *et al.* [5] introduced cascaded MLP ensembles, yet their method struggled with rare attack categories due to insufficient imbalance handling and produced sub-optimal accuracy (<90%). Similarly, Srinivas *et al.* [6] incorporated classical ML and deep learning models; although their DNN achieved competitive performance, the approach demanded high computational resources and did not include systematic feature selection or hybrid optimization strategies. Meanwhile, Bose *et al.*'s [7] SDN-based BiLSTM-Attention model showcased strong temporal modeling ability but suffered from scalability challenges and dependency on SDN-specific traffic patterns, limiting cross-environment adaptability. In contrast, the proposed BLEX framework directly addresses these shortcomings through a combination of hybrid feature selection (BOMI) to eliminate redundant and irrelevant attributes, Borderline-SMOTE to properly rebalance minority attack classes, and Bayesian hyperparameter optimization (BATO) to robustly tune ensemble components without manual intervention. Furthermore, BLEX utilizes a stacked ExtraTrees–CatBoost ensemble, enabling improved detection of nonlinear, high-variance patterns while maintaining computational efficiency. As a result, BLEX overcomes the major limitations of prior models—including lack of robust feature engineering, poor imbalance handling, insufficient optimization, and dataset dependency—achieving superior accuracy, F1-Score, and generalization performance. Table II and Fig. 7 illustrates the comparative analysis of IDS models.

TABLE II. COMPARATIVE PERFORMANCE ANALYSIS OF IDS MODELS

Author	Dataset	Accuracy (%)	Key Techniques Used
Kumar *et al.* [4]	KDD-99	90.07	PCA + KNN (Cosine)
Sarkar *et al.* [5]	NSL-KDD	87.63	Cascaded MLP + Hyperparameter Tuning
Srinivas *et al.* [6]	NSL-KDD	93.20	Random Forest, DNN (5 layers)
Bose *et al.* [7]	In-SDN	86.00	BiLSTM + Attention + Convolutional Layers
Proposed BLEX (BOMI + BATO)	Kaggle Dataset	97.28	Boruta + MI (BOMI), ET + CB Ensemble, TPE Tuning (BATO)

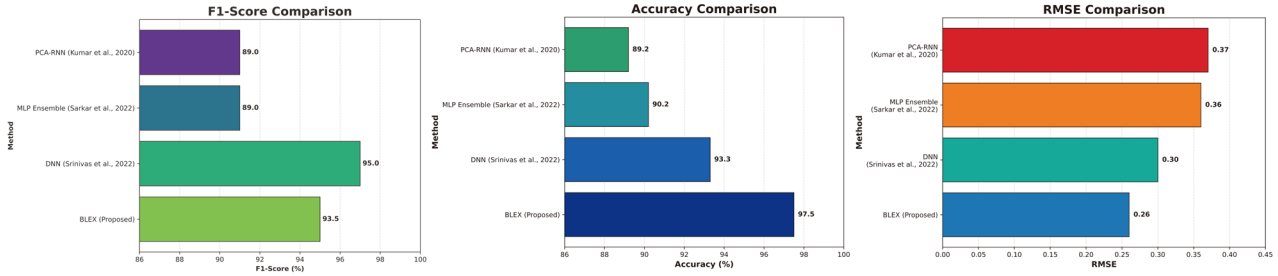

Fig. 7. Comparison of Intrusion Detection System (IDS) models.

V. DISCUSSION

Recent advances in AI-driven cybersecurity and network intelligence have explored federated intrusion detection for 6G IoT systems [24], unified frameworks for cyberattack detection in healthcare networks [25], neural network approaches for stroke prediction [26], and hybrid PSO-LightBoost methods for IoT intrusion detection [27] Robustness of the model was also secured by hyperparameter optimization using BATO. Bayesian Optimization was effective in both CatBoost and Extra Trees classifiers in exploring hyperparameter hyperspaces to reach the best configurations. This tuning process was to modify the decision boundaries and, therefore, generalize much better to previously inaccessible data. BLEX with a significantly bigger increase in accuracy (1.67%) and greater than 2.4% in F1-Score, demonstrated the effect of systematic tuning on the performance of an ensemble, compared to the untuned version (ET + CB). The visual comparisons demonstrate further the unanimity and supremacy of BLEX in any particular dimension of individual performance. Although the constituent models by themselves showed good performance, their

combination tended to integrate the advantages of both detection models, providing a flexible and dynamic decision strategy that could be used in a real-time intrusion detection network.

Despite the strong performance and methodological improvements introduced by the BLEX framework, several open challenges remain. First, the model is trained and evaluated on an offline Kaggle dataset, which may not fully reflect the variability, noise, and unpredictability of real-time network traffic; thus, its performance in live deployment scenarios requires further investigation. Second, although the combination of Borderline-SMOTE and BOMI improves class balance and feature relevance, the approach may still struggle with previously unseen zero-day attacks or rapidly evolving intrusion patterns that fall outside the statistical structure of the training data. Third, the ensemble architecture—while effective—introduces additional computational overhead during training, which may limit scalability for resource-constrained edge or IoT environments. Fourth, the framework has not yet been tested against adversarial manipulation techniques, such as evasion or poisoning attacks, which represent a growing threat to machine learning–driven IDS models. Finally, although Bayesian optimization enhances model robustness, it remains sensitive to search-space design and may require further refinement to ensure stability across diverse network domains. Addressing these challenges forms the basis for future extensions of the BLEX framework toward real-time, cross-domain, and adversarially resilient intrusion detection.

The proposed BLEX framework enhances IDS security by addressing key adversarial and operational challenges. Its hybrid feature selection (BOMI) makes the model more resistant to evasion attempts by reducing dependence on weak or easily manipulated features, while IQR-based outlier removal and Borderline-SMOTE improve robustness against noisy or inconsistent traffic. The ensemble design (ExtraTrees + CatBoost) distributes decision-making across multiple learners, reducing the impact of data poisoning and limiting the influence of corrupted inputs. BATO-driven hyperparameter optimization further stabilizes the model under adversarial conditions, ensuring consistent performance across variable traffic patterns. Although no IDS can fully prevent zero-day or sophisticated poisoning attacks, BLEX offers stronger resilience, better generalization, and improved reliability compared to traditional IDS approaches.

VI. Conclusion

The present study introduces BLEX, a high-performance intrusion detection framework that integrates a hybrid feature selection strategy (BOMI) with an ensemble learning approach and hyperparameter optimization using Bayesian Tree-structured Parzen Estimators (BATO). Through rigorous experimentation on benchmark network traffic data, BLEX demonstrated superior performance in accuracy (97.28%), precision (93.49%), recall (91.88%), F1-Score (94.23%), and RMSE (0.2592), clearly outperforming individual classifiers and their untuned ensemble. The BOMI approach successfully isolated the most significant features, reducing dimensionality and enhancing classification precision, while BATO-driven tuning effectively optimized the learning parameters of the ensemble model, improving its generalization capability. The strategic combination of CatBoost and Extra Trees offered a robust and adaptive decision boundary that proved highly effective in detecting diverse attack types. Furthermore, the visual and tabular comparisons validated the robustness and reliability of BLEX across multiple metrics. As a result, the study contributes an intelligent, interpretable, and scalable model suitable for modern, real-time intrusion detection systems. Future work may focus on extending the BLEX framework to handle streaming data environments, integrating deep learning components for richer feature representation, and validating the model across cross-domain and adversarial settings to enhance its resilience and adaptability against evolving cyber threats.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

SPP conceptualized the study, designed the methodology, and wrote the original draft; SB contributed to the experimental design and data collection; NSKMKT performed the data analysis and validation; DAD reviewed and edited the manuscript and contributed to the interpretation of results; TBK supervised the research, provided critical feedback, and finalized the manuscript; all authors reviewed and approved the final version.

References

[1] A. Ahmim, L. Maglaras, M. A. Ferrag *et al.,* "A novel hierarchical intrusion detection system based on decision tree and rules-based models," in *Proc. 15th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, Santorini, 2019.

[2] M. Saber, S. Chadli, M. Emharraf *et al.,* "Modeling and implementation approach to evaluate the intrusion detection system," in *Proc. Int. Conf. Netw. Syst.*, 2015, pp. 513–517.

[3] W.-C. Lin, S.-W. Ke, and C.-F. Tsai, "CANN: An intrusion detection system based on combining cluster centers and nearest neighbors," *Knowl. Based Syst.*, vol. 78, pp. 13–21, 2015.

[4] I. Kumar, N. Mohd, C. Bhatt *et al.,* "Development of IDS using supervised machine learning," in *Advances in Intelligent Systems and Computing*, Springer, pp. 565–577, 2020. doi: 10.1007/978-981-15-4032-5_52

[5] A. Sarkar, H. S. Sharma, and M. M. Singh, "A supervised machine learning-based solution for efficient network intrusion detection using ensemble learning based on hyperparameter optimization," *Int. J. Inf. Technol.*, vol. 15, no. 1, pp. 423–434, 2022. doi: 10.1007/s41870-022-01115-4

[6] K. Srinivas, N. Prasanth, R. Trivedi *et al.,* "A novel machine learning inspired algorithm to predict real-time network intrusions," *Int. J. Inf. Technol.*, vol. 14, no. 7, pp. 3471–3480, 2022. doi: 10.1007/s41870-022-00925-w

[7] S. Bose, G. Gokulraj, N. Maheswaran *et al.,* "Multi-layered security framework for intrusion detection system in software defined networking environment using machine learning," in *Proc. 15th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2024, pp. 1–7. doi: 10.1109/icccnt61001.2024.10724112

[8] N. S. K. M. K. Tirumanadham, V. Priyadarshini, S. P. Praveen *et al.,* "Optimizing lung cancer prediction models: A hybrid methodology using GWO and random forest," in *Studies in*

Computational Intelligence, Springer, 2025, pp. 59–77. doi: 10.1007/978-3-031-82516-3_3

[9] P. I. Priyadarsini, "ABC-BSRF: Artificial bee colony and borderline-SMOTE RF algorithm for intrusion detection system on data imbalanced problem," in *Lecture Notes on Data Engineering and Communications Technologies*, Springer, 2020, pp. 15–29. doi: 10.1007/978-981-15-8767-2_2

[10] N. S. K. M. K. Tirumanadham, S. Thaiyalnayaki, and M. Sriram, "Improving predictive performance in e-learning through hybrid 2-tier feature selection and hyper parameter-optimized 3-tier ensemble modeling," *Int. J. Inf. Technol.*, vol. 16, no. 8, pp. 5429–5456, 2024. doi: 10.1007/s41870-024-02038-y

[11] M. Alalareth and S.-C. Hong, "An improved mutual information feature selection technique for intrusion detection systems in the internet of medical things," *Sensors*, vol. 23, no. 10, 4971, 2023. doi: 10.3390/s23104971

[12] A. Sharaff and H. Gupta, "Extra-tree classifier with metaheuristics approach for email classification," in *Advances in Intelligent Systems and Computing*, Springer, 2019, pp. 189–197. doi: 10.1007/978-981-13-6861-5_17

[13] Y. Li, S. Han and P. Niu, "Wireless sensor network intrusion detection based on Borderline-SMOTE and deep ensemble learning," in *Proc. 7th Int. Conf. Electron. Electr. Eng. Technol. (EEET)*, 2024, pp. 1–6. doi: 10.1109/eeet64351.2024.00011

[14] B. Thuraka, V. Pasupuleti, C. S. Kodete *et al.,* "Enhancing diabetes prediction using hybrid feature selection and ensemble learning with AdaBoost," in *Proc. 8th Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud)*, 2024, pp. 1132–1139. doi: 10.1109/i-smac61858.2024.10714776

[15] H. Gharoun, N. Yazdanjoe, M. S. Khorshidi *et al.,* "Noise-augmented Boruta: The neural network perturbation infusion with Boruta feature selection," arXiv preprint, arXiv:2309.09694, 2023.

[16] K. V. Rajkumar, K. S. Nithya, C. T. S. Narasimha *et al.,* "Scalable web data extraction for Xtree analysis: Algorithms and performance evaluation," in *Proc. 2nd Int. Conf. Inventive Comput. Inform. (ICICI)*, 2024, pp. 447–455. doi: 10.1109/icici62254.2024.00079

[17] R. R. Pbv, T. N. S. K. M. Kumar, S. P. Praveen *et al.,* "Optimizing lung cancer detection: The synergy of support vector machine and random forest," in *Proc. Int. Conf. Decision Aid Sci. Appl. (DASA)*, 2024, pp. 1–7. doi: 10.1109/dasa63652.2024.10836455

[18] H. B. Bandela, S. Sikindar, C. R. Swaroop *et al.,* "An optimized bagging ensemble learning of machine learning algorithms for early detection of diabetes," in *Proc. Int. Conf. Self Sustain. Artif. Intell. Syst. (ICSSAS)*, 2023, pp. 274–281. doi: 10.1109/icssas57918.2023.10331844

[19] R. Islam, A. Sultana, and M. N. Tuhin, "A comparative analysis of machine learning algorithms with tree-structured Parzen estimator for liver disease prediction," *Healthcare Anal.*, vol. 6, 100358, 2024. doi: 10.1016/j.health.2024.100358

[20] S. Sieradzki and J. Mańdziuk, "Modified adaptive tree-structured Parzen estimator for hyperparameter optimization," arXiv preprint, arXiv: 2502.00871, 2025.

[21] R. Nagarajan, M. Batumalay, and Z. Xu, "IoT based intrusion detection for edge devices using augmented system," *J. Appl. Data Sci.*, vol. 5, no. 3, pp. 1412–1423, 2024.

[22] D. Z. Rodriguez, O. D. Okey, S. S. Maidin *et al.,* "Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic explainable feature selection," *PLoS One*, vol. 18, no. 10, e0286652, 2023.

[23] M. Fahim-Ul-Islam, A. Chakrabarty, M. G. R. Alam *et al.,* "A resource-efficient federated learning framework for intrusion detection in IoMT networks," *IEEE Trans. Consum. Electron.*, vol. 71, no. 2, pp. 4508–4521, 2025.

[24] S. P. Praveen, K. Sharma, D. Parashar *et al.,* "Design of an iterative method for adaptive federated intrusion detection for energy-constrained edge-centric 6G IoT cyber-physical systems," *Sci. Rep.*, vol. 15, no. 1, 41387, 2025.

[25] S. P. Praveen, M. Kamalrudin, M. Musa *et al.,* "A unified AI framework for confidentiality preserving cyberattack detection in healthcare cyber physical networks," *Int. J. Innov. Technol. Interdiscip. Sci.*, vol. 8, no. 3, pp. 818–841, 2025.

[26] S. P. Praveen, J. S. Mantena, U. Sirisha *et al.,* "Navigating heart stroke terrain: A cutting-edge feed-forward neural network expedition," *J. Appl. Data Sci.*, vol. 6, no. 3, pp. 2111–2126, 2025.

[27] S. P. Praveen, S. Lalitha, P. Sarala *et al.,* "Optimizing intrusion detection in Internet of Things (IoT) networks using a hybrid PSO-LightBoost approach," *Int. J. Intell. Eng. Syst.*, vol. 18, no. 3, 2025.