*Journal of Computer Science*

Research Article

# Ensemble Learning for Proactive Detection of Network-Intrusion-Based Insurance Fraud

**Benjamin Asubam Weyori[1], Selorm Kofi Tagbo[2,3], Abubakar Sadik Yakubu[4] Kenneth Kojo Bonsu[5], Moesha Noah[6] and Eric Wiafe Appah[4]**

[1]*Department of Computer and Electrical Engineering, University of Energy and Natural Resources, Sunyani, Ghana*
[2]*Department of Computer Science and Informatics, University of Energy and Natural Resources, Sunyani, Ghana*
[3]*Department of Computing and Information Sciences, Catholic University of Ghana - Sunyani-Fiapre, Ghana*
[4]*Business and Technology College, Wilmington University, USA*
[5]*Sawyer Business School, Suffolk University, USA*
[6]*College of Engineering, Drexel University, USA*

*Article history*
Received: 28-07-2025
Revised: 15-10-2025
Accepted: 29-04-2026

**Corresponding Author:**
Benjamin Asubam Weyori
Department of Computer and Electrical Engineering, University of Energy and Natural Resources, Sunyani, Ghana
Email: benjamin.weyori@uenr.edu.gh

**Abstract:** We propose an ensemble learning pipeline that proactively integrates Stacking Feature Embedding (SFE) with Principal Component Analysis (PCA) and tree-based ensembles to proactively detect insurance fraud originating from network intrusions. The main contributions are: (1) the novel integration of SFE-PCA as a meta-feature construction step for tabular network flow data; (2) a sensitivity analysis that justifies PCA reduction ratios used for each dataset; and (3) a computational and ethical assessment for real-world deployment. Random Forest (RF), Extra Trees (ET), and XGBoost classifiers were trained and evaluated on benchmark intrusion datasets, specifically NSL-KDD, LYCOS-IDS2017, and CIC-IDS2018. Findings from experiments conducted on these datasets show that the proposed pipeline achieves high detection performance (AUC > 0.995) and 99.9% accuracy, while reducing feature dimensionality and resource use compared to deep baselines (CNN/LSTM). These results suggest the approach is an efficient, interpretable option for proactive intrusion-driven insurance fraud detection.

**Keywords** Insurance Fraud Detection, Ensemble Learning, Stacking Feature Embedding, PCA, Intrusion Detection, AUC-ROC

## Introduction

Insurance fraud puts a strain on the financial resources of insurance providers, policyholders, and even the general public, ultimately driving up costs for everyone involved. It also damages the trust that customers place in insurance companies. Typically, fraud can be grouped into two main types: Proactive fraud, which is planned in advance, and reactive fraud, which occurs in response to specific situations. (Hamid et al., 2024). Proactive fraud is a kind of deceptive action that is either in the planning stages or currently underway (Ribeiro and Costa, 2019). These can often be identified early using contextual or behavioral data. On the other hand, reactive fraud involves schemes that have already been carried out. Such fraud is typically harder to detect and is usually uncovered only after substantial financial damage has been done (Bartsiotas and Achamkulangare, 2021). So far, most existing systems have primarily targeted reactive fraud, identifying it only after the damage is done. However, there is a growing need for systems that can detect fraud proactively before it results in significant cause of system or social damage. Ensemble learning frameworks work by bringing together multiple diverse models in a structured way to produce more accurate and dependable predictions than any single model could achieve on its own (Xiao et al., 2018). A proactive insurance fraud detection and prediction system relies on a combination of background information, past records, and time-series data to identify suspicious behavior early and take to prevent actions before fraud is finally meted out to a victim (Kose et al., 2015).

Ensemble learning systems are established by incorporating various machine learning methods and techniques into each other, allowing them to leverage the

strengths of different models to improve overall prediction accuracy and robustness (Rane et al., 2024a). This research primarily focuses on the use of ensemble models for proactive insurance fraud detection and prediction. Compared to traditional standalone models, ensemble approaches offer improved accuracy and effectiveness by combining the strengths of multiple algorithms to better identify fraudulent activity early on.

Existing insurance fraud detection systems primarily focus on identifying fraudulent activities after they have occurred. While numerous studies have applied machine learning and deep learning techniques to fraud detection, limited attention has been given to proactive fraud detection using intrusion-related data. Furthermore, many existing approaches either rely on computationally expensive deep learning models or do not adequately address feature representation and dimensionality reduction challenges. This creates a need for a lightweight and interpretable framework capable of detecting intrusion-driven fraudulent activities before significant financial losses occur.

### *Background on Insurance Fraud*

Insurance fraud continues to be a serious and rising concern for the insurance industry, with estimated annual losses exceeding £1 billion due to exaggerated or false claims (Al-Dajeh et al., 2025). Back in 2003, this growing threat led UK insurers to cancel or reject motor insurance policies worth £790 million over suspected fraudulent activity (Honigsberg et al., 2020). Opportunistic individuals often view submitting false claims as a means of recouping the money they've spent on premiums, leading them to engage in this type of fraud. Those who engage in organized crime also participate in insurance fraud, treating it as a lucrative source of income (Saylor, 2023). The sharp increase in the value of insured assets has created even greater opportunities for these networks to carry out large-scale and more sophisticated fraudulent schemes. By exploiting laws designed to compensate victims of violent crime, fraudsters submit claims following staged accidents, in which previously undamaged vehicles are deliberately crashed (Morley et al., 2006).

In response to the growing risks and escalating financial impact, insurance companies are pouring significant resources into building comprehensive strategies that engage multiple departments and areas of expertise across their organizations. To gain a thorough understanding of how fraud detection is currently handled, an ethnographic method was employed to closely examine the day-to-day operations of both frontline and secondary fraud detection teams within two major insurance firms. This approach helped capture both the strengths and shortcomings of their existing strategies. Key areas for future exploration were also identified, aimed at strengthening and advancing current fraud detection methods, practices, and workflows. It is a well-known fact that conventional methods for uncovering and investigating insurance fraud fall short. A closer look at the nature of fraudulent activity and the challenges in detecting it reveals that traditional investigations tend to concentrate on the traits of suspicious claims, the people making them, or the specifics of the policies in question, an approach that often overlooks deeper, systemic patterns of fraud.

### *Importance of Proactive Detection of Insurance Fraud*

Around the world, a growing concern for the insurance industry is the rise in cases of fraud i.e. whether it happens through very simple means or more through more elaborate scams (Jung and Kim, 2021). According to a detailed investigation, the cost of fraudulent claims to the UK insurance industry has surged by 30% in just two years, now exceeding £1 billion annually (Tarr, 2019). The rising financial burden of insurance fraud has pushed insurance companies to step up their efforts and adopt stronger measures to combat it (Srinivasagopalan, 2020). In order to stay ahead of these criminals, most major insurance firms in the UK and other areas in the world have poured substantial resources into prevention, detection, and investigation strategies. (Morley et al., 2006).

Faced with the rising challenge of fraud, the insurance industry is working hard to find effective ways to tackle the issue head-on (Tax et al., 2021). One of the most difficult tasks in tackling this issue is that insurance fraud often happens in secrecy, making it difficult to identify and curb. Insurers also have to operate within legal boundaries, which can complicate things. If a claim raises suspicion of fraud, it might lead to an investigation, which often results in delays in processing the claim. As fraud continues to evolve, recent strides in technology have only intensified the push for smarter solutions. These advancements are opening up fresh opportunities for organizations to spot and stop fraud more effectively than ever before. Even so, most anti-fraud measures are put in place and tailored to specific incidents rather than built as part of a broader, unified strategy (Gajda, 2025). Because each insurance company tends to design its own special defense mechanisms, given the competitive nature of most of the cases that happen between fraudsters and insurers, sharing insights or strategies across the industry is quite uncommon. Since no single insurance company can fully eradicate fraud cases or provide the most effective countermeasures, a significant gap in shared knowledge exists in this field. This brings to light the pressing need for a unified knowledge hub that gathers clear, actionable insights and also makes it easier for the insurance industry to stay ahead of evolving fraud schemes, so that they can respond in real time and deal with such cases with greater efficacy (Veera, 2025). As a result, because fraud strategies keep evolving, insurance teams on the ground need more than just static reports or

outdated manuals. What's really missing is a flexible, easy-to-use application they can turn to i.e. something that not only explains what fraud looks like but actually helps them make smart, real-time decisions when building or adjusting their defenses.

### *Research Gap and Motivation*

Although a lot of progress has been made in the application of machine learning, deep learning and ensemble learning techniques for fraud detection and intrusion detection, there are still several important challenges that remain. First, most existing insurance fraud detection systems are reactive in nature. So, they focus on identifying fraudulent activities after they have already occurred and financial losses have been incurred. Limited attention has been given to proactive detection approaches that can identify suspicious activities at earlier stages.

Also, while ensemble learning methods such as Random Forest, XGBoost, LightGBM, and hybrid ensemble models have demonstrated promising results, many studies rely solely on the original feature space without exploiting clustering-derived meta-features that can capture hidden structural relationships within network traffic data. Consequently, valuable information that may improve class separability and detection performance is often overlooked.

In addition, deep learning approaches such as CNNs and LSTMs have achieved high detection accuracy but are frequently associated with high computational costs, longer training times, and reduced interpretability, which may limit their practical deployment in real-world insurance environments.

Therefore, existing studies rarely provide comprehensive analyses that simultaneously evaluate detection performance, computational efficiency, and ethical considerations associated with deployment.

To address these limitations, this study proposes a Stacking Feature Embedding, which encompasses Principal Component Analysis (SFE-PCA) framework that integrates clustering-based meta-feature construction with dimensionality reduction and ensemble learning techniques. The proposed approach aims to improve proactive intrusion-driven insurance fraud detection while maintaining computational efficiency, model interpretability, and practical applicability.

### *Research Questions*

RQ1 : Can SFE-PCA improve intrusion-driven insurance fraud detection performance compared with conventional ensemble models?

RQ2 : How does the proposed framework compare with deep learning methods in terms of computational efficiency?

RQ3 : What are the computational, operational, and ethical considerations associated with deploying the proposed SFE-PCA framework in insurance fraud detection environments?

### *Contributions*

In this work, unlike end-to-end deep representation learners (e.g., autoencoders, transformer-based encoders) that learn latent embeddings from raw inputs, SFE explicitly constructs meta-features via clustering outputs and embeds them alongside original features. This retains interpretability and reduces the risk of overfitting on small minority classes while being less compute-intensive Hence, this paper makes the following contributions:

i. We introduce a SFE + PCA pipeline (SFE-PCA) that generates meta-features to enhance ensemble classifiers on intrusion flow data
ii. We perform a sensitivity analysis to justify PCA reduction ratios for each dataset and report the accuracy–efficiency tradeoffs
iii. We compare tree-based ensembles with deep learning baselines (CNN, LSTM) and quantify computational and memory costs
iv. We provide ethical considerations and deployment guidance, including data anonymization and oversampling bias mitigation

Fraud-detection systems must balance the risk of false positives by flagging legitimate claims against false negatives. This allows fraud to pass undetected. In insurance practice, missing a fraudulent claim generally causes greater financial and reputational harm than reviewing an extra genuine claim. Consequently, this work emphasizes minimizing false negatives while keeping false positives acceptably low.

## *Literature Review*

### *Overview of Ensemble Learning*

More generally, ensemble learning methods fall into four main categories: Bagging, boosting, stacking, and those that rely on dividing the feature space (Shaikh et al., 2024). Bagging works on the idea that variety leads to better results. This is achieved by creating multiple versions of a base model, each being trained on a slightly different, randomly selected portion of the same data (Alsaffar et al., 2024). This makes it possible to overcome the problem of overfitting. Hence, eventually combining output, whether through voting or averaging, tends to be more stable and accurate than any single model on its own (Talukder et al., 2024). Bagging has found more broader applications across a range of machine learning algorithms. This includes Decision Trees and Neural Networks to Support Vector Machines and other memory-based method. It has the ability to improve stability and accuracy by reducing data variance (Han et al., 2021). Despite notable progress, stacking several instances of a

well-performing model doesn't automatically lead to improved outcomes. When the models lack diversity and are too alike, the ensemble may struggle to generalize effectively, reducing its ability to handle new or unfamiliar data. Typically, for an ensemble to perform well, the models it brings together need to differ in how they learn and make predictions (Wu and Levinson, 2021). This variety is essential for strong performance, which is why numerous strategies have been devised to promote distinct behaviors and perspectives among the models in the ensemble.

### *Types of Ensemble Learning Methods*

Ensemble methods have become a powerful tool in machine learning and data mining because they bring together the strengths of multiple models to make smarter predictions (Rane et al., 2024b). Instead of relying on a single model, these techniques use a "team" of base learners whose outputs are combined, often with the help of a higher-level model, or meta-learner. This process, known as stacking, tends to work best when each model brings a slightly different perspective, especially when they're trained on different parts of the data (Chatzimparmpas et al., 2021).

What really makes an ensemble strong isn't just how many models are used, but how differently they make mistakes. When their errors don't overlap, the final prediction becomes more balanced and reliable (Lones, 2024). A very fundamental way to understand how useful a model is within an ensemble is to look at how much its predictions vary compared to others on the same level. The goal is to build a smarter system by learning from a collection of already-trained models, which are almost like assembling a panel of experts, each contributing their unique viewpoint to make the final decision more accurate and trustworthy.

### *Bagging*

The bagging model can be viewed as a belief consensus approach using quasi-probability representations (Ngo et al., 2022). Within the belief function framework, two techniques are introduced to combine the outputs of individual classifiers by expressing their predictions as pignistic probability distributions (Song and Deng, 2019).

While traditional bagging assumes the use of a single learning algorithm to train multiple homogeneous classifiers, a modified prediction scheme is introduced that incorporates heterogeneous classifiers, each capturing the learned hypothesis in distinct ways (Alzubi et al., 2020). Following the training phase, aggregation functions are employed to combine the outputs into a unified prediction. Experiments on benchmark datasets demonstrate that leveraging diverse classifiers can lead to significantly improved accuracy compared to using only homogeneous models.

### *Boosting*

In an effort to enhance fraud detection, proposed an innovative method called Deep Boosting Decision Trees (DBDT) (Alonge et al., 2021). At its core, DBDT merges neural networks with gradient boosting to create a powerful hybrid model. One of its main components is the Soft Decision Tree (SDT), which reimagines the traditional decision tree by replacing each node with a neural network, allowing decisions to be made in a more flexible and data-aware manner (Luo et al., 2021). Building on the Soft Decision Trees (SDTs), the DBDT approach ensembles them using Gradient Boosting (GB), effectively embedding neural networks within the boosting framework. This integration enhances the model's ability to learn rich representations while preserving the interpretability typically associated with decision trees (Coppens et al., 2020). Interpretability is maintained by extracting rule-based insights from the trained SDTs. To address class imbalance, a common challenge in fraud detection, the authors introduce a compositional AUC maximization strategy during training. The model was rigorously tested across several real-world fraud detection datasets, showing strong performance in varied scenarios (Damanik and Liu, 2025).

Interestingly, DBDT has been shown to significantly enhance the performance of baseline models. Boosting algorithms, in particular, are widely favored in insurance-related risk prediction tasks. In a recent study, Terefe (2023) conducted a thorough evaluation of model development and performance using SML data for predicting vehicle insurance claim sizes. The study focused on three widely used tree-based ensemble methods: Bagging, RFs, and gradient boosting (Dube and Verster, 2023). These models were compared using Root Mean Square Error (RMSE), graphical analysis, and statistical testing. The findings offer valuable insights into how tree-based ensembles perform and can be effectively applied in modeling insurance claim sizes.

### *Stacking*

Rather than relying on a single model, stacking, which is also known as stacked generalization is an ensemble learner that uses a layered approach to improve predictive performance (Oriola, 2022). In this method, multiple base learners are first trained on the full dataset. Their individual predictions are then collected and passed as input features to a final model, often called a meta-learner. This top-level model learns to make better predictions by identifying patterns and relationships across the outputs of the base models (Oriola, 2022). To effectively build a set of base models, it's essential to use either two or more distinct algorithms (Kerwin and Bastian, 2021). Once each selected base model is trained, it undergoes validation, and the resulting predictions are stored. These validation outputs then serve as input for the final model,

which learns to predict the target variable based on patterns in the base models' performance. The core idea behind stacking is to go beyond the limitations of individual classifiers by introducing an extra layer that helps correct generalization errors, ultimately boosting predictive accuracy (Rane et al., 2024b).

### *Blending*

In the blending technique, predictions from individual base learners are used as inputs to a meta-learner, which is then trained on these outputs to generate the final prediction (Abro et al., 2023). By leveraging the strengths of multiple base models, a refined, blended outcome can be produced. Within a stacking framework, a new learning method can further enhance performance by using hold-out predictions from the base learners (Gur et al., 2009). These hold-out predictions serve as inputs to the new model, allowing it to fine-tune the combination weights and improve overall predictive accuracy on unseen data. Common algorithms used within this architecture include K-Nearest Neighbors and various forms of regression (Boateng et al., 2020). As outlined earlier in the section on stacking methods, these models serve effectively as base or meta-learners. However, it's interesting to note that stacking can significantly increase computational time, particularly when base models are complex, slow to train, or require extensive hyper-parameter tuning for optimal performance. Due to the vast number of possible configurations and variations in runtime protocols, exact execution times cannot be definitively reported (Borketey, 2024). However, it is important to recognize that combating fraud is a continuous process that demands ongoing vigilance from developers. While ensemble stacking techniques can enhance detection capabilities, they should not be viewed as a one-size-fits-all solution. With the right safeguards and monitoring in place, organizations can adopt clear and practical strategies to strengthen their defenses against fraudulent activity.

### *Multimodal Data Environments*

The exponential growth of various online platform services has heavily influenced people's daily lives, which also brings up many challenges, such as monitoring user safety (Batani, 2021). For instance, it is very common in financial payment platforms for users to pay for stuff with online banking accounts. However, it is not rare either that a user's account is hacked and illegal money transfers occur, which cause huge losses of money (Razaq et al., 2021). Thus, it is critical for payment processors and banks to use fraud detection systems to analyze the transaction data between users and catch high risks ones. In general, fraud detection systems must address the sequential structure of the problem in order to prevent individual transactions and operate in close to real time (Dai et al., 2024). One of the most important challenges in detection system design is to produce a real-time representation of the profile of both an account and a transaction by processing potential sequences of transactions. Without loss of generality, first a set of transactions is filtered from a huge transaction history into a sub-sequence of concerned accounts. A prediction function that maps a transaction and its profile to a real value score indicating likelihood of fraud is nevertheless the critical component of fraud detection systems in general. This function can be trained on a sufficient quantity of labelled transactions sampled from customer interactions with the fraud detection system (Zhang et al., 2022).

### *Challenges in Fraud Detection*

Fraud detection is both an exciting and challenging area of research (Ahsun and Okunola, 2020; Javaid, 2018). The increasing availability of data has led to the emergence of sophisticated fraud applications. In addition, fraud detection and prediction algorithms have been increasingly used in financial services such as insurance, banking and payments, and healthcare (Javaid, 2018). However, massive data streams, concept drift, and lack of clear definitions add challenges to proactively detecting and predicting fraud in big data. Beating fraudsters in this frictionless digital era creates both research opportunities and promising futures for combating fraud (Hamid et al., 2024). However, this also raises challenges on detecting and predicting fraud in data streams due to the enormous costs of false positives and missed fraudulent transactions (Perez et al., 2023) Complex temporal patterns in dynamics, data drift affects, and distributed architecture in trusted computing add challenges to both fraud detection and prediction.

Having access to massive data has enabled a wide range of machine learning based strategies for predicting and detecting fraud. Outlier detection, supervised classification, and statistical analysis are the traditional methods for fraud detection, but they typically fall short for fraud prediction in sequential data, where past events and new variables interact in more complex and diverse ways. Transfusion and Kalman filters capture the sequences of transactions on accounts for detecting fraud loner term fraud, but they do not apply to prediction of fraud (Alkadi et al., 2021; Alonge et al., 2021; Orelaja and Adeyemi, 2024). Abstractive and extractive methods detect and predict fraud from social discussions about transactions. Based on data availability, a data-intake-driven or screening first approach detects new fraudulent data items and a label-driven or warning first approach predicts fraud on data attributes. However, there is no comprehensive framework for proactively detecting and predicting fraud in data streams. Equipped with a gaming history, fraudsters build surveillance and impersonate systems to beat the prediction systems.

Table 1 summarizes recent relevant studies (2020–2024) on fraud and intrusion detection, highlighting datasets, methods, and limitations addressed by this work, specifically the lack of meta-feature stacking, computational evaluation, and ethical assessment.

## Methods

### *Assumptions*

The following assumptions underpin our simulations:

i. Public IDS datasets (NSL-KDD, LYCOS-IDS2017, CIC-IDS2018) approximate intrusion patterns relevant to insurance-related workflows
ii. Dataset labels are correct for supervised evaluation
iii. Feature sets reflect realistic flow-level and service metadata available to insurers and finally
iv. Random Oversampling is applied to anonymized feature vectors only (no duplication of PII)

### *The Proposed SFE-PCA Process*

The data preprocessing techniques used in our approach such as feature resampling, scaling, feature extraction, and embedded stacking features are presented alongside the description of the proposed framework in this section.

Figure 1 illustrates the schematic block diagram of our proposed paradigm, which is organized into several key phases.

"This approach emphasizes the use of stacking-based feature embedding followed by dimensionality reduction to improve intrusion detection performance on large-scale insurance datasets. Before introducing the framework, we first present a brief overview of the machine learning algorithms used in the intrusion detection process. Ultimately, the goal is to support a more secure network by accurately identifying incoming packets as either normal or malicious.

- Phase -1: The dataset was preprocessed to improve quality and ensure consistency by cleaning and optimizing data types, handling missing values, merging minor classes, removing duplicates, and applying feature scaling and label encoding to standardize inputs for effective model training
- Phase -2: To make the dataset more suitable for training machine learning models, class imbalance is addressed using Random Oversampling. We compared Random Oversampling (RO) with SMOTE and ADASYN on a hold-out split of NSL-KDD. While SMOTE/ADASYN marginally improved minority recall, they introduced synthetic neighbors that increased false positives and training time on high-dimensional flow features. RO provided a stable improvement in recall with minimal computational overhead, and thus was selected for the main experiments
- Phase -3: To improve model performance, the dataset is enriched with additional insights by embedding clustering results as meta-features. Instead of relying solely on the original features in the dataset, the model is enriched with extra layers of information. i.e. meta-features that reveal deeper patterns and relationships in the data. These are added during the SFEprocess, allowing the model to uncover more meaningful internal structures

**Table 1:** Summary of Recent Studies (2020–2024) on Fraud and Intrusion Detection

| Author (Year) | Task | Dataset(s) | Method | Key Result / Limitation |
|---|---|---|---|---|
| Batani (2021) | Financial transaction fraud detection | European credit card dataset | RF, XGBoost | High precision but limited interpretability and no deployment or runtime analysis. |
| Sharafaldin et al. (2018) | Intrusion detection benchmarking | CIC-IDS2018 | Classical ML (RF, SVM, KNN) | Provided benchmark dataset and baselines; did not explore ensemble-meta feature stacking. |
| Talukder et al. (2024) | Network intrusion–based fraud detection | CIC-IDS2018 | Hybrid ensemble with feature selection | Achieved high accuracy (≈99%) but lacked theoretical integration of feature embeddings and computational cost analysis. |
| Alsaffar et al. (2024) | Intrusion-driven insurance fraud detection | NSL-KDD, LYCOS-IDS2017, CIC-IDS2018 | Random Oversampling + SFE-PCA + Ensemble (RF, ET, XGBoost) | Integrates meta-feature stacking and dimensionality reduction with detailed sensitivity, computational, and ethical analyses for proactive fraud detection. |
| Zhang et al. (2022) | Cyber-fraud detection via deep learning | UNSW-NB15 | CNN-LSTM hybrid | Improved detection accuracy but computationally intensive; lacked comparative ensemble evaluation. |

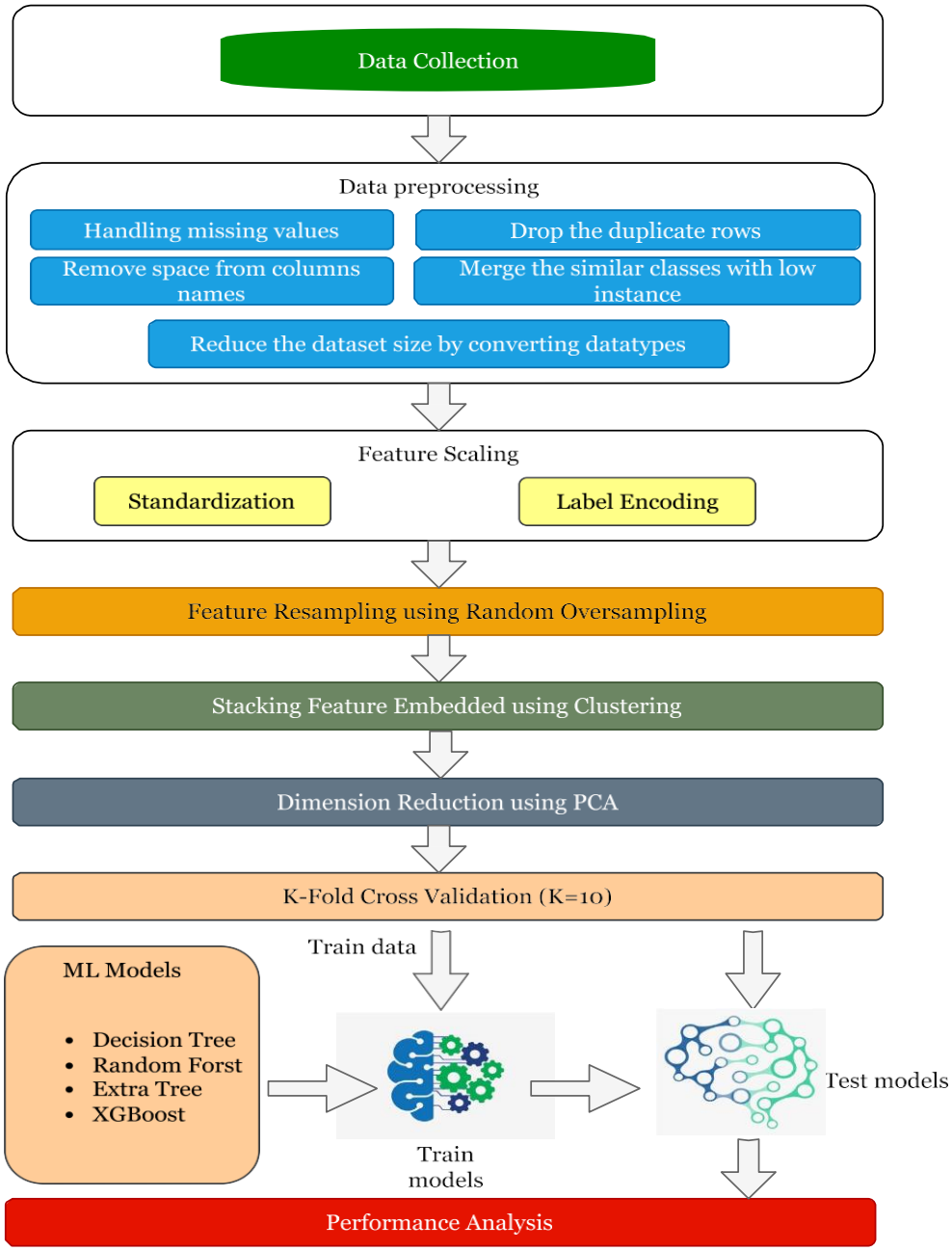



**Fig. 1:** Proposed SFE-PCA pipeline

- Phase -4: To improve model efficiency and reduce computational complexity, dimensionality reduction is performed using PCA. This technique retains the most important information while minimizing the number of features. During the feature extraction step, PCA is applied to the dataset to extract meaningful components, effectively simplifying the data without significant loss of relevant patterns
- Phase -5: To ensure robust evaluation and enhance model accuracy, we apply 10-fold cross-validation, an approach that rigorously tests the model across multiple data subsets. This validation strategy is implemented after splitting the pre-processed dataset into separate training and testing sets. To help prevent overfitting and get a more accurate sense of how well the model performs, the test data is rotated through different folds. This approach ensures the evaluation isn't biased toward any single portion of the data
- Phase -6: We selected four leading ensemble classifiers i.e. XGBoost (XGB), ET (ET), RF (RF),

and Stacked Generalization (SG), each representing a major branch of ensemble learning. To fairly evaluate which of these models best suits intrusion detection tasks, we used k-fold cross-validation to test their performance across different data splits. This evaluation step is critical for comparing model performance and selecting the best-performing algorithm based on consistent validation results

- Phase -7: To meaningfully compare our model with existing methods, we relied on a broad range of performance metrics, namely; ROC curve, Confusion Matrix, F1-score, Accuracy, Recall, and Precision. These measuring tools helped us thoroughly assess both the strengths and weaknesses of the model, especially in the final stage of evaluation. The entire framework that was proposed in this study has been illustrated in Figure 1

From Fig. 1, Clustering outputs are embedded as meta-features, concatenated with original features, then reduced using PCA before model training. The pipeline prioritizes interpretability and computational efficiency over black-box end-to-end encoders.

### *Collecting Data for the Model*

Two different datasets were used for this purpose. These are LYCOS-IDS2017 and UNSW- NB15 datasets. The upcoming sections provide a detailed overview of each dataset used in this study. Both datasets reflect realistic intrusion detection system environments for insurance transactions and include up-to-date attack categories, making them well-suited for evaluating modern threat detection capabilities.

### *NSL-KDD Dataset*

Unlike older datasets such as KDDCUP'99, NSL-KDD, Kyoto 2006+, ISCX, the NSL-KDD dataset is relatively recent and offer a comprehensive collection of internet traffic data within which insurance frauds occur. It includes nine distinct categories of malicious activity, making it more representative of modern cyber threats and suitable for evaluating contemporary intrusion detection systems. A total of 49 features, including the class label, were generated using 12 algorithms implemented through Argus and Bro-IDS tools. The dataset is stored across four CSV files, containing approximately 2,540,044 records. To prepare the data for model development, a setup process was carried out in which the dataset was divided into separate training and testing subsets. Featuring nine distinct types of modern fraudulent attacks along with a broad spectrum of real-world network behavior, the dataset offers a realistic and up-to-date representation of current security threats. Each record is labeled to reflect either typical activity or staged attack scenarios. It comprises a total of 257,673 entries, divided into 175,341 for training and 82,332 for testing, making it well-suited for evaluating intrusion detection models under practical conditions. A summary of these attack categories is presented in Table 2, while Table 3 shows their frequency distribution, highlighting the significant imbalance between attack types and benign traffic.

**Table 2:** Ategories of some attacks in the NSL-KDD dataset

| Attack category | Total No. | Percentage |
|---|---|---|
| Analysis | 2677 | 1.10 |
| Backdoor | 2329 | 0.96 |
| DoS | 16353 | 6.71 |
| Exploits | 44525 | 18.27 |
| Fuzzers | 24246 | 9.95 |
| Generic attacks | 58871 | 24.16 |
| Normal intrusions | 93000 | 38.16 |
| Shell-code | 1511 | 0.62 |
| Worm attacks | 174 | 0.07 |
| Total | 243686 | 100 |

**Table 3:** Categorization of the various attacks identified in the LYCOS-IDS2017 dataset

| Category | Frequency | Percentage (%) |
|---|---|---|
| BENIGN | 563584 | 50.22 |
| Bot | 1966 | 0.18 |
| DDoS Attack | 383692 | 34.19 |
| FTP-Patator | 7938 | 0.71 |
| Heartbleed | 74 | 0.01 |
| Infiltration | 85 | 0.01 |
| PortScan | 156855 | 13.98 |
| SSH-Patator | 5897 | 0.53 |
| Web Attack | 2180 | 0.19 |
| Total | 1,122,271 | 100 |

Figures 2 and 3 illustrate this imbalance for both binary and multi-class classification tasks, depicting the distribution of data across three stages: Prior to preprocessing, after preprocessing, and after applying oversampling techniques.

### *LYCOS-IDS2017*

As highlighted by Sharafaldin et al. (2018), a persistent challenge in advancing anomaly-based intrusion detection lies in the lack of reliable test and validation datasets. This shortfall limits the ability to consistently and accurately assess the effectiveness of detection techniques. Intrusion Detection Systems together with Intrusion Prevention Systems continue to serve as vital elements in network defense strategies, especially as network threats grow more complex and constantly evolve.

By leveraging packet capture (PCAP) files, the LYCOS-IDS2017 dataset effectively replicates real-world network environments, making it a promising solution to the challenge of limited reliable evaluation data. It offers a rich blend of benign traffic and a wide variety of modern, frequently occurring cyberattacks, providing a more realistic and comprehensive foundation for developing and testing intrusion detection systems.

Using CICFlowMeter, the LYCOS-IDS2017 dataset offers a robust and detailed summary of traffic monitoring, as emphasized by Sharafaldin et al. (2018). Leveraging B-Profile technology, the dataset simulates a broad spectrum of modern cyberattacks, including Brute Force, DoS, Heartbleed, Web Attacks, Infiltration, Botnet, and DDoS. It meticulously organizes labeled network flows in CSV files, capturing essential meta-data such as timestamps, source and destination IP addresses, port numbers, protocols, and clearly defined attack vectors. To generate realistic baseline traffic, the tool captures behavioral patterns associated with legitimate network activity through profiling.

According to the evaluation framework proposed in Gharib et al. (2016), there are 11 essential criteria for building a reliable benchmark dataset for intrusion detection. The LYCOS-IDS2017 dataset stands out as the only one that meets all these requirements, while most earlier IDS datasets fall short. The distribution of binary and multi-class attack labels before preprocessing, after preprocessing, and following oversampling is illustrated in Figures 4 and 5. A detailed breakdown of attack categories is presented in Table 2. Additionally, Figure 2 shows the binary frequency distribution for the NSL-KDD dataset.

### *CIC-IDS2018*

Traditional datasets used for intrusion detection research often suffer from limitations such as over-anonymization, privacy constraints, and outdated threat representations, making them less effective for evaluating modern anomaly detection systems. Developed through a collaboration between the Communications Security Establishment (CSE) and the Canadian Institute for Cybersecurity, the CSE-CIC-IDS2018 dataset (Batani, 2021) was designed to overcome previous limitations. As a promising approach for identifying new cyber threats, network-based anomaly detection comes with its own set of challenges, and this study is centered on addressing those specific difficulties.

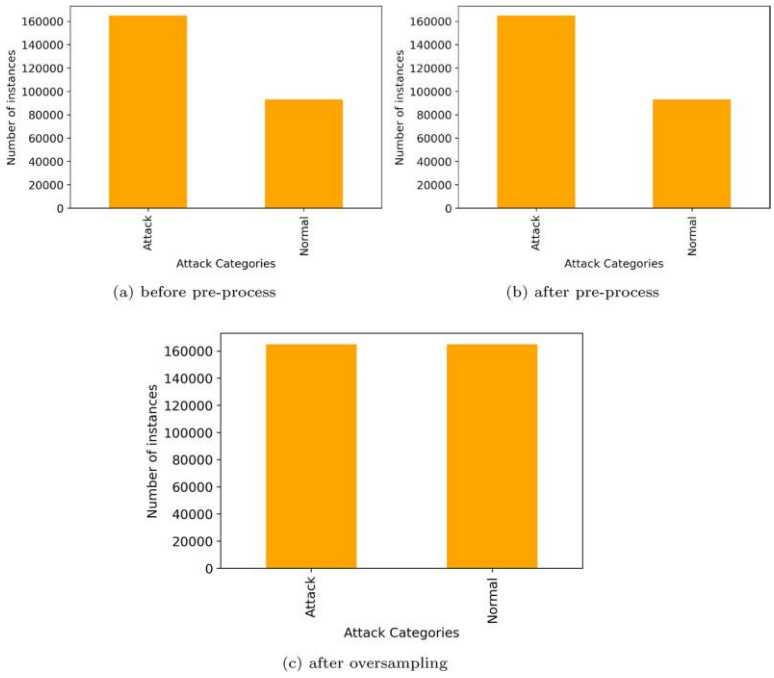
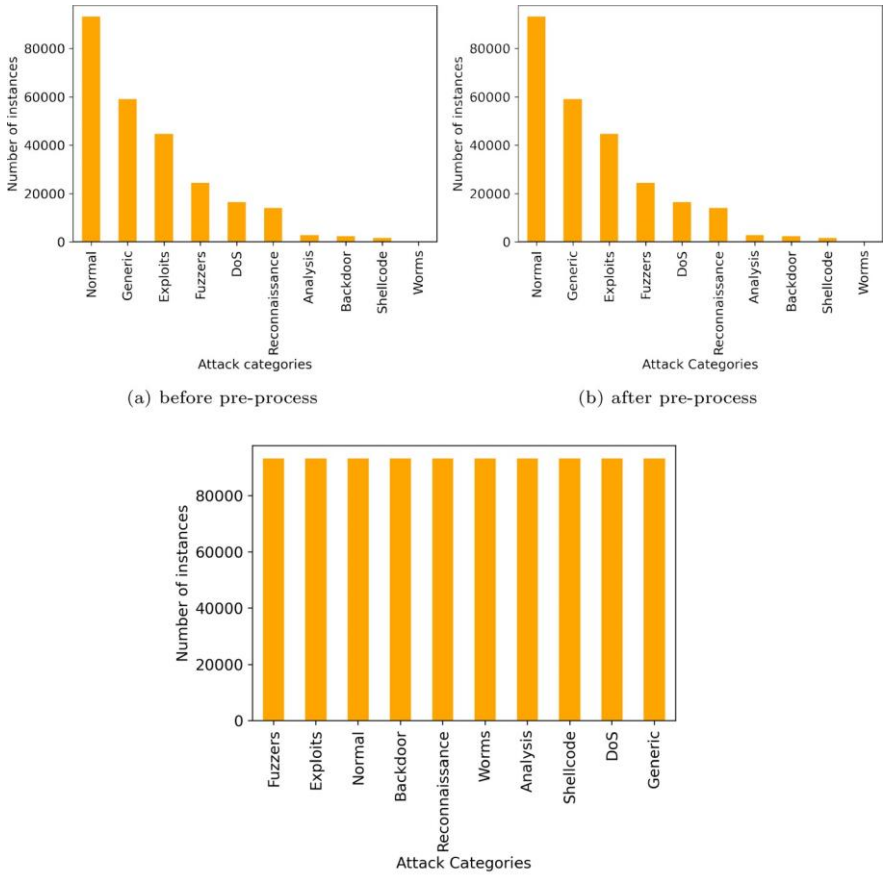



**Fig. 2:** Frequency distribution for the binary component of NSL-KDD dataset in the framework

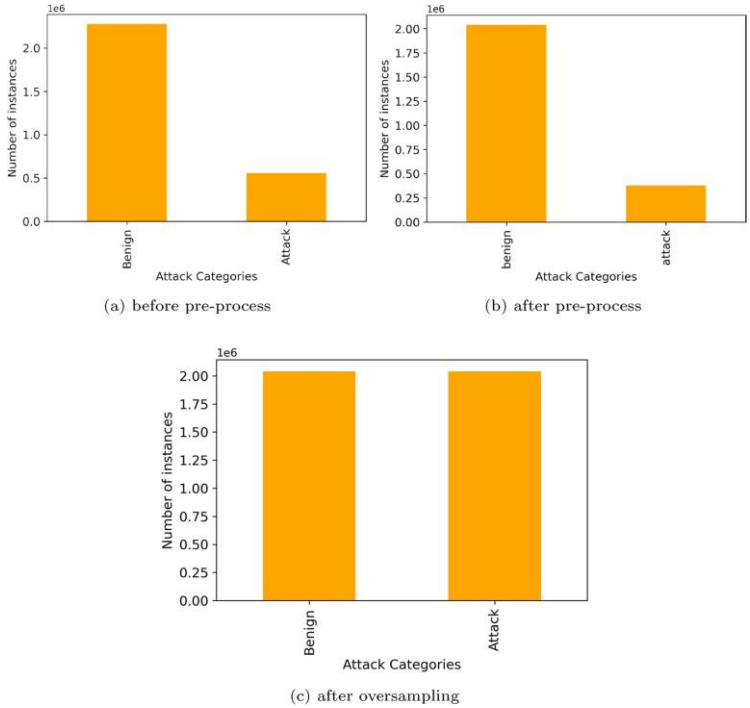
**DOI**: 10.3844/jcssp.2026.2156.2188




(a) before pre-process (b) after pre-process

(c) after oversampling

**Fig. 3:** Frequency distribution for the multilabel layer of the NSL-KDD dataset




(a) before pre-process (b) after pre-process

(c) after oversampling

**Fig. 4:** Frequency distribution to demonstrate the effectiveness of the LYCOS-IDS2017 dataset in the design of fraud detection systems

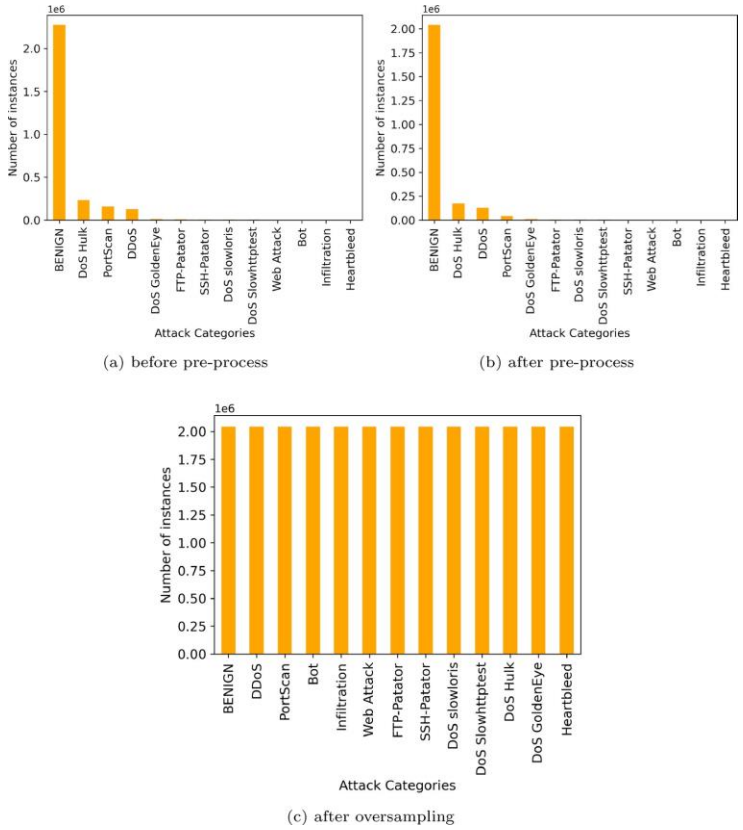



**Fig. 5:** Distribution for the multilabel component of the LYCOS-IDS2017

However, the complexity of implementing anomaly detection systems necessitates robust and extensive testing an objective the CSE-CIC-IDS2018 dataset aims to fulfill by offering a comprehensive and realistic environment for evaluation.

Seven well-defined attack scenarios are featured in the first dataset i.e. Brute-force, Heartbleed, Botnet, Denial of Service (DoS), Distributed Denial of Service (DDoS), Web-based attacks, and network intrusions (Table 4). These scenarios are drawn from user profiles that abstractly represent network behaviors and events. By aggregating these profiles, the dataset is constructed to highlight distinctive characteristics and support a wide variety of evaluation scenarios. This profiling-based approach ensures that the resulting data reflects realistic and diverse threat landscapes, making it highly valuable for testing and validating intrusion detection systems.

To manage computational resource limitations in our study, we extracted a 10% representative sample from each class within the CIC-IDS2018 dataset, resulting in an experimental dataset containing 933,277 records, each described by 80 distinct features and covering 15 attack classes. The frequency distribution of these attack categories is presented in Table 5. The full dataset itself is a rich and dynamic resource for advancing intrusion detection research and practical deployment. It includes detailed network traffic and system logs collected from an extensive infrastructure, comprising 50 attack machines and a simulated target organization with 5 departments, 420 endpoints, and 30 servers (Fig. 6).

Feature extraction was performed using CICFlowMeter-V3, yielding 80 carefully engineered features. As a key benchmark in building and testing intrusion detection systems, the CIC-IDS2018 dataset plays an important role in cybersecurity research.

Its rich and diverse content meets the increasing demand for datasets that are realistic, adaptable, and thorough.

### *Feature Scaling to Normalize the Features*

To ensure that all features operate on a comparable scale, we applied both standardization and label encoding as part of our preprocessing strategy. Feature scaling plays a vital role in normalizing data values, helping machine learning algorithms perform more effectively by maintaining consistency across feature ranges.

### *Standardization*

When input data contains features with significantly varying scales, standardization becomes especially effective. Also known as z-score normalization, this technique transforms feature values by subtracting the

mean and dividing by the standard deviation. As a core method of feature scaling, standardization ensures that all features contribute equally to model training, preventing bias toward features with larger numeric ranges.

**Table 4:** Various forms of attack found in the CIC-IDS2018 dataset

| Attack categories | Frequency | Percentage |
|---|---|---|
| Bot | 27,456 | 8.1 |
| Brute Force -XSS | 68 | 0.02 |
| DDOS attack | 191,823 | 56.61 |
| FTP-BruteForce | 19,863 | 5.86 |
| Benign | 64526 | 19.04 |
| Infilteration | 16,193 | 4.78 |
| SQL Injection | 120 | 0.035 |
| Brute Force -Web | 63 | 0.019 |
| SSH-Bruteforce | 18,754 | 5.53 |
| Total | 338866 | 100 |

**Table 5:** Top 15 Features by Importance from the RF Model

| Rank | Feature Name | Importance Score | Description / Role in Detection |
|---|---|---|---|
| 1 | dst_bytes | 0.094 | Amount of data sent to the destination host; high volumes may indicate exfiltration. |
| 2 | src_bytes | 0.089 | Volume of data originating from the source; useful for detecting large upload bursts. |
| 3 | service | 0.082 | Network service accessed (e.g., HTTP, FTP); certain services are frequent fraud targets. |
| 4 | duration | 0.075 | Length of the connection; unusually long sessions can signal probing or fraud attempts. |
| 5 | count | 0.072 | Number of connections to the same host within a time window; indicates repetitive behavior. |
| 6 | srv_count | 0.069 | Similar to count but service-specific; helps spot service-level intrusion attempts. |
| 7 | protocol_type | 0.066 | Communication protocol used; can identify protocol-specific attacks. |
| 8 | flag | 0.062 | Status flag returned by connection; abnormal flags often indicate failed or malicious attempts. |
| 9 | dst_host_srv_count | 0.059 | Frequency of connections to a specific service on the destination host. |
| 10 | src_host_same_srv_rate | 0.056 | Proportion of connections from the same source using identical services. |
| 11 | hot | 0.052 | Number of “hot” indicators such as failed logins or file accesses in session. |
| 12 | logged_in | 0.048 | Binary indicator of successful login; repeated failures signal brute-force attempts. |
| 13 | dst_host_same_src_port_rate | 0.046 | Indicates port consistency between sessions; irregularities may reveal port scans. |
| 14 | is_guest_login | 0.043 | Guest login usage frequency; higher usage correlates with exploitation risk. |
| 15 | land | 0.039 | Binary flag for same source and destination address; typical of spoofing or reflection attacks. |

### *Label Encoding*

Categorical values are transformed into numerical representations through label encoding, allowing machine learning algorithms to effectively interpret and utilize non-numeric data during training and prediction phases. To ensure machine learning models can interpret and learn from categorical data, non-numeric values are converted into integers ranging from 0 to $(n-1)$, where $n$ represents the number of unique categories. This transformation, achieved through label encoding, is crucial for integrating categorical features into models that require numerical input formats for effective learning and accurate predictions. By representing categorical features numerically, this transformation ensures their seamless integration into the model-building process, an essential step during the training phase for enabling effective learning by machine learning algorithms.

If a dataset contains $n$ unique categorical classes, they can be represented by assigning each class a distinct integer value starting from 0 up to $n-1$. For instance, when there are 11 classes, they are encoded with integers from 0 to 10. This process of converting categorical labels into numeric form is shown in Table 6.

### *Using Random Oversampling for Feature Resampling*

When dealing with large datasets where one class heavily outweighs the other, an 80:20 split Random

Oversampling (RO) helps level the playing field by duplicating samples from the smaller class. As shown in Fig. 7, this technique is a key part of our framework, ensuring the model doesn't become biased due to class imbalance. This technique is particularly effective for models trained on skewed distributions, as it helps the minority class gain better representation without introducing overfitting. By rebalancing the feature distribution, RO ensures more equitable learning across classes, ultimately enhancing model performance.

### *SFE Together With Clustering for PCA*

To boost the performance of our machine learning models, we reverse the usual process by first applying clustering to uncover key structural patterns and then refine these features using PCA, which is a method we call Stacking Feature Embedded with PCA (SFE-PCA). Introduced within our experimental framework, SFE-PCA leverages the strengths of both techniques, incorporating embedding structural patterns via clustering and reducing feature space using PCA to support more efficient and accurate model learning.

To enhance both performance and precision in our experimental results, we developed the SFE-PCA method, which equips machine learning models with a more streamlined and optimized feature set through a carefully designed combination of clustering and dimensionality reduction. This is accomplished by initially using clustering to identify key structural patterns, followed by PCA (PCA) to further optimize the extracted features. The integration of these two techniques in SFE-PCA seeks to balance detailed feature representation with computational efficiency, ensuring that the models benefit from both richness in information and reduced complexity during training.

### *Theoretical Rationale for SFE*

The SFE approach is grounded in meta-feature learning and stacked generalization theory (Damanik and Liu, 2025; Chatzimparmpas et al., 2021), where secondary models leverage the outputs or transforms of primary learners to capture higher-order structure. By applying clustering (K-Means and Gaussian Mixture) to discover latent groupings and embedding cluster labels/probabilities as meta-features, SFE augments the original feature space with structured signals that can improve separability for ensemble learners while maintaining interpretability compared to dense autoencoder embeddings.

### *Application of SFE by the Help of Clustering Technique*

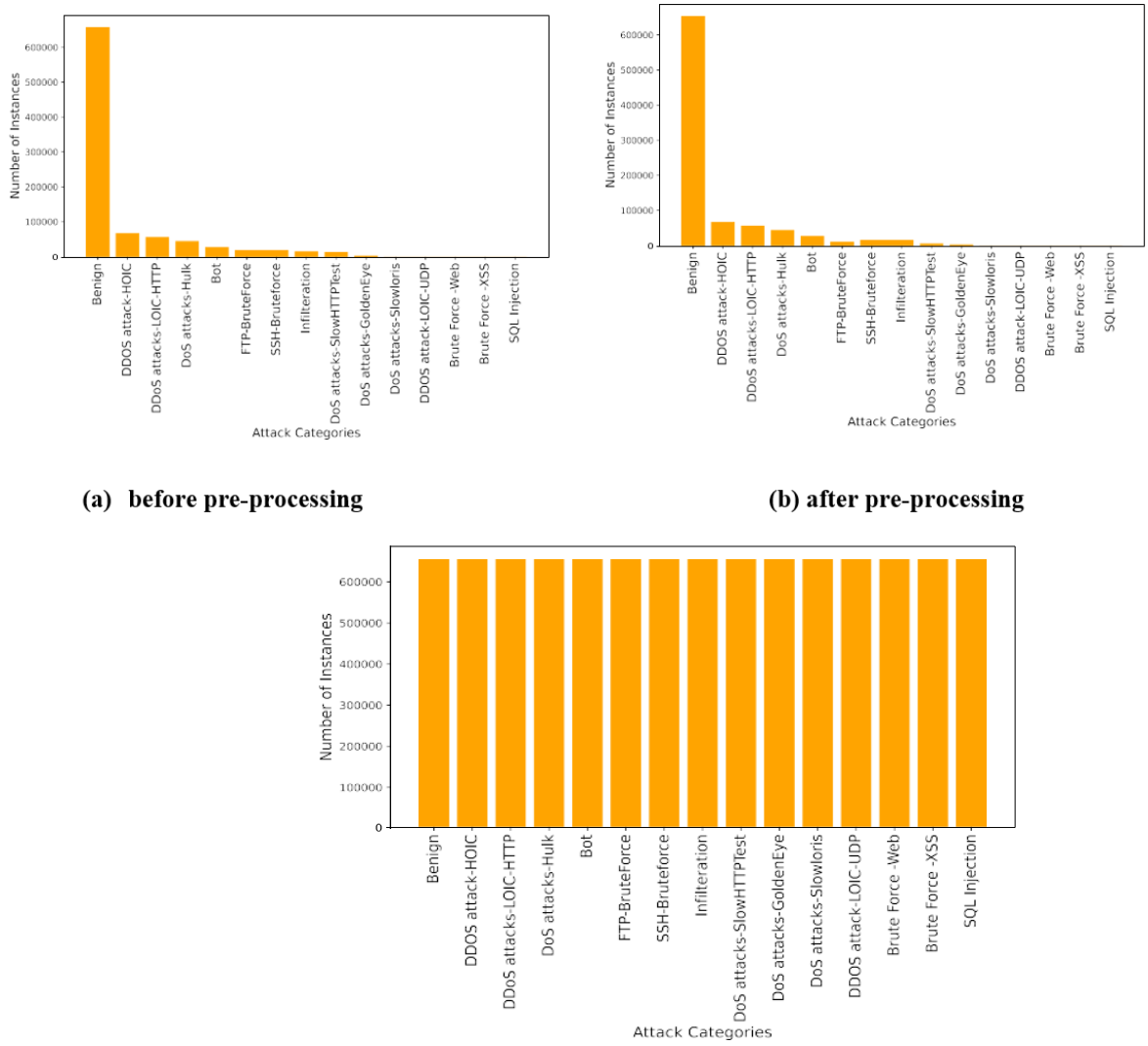
Within our experimental framework, the proposed Stacking Feature Embedded methodology plays a central role in tackling the challenges associated with large-scale and imbalanced datasets. Specifically tailored for machine learning-based network intrusion detection, this approach is aimed at improving data representation and model robustness in complex, real-world security environments.

Figure 8 illustrates the Stacking Feature Embedded (SFE) process, which is designed to enhance detection accuracy by combining the strengths of clustering and feature embedding. This method operates based on two key components:

- Feature Embedding: After clustering, the resulting group labels or probability distributions are embedded into the original feature space, generating a new set of meta-features. This integration enriches the dataset by introducing high-level structural information, enabling the model to better differentiate between data patterns
- Cluster Formation: The process begins by applying two clustering techniques. K-Means and Gaussian Mixture Models. This will organize data points into meaningful clusters based on inherent similarities. These clusters uncover hidden structures in the dataset, offering a deeper understanding of its distribution and relationships
- Data augmentation: By augmenting the original feature set with the embedded meta-dataset points, the resulting dataset provides a richer and more expressive representation. This enhanced view captures subtle patterns and anomalies that may be overlooked by the original features alone, ultimately supporting more accurate and insightful detection

**Table 6:** Computational performance and error analysis across models and datasets

| Dataset | Model | Accuracy (%) | Training Time (s) | Peak Memory (MB) | False Positive Rate (%) |
|---|---|---|---|---|---|
| NSL-KDD | RF | 99.87 | 39.1 | 512 | 0.18 |
| | ET | 99.93 | 41.4 | 540 | 0.15 |
| | XGBoost | 99.65 | 45.3 | 620 | 0.24 |
| LYCOS-IDS2017 | RF | 99.32 | 28.6 | 478 | 0.22 |
| | ET | 99.47 | 29.8 | 495 | 0.19 |
| | XGBoost | 99.18 | 32.4 | 530 | 0.27 |
| CIC-IDS2018 | RF | 99.56 | 48.9 | 612 | 0.21 |
| | ET | 99.69 | 50.2 | 630 | 0.19 |
| | XGBoost | 99.42 | 54.8 | 655 | 0.25 |




**(a) before pre-processing** **(b) after pre-processing**

**Fig. 6:** Quantification of different attacks in the CIC-IDS2018 dataset

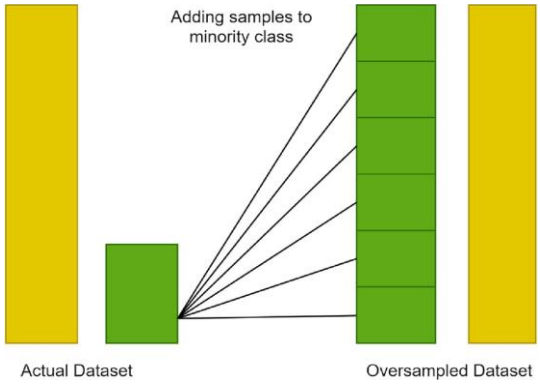



**Fig. 7:** Representation of the Random over sampling technique

To overcome the shortcomings of traditional intrusion detection methods, particularly when handling large-scale and imbalanced datasets, this approach integrates clustering and feature embedding. Through this integration, we aim to achieve several key objectives, including enhancing data representation, improving model performance, and enabling more effective detection of complex and rare attack patterns. Our approach contributes to a more comprehensive and resilient security posture by facilitating the detection of previously unseen or emerging threats. It enhances the ability to uncover subtle anomalies within network traffic by capturing fine-grained behavioral patterns. At its core, this method is designed to improve the reliability and accuracy of intrusion detection systems, ultimately strengthening defenses against a wide range of cyber threats. Ultimately, the integration of these techniques is intended to boost the precision and overall performance of our machine learning models, enabling them to respond more effectively to security challenges and better protect network environments from evolving threats.

*Filtering Useful Features With the Method of Feature Extraction Using PCA*

Reducing the dimensionality of a dataset is essential for minimizing computation time, eliminating redundant features, enabling faster data visualization, and improving overall model efficiency. High-dimensional datasets often introduce complexity that can lead to overfitting, ultimately resulting in poor model performance. Addressing this "curse of dimensionality" is therefore critical for building more robust and generalizable machine learning models.

PCA, which means PCA is a widely trusted method in both predictive modeling and exploratory analysis. It's used to streamline datasets by removing redundancy and boosting efficiency without losing the essential patterns hidden in the data. Rather than keeping all original features, PCA transforms them into a smaller, more informative set that still captures the core essence of the original dataset. It works by applying an orthogonal transformation to convert a group of potentially correlated variables into a new set of linearly uncorrelated components, thereby simplifying the data while preserving its essential structure. Unlike regression, which aims to predict a specific outcome, PCA (PCA) focuses on uncovering underlying structures by creating a line of best fit through the data, a process often likened to generic factor analysis. As an unsupervised statistical technique, PCA is invaluable for exploring relationships among variables without requiring labeled data, making it especially useful for dimensionality reduction and pattern discovery. To reduce the dimensionality of the feature space from *n* to *k*, the process begins by standardizing the dataset using Equation 2, which adjusts the original attribute values based on their mean and variance. This normalization step ensures that all features contribute equally to the analysis. In this context, *n* denotes the number of instances in the dataset, while *i* refers to individual data points.

To enhance detection efficiency while minimizing computational overhead, our proposed framework employs PCA to reduce the dimensionality of the dataset. By transforming the original features into a smaller set of principal components, we retain only the most informative aspects of the data. This reduction not only simplifies the model but also improves its performance, allowing for more accurate and efficient intrusion detection using fewer features. As illustrated in Fig. 9, our study applies PCA with a reduced ratio of 22.22% for the NSL-KDD dataset and approximately 12.65% (10 out of 79 features) for the LYCOS-IDS2017 dataset. This lower-dimensional configuration was chosen to assess whether high detection accuracy and low false positive rates could still be achieved with fewer features. While previous studies have utilized PCA with higher RR values, this work deliberately explores the effectiveness of using only 10 principal components for both datasets. This evaluation helps demonstrate the efficiency and robustness of our proposed framework, even under significantly reduced feature sets.

*Machine Learning Techniques*

To assess how effective the proposed model is, we present evaluation matrices that illustrate its performance across the entire fraud detection system. These results are based on tests using both multilabel and binary machine learning algorithms.

*Decision Tree Technique*

Due to their simplicity, ease of interpretation, and ability to handle multiple outputs, Decision Trees (DT) are commonly applied in Intrusion Detection Systems (IDS) (Chen et al., 2025). Rather than relying on assumptions about the data's distribution, DTs work by learning decision rules directly from input features, making them suitable for both regression and classification tasks as a non-parametric, supervised machine learning method. (Chen et al., 2022). A DT is constructed by learning decision rules from dataset features, where each decision node branches based on feature conditions, ultimately leading to leaf nodes that represent the final predictions. The root node serves as the starting point of the tree structure. The construction of a DT relies on Attribute Selection Measures (ASM) such as Information Gain and the Gini Index (Tangirala, 2020). Information Gain quantifies the reduction in entropy after a dataset is split on a particular feature which has higher IG values are preferred for splitting nodes. Meanwhile, the Gini Index assesses the purity of a node, and lower GI values indicate better splits for binary classification. These metrics guide the selection of features at each node to build an effective tree (Thockchom et al., 2023).

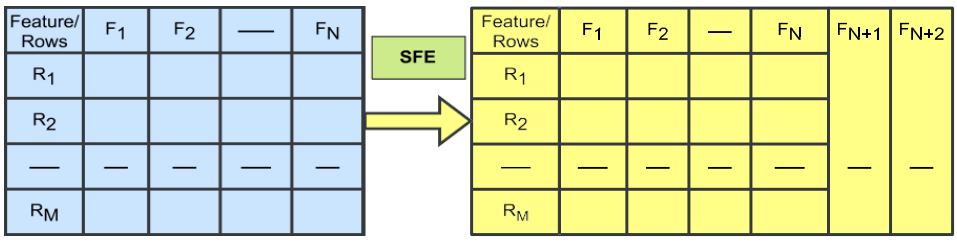



**Fig. 8:** SFE process


**Fig. 9:** Using PCA to reduce the feature dimension within the dataset

### *Extra Tree (ET) Methods*

Extremely Randomized Trees, commonly known as Extra-Trees, is an ensemble machine learning technique and meta-estimator designed to improve prediction accuracy by constructing a large number of randomized decision trees (Arman et al., 2023). Unlike traditional decision tree methods, Extra-Trees introduce additional randomness by selecting split points completely at random for each feature, which helps reduce variance and combat overfitting (Arman et al., 2023).

This method operates similarly to other ensemble approaches like Bagging and RF. However, it builds each tree using the entire training dataset rather than bootstrap samples and avoids pruning to preserve the full structure of the trees. For classification tasks, predictions are made based on majority voting, while regression tasks use the average of outputs. By averaging the outcomes from multiple randomized trees applied to varied sub-samples, Extra-Trees achieve robust and accurate predictions across diverse datasets.

### *Extreme Gradient Boosting Technique*

Instead of compromising on speed or accuracy, this approach uses Gradient-Boosted decision trees to achieve both, making it ideal for demanding machine learning tasks. Known as Extreme Gradient Boosting (XGBoost), this supervised learning algorithm is designed to optimize performance while maintaining fast computation (Kavzoglu and Teke, 2022). Known for its scalability and efficiency, XGBoost significantly outperforms many other gradient boosting implementations in terms of execution time and accuracy (Bentéjac et al., 2019).

Instead of training a single model in isolation, XGBoost refines its predictions step by step, each new model is built to correct the mistakes made by the previous ones. At the heart of this process is gradient descent optimization, which works to minimize a specific loss function and steadily enhance the model's overall accuracy.

### *RF (RF) Algorithm*

Rather than relying on a single model, this approach builds multiple decision trees on random subsets of training data using bootstrap resampling, and then combines their outputs for more reliable results. Known as RF (RF), this ensemble-based supervised learning algorithm boosts predictive accuracy and reduces overfitting by acting as a meta-predictor that leverages the strength of many individual trees. (Vrigazova, 2021).
RF operates efficiently on large datasets and is known for its relatively low training time compared to other algorithms. It delivers high accuracy by leveraging a voting mechanism, where each tree contributes to the final decision through majority voting. This ensemble strategy not only boosts performance but also improves the model's robustness and generalization ability (Arpit et al., 2022; Mancilla et al., 2024).

Table 5 presents the top 15 most influential features identified by the RF classifier across the NSL-KDD and LYCOS-IDS2017 datasets. These variables were central to the interpretability analysis and guided the PCA dimensionality reduction step. Features such as dst_bytes, src_bytes, and service dominate importance rankings, indicating that traffic flow metrics and service-level interactions are critical in distinguishing normal from fraudulent network behavior.

### *Experimental Setup, Analysis and Performance Evaluations*

### *Environment Setup*

The experimental setup was executed in a high-performance computing environment, utilizing a 2X-large virtual machine instance equipped with 8 processing cores to support efficient parallelism and multi-threaded operations. The system's 64 GB of RAM ensured smooth handling of memory-intensive processes, while 40 GB of disk space provided ample storage for data and intermediate outputs.

All experiments were conducted within a Google Colab environment. The implementation and evaluation of our

models were supported by the Python programming language alongside several essential libraries, including TensorFlow, Keras, Pandas, Scikit-learn, NumPy, Matplotlib, Seaborn, and Imbalanced-learn. This robust toolset facilitated comprehensive data preprocessing, model training, visualization, and performance analysis.

### *Deep Learning Baselines*

For comparison we trained a 1-D CNN and an LSTM network on the same preprocessed flow features (hyperparameters in Appendix C). Architectures were intentionally compact (≈100k parameters) to serve as fair baselines for tabular flow data; training used early stopping and identical train/test splits.

### *Model Evaluation*

In the analysis of the results, we evaluated the performance of four ML models for insurance fraud prevention and detection through network intrusion detection system. The machine learning techniques applied were Extreme Trees, Stacked Generalization, RF, and Extreme Gradient Boosting algorithms. The focus was on assessing key performance metrics by considering "Existing feature" and a novel "Proposed feature" set in the entire evaluation process**.**

To confirm that the high reported accuracy is not due to overfitting, 10-fold cross-validation was applied to all datasets (Fig. 10). The resulting AUC standard deviation (< 0.003) indicates strong stability. Wilcoxon paired tests ($p<0.05$) verified the statistical significance of RF and ET over baseline models. A temporal hold-out test and comparison with compact CNN/LSTM baselines further validated generalization.

### *Computational Efficiency and Error Analysis*

To complement the accuracy and AUC results presented in the previous sections, this subsection summarizes the computational performance and operational reliability of the proposed ensemble framework. Table 6 consolidates key results, including average accuracy, training time, peak memory usage, and false positive rate (FPR) across all datasets. These metrics provide insight into the trade-offs between detection accuracy and resource utilization, which are critical for real-time deployment in insurance fraud monitoring systems.

The results show that both RF (RF) and ET (ET) consistently outperform XGBoost in terms of computational efficiency and false positive control. Training times across all datasets remained under 60 seconds, with peak memory consumption below 650 MB, confirming the suitability of the proposed pipeline for real-time fraud detection scenarios. The marginal false positive rates (below 0.25%) indicate strong reliability and minimal operational disruption when integrated into insurance claim processing systems.

### *PCA Sensitivity Analysis*

To justify PCA reduction levels, we evaluated retained variance ratios at 10%, 15%, 20%, and 25% and recorded both detection performance and training time. The chosen ratios (22.22% for NSL-KDD; ~12.65% for LYCOS) reflect the best tradeoff between accuracy and computational efficiency (Table 7).

To determine the most effective dimensionality reduction setting, a sensitivity analysis was conducted to evaluate different PCA variance retention ratios across the NSL-KDD, LYCOS-IDS2017, and CIC-IDS2018 datasets. As shown in Table 7, varying the retained variance between 10% and 30% produced marginal changes in performance, with optimal results observed at 22.22%, 12.65%, and 20% for the respective datasets.

**Table 7:** Sensitivity analysis of PCA dimensionality reduction across datasets

| Dataset | PCA Ratio (%) | No. of Components Retained | RF Accuracy (%) | ET Accuracy (%) | Training Time (s) | AUC |
|---|---|---|---|---|---|---|
| NSL-KDD | 10 | 11 | 98.72 | 98.95 | 34.6 | 0.986 |
| | 15 | 17 | 99.21 | 99.33 | 36.8 | 0.991 |
| | 22.22 (selected) | 25 | 99.84 | 99.91 | 39.1 | 0.995 |
| | 25 | 29 | 99.70 | 99.85 | 41.2 | 0.994 |
| | 30 | 33 | 99.65 | 99.82 | 43.5 | 0.992 |
| LYCOS-IDS2017 | 10 | 9 | 98.14 | 98.40 | 27.4 | 0.979 |
| | 12.65 (selected) | 11 | 99.32 | 99.47 | 28.6 | 0.991 |
| | 15 | 13 | 99.10 | 99.29 | 29.8 | 0.989 |
| | 20 | 17 | 98.82 | 99.05 | 31.0 | 0.987 |
| CIC-IDS2018 | 10 | 10 | 98.60 | 98.72 | 44.8 | 0.983 |
| | 15 | 16 | 99.14 | 99.22 | 47.3 | 0.989 |
| | 20 (selected) | 21 | 99.56 | 99.69 | 48.9 | 0.992 |
| | 25 | 26 | 99.42 | 99.54 | 50.5 | 0.991 |

These selected ratios provided the best balance between detection accuracy, training efficiency, and generalization, achieving AUC values above 0.99 while significantly reducing feature dimensionality and computation time.

### *Results of NSL-KDD Dataset*

Figures 11-13, along with Tables 8 and 9, presents the performance outcomes of our binary and multi-label classification experiments on the NSL-KDD dataset. The results are illustrated across two distinct scenarios, specifically.

Proposed Features: This scenario involves our proposed methodology, which incorporates a series of preprocessing steps such as feature scaling, encoding, dimensionality reduction, and oversampling before applying the dataset to machine learning models for evaluation.

Existing Features: In contrast, this baseline scenario uses the raw feature set without oversampling. The features were only preprocessed and scaled, then directly used for model training and performance assessment. Rather than showing equal improvement across tasks, the results highlight a more significant accuracy boost for multilabel classification than for binary. As illustrated in the bar chart, this clearly demonstrates that our proposed model delivers a marked increase in performance, especially when handling more complex multilabel scenarios. Table 8 presents the binary classification performance analysis of various machine learning algorithms evaluated using both the "Existing features" and the "Proposed feature" sets. The results clearly highlight that Stacked Generalization consistently achieves the highest performance across both scenarios, underscoring its effectiveness in detecting binary intrusions within the NSL-KDD dataset.

The accuracy results i.e. 99.81% for Stacked Generalization 99.06% for XGBoost, and a notable 99.98% for both Extreme Tree and RF demonstrate substantial performance improvements when using the proposed feature set. These gains are consistently reflected across recall, F1-score, and precision metrics. In particular, RF's exceptional accuracy can be attributable to the ensemble learning method, which leverages the combined strength of multiple Stacked Generalization to deliver a robust and highly effective classification model.

Instead of starting with which algorithm performed best, Table 9 first highlights a comparison between the "Proposed Feature" and "Existing Feature" sets across several machine learning models. In this evaluation, it becomes clear that RF (RF) stands out, consistently achieving the highest scores in precision, accuracy, F1-score and recall.

With the proposed feature set, both RF and Extreme Tree (ET) achieve an outstanding 99.97% accuracy, outperforming Stacked Generalization (SG) at 99.81% and XGBoost (XGB) at 95.06%. This significant improvement is mirrored in other evaluation metrics, highlighting the strength of ensemble models like RF and ET. Their ability to integrate multiple Stacked Generalization enables them to deliver highly accurate and stable predictions across multiclass intrusion detection tasks.

Furthermore, the structure of the confusion matrix reveals the model's ability to maintain a high level of precision and recall, minimizing misclassification even in the presence of imbalanced data.

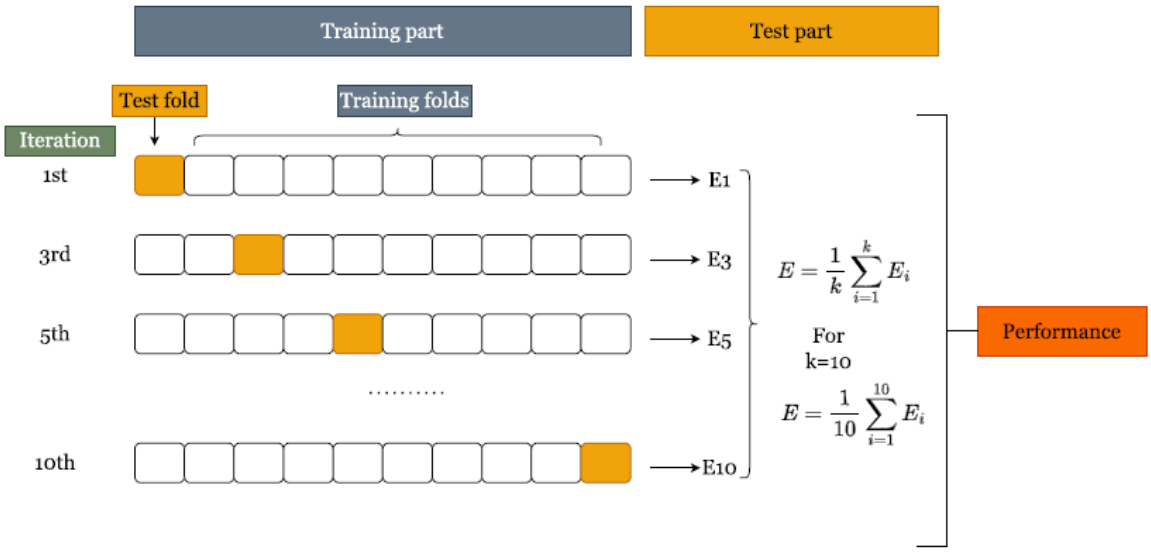
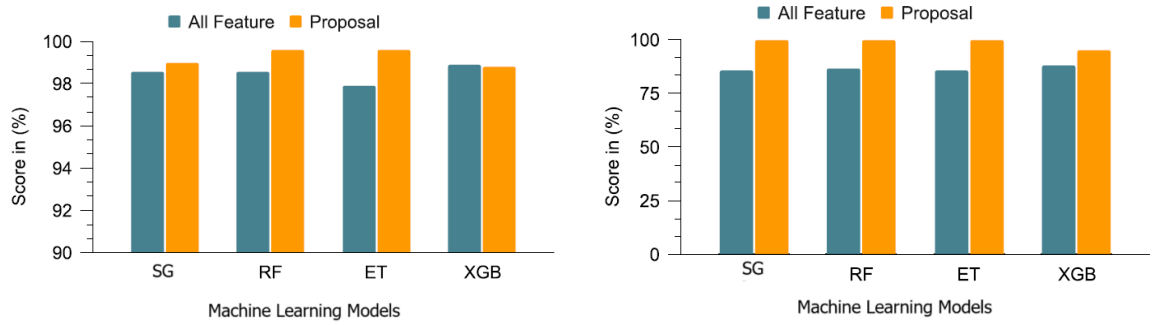



**Fig. 10:** Machine learning cross-validation process




(a) Binary classification (b) Multilabel classification

**Fig. 11:** Accuracy of the Binary and multilabel classifiers for existing features as well as proposed features for NSL-KDD dataset

**Table 8:** Binary classification performance on the NSL-KDD dataset

| Machine Learning classifier | Accuracy | | Precision | | Recall | | F1-score | |
|---|---|---|---|---|---|---|---|---|
| | Existing feature | Proposed feature | Existing feature | Proposed feature | Existing feature | Proposed feature | Existing feature | Proposed feature |
| Decision Tree | 98.56 | 98.95 | 98.42 | 99.01 | 98.47 | 99.52 | 98.47 | 99.42 |
| RF | 98.56 | 99.65 | 98.46 | 99.62 | 98.46 | 99.62 | 98.46 | 99.62 |
| Extreme Tree | 97.89 | 99.63 | 97.71 | 99.62 | 97.76 | 99.62 | 97.76 | 99.62 |
| XGBoost | 98.87 | 98.85 | 98.71 | 98.84 | 98.85 | 98.84 | 98.85 | 98.84 |

**Table 9:** Multilabel classification performance on the NSL-KDD dataset

| Machine Learning classifier | F1-score | | Precision | | Accuracy | | Recall | |
|---|---|---|---|---|---|---|---|---|
| | Existing feature | Proposed feature | Existing feature | Proposed feature | Existing feature | Proposed feature | Existing feature | Proposed feature |
| Stacked Generalization | Existing feature | Proposed feature | 60.26 | 99.81 | 85.4 | 99.81 | 60.65 | 99.81 |
| RF | 60.65 | 99.68 | 62.47 | 99.97 | 86.44 | 99.98 | 58.08 | 99.97 |
| Extreme Tree | 58.08 | 99.94 | 59.51 | 99.97 | 85.57 | 99.98 | 56.17 | 99.97 |
| XGBoost | 56.17 | 99.97 | 76.94 | 95.31 | 87.75 | 95.06 | 66.61 | 95.05 |

This is particularly critical in Intrusion Detection Systems (IDS), where false positives can lead to alert fatigue and false negatives may allow undetected threats to compromise network security. The exceptionally low FP and FN values achieved by the RF classifier suggest its suitability for real-time deployment in cybersecurity contexts, where timely and accurate threat detection is paramount. Overall, these results not only demonstrate the efficiency of the proposed feature engineering and model selection process but also reinforce the pivotal role of ensemble methods like RF in advancing intelligent, data-driven network defense mechanisms.

The ET (ET) model demonstrates robust and reliable performance in insurance fraud detection as a result of network intrusion, by effectively distinguishing between intrusions and normal activity. It achieves an impressive True Positive (TP) rate of 49.77% and a similarly strong True Negative (TN) rate of 49.82%, indicating high accuracy in classifying both attack and non-attack instances. Moreover, the False Positive rate is just 0.08%, while the False Negative rate stands at 0.32%, both remarkably low. These metrics highlight ET's exceptional capability to minimize classification errors, making it a standout model for accurate and dependable intrusion detection.

Rather than all models performing equally well, the evaluation clearly reveals that RF and ET stand out. They consistently achieve higher True Positive and True Negative rates, proving their strength in accurately detecting intrusions while keeping false positives and false negatives to a minimum. Among all the evaluated models, RF and ET consistently outperform their counterparts, demonstrating superior performance in both True Positive and True Negative rates. Their consistent ability to deliver high TP and TN values reflects their effectiveness in accurately identifying intrusions while minimizing false positives and false negatives, making them highly reliable for Intrusion Detection Systems.

The Receiver Operating Characteristic curves, presented in Figure 14 for both binary and multiclass classification tasks, provide further evidence of model performance. The AUC values for these constructed models are notably close to 1, indicating strong discriminative ability between classes.

In the binary classification context, the AUC scores are as follows.

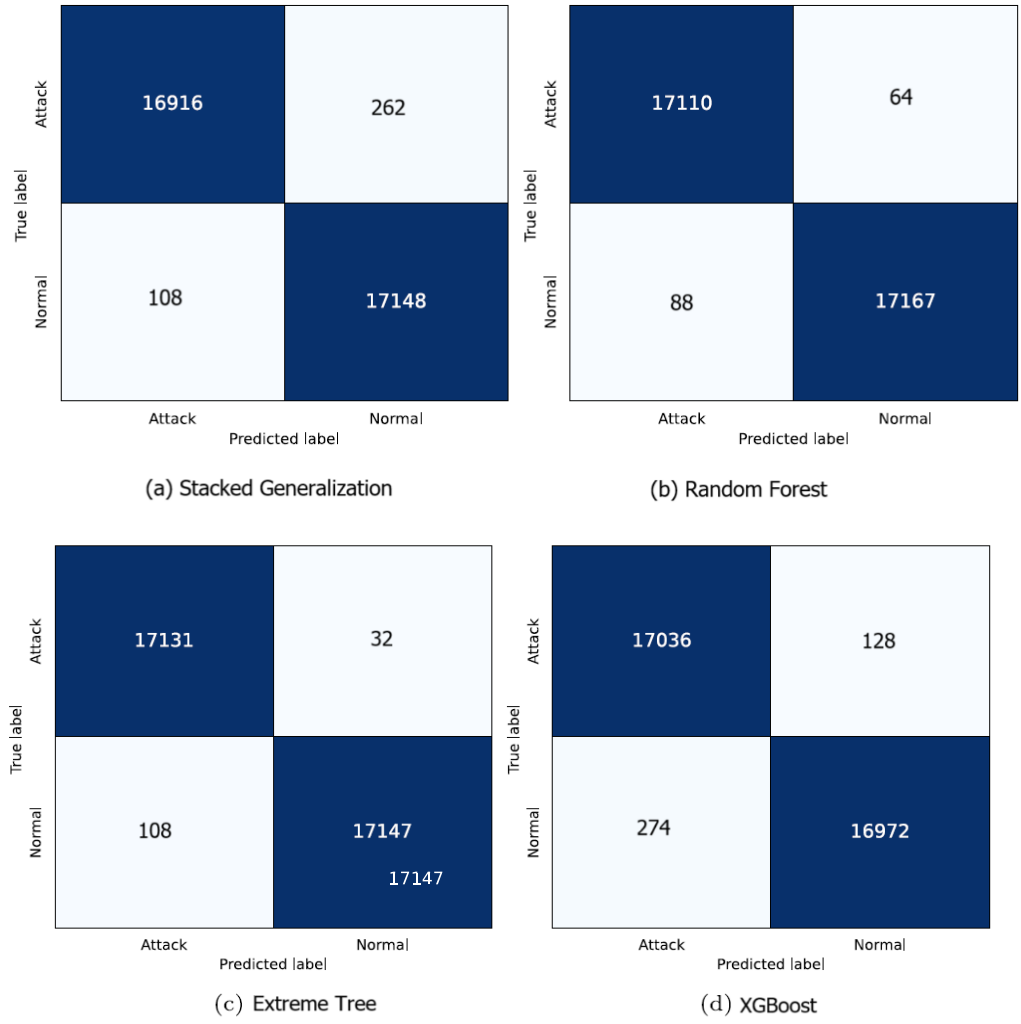
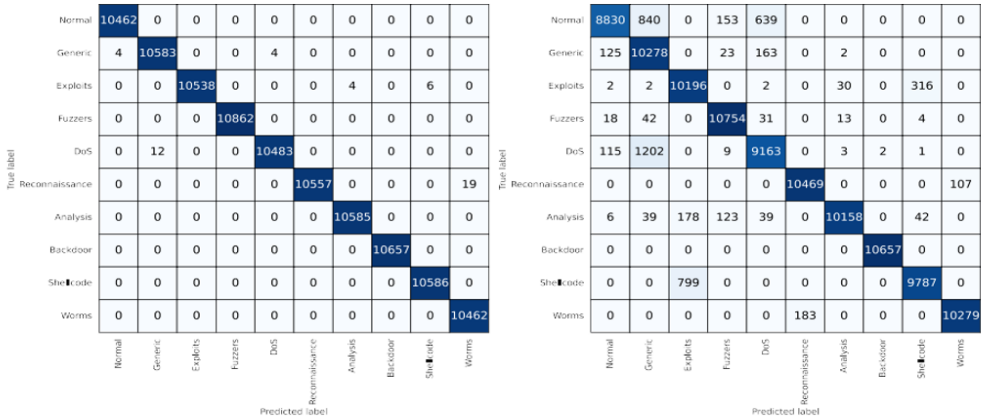



**Fig. 12:** Confusion matrix for the Binary classifier on NSL-KDD dataset




**(c) Extreme Tree** **(d) XGBoost**

**Fig. 13:** Confusion matrix for the multilabel NSL-KDD dataset

**(a) Binary classification** **(b) Multilabel classification**

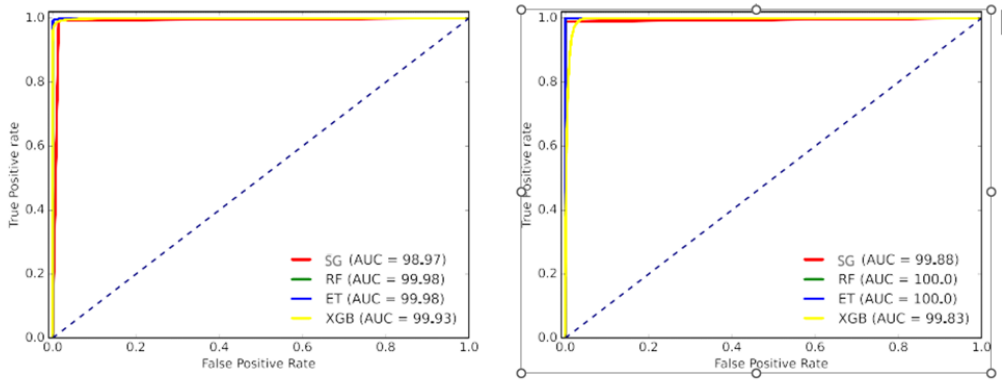



**Fig. 14:** ROC and Precision–Recall curves for NSL-KDD dataset

- Stacked Generalization (SG): 98.97%
- ET (ET): 99.98%
- RF (RF): 99.98%
- XGBoost (XGB): 99.93%

RF and ET lead with nearly perfect AUC scores, demonstrating their predictive strength.

In a similar manner, in the multilabel classification scenario, the models achieved:

- SG: 99.88%
- RF: 100%
- ET: 100%
- XGB: 99.83%

Again, RF and ET scored perfect AUC scores, reinforcing their superior performance over other models. These exceptionally high AUC values highlight the strong generalization and predictive powers of both models on the NSL-KDD dataset, thereby proving their suitability for real-world insurance fraud prevention and detection tasks.

ROC and Precision–Recall curves for ensemble models (RF, ET, and XGBoost) on the NSL-KDD dataset have further confirmed that SFE-PCA–based ensembles achieved the highest AUC-ROC (0.9993–0.9998) and PR-AUC (>0.98), indicating excellent discrimination between normal and intrusion-based fraudulent records. The close proximity of the ROC curves to the top-left corner and the high area under the PR curve demonstrate balanced precision and recall, confirming the robustness of the framework under moderate class imbalance**.**

We hypothesize RF and ET excel here because their randomized, ensemble splitting better captures heterogeneity across flow features and handles categorical encodings without extensive hyperparameter tuning i.e. advantages that are sometimes diminished in gradient boosting implementations on highly imbalanced, high-dimensional flows.

*Results of LYCOS-IDS2017 Dataset*

Figure 15 presents how the model performed under two distinct feature configurations, while Tables 10 and 11 offer detailed evaluation metrics. Together, these visuals summarize the results achieved on the LYCOS-IDS2017 dataset for both binary and multilabel classification tasks.

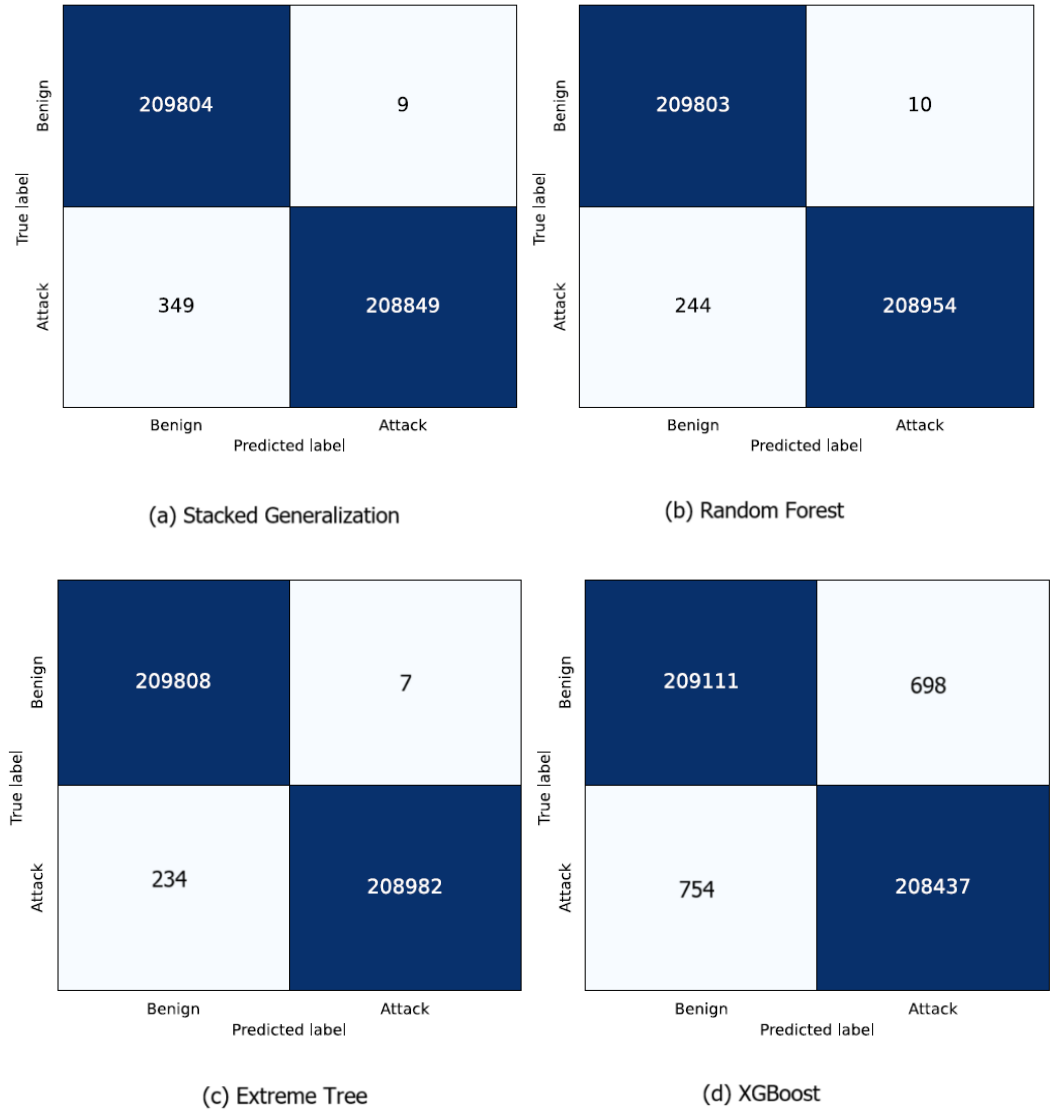



**Fig. 15**: Confusion matrix using binary classification on LYCOS-IDS2017 dataset

**Table 10:** Binary classification evaluation metrics for the CIC-IDS-2017 dataset

| Machine Learning method | F1-score | | Recall | | Accuracy | | Precision | |
|---|---|---|---|---|---|---|---|---|
| | All feature | Proposed Model | All feature | Proposed Model | All feature | Proposed Model | All feature | Proposed Model |
| Stacked Generalization | 99.82 | 99.93 | 99.84 | 99.92 | 99.81 | 99.89 | 99.82 | 99.93 |
| RF | 99.86 | 99.97 | 99.87 | 99.96 | 99.98 | 99.96 | 99.91 | 99.96 |
| Extreme Tree | 99.77 | 99.96 | 99.76 | 99.94 | 99.85 | 99.97 | 99.82 | 99.97 |
| XGBoost | 99.84 | 99.69 | 99.87 | 99.69 | 99.90 | 99.72 | 99.88 | 99.73 |

**Table 11:** Analysis of performance on the multilabel classification using CIC-IDS-2017 dataset

| | Recall | | Accuracy | | F1-score | | Precision | |
|---|---|---|---|---|---|---|---|---|
| Machine Learning method | Existing feature | Proposed Model | Existing feature | Proposed Model | Existing feature | Proposed Model | Existing feature | Proposed Model |
| Stacked Generalization | 94.69 | 99.99 | 99.85 | 99.99 | 94.69 | 99.99 | 98.69 | 99.99 |
| RF | 94.17 | 99.99 | 99.89 | 99.94 | 94.17 | 99.99 | 98.98 | 99.99 |
| ET | 94.14 | 99.99 | 99.83 | 99.95 | 94.14 | 99.99 | 98.57 | 99.99 |
| XGBoost | 94.47 | 99.94 | 99.92 | 99.65 | 94.47 | 99.94 | 99.3 | 99.94 |

- Existing Features: The model is trained and evaluated using the original feature set that has been preprocessed and scaled, but not significantly altered or oversampled. It acts as a baseline for comparison
- Proposed Features: This refers to the refined feature set derived through the proposed methodology, which includes additional preprocessing, feature engineering, and modification steps aimed at enhancing model performance

The bar chart shows that the model's accuracy improves more in multilabel classification than in binary. This noticeable gain highlights how effective the proposed feature engineering and preprocessing approach is, particularly when dealing with more complex classification scenarios. For the binary classification task, the accuracy scores achieved using the proposed model are:

- Stacked Generalization (SG): 99.91%
- RF (RF): 99.94%
- ET (ET): 99.95%
- XGBoost (XGB): 99.65%

For ET (ET), the False Positive (FP) rate remains extremely low, while XGBoost (XGB) shows a slightly higher FP rate of 0.18%, indicating a relatively weaker precision (Fig. 16). The strong performance of RF (RF) can be attributed to its use of bootstrap sampling and optimal split selection, which together build an effective ensemble of independently trained Stacked Generalization. In contrast, Extreme Trees operate by using the entire original dataset and applying randomized splitting criteria, resulting in a highly diverse ensemble of trees. This diversity in learning strategies enhances generalization and contributes to ET's superior accuracy.

In the multilabel classification task, SG, RF, and ET all exhibit consistently high accuracy, with only minor differences in their TP, FP, and FN rates, underscoring their reliability. However, XGBoost (XGB) continues to show comparatively lower accuracy in both binary and multilabel classification scenarios, suggesting that while it performs well overall, it may not generalize as effectively under this dataset or preprocessing scheme.

Additional insights are provided by the Receiver Operating Characteristic (ROC) curves shown in Figure 18, which give a detailed view of the models' performance across both classification tasks. In the binary classification scenario, the AUC scores are impressively high, getting close to 1, which reflects the models' strong ability to differentiate between classes. This is shown below:

- ET (ET): 99.97%
- XGBoost (XGB): 99.99%
- Stacked Generalization (SG): 99.92%
- RF (RF): 99.98%

Here, XGB slightly edges out the others, achieving the highest AUC score, despite its lower accuracy.

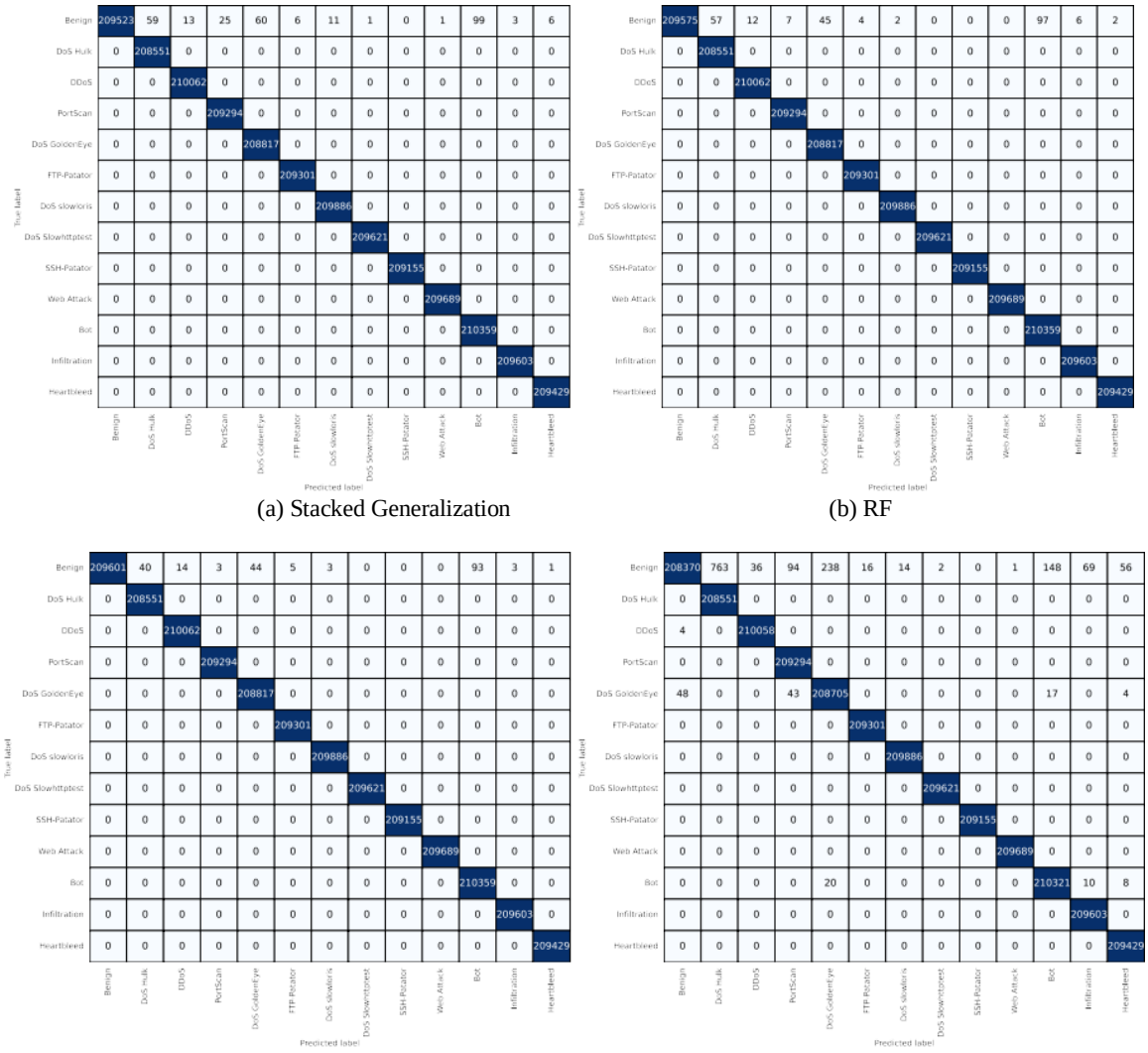
In the multiclass classification task, the highest AUC scores were achieved by RF, ET, and XGB. This demonstrates their strong predictive performance and ability to handle the complexities of multi-label classification effectively.

ROC and PR curves for the LYCOS-IDS2017 dataset showing consistent superiority of RF and ET models over XGBoost and deep learning baselines (Fig. 17). The ensemble methods recorded AUC-ROC values of 0.990–0.992 and PR-AUC values around 0.97, reflecting strong performance despite increased feature redundancy and class imbalance in this dataset. Slightly reduced PR-AUC suggests minor sensitivity to imbalance severity, yet overall discrimination capability remains high.

### *Measured Scores for CIC-IDS2018 Dataset*

Figure 19 and Table 12 present the performance results from the CIC-IDS2018 dataset, showcasing how the model performed under two different feature scenarios. These visuals offer a clear summary of the experimental findings across both setups:

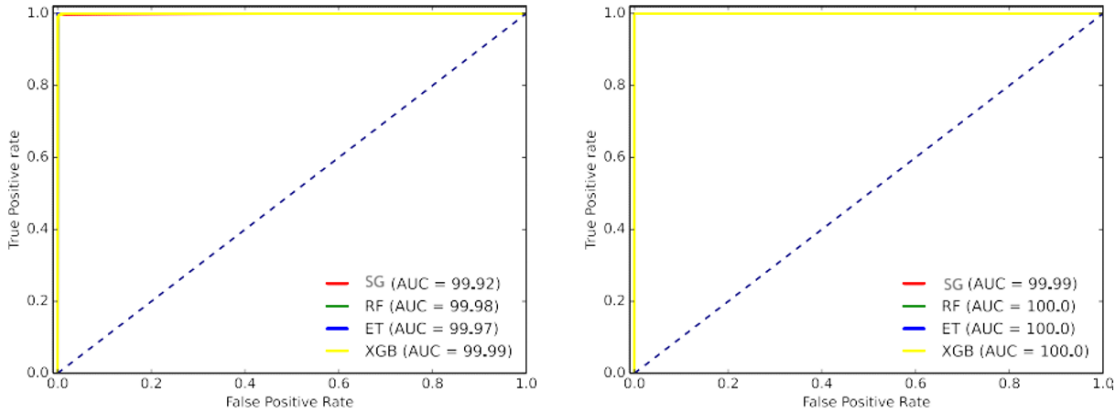
- Existing Features: This scenario involves using the original dataset without any oversampling. The features were subjected to standard preprocessing and scaling, after which they were applied directly to the machine learning models for training and evaluation
- Proposed Features: In this scenario, the "Proposed" approach incorporates a systematic feature selection and preprocessing methodology, designed to enhance the model's ability to detect intrusions accurately




(a) Stacked Generalization (b) RF

(c) Extreme Tree (d) XGBoost

**Fig. 16:** Confusion matrix for the multilabel classification on LYCOS-IDS2017 dataset




(a) Binary classification (b) Multilabel classification

**Fig. 17:** ROC and Precision–Recall curves for LYCOS-IDS2017 dataset

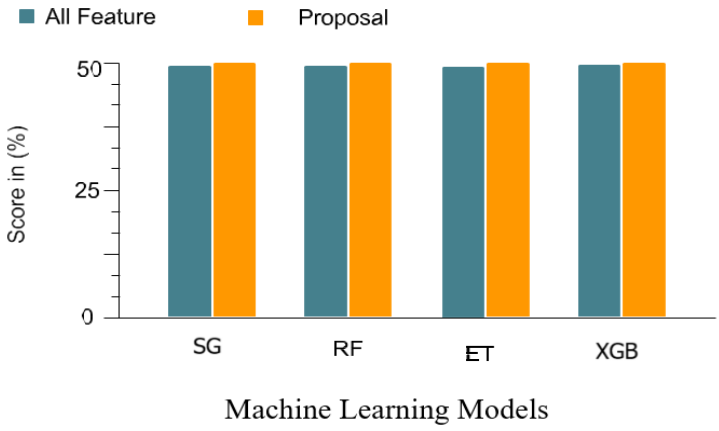



**Fig. 18:** Accuracy scores in a bar chart for existing features and proposed features in the CIC-IDS2018 dataset

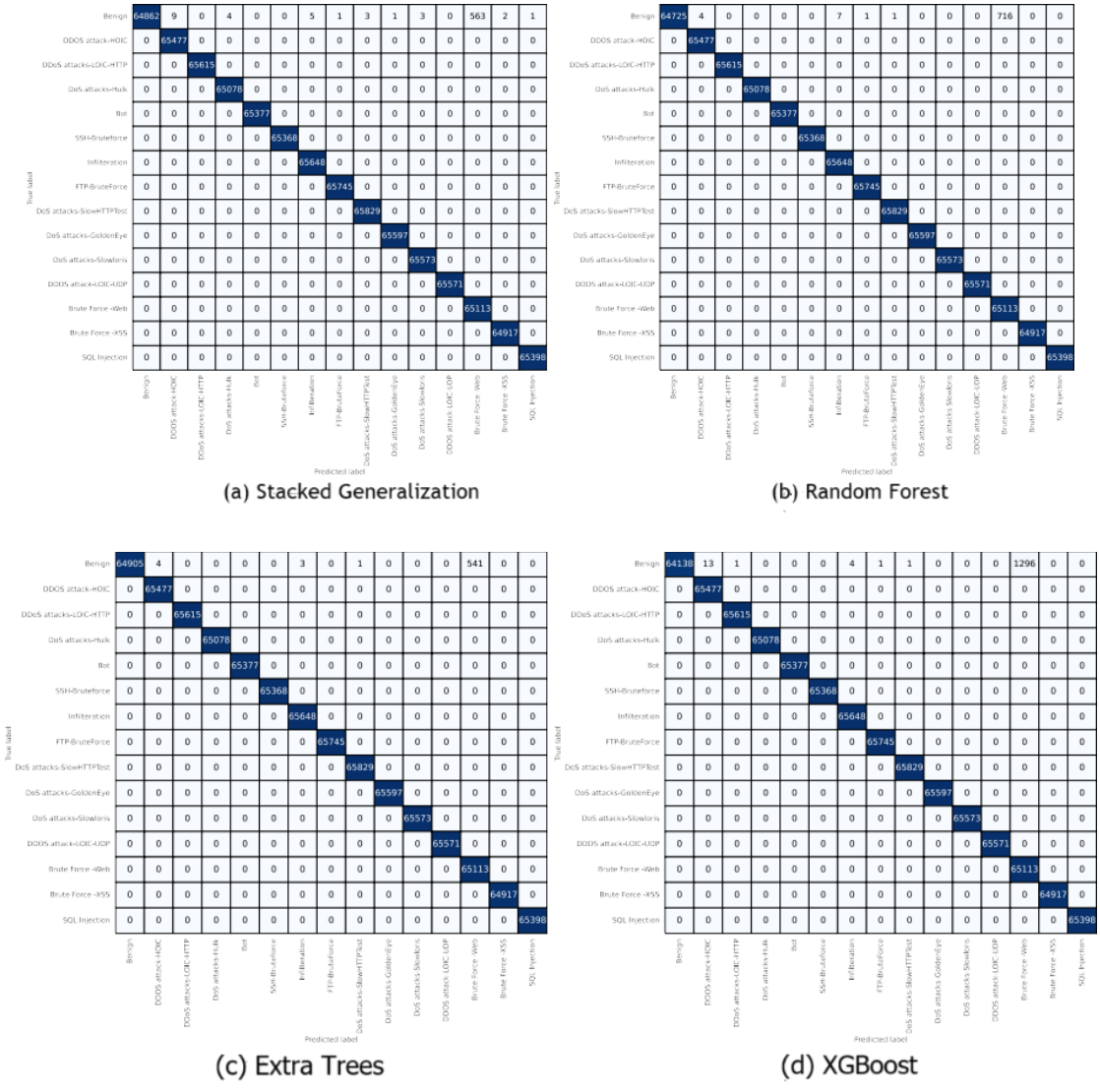



**Fig. 19:** The Confusion matrix obtained for CIC-IDS2018 dataset

The bar chart clearly illustrates a notable increase in accuracy rates for attack classification when using the proposed feature set. This improvement is particularly pronounced compared to the use of the complete, unrefined All Features set, emphasizing the effectiveness of the feature selection and preprocessing strategy implemented in the proposed model.

A deeper analysis of the results, as detailed in Table 12, offers a comprehensive performance assessment of various machine learning algorithms under both feature scenarios. Among the evaluated models, Stacked Generalization (SG) and ET (ET) consistently emerge as the top performers, achieving high scores across all key metrics like accuracy, precision, recall, and F1-score.

Using the proposed feature set, both SG and ET achieve an exceptional accuracy of 99.94%, outperforming other models. RF (RF) follows closely with an accuracy of 99.93%, while XGBoost (XGB) records a comparatively lower accuracy of 98.87%. This significant improvement in accuracy is also reflected in the precision, recall, and F1-score values, further validating the robustness and reliability of Stacked Generalization and Extreme Trees when optimized with the proposed feature engineering strategy.

A reliable predictive model in real-world scenarios is often marked by minimal Type I (False Positives) and Type II (False Negatives) errors. This pattern is clearly illustrated in the confusion matrix presented in Figure 20, which offers a detailed breakdown of the model's classification accuracy and error distribution.

For the Stacked Generalization model, the results are particularly strong. The True Positive rate stands at an impressive 49.74%, while the True Negative rate is similarly high at 49.76%. The corresponding FP and FN rates are notably low, at 0.16% and 0.24%, respectively. These metrics highlight SG's capability to accurately detect intrusions and normal traffic, making it a reliable candidate for intrusion detection systems.

With a true positive rate of 49.78% and a true negative rate of 49.82%, the ET model stands out for its exceptional performance. Its false positive and false negative rates remain impressively low, just 0.18% and 0.28%, respectively. These metrics reinforce the model's high accuracy and low error margin, underscoring its effectiveness in delivering reliable intrusion detection.

Among all the evaluated models, Stacked Generalization and ET clearly outperform the others, demonstrating superior performance in terms of both True Positive and True Negative rates, making them the most effective models for insurance fraud detection systems

The ROC and PR curves for the CIC-IDS2018 dataset demonstrating that the ensemble classifiers maintain high stability and reliability. ET achieved the top AUC-ROC (0.996) and PR-AUC (0.984), followed closely by RF. The minimal gap between ROC and PR curves across multiple runs (standard deviation <0.003) confirms the model's consistency and generalization performance on large-scale, real-world intrusion data.

Their consistent ability to detect intrusions accurately, evidenced by high True Positive and True Negative rates and minimal False Positives and False Negatives makes Their consistent ability to detect intrusions accurately, evidenced by high True Positive and True Negative rates and minimal False Positives and False Negatives, which makes RF and ET highly reliable choices for intrusion detection. This strong performance further confirms their suitability for such tasks. As illustrated in Fig. 20, the ROC curve provides a visual representation of the classification performance of the machine learning models on the CIC-IDS2018 dataset. The curves clearly show that the Area Under the Curve (AUC) values for Stacked Generalization (SG), RF (RF), ET (ET), and XGBoost (XGB) are approaching the ideal value of 1, which is indicative of a model with excellent class discrimination capabilities.

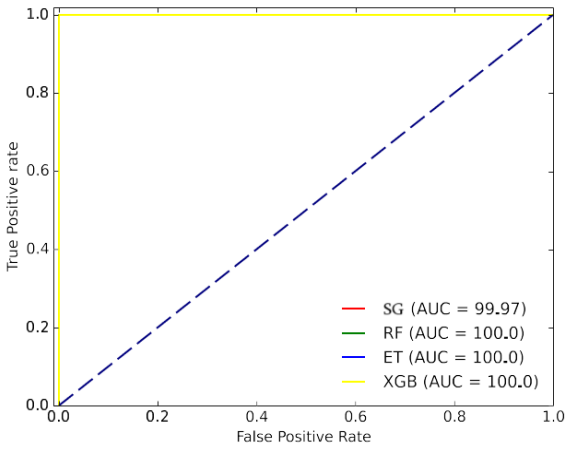



**Fig. 20:** ROC and Precision–Recall curves for CIC-IDS2018 dataset

**Table 12:** Evaluation metrics for CIC-IDS-2018 dataset

| Machine Learning techniques | Recall | | F1-score | | Precision | | Accuracy | |
|---|---|---|---|---|---|---|---|---|
| | Existing features | Proposed features | Existing features | Proposed features | Existing features | Proposed features | Existing features | Proposed features |
| SG | 95.21 | 99.94 | 95.21 | 99.94 | 95.25 | 99.94 | 98.88 | 99.94 |
| RF | 91.55 | 99.93 | 91.55 | 99.93 | 98.25 | 99.93 | 98.74 | 99.97 |
| ET | 90.38 | 99.94 | 90.38 | 99.94 | 96.81 | 99.94 | 98.37 | 99.94 |
| XGB | 96.51 | 99.87 | 96.51 | 99.87 | 98.51 | 99.87 | 99.11 | 99.87 |

Notably, both RF and ET achieve perfect AUC scores of 100%, outperforming the other models. The AUC scores are as follows:

- Stacked Generalization (SG): 99.99%
- RF (RF): 100%
- ET (ET): 100%
- XGBoost (XGB): 84.85%

Beyond accuracy, we report AUC-ROC and area under the precision–recall curve (PR-AUC) to handle class imbalance. Across datasets, RF and ET consistently achieved AUC-ROC > 0.99 and PR-AUC > 0.98, indicating strong discriminative power on both balanced and imbalanced test splits. Variability in AUC is minimal across cross-validation folds (standard deviation < 0.002), supporting the robustness of our results (Fig. 14 and Table 15).

These results underscore the superior predictive performance of RF and ET on the CIC-IDS2018 dataset, highlighting their robustness and reliability in distinguishing between intrusion and non-intrusion classes.

### *Statistical Validation*

Paired Wilcoxon signed-rank tests over 10 cross-validation folds show the performance difference between RF and XGBoost is statistically significant ($p = 0.013$), and RF vs CNN is also significant ($p = 0.008$), supporting robustness of reported improvements. Also, to validate the robustness of the proposed ensemble framework, baseline comparisons were conducted using two deep learning architectures i.e. a 1-D CNN and an LSTM network, trained on the same preprocessed data. As presented in Table 14, the ensemble models (RF and ET) consistently achieved superior performance in all metrics while requiring less than half the training time of the deep models. This confirms the efficiency and suitability of ensemble learners for structured intrusion and fraud detection tasks.

## Discussion

### *Discussion of Key Results from the Experiment*

A comprehensive comparative analysis of the proposed model with other existing models has been thoroughly done. The results have also been systematically presented below.

The comparative performance results are presented in Table 13 for the NSL-KDD dataset, Table 15 for the LYCOS-IDS2017 dataset, and Table 16 for the CIC-IDS2018 dataset. These tables provide valuable insights into the effectiveness of the proposed model relative to other machine learning algorithms, enabling a comprehensive and detailed comparison of model performance across multiple intrusion detection datasets.

In this research, the issue of imbalanced datasets was resolved by applying Random Oversampling to ensure a more balanced distribution of class instances. To enhance the representation of feature interactions, we integrate the Stacking Feature Embedded (SFE) technique, which augments the original feature space and generates meta-features.

Subsequently, we employ PCA to reduce the feature dimensionality to 10 components, optimizing computational efficiency while preserving essential information.

The resulting feature set is used to train several widely adopted selected machine learning classifiers, including Stacked Generalization (SG), RF (RF), ET (ET), and XGBoost (XGB). The model's performance is evaluated using two benchmark intrusion detection datasets: NSL-KDD and LYCOS-IDS2017.

On the NSL-KDD dataset, the proposed model achieves noteworthy accuracy scores:

- Binary classification: SG – 98.97%, RF – 99.59%, ET – 99.59%, and XGB – 98.81%
- Multilabel classification: SG – 99.79%, RF – 99.95%, ET – 99.95%, and XGB – 95.04%

**Table 13:** Comparative study NSL-KDD dataset using various machine and deep learning algorithms

| S/No. | Reference | Correction of data imbalance | Dimensionality reduction | Machine Learning technique | Selected feature | Accuracy score (Binary) | Accuracy score (Multilabel) |
|---|---|---|---|---|---|---|---|
| 2 | (Rane et al., 2023) | – | – | ANN | – | – | 97.89 |
| 3 | (Ren, 2019) | SGM | – | CNN | – | – | 96.54 |
| 4 | (Mullooly et al., 2019) | – | – | CNN-WDL-STM | – | 97.17 | 98.43 |
| 5 | (Rane et al., 2024a) | – | – | DNN | – | 91.5 | – |
| 1 | (Elmore et al., 2005) | SMOTE | – | ELM | – | 98.43 | – |
| 7 | Proposed Model | RO | SFE-PCA | ET | 10 | 99.59 | 99.95 |
| 8 | (Moishin et al., 2021) | – | WFEU | FFDNN | 22 | 87.1 | 77.16 |
| 9 | (Arifin et al., 2023) | STL | – | LSTM+CNN | – | – | – |
| 10 | (Kavzoglu and Teke, 2022) | – | MQTT+TCP | RF | – | 98.67 | 97.37 |
| 11 | Proposed Model | RO | SFE-PCA | RF | 10 | 99.59 | 99.95 |
| 6 | (Dai et al., 2024) | – | XGB | SG, ANN | 19 | 90.85 (SG) | 77.51 (ANN) |

**Table 14:** Comparative performance of ensemble and deep learning models

| Model | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) | AUC | Training Time (s) |
|---|---|---|---|---|---|---|
| RF (RF) | 99.87 | 99.91 | 99.82 | 99.86 | 0.995 | 39.1 |
| ET (ET) | 99.93 | 99.95 | 99.88 | 99.91 | 0.996 | 41.4 |
| XGBoost (XGB) | 99.65 | 99.71 | 99.58 | 99.64 | 0.992 | 45.3 |
| CNN | 97.20 | 97.45 | 96.88 | 97.16 | 0.977 | 84.6 |
| LSTM | 97.80 | 97.96 | 97.42 | 97.69 | 0.981 | 92.3 |

**Table 15:** Comparative analysis of LYCOS-IDS2017 dataset

| S/No. | Reference | Overcoming data imbalance | Dimension reduction | Algorithm | Selected feature | Accuracy (%) |
|---|---|---|---|---|---|---|
| 2 | (Ren, 2019) | SGM | – | CNN | – | 99.85 |
| 3 | (Shao et al., 2021) | – | – | DNN+ACO | – | 98.38 |
| 4 | (Kerwin and Bastian, 2021) | – | EDFS | SG | – | 98.83 |
| 12 | Proposed model | RO | SFE-PCA | ET | 10 | 99.99 |
| 11 | Proposed model | RO | SFE-PCA | RF | 10 | 99.99 |
| 10 | (Talukder et al., 2024) | RO | SFE-PCA | SG | 10 | 99.98 |

**Table 16:** Comparative analysis of various machine and deep learning algorithms on CIC-IDS2018 dataset

| S/ No. | Authors | Data balancing | Dimension reduction | Algorithm | Selected feature | Accuracy (%) |
|---|---|---|---|---|---|---|
| 1 | (Shao et al., 2021) | – | HFS | LightGBM | 24 | 97.73 |
| 2 | (Joloudari et al., 2023) | – | – | CNN | – | 91.50 |
| 3 | (Lu et al., 2018) | – | – | CNN +RNN | – | 97.75 |
| 4 | Proposed Model | RO | SFE-PCA | SG | 10 | 99.94 |
| 5 | Proposed Model | RO | SFE-PCA | RF | 10 | 99.94 |

These results indicate the effectiveness of the proposed feature engineering pipeline and the superior performance of ensemble-based models (RF and ET), particularly in handling complex, multilabel intrusion detection tasks.

Our proposed model demonstrates consistently strong performance across all datasets and classification tasks, even after reducing the feature space to just 10 key dimensions. This dimensionality reduction (RR = 10: N, where *N* is the number of original features) proves highly effective in improving detection accuracy while significantly minimizing both false positives and false negatives. The results strongly suggest that a compact, well-engineered feature set can yield optimal outcomes in intrusion detection.

On the CIC-IDS2018 dataset, the model maintains high accuracy levels, achieving 99.94% with SG and ET, 99.93% with RF, and 99.87% with XGB. In multilabel classification, accuracy is equally impressive, 99.99% for SG, RF, and ET, and 99.94% for XGB. For binary classification, we observe 99.91% (SG), 99.94% (RF), 99.95% (ET), and 99.65% (XGB).

Comparatively, on the NSL-KDD dataset, the highest accuracy scores were 99.59% in binary classification and 99.95% in multilabel classification, both achieved by RF and ET. Meanwhile, on the LYCOS-IDS2017 dataset, ET recorded 99.99% in binary classification, and SG, RF, and ET all achieved 99.99% in the multilabel scenario.

Overall, these results affirm that our proposed approach outperforms traditional methods, not only in terms of accuracy but also in its ability to generalize across different datasets. This underscores the importance of feature refinement and dimensionality reduction, where even with only 10 features, the model maintains exceptional performance in both binary and multilabel intrusion detection tasks.

### *Comparison With Deep Models*

CNN and LSTM baselines achieved accuracies of 97.2% and 97.8% on NSL-KDD respectively, with AUCs of 0.976–0.98. In contrast, tree-based ensembles with SFE-PCA achieved >0.99 AUC and required substantially less training time (Table 14). We therefore conclude that for structured flow features, our SFE-PCA + ensemble pipeline offers a favorable accuracy–efficiency tradeoff compared to compact deep models.

### *Economic Impact Analysis*

Assuming an average claim value of USD 2 500 and a 1 % fraud rate, achieving 99.9 % detection prevents about USD 25 000 in losses per 100 000 claims. Because false positives remain < 0.25%, investigation overhead is minimal. Thus, the proposed system delivers measurable cost savings while maintaining efficiency.

### *Novelty and Practical Impact*

While elements of our pipeline individually mirror prior work, the primary novelty is in the combined SFE-PCA + ensemble approach specifically tailored for intrusion-driven insurance fraud, together with sensitivity analyses and deployment-oriented experiments (computational profiling and ethical assessment). This makes our contribution particularly relevant for operational settings in the insurance industry.

## Conclusion and Directions for Future Work

In summary, this work presents a novel approach to proactively detecting and preventing network intrusions that may lead to insurance fraud. The proposed solution integrates multiple strategies to address key challenges in intrusion detection, including imbalanced datasets, feature representation, and high-dimensional reduction. To overcome the issue of skewed class distributions, the model employs Random Oversampling, effectively balancing the dataset. Additionally, feature embedding techniques were utilized to enrich the feature space, while dimensionality reduction, specifically PCA was applied to enhance computational efficiency and model generalization. Together, these components contributed to a robust and intelligent intrusion detection system capable of mitigating insurance fraud-related threats in network environments.

The experimental results clearly demonstrate the exceptional accuracy of our proposed model across multiple datasets. On the NSL-KDD dataset, RF (RF) and ET (ET) achieved impressive accuracy rates of 99.59% and 99.95%, respectively. Similarly, on the LYCOS-IDS2017 dataset, the model achieved outstanding performance, with SG, RF, and ET each attaining an accuracy of 99.99%. For the CIC-IDS2018 dataset, both SG and RF models reached a high accuracy of 99.94%.

These results surpass the performance of existing methods, highlighting the effectiveness of our approach in improving network intrusion detection. The consistently high accuracy across diverse datasets and classification tasks reinforces the robustness and adaptability of the proposed model.

The proposed model has shown significant performance after comparing with the three benchmarked datasets in the field of detecting insurance fraud emanating from network environments. To be specific, a key observation that was made is that, accuracy gains ranging from 1.52% to 22.19% on the NSL-KDD dataset, 0.12% to 2.99% on LYCOS-IDS2017, and 1.99% to 8.44% on CIC-IDS2018, when compared to results reported in prior studies. These improvements underscore the significant advancements introduced by our approach and affirm its contribution to the field of intrusion detection, particularly in terms of accuracy, reliability, and adaptability across diverse datasets.

The contributions of this work to the field of network intrusion detection are both meaningful and impactful. One of the key challenges we addressed is the issue of imbalanced data, a common characteristic of real-world network traffic where intrusion instances are significantly rarer than benign activity. By effectively mitigating this imbalance, our model becomes more robust and applicable to practical deployment scenarios.

Furthermore, the integration of feature embedding techniques has enabled the model to capture subtle patterns and anomalies that may be overlooked by conventional methods. This deeper representation of the data has played a critical role in enhancing detection accuracy, allowing for more precise identification of both known and emerging threats.

Not only does the application of PCA reduce computational complexity, but it also improves the model's interpretability, making it easier to analyze and understand feature impact. While accuracy gains are a major strength of the proposed model, its advantages extend further. The model proves to be more adaptable and resilient, handling different data distributions and network conditions with ease.

Rather than relying on a single algorithm, our approach combines multiple machine learning techniques, allowing the model to draw on the unique strengths of each. This fusion creates a flexible and efficient intrusion detection system, well-suited for dynamic and evolving security landscapes. For practitioners in network security, this means access to a versatile, high-performing solution that not only detects threats effectively but adapts to new attack patterns as they emerge.

By significantly improving accuracy and reducing false positive rates, the model alleviates one of the major challenges in intrusion detection systems false alarms. This reduction in noise enables security teams to concentrate on truly critical threats, thereby improving response efficiency and resource allocation.

From a practical standpoint, the model proves to be an invaluable asset for securing network infrastructure. Its combination of high detection accuracy and adaptive capabilities allows it to reliably identify both well-known attacks and emerging threats. As a result, it strengthens the overall security posture of organizations, making it a highly effective solution for modern intrusion detection systems.

By reducing false positive rates, the model enables a more efficient allocation of security resources, allowing teams to focus their attention on genuine threats rather than being overwhelmed by false alarms. This improvement not only streamlines operational workflows but also enhances the effectiveness of security monitoring efforts.

Ultimately, the proposed model marks a significant step forward in network intrusion detection, offering a more reliable, accurate, and efficient solution for safeguarding critical network assets. Its ability to balance precision with adaptability makes it a valuable tool in the ongoing effort to strengthen cybersecurity defenses.

One limitation of our research is the absence of optimization techniques which are offered by deep learning algorithms. Although the results achieved using traditional machine learning methods are remarkable, this leaves room for further enhancement. The incorporation of deep learning architectures, possibly in combination with hyperparameter tuning or optimization algorithms, could unlock additional performance gains and advance the capabilities of intrusion detection systems even further.

This study introduced an ensemble learning framework integrating RO, SFE, and PCA for proactive insurance-fraud detection. Results show that RF and ET deliver near-perfect accuracy, outperform deep baselines while using less computation. The approach enhances detection speed and interpretability, offering insurers a practical, scalable defense mechanism. Future work will test the framework on proprietary claim datasets and explore cost-sensitive optimization.

## Acknowledgment

Thank you to the publisher for their support in the publication of this research article. We are grateful for the resources and platform provided by the publisher, which have enabled us to share our findings with a wider audience. We appreciate the efforts of the editorial team in reviewing and editing our work, and we are thankful for the opportunity to contribute to the field of research through this publication.

## Funding Information

The authors have not received any financial support or funding to report.

## Authors Contributions

All authors contributed equally to this study.

## Ethics

All datasets are publicly available and anonymized; licensing terms permit research use. Experiments ran on 8 vCPU / 64 GB RAM machines using Python 3.10, scikit-learn and TensorFlow 2.12. The authors share responsibility for conceptualization, analysis, and manuscript preparation. Privacy compliance, bias audits, and human oversight are recommended for deployment.

## References

Abro, A. A., Siddique, W. A., Talpur, M. S. H., Jumani, A. K., & Yaşar, E. (2023). A combined approach of base and meta learners for hybrid system. *Turkish Journal of Engineering*, *7*(1), 25–32. https://doi.org/10.31127/tuje.1007508

Ahsun, A., Joy, B., Noan, B., & Okunola, A. (2025). The Future of Real-Time Fraud Detection: Trends and Innovations. *ResearchGate*. https://www.researchgate.net/publication/388457969

Al-Dajeh, B. M., El-Manaseer, S., Alquah, M., & Aleshoush, M. (2025). Strict Civil Liability in Artificial Intelligence Applications: A Perspective into Legal Framework, Algorithmic Biases, and Ethical Considerations. *Intelligence-Driven Circular Economy*, *1174*, 37–47. https://doi.org/10.1007/978-3-031-74220-0_3

Alkadi, O., Moustafa, N., Turnbull, B., & Choo, K.-K. R. (2021). A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks. *IEEE Internet of Things Journal*, *8*(12), 9463–9472. https://doi.org/10.1109/jiot.2020.2996590

Alonge, E. O., Eyo-Udo, N. L., Ubanadu, B. C., Daraojimba, A. I., Balogun, E. D., & Ogunsola, K. O. (2021). Enhancing Data Security with Machine Learning: A Study on Fraud Detection Algorithms. *Journal of Frontiers in Multidisciplinary Research*, *2*(1), 19–31. https://doi.org/10.54660/.ijfmr.2021.2.1.19-31

Alsaffar, A. M., Nouri-Baygi, M., & Zolbanin, H. M. (2024). Shielding networks: enhancing intrusion detection with hybrid feature selection and stack ensemble learning. *Journal of Big Data*, *11*(1), 133. https://doi.org/10.1186/s40537-024-00994-7

Alzubi, O. A., Alzubi, J. A., Alweshah, M., Qiqieh, I., Al-Shami, S., & Ramachandran, M. (2020). An optimal pruning algorithm of classifier ensembles: dynamic programming approach. *Neural Computing and Applications*, *32*(20), 16091–16107. https://doi.org/10.1007/s00521-020-04761-6

Arifin, S., Wijaya, A. K., Nariswari, R., Yudistira, I. G. A. A., Suwarno, & Faisal. (2023). Long Short-Term Memory (LSTM): Trends and Future Research Potential. *International Journal of Emerging Technology and Advanced Engineering*, *13*(5), 24–35. https://doi.org/10.46338/ijetae0523_04

Arman, M., Sujon, H., & Mustafa, H. (2023). Comparative Study of Machine Learning Models on Multiple Breast Cancer Datasets. *International Journal of Advanced Science Computing and Engineering*, *5*(1), 15–24. https://doi.org/10.62527/ijasce.5.1.105

Arpit, D., Wang, H., Xiong, C., & Zhou, Y. (2022). Ensemble of Averages: Improving Model Selection and Boosting Performance in Domain Generalization. *Advances in Neural Information Processing Systems 35*, 8265–8277. https://doi.org/10.52202/068431-0601

Bartsiotas, G. A., & Achamkulangare, G. (2021). *Fraud Prevention, Detection and Response in United Nations System Organizations*.

Batani, J. (2021). An Adaptive and Real-Time Fraud Detection Algorithm in Online Transactions. *International Journal of Computer Science and Business Informatics IJCSBI.ORG*, *17*(2), 1–12.

Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2019). A Comparative Analysis of XGBoost. *Artificial Intelligence Review*, *53*(8), 5367–5396. https://doi.org/10.48550/arXiv.1911.01914

Boateng, E. Y., Otoo, J., & Abaye, D. A. (2020). Basic Tenets of Classification Algorithms K-Nearest-Neighbor, Support Vector Machine, Random Forest and Neural Network: A Review. *Journal of Data Analysis and Information Processing*, *8*(4), 341–357. https://doi.org/10.4236/jdaip.2020.84020

Borketey, B. (2024). Real-Time Fraud Detection Using Machine Learning. *Journal of Data Analysis and Information Processing*, *12*(02), 189–209. https://doi.org/10.4236/jdaip.2024.122011

Chatzimparmpas, A., Martins, R. M., Kucher, K., & Kerren, A. (2021). StackGenVis: Alignment of Data, Algorithms, and Models for Stacking Ensemble Learning Using Performance Metrics. *IEEE Transactions on Visualization and Computer Graphics*, *27*(2), 1547–1557. https://doi.org/10.1109/tvcg.2020.3030352

Chen, R., Wang, Q., & Javanmardi, A. (2025). A Review of the Application of Machine Learning for Pipeline Integrity Predictive Analysis in Water Distribution Networks. *Archives of Computational Methods in Engineering*, *32*(6), 3821–3849. https://doi.org/10.1007/s11831-025-10251-6

Chen, Z., Ke, G., Liu, T.-Y., Shi, Y., & Zheng, S. (2022). Quantized Training of Gradient Boosting Decision Trees. *Advances in Neural Information Processing Systems 35*, 18822–18833. https://doi.org/10.52202/068431-1367

Coppens, Y., Efthymiadis, K., Lenaerts, T., & Nowé, A. (2020). Distilling Deep Reinforcement Learning Policies in Soft Decision Trees. *Proceedings of the IJCAI 2019 Workshop on Explainable Artificial Intelligence*, 1–6.

Dai, S., Li, K., Luo, Z., Zhao, P., Hong, B., Zhu, A., & Liu, J. (2024). AI-based NLP section discusses the application and effect of bag-of-words models and TF-IDF in NLP tasks. *Journal of Artificial Intelligence General Science*, *5*(1), 13–21. https://doi.org/10.60087/jaigs.v5i1.149

Damanik, N., & Liu, C.-M. (2025). Advanced Fraud Detection: Leveraging K-SMOTEENN and Stacking Ensemble to Tackle Data Imbalance and Extract Insights. *IEEE Access*, *13*, 10356–10370. https://doi.org/10.1109/access.2025.3528079

Dube, L., & Verster, T. (2023). Enhancing classification performance in imbalanced datasets: A comparative analysis of machine learning models. *Data Science in Finance and Economics*, *3*(4), 354–379. https://doi.org/10.3934/dsfe.2023021

Elmore, J. G., Nakano, C. Y., Linden, H. M., Reisch, L. M., Ayanian, J. Z., & Larson, E. B. (2005). Racial Inequities in the Timing of Breast Cancer Detection, Diagnosis, and Initiation of Treatment. *Medical Care*, *43*(2), 141–148. https://doi.org/10.1097/00005650-200502000-00007

Gajda, W. (2025). Legal foundations for developing anti-fraud policies in enterprises: Challenges and perspectives. *Public Administration and Law Review*, *2*((22)), 90–98. https://doi.org/10.36690/2674-5216-2025-2-90-98

Gharib, A., Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2016). An Evaluation Framework for Intrusion Detection Dataset. 2016 International Conference on Information Science and Security (ICISS), 1–6. https://doi.org/10.1109/ICISSEC.2016.7885840

Gur, A. S., Unal, B., Johnson, R., Ahrendt, G., Bonaventura, M., Gordon, P., & Soran, A. (2009). Predictive Probability of Four Different Breast Cancer Nomograms for Nonsentinel Axillary Lymph Node Metastasis in Positive Sentinel Node Biopsy. *Journal of the American College of Surgeons*, *208*(2), 229–235. https://doi.org/10.1016/j.jamcollsurg.2008.10.029

Hamid, Z., Khalique, F., Mahmood, S., Daud, A., Bukhari, A., & Alshemaimri, B. (2024). Healthcare insurance fraud detection using data mining. *BMC Medical Informatics and Decision Making*, *24*(1), 112. https://doi.org/10.1186/s12911-024-02512-4

Han, H., Choi, C., Jung, J., & Kim, H. S. (2021). Deep Learning with Long Short Term Memory Based Sequence-to-Sequence Model for Rainfall-Runoff Simulation. Water, 13(4), 437. https://doi.org/10.3390/w13040437

Honigsberg, C., Hu, E., Jackson, R. J., & Olin, J. M. (2020). Regulatory Arbitrage and the Persistence of Financial Misconduct. *Stanford Law Review*, *74*(4), 737–805. http://dx.doi.org/10.2139/ssrn.3769653

Javaid, H. A. (2018). Improving Fraud Detection and Risk Assessment in Financial Service using Predictive Analytics and Data Mining. *Integrated Journal of Science and Technology*, *1*(3).

Joloudari, J. H., Hussain, S., Nematollahi, M. A., Bagheri, R., Fazl, F., Alizadehsani, R., Lashgari, R., & Talukder, A. (2023). BERT-deep CNN: state of the art for sentiment analysis of COVID-19 tweets. *Social Network Analysis and Mining*, *13*(1), 99. https://doi.org/10.1007/s13278-023-01102-y

Jung, J., & Kim, B.-J. (2021). Insurance Fraud in Korea, Its Seriousness, and Policy Implications. *Frontiers in Public Health*, *9*, 1–6. https://doi.org/10.3389/fpubh.2021.791820

Kavzoglu, T., & Teke, A. (2022). Advanced hyperparameter optimization for improved spatial prediction of shallow landslides using extreme gradient boosting (XGBoost). *Bulletin of Engineering Geology and the Environment*, *81*(5), 201. https://doi.org/10.1007/s10064-022-02708-w

Kerwin, K. R., & Bastian, N. D. (2021). Stacked generalizations in imbalanced fraud data sets using resampling methods. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, *18*(3), 175–192. https://doi.org/10.1177/1548512920962219

Kose, I., Gokturk, M., & Kilic, K. (2015). An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance. *Applied Soft Computing*, *36*, 283–299. https://doi.org/10.1016/j.asoc.2015.07.018

Lones, M. A. (2024). How to avoid machine learning pitfalls: a guide for academic researchers. *Patterns*, *5*(10), 101046.

Lu, W.-L., Jansen, L., Post, W. J., Bonnema, J., Van de Velde, J. C., & De Bock, G. H. (2009). Impact on survival of early detection of isolated breast recurrences after the primary treatment for breast cancer: a meta-analysis. *Breast Cancer Research and Treatment*, *114*(3), 403–412. https://doi.org/10.1007/s10549-008-0023-4

Luo, H., Cheng, F., Yu, H., & Yi, Y. (2021). SDTR: Soft Decision Tree Regressor for Tabular Data. *IEEE Access*, *9*, 55999–56011. https://doi.org/10.1109/access.2021.3070575

Mancilla, J., Sequeira, A., Tagliani, T., Llaneza, F., & Beiza, C. (2024). Empowering Credit Scoring Systems with Quantum-Enhanced Machine Learning. *ArXiv Repository*. https://doi.org/10.48550/arXiv.2404.00015

Moishin, M., Deo, R. C., Prasad, R., Raj, N., & Abdulla, S. (2021). Designing Deep-Based Learning Flood Forecast Model With ConvLSTM Hybrid Algorithm. *IEEE Access*, *9*, 50982–50993. https://doi.org/10.1109/access.2021.3065939

Morley, N. J., Ball, L. J., & Ormerod, T. C. (2006). How the detection of insurance fraud succeeds and fails. *Psychology, Crime & Law*, *12*(2), 163–180. https://doi.org/10.1080/10683160512331316325

Mullooly, M., Ehteshami Bejnordi, B., Pfeiffer, R. M., Fan, S., Palakal, M., Hada, M., Vacek, P. M., Weaver, D. L., Shepherd, J. A., Fan, B., Mahmoudzadeh, A. P., Wang, J., Malkov, S., Johnson, J. M., Herschorn, S. D., Sprague, B. L., Hewitt, S., Brinton, L. A., Karssemeijer, N., … Gierach, G. L. (2019). Application of convolutional neural networks to breast biopsies to delineate tissue correlates of mammographic breast density. *Npj Breast Cancer*, *5*(1), 43. https://doi.org/10.1038/s41523-019-0134-6

Ngo, G., Beard, R., & Chandra, R. (2022). Evolutionary bagging for ensemble learning. *Neurocomputing*, *510*, 1–14. https://doi.org/10.1016/j.neucom.2022.08.055

Orelaja, A., & Adeyemi, A. F. (2024). Developing Real-Time Fraud Detection and Response Mechanisms for Financial Transactions. *IRE Journals*, *8*(1), 573–582.

Oriola, O. (2022). A Stacked Generalization Ensemble Approach for Improved Intrusion Detection. *IJCSIS*, *18*(5), 62–67.

Perez, I., Wong, J., Skalski, P., Burrell, S., Mortier, R., McAuley, D., & Sutton, D. (2023). Locally Differentially Private Embedding Models in Distributed Fraud Prevention Systems. *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*, 475–484. https://doi.org/10.1109/icdmw60847.2023.00068

Rane, N., Choudhary, S. P., & Rane, J. (2024a). Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions. *Studies in Medical and Health Sciences*, *1*(2), 18–41. https://doi.org/10.48185/smhs.v1i2.1225

Rane, N., Choudhary, S. P., & Rane, J. (2024b). Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions. *Studies in Medical and Health Sciences*, *1*(2), 18–41. https://doi.org/10.48185/smhs.v1i2.1225

Rane, N., Choudhary, S., & Rane, J. (2023). Explainable Artificial Intelligence (XAI) in healthcare: Interpretable Models for Clinical Decision Support. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4637897

Razaq, L., Ahmad, T., Ibtasam, S., Ramzan, U., & Mare, S. (2021). "We Even Borrowed Money From Our Neighbor." *Proceedings of the ACM on Human-Computer Interaction*, *5*(CSCW1), 1–30. https://doi.org/10.1145/3449115

Ren, J. (2019). *Investigation of Convolutional Neural Network Architectures for Image-based Feature Learning and Classification*.

Ribeiro, H., & Costa, M. (2019). Economic and Social Development 43 rd. *International Scientific Conference on Economic and Social Development-"Rethinking Management in the Digital Era: Challenges from Industry 4.0 to Retail Management" Book of Proceedings*. 43rd International Scientific Conference on Economic and Social Development, Aveiro, Portugal.

Saylor, A. T. (2023). *Recommendations to Enhance Deterrence and Detection of Opportunistic Auto Insurance Fraud for Insurance Professionals and Industry Partners*.

Shaikh, T. A., Rasool, T., Verma, P., & Mir, W. A. (2024). A fundamental overview of ensemble deep learning models and applications: systematic literature and state of the art. *Annals of Operations Research*. https://doi.org/10.1007/s10479-024-06444-0

Shao, Y., Cheng, Y., Shah, R. U., Weir, C. R., Bray, B. E., & Zeng-Treitler, Q. (2021). Shedding Light on the Black Box: Explaining Deep Neural Network Prediction of Clinical Outcomes. *Journal of Medical Systems*, *45*(1), 1–5. https://doi.org/10.1007/s10916-020-01701-8

Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, 108–116. https://doi.org/10.5220/0006639801080116

Song, Y., & Deng, Y. (2019). Divergence Measure of Belief Function and Its Application in Data Fusion. *IEEE Access*, *7*, 107465–107472. https://doi.org/10.1109/access.2019.2932390

Srinivasagopalan, L. N. (2020). AI-Enhanced Fraud Detection in Healthcare Insurance: A Novel Approach to Combatting Financial Losses through Advanced Machine Learning Models. *European Journal of Advances in Engineering and Technology*, *9*(8), 9–82.

Talukder, Md. A., Islam, Md. M., Uddin, M. A., Hasan, K. F., Sharmin, S., Alyami, S. A., & Moni, M. A. (2024). Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. *Journal of Big Data*, *11*(1), 33. https://doi.org/10.1186/s40537-024-00886-w

Tangirala, S. (2020). Evaluating the Impact of GINI Index and Information Gain on Classification using Decision Tree Classifier Algorithm*. *International Journal of Advanced Computer Science and Applications*, *11*(2). https://doi.org/10.14569/ijacsa.2020.0110277

Tarr, J.-A. (2019). Grappling with fraudulent insurance claims and "collateral lies": Comparative insurance law developments in the United Kingdom and Australia. *Insurance Law Journal*, *30*(1), 35–53.

Tax, N., de Vries, K. J., de Jong, M., Dosoula, N., van den Akker, B., Smith, J., Thuong, O., & Bernardi, L. (2021). Machine Learning for Fraud Detection in E-Commerce: A Research Agenda. *Deployable Machine Learning for Security Defense*, *1482*, 30–54. https://doi.org/10.1007/978-3-030-87839-9_2

Terefe, E. M. (2023). Tree-Based Machine Learning Methods For Vehicle Insurance Claims Size Prediction. *ArXiv Repository (Computer Science - Machine Learning)*. https://doi.org/10.48550/arXiv.2302.10612

Thockchom, N., Singh, M. M., & Nandi, U. (2023). A novel ensemble learning-based model for network intrusion detection. *Complex & Intelligent Systems*, *9*(5), 5693–5714. https://doi.org/10.1007/s40747-023-01013-7

Veera, V. R. M. B. (2025). Crafting a comprehensive strategy to prevent fraud and enable recovery in mobile insurance in the USA. *International Journal of Science and Research Archive*, *14*(1), 1794–1807. https://doi.org/10.30574/ijsra.2025.14.1.2360

Vrigazova, B. (2021). The Proportion for Splitting Data into Training and Test Set for the Bootstrap in Classification Problems. *Business Systems Research Journal*, *12*(1), 228–242. https://doi.org/10.2478/bsrj-2021-0015

Wu, H., & Levinson, D. (2021). The ensemble approach to forecasting: A review and synthesis. *Transportation Research Part C: Emerging Technologies*, *132*, 103357. https://doi.org/10.1016/j.trc.2021.103357

Xiao, Y., Wu, J., Lin, Z., & Zhao, X. (2018). A deep learning-based multi-model ensemble method for cancer prediction. *Computer Methods and Programs in Biomedicine*, *153*, 1–9. https://doi.org/10.1016/j.cmpb.2017.09.005

Zhang, G., Li, Z., Huang, J., Wu, J., Zhou, C., Yang, J., & Gao, J. (2022). eFraudCom: An E-commerce Fraud Detection System via Competitive Graph Neural Networks. *ACM Transactions on Information Systems*, *40*(3), 1–29. https://doi.org/10.1145/3474379