Expert Systems With Applications 213 (2023) 119030

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Dual-IDS: A bagging-based gradient boosting decision tree model for network anomaly intrusion detection system

Maya Hilda Lestari Louk [a], Bayu Adhi Tama [b,*]

[a] Department of Informatics Engineering, Universitas Surabaya, Jl. Ngagel Jaya Selatan No. 169, Surabaya 60284, Indonesia
[b] Department of Information Systems, University of Maryland Baltimore County (UMBC), 1000 Hilltop Circle, Baltimore, MD 21250, USA

ARTICLE INFO

Keywords:
Gradient boosting tree
Intrusion detection
Anomaly detection
Bagging
Dual ensemble

ABSTRACT

The mission of an intrusion detection system (IDS) is to monitor network activities and assess whether or not they are malevolent. Specifically, anomaly-based IDS can discover irregular activities by discriminating between normal and anomalous deviations. Nonetheless, existing strategies for detecting anomalies generally rely on single classification models that are still incapable of reducing the false alarm rate and increasing the detection rate. This study introduces a dual ensemble model by combining two existing ensemble techniques, such as bagging and gradient boosting decision tree (GBDT). Multiple dual ensemble schemes involving various fine-tuned GBDT algorithms such as gradient boosting machine (GBM), LightGBM, CatBoost, and XGBoost, are extensively appraised using multiple publicly available data sets, such as NSL-KDD, UNSW-NB15, and HIKARI-2021. The results indicate that the proposed technique is a reasonable solution for the anomaly-based IDS task. Furthermore, we demonstrate that the combination of Bagging and GBM is superior to all alternative combination schemes. In addition, the proposed dual ensemble (e.g., Bagging-GBM) is considerably more competitive than similar techniques reported in the current literature.

1. Introduction

As computer networks, e-commerce, and web services have grown at a rapid pace; there has been a great demand for cyber security attack prevention and detection measures. An intrusion detection system (IDS) is one solution that enables monitoring network traffic activities and identifying whether or not such activities or contents are malicious (Denning, 1987; Sommer & Paxson, 2010). An IDS is also meant to give alarms and generate information upon detecting malicious activities (Wang, Fok, & Thing, 2022; Zhang & Liu, 2022). While the vast majority of intrusion detection systems are meant to identify and report suspicious network activity, it is notable that advanced solutions can block potentially malicious network traffic (Scarfone, Mell, et al., 2007).

An IDS can be broadly broken down into two distinct categories according to its objectives, i.e., anomaly and signature-based detectors (Alazzam, Sharieh, & Sabri, 2020). An anomaly-based detector identifies irregular activities by locating discrepancies in normal network traffic. Despite the fact that anomaly-based detectors are more effective in recognizing new sorts of attacks, they are commonly disadvantaged by a high level of false alarm rate. Conversely, a signature-based detector can efficiently differentiate between legitimate and fraudulent activity based on predefined traffic patterns. While signature-based detectors are convincing at recognizing known attacks, they are incapable of discovering unexpected attacks or variants of existing ones (Yang, Zheng, Wu, Yang, & Wang, 2020; Zhang, Huang, Wu, & Li, 2020; Zhong et al., 2020).

A lot of machine learning (ML) algorithms have been developed with the aim of improving the performance of IDS. They have become the backbone of IDS due to their learning capability (Mishra, Varadharajan, Tupakula, & Pilli, 2018). However, several prior researches have been focus on the individual classifiers which are still ineffective in terms of detection rate and false alarm rate. Several learning algorithms have been taken into account for building IDS, such as DT (Nancy et al., 2020), SVM (Zhang, Li, Lv, Sangaiah, & Huang, 2020), genetic algorithm (Sai Satyanarayana Reddy, Chatterjee, & Mamatha, 2019), particle swarm optimization (Kan et al., 2021), statistical approach (Amma, Selvakumar, & Velusamy, 2020), and other swarm intelligence techniques (Nasir, Khan, Khan, & Fatima, 2022). There was one issue with the original concept of employing ML in IDS, and that issue was the possibility that a single classifier would not be powerful enough to construct an effective IDS (Aljanabi, Ismail, & Mezhuyev, 2020; Chkirbene et al., 2020). As a results, it is unfeasible for a single classifier to effectively function in all settings and for all.

* Corresponding author.
E-mail addresses: mayalouk@staff.ubaya.ac.id (M.H.L. Louk), bayu@umbc.edu (B.A. Tama).

https://doi.org/10.1016/j.eswa.2022.119030
Received 28 June 2022; Received in revised form 7 October 2022; Accepted 10 October 2022
Available online 17 October 2022

List of Acronyms

ACO	Ant Colony Optimization
BA	Bat Algorithm
CBF	Correlation-based Filter
CFS	Correlation-based Feature Selection
CM	Consistency Method
CNN	Convolutional Neural Network
CR	Conjunctive Rule
CV	Cross Validation
DL	Deep Learning
DNN	Deep Neural Network
DSSTE	Difficult Set Sampling Technique
DT	Decision Tree
FCBF	Fast Correlation-based Filter
FOS	Feature-oriented Stacking
GA	Genetic Algorithm
GBDT	Gradient Boosting Decision Tree
GBM	Gradient Boosting Machine
GRU	Gated Recurrent Unit
HGB	Histogram-based Gradient Boosting
IBL	Instance-based Learning
IFA	Improved Firefly Algorithm
k-NN	k-Nearest Neighbor
LR	Logistic Regression
LSTM	Long Short Term Memory
ML	Machine Learning
MLP	Multilayer Perceptron
MOE	Mixture-of-Experts
MOIA	Multi-Objective Immune Algorithm
MPV	Maximum Probability Voting
MRN	Multimodal Residual Network
NB	Naive Bayes
NN	Neural Network
PA	Penalizing Attribute
PCA	Principle Component Analysis
PSO	Particle Swarm Optimization
RF	Random Forest
SMOTE	Synthetic Minority Oversampling Technique
SVM	Support Vector Machine

Ensemble learning techniques (Sagi & Rokach, 2018; Zhou, 2021) have been introduced to tackle a plethora of ML issues and challenges, and IDSs are no exception (Tama & Lim, 2021). Ensemble learning refers to the idea of incorporating multiple classification algorithms to make a decision, particularly in supervised ML tasks. The fundamental concept behind ensemble learning is that if multiple classifiers are combined, the prediction errors of a single classifier are deemed to be compensated for by other classifiers. As a consequence, the final prediction of an ensemble will be superior to that of a single classifier. This article copes with a new anomaly-based IDS technique based on a dual ensemble learner, where a classic bagging (Breiman, 1996) ensemble technique is utilized to train other gradient boosting ensembles, i.e., gradient boosting machine (GBM) (Friedman, 2001), LightGBM (Ke et al., 2017), CatBoost (Prokhorenkova, Gusev, Vorobev, Dorogush, & Gulin, 2018), and extreme gradient boosting machine (XGBoost) (Chen & Guestrin, 2016). In addition, we tested the IDS models on three different data sets, i.e., HIKARI-2021 (Ferriyan, Thamrin, Takeda, & Murai, 2021), NSL-KDD (Tavallaee, Bagheri, Lu, & Ghorbani, 2009), and UNSW-NB15 (Moustafa & Slay, 2015) in order to verify the generalizability of the proposed model. We finally appraised and benchmarked the proposed model using accuracy, precision, recall, and Matthew correlation coefficient (MCC) metrics.

In essence, our paper deals with the following goals:

(a) To develop a dual ensemble technique involving a fusion of two different strategies, such as bagging and GBDT ensembles, for detecting network anomalies. A hyperparameter tuning on each GBDT ensemble model has been applied to prevent overfitting.
(b) To compare various ensemble schemes, whether their performances are statistically significant in multiple intrusion data sets.
(c) Lastly, to benchmark our proposed model with the previous studies available in the current literature.

We structure the rest of this paper as follows. Section 2 presents an overview of state-of-the-art techniques for anomaly-based intrusion detection systems, followed by Section 3, which analyzes the research gap in the existing works and breaks down the paper's contribution. Section 4 introduces the proposed dual ensemble learning. Next, the data sets and the performance metrics used for the benchmark are provided in Section 5. Finally, we compare and discuss the experimental results in Section 6, and conclude the paper in Section 7.

2. Related work

Numerous studies have applied ensemble learning techniques to develop anomaly-based intrusion detection systems (IDS). Ensemble learners have shown a significant improvement over individual learning models. Recently, Tama and Lim (2021) provided an overview of how various ensemble learning models are utilized in the IDS domain. On the other hand, Resende and Drummond (2018) presented a more specific method (e.g., random forest (RF))-based network intrusion detection. They surveyed 35 prior publications that utilized RF, either as a classifier or for other non-conventional tasks (e.g., feature selection). More recently, Chou and Jiang (2021) focused on the high-level overview of data-driven network intrusion detection, common data sets, and a taxonomy of important research areas. In the following section, we summarize existing state-of-the-art ensemble techniques for network intrusion detection that were published between 2019 and 2022 (see Table 1). More importantly, we classify each publication with respect to the architecture of the ensemble, base classifier(s), feature selection, validation technique, and data sets.

The ensemble method can be grouped into two main folks: homogeneous (e.g., the same types of classifiers) and different types of classifiers that lead to heterogeneous ensembles (Zhou, 2012). Existing anomaly-based IDS techniques have extensively considered different types of classifiers as base learners, while voting is used as a fusion method. Specifically, majority voting (e.g., dictatorship) (Khonde & Ulagamuthalvi, 2019; Krishnaveni & Prabakaran, 2019; Krishnaveni, Sivamohan, Sridhar, & Prabakaran, 2021; Salo, Nassif, & Essex, 2019; Thaseen et al., 2020), weighted majority voting (Feng & Dou, 2021; Krishnaveni, Sivamohan, Sridhar, & Prabhakaran, 2022; Tian, Han, Hsieh, Li, & Castiglione, 2021), average voting (Zhou, Cheng, Jiang, & Dai, 2020), and quadratic ensemble (Yang, Zheng et al., 2020) have been prevalent in the current literature. Another ensemble architecture employing a meta-combination technique (e.g., meta-learning) has also been implemented. Meta-learning is a process of learning from classifiers. A meta-classifier is trained in two or more stages, instead of a single stage for standard classifiers. Stacking (Wolpert, 1992) is likely the most prevalent meta-learning strategy. Several studies (Folino, Folino, Guarascio, Pisani, & Pontieri, 2021; Jain & Kaur, 2021; Luo et al., 2020; Rashid, Kamruzzaman, Imam, Wibowo, & Gordon, 2022) argue that stacking is able to produce the best performance accuracy, especially when 'strong classifiers' are utilized (Nkenyereye, Tama, & Lim, 2021; Tama & Lim, 2021; Tama, Nkenyereye, Islam, & Kwak,

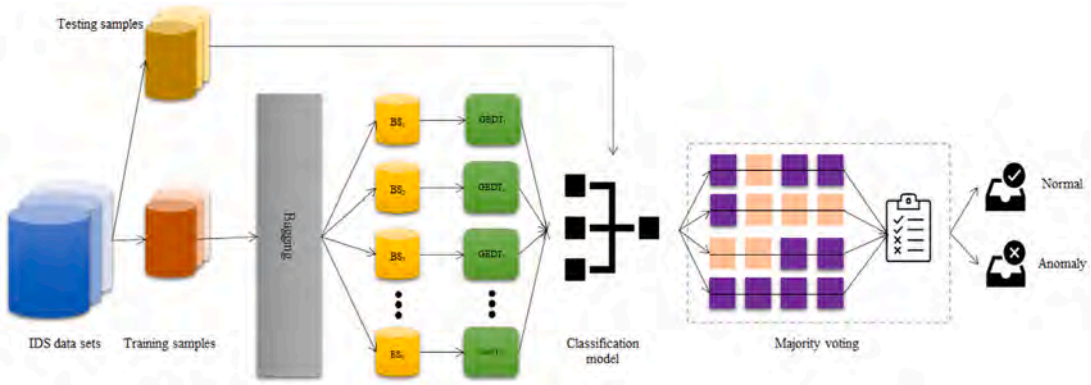


Fig. 1. Proposed dual ensemble model for anomaly-based intrusion detection system.

2020). It is to be noted that the original goal of ensemble method is to boost 'weak classifiers', which can be tree learners, Bayesian algorithms, neural networks, or other kinds of weak learning algorithms. Previous studies have used the term 'strong classifier' to emphasize the concept of an ensemble learner, the base learner of which is another ensemble (Van der Laan, Polley, & Hubbard, 2007).

Furthermore, homogeneous ensembles have been applied in this discussed domain. They include random forest (Abdulhammed, Faezipour, Abuzneid, & AbuMallouh, 2019; Al & Dener, 2021; Gupta, Jindal, & Bedi, 2022; Liu, Wang, Lin, & Liu, 2020; Nazir & Khan, 2021; Seth, Chahal, & Singh, 2021; Wei, Chen, Lin, Ji, & Chen, 2020; Wu & Li, 2021; Zhang, Li, Liu, & Dong, 2021) and boosting ensembles (Gupta et al., 2022; Halim et al., 2021; Liu, Gao, & Hu, 2021; Liu et al., 2020; Mazini, Shirazi, & Mahdavi, 2019; Moustafa, Turnbull, & Choo, 2019; Seth et al., 2021; Tama & Rhee, 2019; Wang, Liu, & Sun, 2022; Wei et al., 2020; Wu & Li, 2021). Particularly, recent development of gradient boosting ensembles such as XGBoost (Chen & Guestrin, 2016) and LightGBM (Ke et al., 2017) have been deployed to improve the detection performance of anomaly-based IDS. Besides, classical boosting algorithms such as Adaboost (Freund, Schapire, & Abe, 1999) and gradient boosting machine (GBM) (Friedman, 2001) can be found in the recent literature.

3. Research gap and contributions

As discussed in Section 2, most anomaly-based IDS techniques focus on using individual classifier ensemble approaches, either utilizing homogeneous or heterogeneous architecture models. More precisely, recent advancements in gradient boosting ensembles such as XGBoost and LightGBM have been taken into account. Such ensemble models, however, are primarily used as a standalone classifier rather than as a base classifier in another ensemble. Moreover, Catboost (Prokhorenkova et al., 2018) has not been seen in this application domain, either as an individual classifier or as a constituent of an ensemble. Inspired by such open research issues, we propose a dual ensemble model for anomaly-based IDS by incorporating two different classifier ensemble models, such as bagging (Breiman, 1996) and state-of-the-art gradient boosting decision tree (GBDT) models (e.g., GBM (Friedman, 2001), LightGBM (Ke et al., 2017), CatBoost (Prokhorenkova et al., 2018), and XGBoost (Chen & Guestrin, 2016)). As far as we know, our dual ensemble model is the first to be designed in this domain of interest; it provides an effective detection model for anomaly-based IDS.

The contributions of this paper can be viewed from the following different angles:

(a) Rather than using GBDT as a standalone ensemble scheme, we employ a bagging procedure to improve the performance of GBDT ensemble models.

(b) We demonstrate that the proposed method significantly improves the detection rate on three distinct intrusion data sets when compared to several state-of-the-art techniques through a detailed performance evaluation.

(c) Performance differences among various configuration ensemble models are verified through a two-step statistical significance test involving the Friedman and Nemenyi tests to illustrate the performance improvement evidenced by the proposed model.

4. Method

This section provides the proposed dual ensemble technique for anomaly-based intrusion detection system. We propose a dual ensemble strategy based on a bagging technique for blending strong classification algorithms for network anomaly detection (see Fig. 1). In this study, we take into consideration the following constituents: GBM (Friedman, 2001), LightGBM (Ke et al., 2017), CatBoost (Prokhorenkova et al., 2018), and XGBoost (Chen & Guestrin, 2016). GBM has been utilized in the previous work (Tama & Rhee, 2019), whereas XGBoost is recognized to be a high-performing classifier in various domains, especially, the domain that uses tabular data (Borisov et al., 2021; Shwartz-Ziv & Armon, 2022).

4.1. Training GBDT algorithms via bagging

A bagging approach (e.g., bootstrap aggregating) (Breiman, 1996) is formulated using Ψ gradient boosting ensemble models constructed from bootstrap replicates (e.g., BS) of the original training set. By sampling with a replacement, an instance of α training instances will be formed from a training data set comprising α training instances. Certain unique instances exist multiple times in the sample, while others are not. The Ψ instances of α training instances are acquired by repeatedly iterating the procedure. Next, using the base learning algorithm (e.g., gradient boosting ensembles (GBDT) in our case), individual gradient boosting model can be trained on each sample.

Assuming Ψ constituents $\{h_1, h_2, \ldots, h_\Psi\}$ are given, and the objective is to blend h_i to predict the class label from a set of τ potential class labels $\{c_1, \ldots, c_\tau\}$. Consider the final class prediction of h_i for a given instance $\mathbf{x}$ to be a τ-dimensional vector $(h_i^1(\mathbf{x}), \ldots, h_i^\tau(\mathbf{x}))^\Psi$, where $h_i^j(\mathbf{x})$ is the result of h_i for the class label c_j. Consequently, $h_i^j(\mathbf{x}) \in \{0, 1\}$ has the value 1 if h_i predicts c_j as the class label, and 0 otherwise. Majority voting allows each constituent to vote for one class label, and final class prediction $H(\mathbf{x})$ that obtains more than half of the votes is elected. Algorithm 1 summarizes the procedure of a bagging-based gradient boosting ensemble model.

Table 1
Classification of previous anomaly-based intrusion detectors using ensemble learning, chronologically ordered by year of publication.

Author(s)	Ensemble scheme	Base learner(s)	Feature selection	Validation method	Data set	Remarks
Abdulhammed et al. (2019)	RF	–	–	–	CICIDS 2017	The proposed system can detect attacks when handling the imbalanced class distribution with fewer samples
Khonde and Ulagamuthalvi (2019)	Voting	NN, k-NN, DT, RF, SVM	Gini index	Hold-out	NSL-KDD	All classifiers operate on a distributed network where every identified anomaly is transformed into a signature to prevent subsequent attacks.
Krishnaveni and Prabakaran (2019)	Voting	Naive Bayes, SVM, Logistic regression	–	Hold-out	Honeypot	The proposed method is deployed on cloud environment.
Mazini et al. (2019)	Adaboost	–	Artificial bee colony	Hold-out	NSL-KDD, ISCX 2021	The proposed technique improves detection performance in terms of accuracy and detection rate.
Moustafa et al. (2019)	Adaboost	NN, Naive Bayes, DT	Correlation coefficient	CV	UNSW-NB15, NIMS botnet	The proposed features have the potential for both characteristics of normal and malicious behavior.
Salo et al. (2019)	Voting	SVM, k-NN, MLP	Information gain, PCA	Hold-out	NSL-KDD, Kyoto 2006, ISCX 2012	The proposed technique integrate multiple feature reduction approaches.
Tama, Comuzzi, and Rhee (2019)	Rotation forest, bagging	CR	PSO, ACO, GA	Hold-out	NSL-KDD, UNSW-NB15	A two-stage ensemble approach is proposed.
Tama and Rhee (2019)	GBM	–	–	Hold-out, CV	NSL-KDD, UNSW-NB15, Wi-Fi intrusion dataset	The performance of the anomaly detector is obtained through the fine-tuning of a GBM.
Zhou et al. (2020)	Average voting	C4.5, RF, forest PA	CFS+BA	CV, Hold-out	NSL-KDD, AWID, CICIDS2017	A heuristic-based feature dimensionality reduction technique is proposed.
Liu et al. (2020)	RF, XGBoost	–	–	Hold-out	NSL-KDD, CICIDS2018	A novel DSSTE algorithm is proposed to overcome class imbalance problem.
Tama et al. (2020)	Stacking	RF, GBM, XGBoost	–	CV, Hold-out	CICIDS2017, NSL-KDD, UNSW-NB15	Stacking is built using strong learners.
Wei et al. (2020)	RF, XGBoost	–	MOIA	CV	NSL-KDD, UNSW-NB15	Ensemble learners are not considered as the main classifiers.
Thaseen et al. (2020)	Voting	LSTM	GA	CV	NSL-KDD, UNSW-NB15	Homogeneous ensemble of LSTM is proposed.
Yang, Sheng, and Wang (2020)	Quadratic ensemble	GBDT, GRU	–	Hold-out	CICIDS2017	Temporal and spatial features are combined.
Luo et al. (2020)	Stacking	MRN, LSTM, CNN	–	Hold-out	CSIC2010	Various DL models are utilized.
Feng and Dou (2021)	Weighted voting	11 classifiers	–	CV	KDD99, CICIDS2017	A dynamic selection method is used to select the base classifiers.
Nkenyereye et al. (2021)	Stacking	DNN	–	CV and Hold-out	NSL-KDD, CICIDS2017, UNSW-NB15	An ensemble of deep learning models is proposed.
Seth et al. (2021)	RF, XGBoost, HGB, LightGBM	–	RF+PCA	CV	CICIDS2018	A hybrid of SMOTE and undersampling is proposed to address the class imbalance problem.
Wu and Li (2021)	RF, Boosting	–	FCBF, CBF, CM	CV	KDD99	Ensemble learners are not considered as the main classifiers.
Halim et al. (2021)	XGBoost	–	GA	CV	CIRA-CIC-DoHBrw-2020, Bot-IoT, UNSW-NB15	GA-based feature selection with a new fitness function is proposed.
Nazir and Khan (2021)	RF	–	Gain ratio, Chi-squared, Pearson correlation	Hold-out	UNSW-NB15	A hybrid of Tabu search and RF is employed for feature selection.
Liu et al. (2021)	LightGBM	–	–	CV	NSL-KDD, UNSW-NB15, CICIDS2017	An adaptive synthetic oversampling method is used to tackle imbalanced data issue.
Al and Dener (2021)	RF	–	–	Hold-out	CIDDS-001, UNSW-NB15	SMOTE+Tomek Links are proposed to overcome the class imbalance issue.
Jain and Kaur (2021)	Stacking	RF, LR	K-means	Hold-out	NSL-KDD, CIDDS2017, Testbed	Distributed ML-based ensemble technique is proposed.
Tian et al. (2021)	Weighted voting	C4.5, MLP, IBL	IFA	CV	NSL-KDD, UNSW-NB15	An improved firefly algorithm is proposed for feature selection.
Zhang et al. (2021)	RF	–	–	CV	NSL-KDD, UNSW-NB15, CICIDS2017	A multi-dimensional feature fusion and stacking ensemble mechanism is proposed.
Folino et al. (2021)	MPV, Stacking, FOS, MOE	DNN	–	Hold-out	ISCX, CICIDS2017	Data resampling, imbalance aware loss function are proposed to address the class imbalance.
Krishnaveni et al. (2021)	Majority voting	SVM, NB, LR, DT	Filter and univariate ensemble	CV	Honeypot, NSL-KDD, Kyoto	Univariate ensemble-based feature selection is proposed.

(*continued on next page*)

Table 1 (*continued*).

Author(s)	Ensemble scheme	Base learner(s)	Feature selection	Validation method	Data set	Remarks
Krishnaveni et al. (2022)	Weighted majority voting	SVM, LR, NB, DT	Gain-ratio, chi-squared, information gain	Hold-out	Honeypot, NSL-KDD, Kyoto	An improved weighted majority voting ensemble is proposed.
Rashid et al. (2022)	Stacking	DT, RF, XGBoost	*SelectKbest*	CV	NSL-KDD, UNSW-NB15	A tree-based stacking ensemble is proposed.
Gupta et al. (2022)	XGBoost, RF	–	–	Hold-out	NSL-KDD, CIDDS-001, CICIDS2017	The proposed models is made up of cost sensitive DL and ensemble learning.
Wang, Liu et al. (2022)	LightGBM	–	DNN	Hold-out	KDD99, NSL-KDD, UNSW-NB15	A novel ensemble feature selection-based DNN is proposed.

Algorithm 1: A bagging-based gradient boosting ensemble model for anomaly-based intrusion detection system

Training:
Require: Original training set $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_\alpha, y_\alpha)\}$; Base learning algorithms, e.g., GBDT; Number of base learning algorithm Ψ; Sample size μ.
1. $\psi \leftarrow 1$
2. **repeat**
3. $D_\psi \leftarrow$ a sample of μ instances from D with replacement.
4. Build classifier h_ψ using GBDT with D_ψ as the training set.
5. $\psi \leftarrow \psi + 1$
6. **until** $\psi > \Psi$
Testing:
Require: An instance to be classified **x**.
Output: Predicted class label c
1. $Counter_1, ..., Counter_y \leftarrow 0$
2. **for** $i = 1$ to Ψ **do**
3. $vote_i \leftarrow h_i(\mathbf{x})$
4. $Counter_{vote_i} \leftarrow Counter_{vote_i} + 1$
5. **end for**
6. $c \leftarrow$ the class label with the largest number votes.
7. Return c

4.2. Base GBDT algorithms

Next, we describe the constituents of the bagging ensemble as follows. Gradient boosting ensemble, also known as gradient boosting decision tree (GBDT), is a classifier ensemble that integrates multiple individual weak classifiers into a single strong classifier. Typically, it is an additive model (e.g., linear addition of weak classifiers) that utilizes the classification and regression tree (CART) algorithm as its weak model. Let $D = \{(\mathbf{x}_i, y_i)|i \in \{1, \ldots, \alpha\}, \mathbf{x}_i \in \mathbb{R}^\eta, y_i \in \mathbb{R}\}$ be the network intrusion data set containing η features and α instances. Given a collection of τ trees, the prediction output $y(\hat{\mathbf{x}})^\tau$ for an input $\mathbf{x}$ is computed by summing the predictions from each tree $y(\hat{\mathbf{x}})^\tau$ as shown in the following formula.

$$y(\hat{\mathbf{x}})^\tau = \sum_{i=1}^{\tau} f_i(\mathbf{x}) \tag{1}$$

where f_i is the output of the ith regression tree of the $\tau-$tree ensemble. To construct the $(\tau+1)$-th tree, GBDT minimizes a regularized objective function Obj^t as follows.

$$\min\{Obj(f)^t\} = \min\{\Omega(f)^t + \Theta(f)^t\} \tag{2}$$

where $\Omega(f)^t$ is loss function and $\Theta(f)^t$ is a regularization function to control the over-fitting. The loss function $\Omega(f)^t$ measures the difference between the prediction $\hat{y}_i$ and the target y_i. On the other hand, the regularization function is defined as $\Theta(f)^t = \gamma T + \frac{1}{2}\lambda\|w\|^2$, where T and w represent the number of leaves and leaf weights in the tree, respectively.

In the following section, additional details concerning the base gradient boosting ensemble models, their hyperparameter settings to tune, and the default learning parameters are provided.

4.2.1. Gradient boosting machine (GBM)

GBM (Friedman, 2001) was the first version of gradient boosting-based ensemble that adopts a forward learning strategy. Tree are produced in a sequential fashion, where subsequent trees rely on the outcomes of the preceding trees. Formally, GBM is accomplished through the iterative construction of a set of functions $f^0, f^1, \ldots, f^t$, given a loss function $\Omega(y_i, f^t)$. Suppose that function f^t has been constructed, we can optimize our estimates of y_i by discovering another function $f^{t+1} = f^t + h^{t+1}(\mathbf{x})$ such that h^{t+1} diminishes the estimated value of the loss function. Therefore,

$$h^{t+1} = \underset{h \in H}{\operatorname{argmin}}\, \mathbb{E}\Omega\left(y, f^t\right) \tag{3}$$

Where H represents the collection of candidate decision trees being considered for inclusion in the ensemble. In addition, the definition of f_{t+1} enables us to express the expected value of the loss function Ω in terms of f_t and h_{t+1}:

$$\mathbb{E}\Omega(y, f^t) = \mathbb{E}\Omega\left(y, f^t + h^{t+1}\right) \tag{4}$$

To determine a rough approximation of the h^{t+1} that minimizes $\mathbb{E}\,\Omega(y, f^t + h^{t+1})$, we might look at the direction in which the gradient of Ω with respect to f_t is falling the fastest. Given these conditions, h^{t+1} can be estimated as:

$$h^{t+1} \approx \underset{h \in H}{\operatorname{argmin}}\, \mathbb{E}\left(\frac{\partial \Omega y}{\partial f^t} - h\right)^2 \tag{5}$$

This technique is referred known as the original idea of GBM because it utilizes the partial derivatives (e.g., gradients) of the loss function Ω with respect to the function f^t to determine h^{t+1}. The learning hyperparameters of GBM were set as follows. Bernoulli distribution was used for all data sets, while the other hyperparameters such as *number of trees*, *interaction depth*, and learning rate (e.g., *shrinkage*) were obtained through *random* search technique.

4.2.2. LightGBM

LightGBM (Ke et al., 2017) is one of the efficient gradient boosting tree implementations that uses histogram and leaf-wise algorithms to boost both the computational power and prediction accuracy. Histogram approach is used to fuse features that are mutually incompatible. The fundamental concept is to discretize the continuous features into n integers before constructing an n-width histogram. The training data is scanned based on the discretized values of the histogram to find the decision tree. The histogram approach significantly decreases the time complexity. Moreover, in LightGBM, the leaf with the largest splitting gain was identified and subsequently divided using a leaf-wise approach. The downside of leaf-wise is that it may yield overfitting and a deeper decision tree. LightGBM thus adds a maximum depth restriction to leaf-wise to guarantee high efficiency and avoid overfitting.

Typically, the information gain is leveraged to split each node in a decision tree algorithm. LightGBM, on the other hand, utilizes gradient-based one-side sampling (GOSS) to calculate variance gain in order to find the split point. Let S be the training data set on a fixed node in decision tree. The absolute values of the gradients of the training examples are first sorted descendingly, and the top $a \times 100\%$ data samples of gradient values, denoted A, are chosen. Then, from the remaining samples A^c, a subset B of size $b \times |A^c|$ is chosen randomly. At last, the instances are split via the estimated variance V on $A \cup B$. The variance gain of the node's split feature k at point c is defined as:

$$V_k(c) = \frac{1}{n}\left(\frac{\left(\sum_{\mathbf{x}_i \in A_l} \mathcal{G}_i + \frac{1-a}{b}\sum_{\mathbf{x}_i \in B_l} \mathcal{G}_i\right)^2}{n_l^j(c)} + \frac{\left(\sum_{\mathbf{x}_i \in A_r} \mathcal{G}_i + \frac{1-a}{b}\sum_{\mathbf{x}_i \in B_r} \mathcal{G}_i\right)^2}{n_r^j(c)} \right) \quad (6)$$

where $A_l = \{\mathbf{x}_i \in A : \mathbf{x}_{ij} \leq c\}$, $A_r = \{\mathbf{x}_i \in A : \mathbf{x}_{ij} > c\}$, $B_l = \{\mathbf{x}_i \in B : \mathbf{x}_{ij} \leq c\}$, $B_r = \{\mathbf{x}_i \in B : \mathbf{x}_{ij} > c\}$, $\mathcal{G}$ denotes the negative gradient of the loss function, and $\frac{1-a}{b}$ is utilized to normalize the sum of gradients. We set the hyperparamters of LightGBM that were not searched by random search as follows. Number of leaves is 2^{max_depth}, *maximum_bin* = 100, *boosting_type*: 'gbdt', and enabled *force_row_wise*.

4.2.3. CatBoost

CatBoost (Prokhorenkova et al., 2018) is developed based on symmetric decision trees. It is acknowledged as the classification algorithm that can deliver superior result and ten times prediction speed against other approaches without symmetric decision trees. Unlike previous GBDT algorithms, CatBoost can handle gradient bias and prediction shift to increase the prediction accuracy and generalization ability in a large data set. In addition, CatBoost is made up of two essential algorithms: ordered boosting, which computes leaf values during tree structure selection to prevent overfitting, and an unique technique for processing categorical features during the training process.

Instead of employing one-hot encoding to encode categorical variables, CatBoost uses the phrase "ordered target statistic", a value obtained from the ground truth output values matching specific values of an absolute attribute in a dataset. CatBoost selects the data to utilize for fitting h^{t+1} by placing the elements of D in an arbitrary order using a random permutation φ. Let $\varphi(m)$ represents the mth element of D under φ, together with $D_m = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\}$, ordered by the random permutation φ. CatBoost utilizes D_m to fit the decision tree h^{t+1}, and the data D to figure out if h^{t+1} is the decision tree that minimizes $\mathbb{E}\Omega\left(y, f^t + h^{t+1}\right)$.

Another unique technique of CatBoost is called indicator function $\mathbb{I}$, which transforms the values of a categorical feature to a numerical value. More precisely, the indicator function $\mathbb{I}_{x_l^k = x_m^k}$ returns 1 if the kth component of the input vector $\mathbf{x}_l$ and $\mathbf{x}_m$ are equal. Hence, m denotes the mth element based on the random permutation φ applied to D, while k assumes the integer values 1 via $m-1$. Table 3 lists the hyperparameters of CatBoost along with the search space and their optimal values.

4.2.4. Extreme gradient boosting (XGBoost)

XGBoost (Chen & Guestrin, 2016) is a scalable end-to-end tree-boosting technique that produces numerous weak learners that are sequentially learned. Each subsequent learner corrects the errors made by the previous one, resulting in an effective model used for classification task. XGBoost not only enhances the GBDT algorithm for its overfitting issue, but also increases the performance of real operation through sparsity-aware metrices and multithreading.

Since boosting is an iterative process, we can express the goal of the current iteration m in terms of the prediction of the previous iteration $\hat{y}_i^{(m-1)}$, with the help of the most recent tree f_k:

$$Obj^m = \sum_i \Omega(y_i, \hat{y}_i^{(m-1)} + \sum_k \Theta(f_k)) \quad (7)$$

Table 2
The characteristic of intrusion detection data sets used in this work.

Data set	Subset	α^t	α^-	α^+	η	ω_R
HIKARI-2021	–	555,278	517,582	37,696	83	0.0728
NSL-KDD	KDDTrain+	125,973	58,630	67,343	41	0.8706
	KDDTrain+_20%	25,192	13,449	11,743	41	0.8732
UNSW-NB15	UNSW-NB15_Train	82,332	37,000	45,332	44	0.8162

XGBoost uses the weight w_j for each leaf to minimize the objective function. The best leaf weight w_j in a current tree structure is given as:

$$w_j = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} = -\frac{G_j}{H_j + \lambda} \quad (8)$$

where g_i and h_i are the first and second order derivatives respectively of the loss function for instance i. Finally, using the best w_j, the objective function for finding the best tree structure is defined as:

$$Obj^m = -\frac{1}{2}\sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \quad (9)$$

There are several XGBoost hyperparameters to tune as given in Table 3 such as *nrounds*, *colsample_bytree*, *subsample*, *eta*, and *max_depth*.

5. Performance evaluation

The evaluation of the proposed detection model is provided here. Next, we present in detail regarding the experimental settings and the performance metrics considered in the benchmark evaluation.

5.1. Intrusion detection data sets

To provide more generalized results, particularly on how the proposed detection technique performs in various scenarios, we take into account 3 different data sets. The data sets are freely available and have been extensively used for benchmarking purposes in this domain. For each data set, a class label attribute is assigned as the prediction target y of the classification algorithm. Since the objective of anomaly detection task is to solve binary classification problem, the class label attribute is made up of two distinct classes, namely normal and attack. A total of 4 classification tasks were specified from 3 original data sets. Table 2 summarizes the characteristics of the data sets utilized in this study, such as total number of instances (α^t), the number of instances labeled normal (α^-), the number of instances labeled attacked (α^+), the number of input attributes (η), and the imbalance ratio (ω_R), which can be defined as the ratio between the number of minority instances and the number of majority instances. Typically, data sets with an $\omega_R < 0.5$ are considered as highly imbalanced. It is worth mentioning that the HIKARI-2021 data set is the only one data set having $\omega_R < 0.5$. We discuss the data sets in the following section.

(a) HIKARI-2021 (Ferriyan et al., 2021)
We considered this data set as it is one of the most recent benchmark IDS data set available. It is essential to use the up-to-date data set since the network traffic is evolving over time. HIKARI-2021 includes the network traffic with encrypted traces, and was generated with a mixture of ground-truth data. A more than 80 features were mimicked from the CICIDS-2017 data set (Sharafaldin, Lashkari, & Ghorbani, 2018), yet 4 more features were appended, such as a source IP address (*originh*), source port (*originp*), destination IP address, and destination port. To evaluate our proposed model on this data set, we adopted *holdout* validation technique, where the original data set was randomly split up into a training and a testing set. A common split of 80/20 was employed for model evaluation.

Therefore, a total of 444,223 instances were used for training while the remainder is used for testing.

(b) NSL-KDD (Tavallaee et al., 2009)
This data set was introduced to improve the original version of KDD 99 data set, and has several merits such as: (i) classification algorithms would be unaffected and unbiased by more frequently occurring instances since the redundant instances were excluded in the training and testing sets; and (ii) it is easier to undertake an accurate performance evaluation due to the fact that the number of instances were proportionally chosen from each difficulty level group. To make a comparable result, we trained our proposed classifier on the two different training sets, e.g. KDDTrain+ and KDDTrain+_20% and tested the model on the two different testing sets, e.g., KDDTest+ and KDDTest-21. This evaluation scenario allows us to get more reliable result since the classification model is validated on real unseen instances. The KDDTest+ and KDDTest-21 consist of 22,544 and 11,850 instances, respectively.

(c) UNSW-NB15 (Moustafa & Slay, 2015)
The data set was produced in the Cyber Range Lab of the Australian Centre for Cyber Security using a commercial penetration tool, which enables to develop a combination of artificially modern normal traces and emerging attack behaviors from network traffic logs. Furthermore, a total of 100 GB *pcap* files had been collected using a *tcpdump* tool, while class label and features were produced using the Bro-IDS tools. The data set provides predefined training and testing instances, such as UNSW-NB15_Train and UNSW-NB15_Test to evaluate the performance of machine learning algorithms. In this work, a total of 175,341 testing instances was taken into account.

5.2. Performance measures

The performance of dual ensemble model is evaluated based on four various metrics, such as accuracy (Acc), precision ($Prec$), recall (Rec), F1, and Matthews correlation coefficient (MCC). Next, we formally explain those metrics in the following section. A machine learning algorithm typically predicts the class for each data instance and assigns a label (e.g., normal or attack) to each sample based on this prediction. At the end of the classification procedure, each sample belongs to one of these four categories:

- TP: attack instances that are (truly) predicted as attack (True Positives).
- TN: normal instances that are (truly) predicted as normal (True Negatives).
- FP: normal instances that are (falsely) predicted as attack (False Positives).
- FN: attack instances that are (falsely) predicted as normal (False Negatives).

The categorization is commonly visualized in a contingency (e.g., confusion) matrix $M = \begin{pmatrix} TP & FN \\ FP & TN \end{pmatrix}$, which summarizes the outcome of network anomaly detection (see Fig. 2). Let us define $FN + TP = \alpha^+$ and $FP + TN = \alpha^-$, hence, a perfect score of a classifier's performance is achieved when $M = \begin{pmatrix} \alpha^+ & 0 \\ 0 & \alpha^- \end{pmatrix}$. From the contingency matrix M, other performance metrics can be inferred as follows.

Acc is the ratio between all truly predicted instances and the total instances (i.e., α^t) in the dataset:

$$Acc = \frac{TP + TN}{\alpha^t} \tag{10}$$

$Prec$ is calculated as the proportion of true positives (TP) and all attack identifications ($TP + FP$):

$$Prec = \frac{TP}{TP + FP} \tag{11}$$

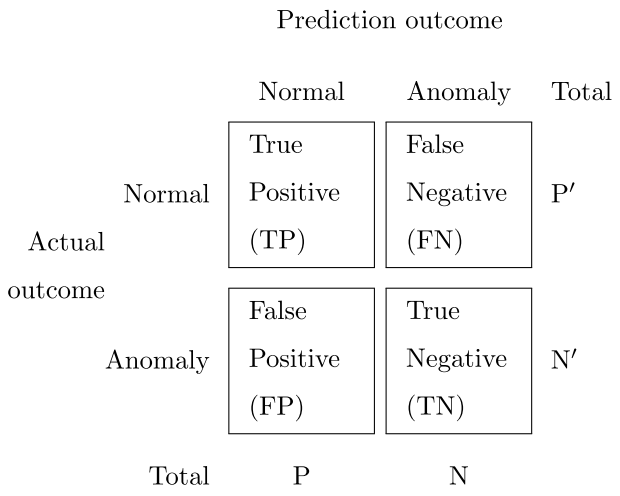


Fig. 2. A contingency table that is generally used to illustrate the outcome of anomaly-based IDS models.

Rec measures how well a classification algorithm detects actual attacks correctly. Formally, it is defined as:

$$Rec = \frac{TP}{\alpha^+} \tag{12}$$

F1 is denoted as the harmonic mean between $Prec$ and Rec. More precisely, it is determined by:

$$F1 = 2 \times \frac{Prec \times Rec}{Prec + Rec} \tag{13}$$

Lastly, Chicco and Jurman (2020) argues that MCC is more informative than $F1$ and Acc in balanced and imbalanced data sets. In addition, MCC is shown to be more appropriate than balanced accuracy, bookmaker informedness, and markedness metrics (Chicco, Tötsch, & Jurman, 2021). $F1$ and Acc, however, yield reliable estimates when applied to balanced data sets, but may give misleading results when applied to imbalanced data sets. Hence, we also utilize MCC metric in our validation scenario to provide a more realistic estimates of the proposed technique. Mathematically, MCC is calculated as follows.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times \alpha^- \times (TN + FN) \times \alpha^+}} \tag{14}$$

6. Result and discussion

In this section, the performance results of the proposed model with various schemes are provided, followed by an in-depth discussion.

6.1. Performance results

We use statistical significant tests to compare the performance of the proposed model to that of the base models (e.g., GBM, LightGBM, CatBoost, and XGBoost). We first analyze the performance obtained by the proposed models and the base models based on four performance metrics determined in the previous section. A random search algorithm (Bergstra & Bengio, 2012) is used to tweak the hyperparameters of every constituent learner. Table 3 contains a description of the search space for hyperparameters and the optimal settings for training each constituent learner on each dataset. We used default setting for the other parameters that are not mentioned in the table. For the implementation, we employ the *mlr3* machine learning framework (Lang et al., 2019), which provides a *R* interface for conducting the experiment.

Fig. 3 compares the performance of individual GBDT models and various proposed model schemes across different intrusion data sets

Table 3
The description of hyperparameters search and their optimal values on each data set.

Base models	Hyperparameters	Search space	Optimal values		
			HIKARI-2021	NSL-KDD	UNSW-NB15
GBM	*interaction.depth*	{3,4,5, …,12}	10	11	10
	n.trees	{100,200,500,1000}	500	200	1000
	shrinkage	{0.3, 0.1, 0.05, 0.01, 0.005}	0.005	0.1	0.1
LightGBM	*max_bin*	{100,255}	100	100	100
	max_depth	{3,4, …,20}	4	16	3
	data_in_leaf	{100,200, …,1000}	800	700	700
	learning_rate	{0.01,0.02, …,0.3}	0.02	0.13	0.2
	lambda_l1	{0,10,20, …,100}	0	30	10
	lambda_l2	{0,10,20, …,100}	0	70	100
	num_iterations	{100,200,500,1000}	1000	100	500
	feature_fraction	{0.5,0.9}	0.5	0.9	0.9
	bagging_fraction	{0.5,0.9}	0.5	0.5	0.9
	path_smooth	{1E−8, 1E-3}	1E−8	1E−3	1E−8
	min_gain_to_split	{0,1, …,15}	0	0	0
CatBoost	*iterations*	{100,500,1000}	1000	1000	1000
	depth	{1,2, …,10}	8	6	6
	learning_rate	{0.01,0.02, …,0.3}	0.01	0.3	0.3
	l2_leaf_reg	{1,3,5,10,100}	1	3	5
	border_count	{5,10,20,32,50,100,200}	20	32	50
	boosting_type	{Ordered,Plain}	Ordered	Plain	Plain
XGBoost	*nrounds*	{100,500,1000}	100	100	1000
	colsample_bytree	{0.5,0.9,0.1}	0.8	0.6	0.6
	subsample	{0.5,0.8,0.1}	0.6	0.7	0.8
	eta	{0,0.1,0.2, …,1.0}	0.3	0.3	0.2
	max_depth	{3,4, …,12}	9	12	11

Table 4
Average performance ranking of individual (e.g., base) classifiers based on the Friedman rank. Best values are denoted in bold.

Metrics	CatBoost	GBM	LightGBM	XGBoost
Accuracy	**2.125**	2.625	2.75	2.5
Precision	**1.625**	2.125	4	2.25
Recall	2.625	**2.375**	2.5	2.5
MCC	**2.125**	2.375	3.25	2.25
F1	**2.125**	2.625	2.75	2.5

Table 5
Average performance ranking of various ensemble schemes based on the Friedman rank. Best values are denoted in bold.

Classifier schemes	Accuracy	Precision	Recall	MCC	F1
Bag-CB-10	8.5	11.5	9.5	8.5	8.75
Bag-CB-20	8	8.5	8.75	8	7.75
Bag-CB-30	10.5	11.5	11.5	10.5	10.25
Bag-CB-50	7.75	8.75	9	7.75	8.25
Bag-CB-100	6.5	8.25	9	6.5	6.5
Bag-GBM-10	7.25	8.25	6.875	6.5	7.25
Bag-GBM-20	8.375	8.375	6.5	7.625	8.375
Bag-GBM-30	8.5	8.25	7.875	7.5	8.5
Bag-GBM-50	**6.25**	6.75	**6.375**	**5.75**	**6.25**
Bag-GBM-100	7.125	7.375	6.375	**6.375**	7.125
Bag-LGB-10	16.5	18	13	16.75	16.5
Bag-LGB-20	16	16.25	13.75	15.25	15.75
Bag-LGB-30	16.25	18.25	14.5	18.25	16.25
Bag-LGB-50	14.875	16.125	13.625	16.125	14.375
Bag-LGB-100	14.625	16.375	13.625	15.625	14.625
Bag-XGB-10	10.75	**6.25**	11.5	10.5	11
Bag-XGB-20	12.25	7.75	10	12.25	9.25
Bag-XGB-30	10.375	7.75	12.25	10.25	10.75
Bag-XGB-50	10.25	8	13.75	10.25	12
Bag-XGB-100	9.375	7.75	12.25	9.75	10.5

and performance metrics. Each proposed model scheme is denoted as Bagging-$\mathcal{X}$, where $\mathcal{X}$ represents an ensemble size (e.g., the number of constituent models used). The bagging ensemble shows its superiority over individual model when GBM is used as the constituent model, irrespective of the performance metrics. The performance improvement of bagging-GBM is more evident on NSL-KDD data set. Similarly, the performance increased can be also obtained when we train CatBoost in a bagging manner. For instance, in terms of recall, F1, and MCC metrics, bagging-CatBoost performs better than individual CatBoost when applied to the Hikari and UNSW-NB15 data sets.

To better understand the behavior of each algorithm in various data set, we adopt the Ward.D hierarchical clustering algorithm (Murtagh & Legendre, 2014) to cluster the algorithms and data sets (see Fig. 4). For horizontal observations (e.g., data sets) and vertical observations (e.g., algorithms), we picked $k = 2$ and $k = 5$, respectively, as the number of clusters to be created. Using precision and recall metrics as examples, there are two distinct groups of data sets, with UNSW-NB15 and Hikari comprising one group while NSL-KDD and NSL-KDD_20% comprising the other. On the other hand, considering MCC as an example, the clustering approach successfully groups the classifiers according to the algorithm family they originally belong to. For instance, individual LightGBM and Bag-LGB-$\mathcal{X}$ share the same proximity (see violet dendogram in Fig. 4d).

We further investigate whether the performance differences are statistically significant. Two statistical tests (e.g., Friedman and Nemenyi test (Demšar, 2006; Friedman, 1940)) are considered in this appraisal procedure. Friedman test is a non-parametric statistical procedure where the rank of each algorithm on each intrusion data set is calculated not based on the explicit performance metrics (Japkowicz & Shah, 2011). Table 4 shows the mean Friedman rank of each algorithm as an individual classifier. CatBoost performs better than any algorithms as indicated a lower rank value, specifically in terms of accuracy, precision, and MCC metrics, while GBM be the best performing algorithm in terms of recall metric. Moreover, Fig. 6 depicts the average rank of numerous ensemble schemes with varying ensemble sizes and base algorithms. The best bagging scheme can be obtained by 50 GBMs, demonstrating the superiority of the ensemble scheme against other rivals across the board, particularly with respect to accuracy, recall, F1, and MCC metrics. In addition, an improved performance can also be obtained when training XGB classifier in bagging, particularly in terms of precision metric (see Table 5).

We extend our analysis by conducting another statistical significance test based on Nemenyi test (Demšar, 2006). Figs. 5 and 6 depict

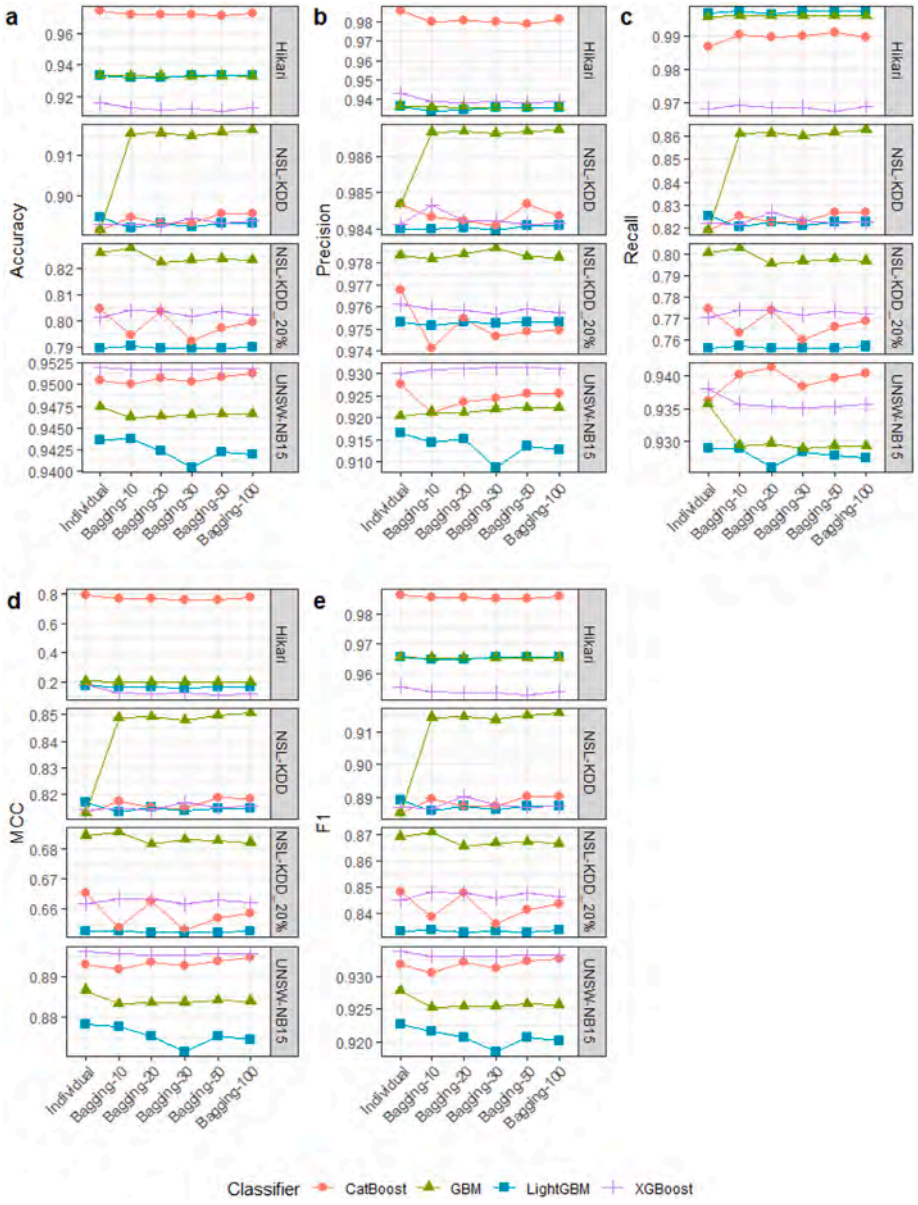


Fig. 3. Performance benchmark between the proposed models (e.g., Bagging-$\mathcal{X}$) and their individual models over various intrusion data sets with respect to accuracy, precision, recall, MCC, and F1 metrics.

the application of significance test derived by combining the significance threshold (e.g., critical difference (CD)) from Nemenyi test with the average ranking from the Friedman test. We plot two horizontal lines denoting the CD values obtained at each significance level ($\alpha = 0.05$ and $\alpha = 0.01$). Therefore, if the rank of an algorithm surpasses the threshold lines, it has performed poorly. Fig. 5 reveals that LightGBM performs worse than the rest algorithms with respect to accuracy, MCC, and precision metrics at all significance levels. However, regarding the recall metric, there are no significant performance differences between algorithms. Similarly, Fig. 6 indicates that LightGBM gains no benefit when it is trained in bagging fashion. For all ensemble schemes containing LightGBM as a base learner, it is evident that they are inferior to any ensemble schemes across the board.

To demonstrate that our proposed model is superior to previously published works, we present a comparative analysis using the same validation method (e.g., holdout) and a complete feature set (e.g., no feature selection task is performed) on each data set. Table 6 and 7 confirm the effectiveness of the proposed model in comparison with other similar works. In particular, our anomaly detector reduces the false alarm rate (FPR) significantly by 1.3% on the KDDTest+ set, while maintaining an excellent precision and F1 scores of 98.67% and 0.915, respectively. In addition, our detection method is the most effective detection model to date on the KDDTest-21 set, as measured by all performance metrics. On the UNSW-NB15_Test set (see Table 8), the proposed model is superior in all performance metrics with the exception of the FPR metric, where (Zhang et al., 2021) slightly outperforms our detection model by 2.11%.

6.2. Discussion

An anomaly-based intrusion detection system (IDS) is a binary classification problem in which a classification algorithm is designed to identify computer and network intrusions by monitoring system activity and characterizing it as either normal or anomaly. The main contribution of this study is the development of dual ensemble learning, in which a classical bagging (Breiman, 1996) technique is employed to enhance the performance of gradient boosting decision tree (GBDT) algorithms. We proposed a novel dual ensemble learning to improve

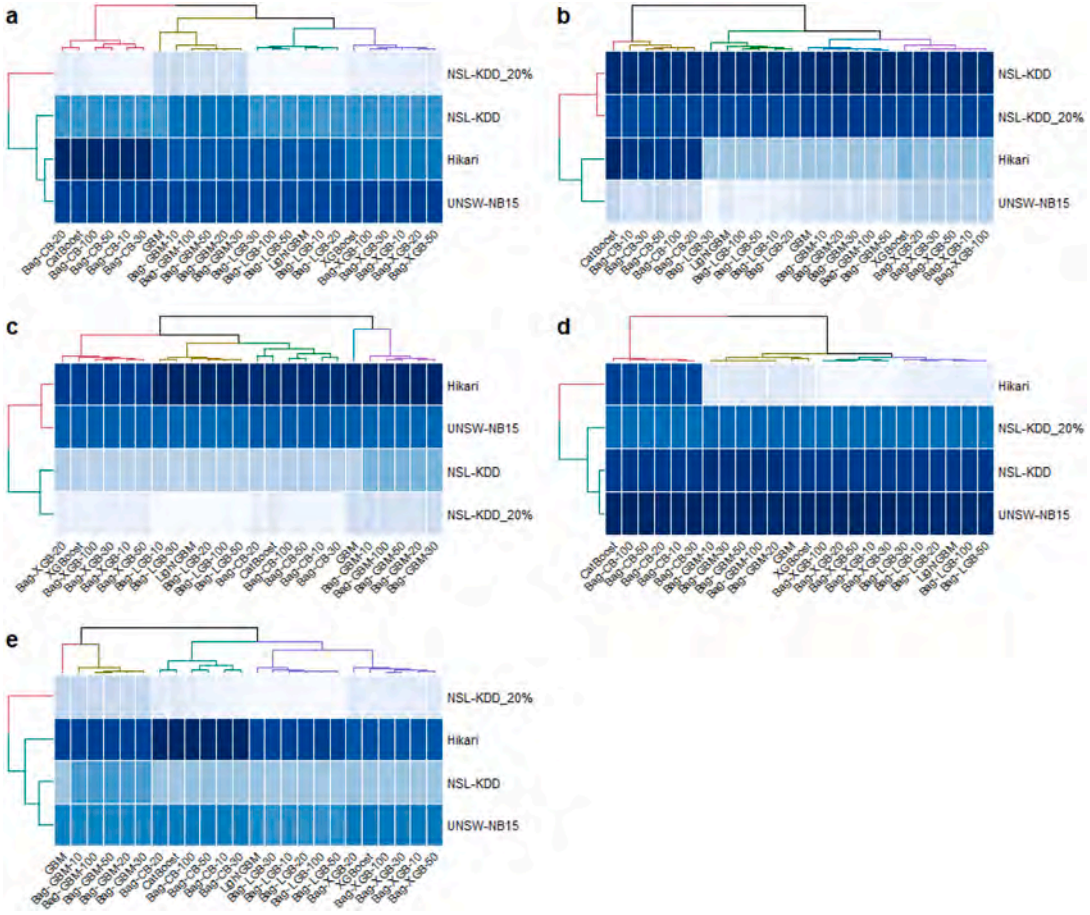


Fig. 4. The result of hierarchical clustering of data sets (horizontal, denoted in two distinguish colors) and algorithms (vertical, denoted in five distinguish colors) with respect to accuracy (a), precision (b), recall (c), MCC (d), and F1 (e) metrics. The corresponding performance score is depicted in blue in each cell (the darker the color the higher the score).

Table 6
Performance comparison between the proposed ensemble model and the existing works on KDDTest+. Best values are denoted in bold.

Ref.	Method	Accuracy (%)	FPR (%)	Precision (%)	Recall (%)	AUC	F1
Tama et al. (2019)	Two-stage ensemble	85.80	11.7	88.0	–	–	–
Alzubi, Anbar, Alqattan, Al-Betar, and Abdullah (2020)	SVM	81.58	–	–	–	–	–
Zhou et al. (2020)	Voting ensemble	87.37	3.19	87.4	–	–	–
Liu et al. (2020)	AlexNet	82.84	–	83.94	82.78	–	0.817
Tama et al. (2020)	Stacking	**92.17**	2.52	–	–	–	–
Ieracitano, Adeel, Morabito, and Hussain (2020)	Autoencoder	84.21	–	–	87.00	–	–
Liu et al. (2021)	LightGBM	89.79	9.13	–	–	–	–
Zhang et al. (2021)	MFFSEM	84.33	24.82	74.61	**97.15**	–	0.841
Krishnaveni et al. (2022)	Weighted majority voting	85.23	12.8	90.3	–	–	0.855
This study	Bagging-GBM	91.57	**1.3**	**98.67**	86.18	**0.981**	**0.915**

Table 7
Performance comparison between the proposed ensemble model and the existing works on KDDTest-21. Best values are denoted in bold.

Ref.	Method	Accuracy (%)	FPR (%)	Precision (%)	Recall (%)	AUC	F1
Tama et al. (2019)	Two-stage ensemble	72.52	18.00	85.0	–	–	–
Zhou et al. (2020)	Voting ensemble	73.57	12.92	73.6	–	–	–
This study	Bagging-GBM	**82.35**	**6.09**	**97.83**	**79.79**	**0.893**	**0.867**

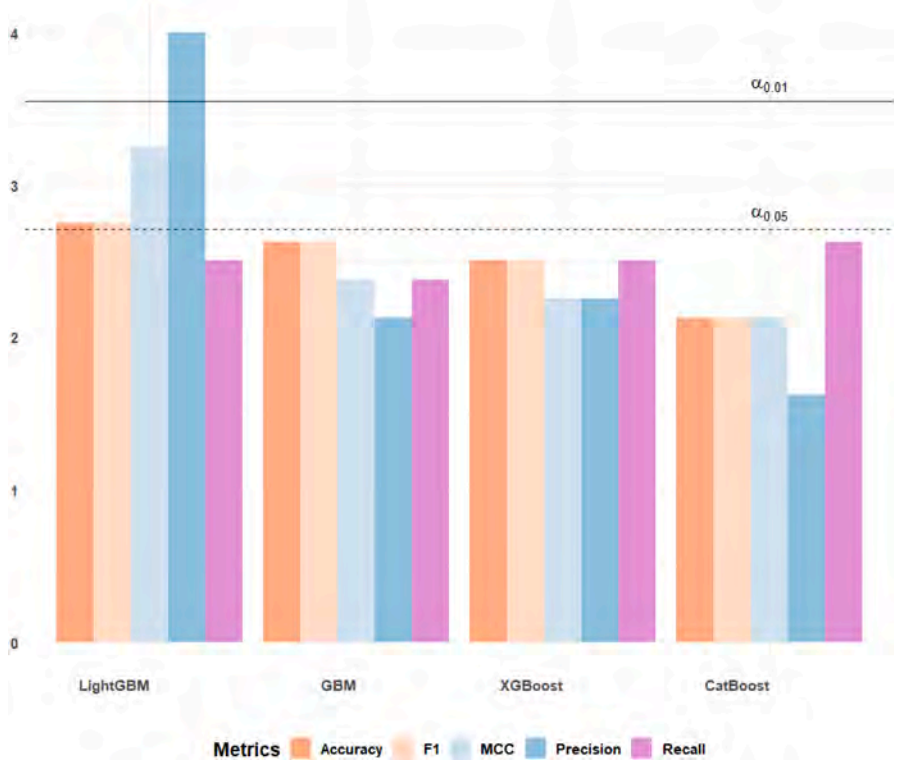


Fig. 5. Critical difference plot of individual classifiers based on the Nemenyi test with two different significance thresholds, e.g., $\alpha = 0.05$ and $\alpha = 0.01$.

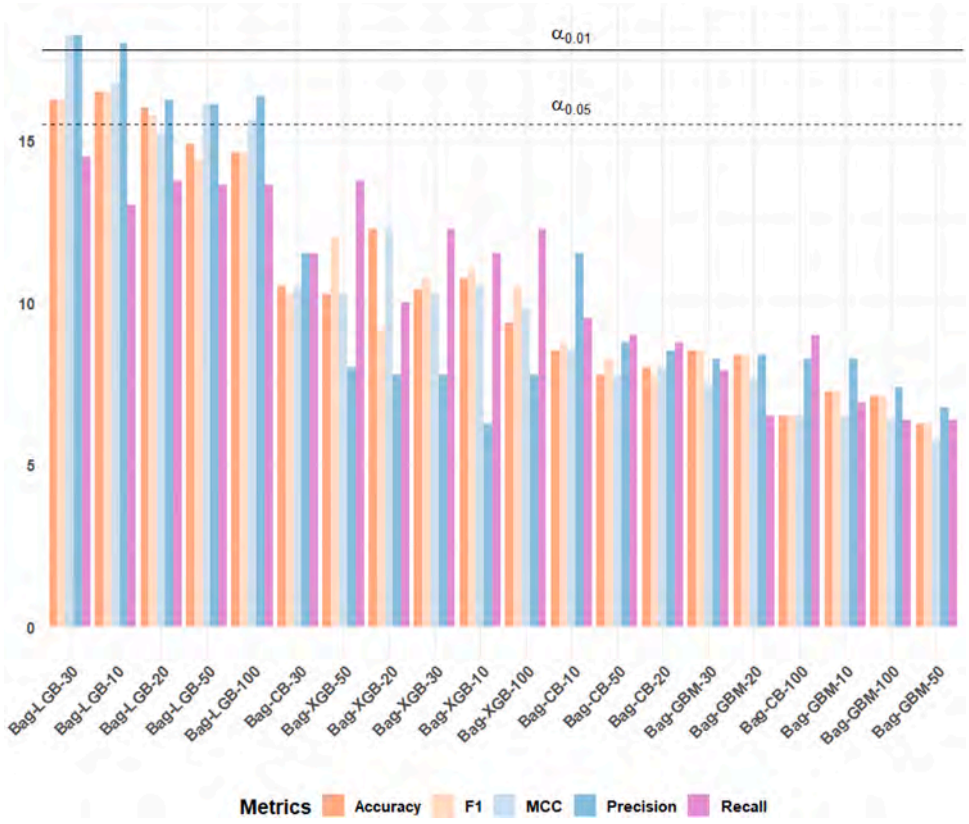


Fig. 6. Critical difference plot of ensemble classifiers based on the Nemenyi test with two different significance thresholds, e.g., $\alpha = 0.05$ and $\alpha = 0.01$.

Table 8
Performance comparison between the proposed ensemble model and the existing works on UNSW-NB15_Test. Best values are denoted in bold.

Ref.	Method	Accuracy (%)	FPR (%)	Precision (%)	Recall (%)	AUC	F1
Tama et al. (2019)	Two-stage ensemble	91.27	8.90	91.60	–	–	–
Tama et al. (2020)	Stacking	92.45	11.3	–	–	–	–
Zhang et al. (2021)	MFFSEM	88.85	**2.27**	–	80.44	–	–
Nazir and Khan (2021)	RF	83.12	3.7	–	–	–	–
Liu et al. (2021)	LightGBM	85.89	14.79	–	–	–	–
Wang, Liu et al. (2022)	LightGBM	88.34	12.46	–	–	–	0.881
This study	Bagging-GBM	**94.66**	4.38	**92.21**	**92.94**	**0.991**	**0.926**

the intrusion detection performance and tested the effectiveness of the model on three different data sets, such as NSL-KDD, UNSW-NB15, and HIKARI-2021. We also tested different GBDT algorithms such as gradient boosting machine (GBM), CatBoost, LightGBM, and XGBoost as base learners. The hyperparameters setting were carefully optimized through a random search. The implications of our findings are discussed as follows.

This study discovered that, among the base algorithms, CatBoost performed the best when evaluated as a single algorithm. CatBoost was the algorithm with the highest accuracy, precision, and MCC metrics. This conclusion slightly contradicts Liu et al. (2021), who suggested that LightGBM was the most effective approach for anomaly-based IDS. Even though Liu et al. (2021) asserted that LightGBM was superior to earlier methods, it should be noted that Liu et al. (2021) relied on a metric (e.g., accuracy) to benchmark the results. This renders the result reported in Liu et al. (2021) less unsuitable, particularly with regard to the usage of such a metric in a binary classification problem. Our analysis, however, considered a more reliable metric (e.g., MCC) that was found to provide more realistic estimations for the proposed model (Chicco et al., 2021). Consequently, our study argues that future works, especially in anomaly-based IDS, should incorporate MCC metric as an evaluation criterion.

Furthermore, our study suggested that a dual ensemble comprised of bagging with 50 GBMs was superior to other combinations. Through an extensive comparison using statistical significance tests, the final dual ensemble scheme could be obtained. The most essential aspect of the statistical significance tests is to demonstrate that there exist performance differences across various schemes and data sets. This analysis helped to improve the generalizability of the proposed model in diverse data sets as the proposed model was deemed to performs consistently in varied environments (Bhuyan, Bhattacharyya, & Kalita, 2013).

Lastly, unlike in other application domains, reducing false positive rate (FPR) in an anomaly-based IDS is challenging when evaluating the classifier's performance. An intrusion detection technique should ideally avoid a high rate of false positives. Nevertheless, it is almost impossible to completely avoid false positives in an anomaly-based IDS (Ahmed, Mahmood, & Hu, 2016; Bhuyan et al., 2013; Fernandes, Rodrigues, Carvalho, Al-Muhtadi, & Proença, 2019). When it comes to reducing false positives, our proposed dual ensemble model achieved significant improvement over existing works such as (Tama et al., 2020; Zhou et al., 2020), among others, on the NSL-KDD data set. On the other hand, our model's performance was still comparable compared to Nazir and Khan (2021), Zhang et al. (2021) on UNSW-NB15 data set.

7. Conclusion

In this study, we developed a dual ensemble technique to address the problem of network anomaly detection tasks. The proposed detection model was built by combining two ensemble strategies, such as bagging and gradient boosting decision tree (GBDT) algorithms. In general, the proposed dual ensemble model outperformed a single GBDT model with fine-tuned parameters. The results of our experiments on multiple intrusion detection data sets demonstrated that bagging with fifty fine-tuned gradient boosting machine (GBMs) performance was superior to other dual ensemble schemes. Compared to other ensemble solutions such as voting, stacking, and two-stage ensemble models, a dual ensemble (e.g., Bagging-GBM) was significantly competitive.

This study has several limitations that can be further addressed in the future. Specifically, only a few data sets were used to validate the proposed model, meaning that the results for other data sets are still questionable. Consequently, future research should consider a wider variety of intrusion data sets in various scenarios and environments. Significant progress has been made in applying deep learning models to tabular data, with research frequently claiming that these models outperform ensemble models, i.e., XGBoost, in some circumstances (Borisov et al., 2021; Shwartz-Ziv & Armon, 2022). In this context, additional study is undoubtedly required, particularly to evaluate if deep learning models perform statistically better on intrusion data sets.

CRediT authorship contribution statement

Maya Hilda Lestari Louk: Conceptualization, Methodology, Validation, Formal analysis, Writing – original draft. **Bayu Adhi Tama:** Writing – review & editing, Supervision, finalization of the manuscript.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data is publicly available

References

Abdulhammed, R., Faezipour, M., Abuzneid, A., & AbuMallouh, A. (2019). Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic. *IEEE Sensors Letters*, *3*(1), 1–4.

Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, *60*, 19–31.

Al, S., & Dener, M. (2021). STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment. *Computers & Security*, *110*, Article 102435.

Alazzam, H., Sharieh, A., & Sabri, K. E. (2020). A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. *Expert Systems with Applications*, *148*, Article 113249.

Aljanabi, M., Ismail, M. A., & Mezhuyev, V. (2020). Improved TLBO-JAYA algorithm for subset feature selection and parameter optimisation in intrusion detection system. *Complexity*, *2020*.

Alzubi, Q. M., Anbar, M., Alqattan, Z. N., Al-Betar, M. A., & Abdullah, R. (2020). Intrusion detection system based on a modified binary grey wolf optimisation. *Neural Computing and Applications*, *32*(10), 6125–6137.

Amma, N. B., Selvakumar, S., & Velusamy, R. L. (2020). A statistical approach for detection of denial of service attacks in computer networks. *IEEE Transactions on Network and Service Management*, *17*(4), 2511–2522.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*(2).

Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems and tools. *Ieee Communications Surveys & Tutorials*, *16*(1), 303–336.

Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., & Kasneci, G. (2021). Deep neural networks and tabular data: A survey. arXiv preprint arXiv:2110.01889.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd Acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Chicco, D., & Jurman, G. (2020). The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, *21*(1), 1–13.

Chicco, D., Tötsch, N., & Jurman, G. (2021). The matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData Mining*, *14*(1), 1–22.

Chkirbene, Z., Erbad, A., Hamila, R., Mohamed, A., Guizani, M., & Hamdi, M. (2020). TIDCS: A dynamic intrusion detection and classification system based feature selection. *IEEE Access*, *8*, 95864–95877.

Chou, D., & Jiang, M. (2021). A survey on data-driven network intrusion detection. *ACM Computing Surveys*, *54*(9), 1–36.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

Denning, D. E. (1987). An intrusion-detection model. *IEEE Transactions on Software Engineering*, (2), 222–232.

Feng, T., & Dou, M. (2021). A weighted intrusion detection model of dynamic selection. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *51*(7), 4860–4873.

Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F., & Proença, M. L. (2019). A comprehensive survey on network anomaly detection. *Telecommunication Systems*, *70*(3), 447–489.

Ferriyan, A., Thamrin, A. H., Takeda, K., & Murai, J. (2021). Generating network intrusion detection dataset based on real and encrypted synthetic attack traffic. *Applied Sciences*, *11*(17), 7868.

Folino, F., Folino, G., Guarascio, M., Pisani, F. S., & Pontieri, L. (2021). On learning effective ensembles of deep neural networks for intrusion detection. *Information Fusion*, *72*, 48–69.

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society for Artificial Intelligence*, *14*(771–780), 1612.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, *11*(1), 86–92.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *The Annals of Statistics*, 1189–1232.

Gupta, N., Jindal, V., & Bedi, P. (2022). CSE-IDS: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems. *Computers & Security*, *112*, Article 102499.

Halim, Z., Yousaf, M. N., Waqas, M., Sulaiman, M., Abbas, G., Hussain, M., et al. (2021). An effective genetic algorithm-based feature selection method for intrusion detection systems. *Computers & Security*, *110*, Article 102448.

Ieracitano, C., Adeel, A., Morabito, F. C., & Hussain, A. (2020). A novel statistical analysis and autoencoder driven intelligent intrusion detection approach. *Neurocomputing*, *387*, 51–62.

Jain, M., & Kaur, G. (2021). Distributed anomaly detection using concept drift detection based hybrid ensemble techniques in streamed network data. *Cluster Computing*, *24*(3), 2099–2114.

Japkowicz, N., & Shah, M. (2011). *Evaluating learning algorithms: a classification perspective*. Cambridge University Press.

Kan, X., Fan, Y., Fang, Z., Cao, L., Xiong, N. N., Yang, D., et al. (2021). A novel IoT network intrusion detection approach based on adaptive particle swarm optimization convolutional neural network. *Information Sciences*, *568*, 147–162.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, *30*.

Khonde, S., & Ulagamuthalvi, V. (2019). Ensemble-based semi-supervised learning approach for a distributed intrusion detection system. *Journal of Cyber Security Technology*, *3*(3), 163–188.

Krishnaveni, S., & Prabakaran, S. (2019). Ensemble approach for network threat detection and classification on cloud computing. *Concurrency Computations: Practice and Experience*, *33*(3), Article e5272.

Krishnaveni, S., Sivamohan, S., Sridhar, S., & Prabakaran, S. (2021). Efficient feature selection and classification through ensemble method for network intrusion detection on cloud computing. *Cluster Computing*, *24*(3), 1761–1779.

Krishnaveni, S., Sivamohan, S., Sridhar, S., & Prabhakaran, S. (2022). Network intrusion detection based on ensemble classification and feature selection method for cloud computing. *Concurrency Computations: Practice and Experience*, *34*(11), Article e6838.

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., et al. (2019). mlr3: A modern object-oriented machine learning framework in R. *Journal of Open Source Software*, *4*(44), 1903.

Liu, J., Gao, Y., & Hu, F. (2021). A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM. *Computers & Security*, *106*, Article 102289.

Liu, L., Wang, P., Lin, J., & Liu, L. (2020). Intrusion detection of imbalanced network traffic based on machine learning and deep learning. *IEEE Access*, *9*, 7550–7563.

Luo, C., Tan, Z., Min, G., Gan, J., Shi, W., & Tian, Z. (2020). A novel web attack detection system for internet of things via ensemble classification. *IEEE Transactions on Industrial Informatics*, *17*(8), 5810–5818.

Mazini, M., Shirazi, B., & Mahdavi, I. (2019). Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms. *Journal of King Saud University-Computer and Information Sciences*, *31*(4), 541–553.

Mishra, P., Varadharajan, V., Tupakula, U., & Pilli, E. S. (2018). A detailed investigation and analysis of using machine learning techniques for intrusion detection. *IEEE Communications Surveys & Tutorials*, *21*(1), 686–728.

Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 military communications and information systems conference*.

Moustafa, N., Turnbull, B., & Choo, K.-K. R. (2019). An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things. *IEEE Internet of Things Journal*, *6*(3), 4815–4830.

Murtagh, F., & Legendre, P. (2014). Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of Classification*, *31*(3), 274–295.

Nancy, P., Muthurajkumar, S., Ganapathy, S., Kumar, S. S., Selvi, M., & Arputharaj, K. (2020). Intrusion detection using dynamic feature selection and fuzzy temporal decision tree classification for wireless sensor networks. *IET Communications*, *14*(5), 888–895.

Nasir, M. H., Khan, S. A., Khan, M. M., & Fatima, M. (2022). Swarm intelligence inspired intrusion detection systems—A systematic literature review. *Computer Networks*, Article 108708.

Nazir, A., & Khan, R. A. (2021). A novel combinatorial optimization based feature selection method for network intrusion detection. *Computers & Security*, *102*, Article 102164.

Nkenyereye, L., Tama, B. A., & Lim, S. (2021). A stacking-based deep neural network approach for effective network anomaly detection. *CMC-Computers Materials & Continua*, *66*(2), 2217–2227.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, *31*.

Rashid, M., Kamruzzaman, J., Imam, T., Wibowo, S., & Gordon, S. (2022). A tree-based stacking ensemble technique with feature selection for network intrusion detection. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, 1–14.

Resende, P. A. A., & Drummond, A. C. (2018). A survey of random forest based methods for intrusion detection systems. *ACM Computing Surveys*, *51*(3), 1–36.

Sagi, O., & Rokach, L. (2018). Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(4), Article e1249.

Sai Satyanarayana Reddy, S., Chatterjee, P., & Mamatha, C. (2019). Intrusion detection in wireless network using fuzzy logic implemented with genetic algorithm. In *Computing and network sustainability* (pp. 425–432). Springer.

Salo, F., Nassif, A. B., & Essex, A. (2019). Dimensionality reduction with IG-PCA and ensemble classifier for network intrusion detection. *Computer Networks*, *148*, 164–175.

Scarfone, K., Mell, P., et al. (2007). Guide to intrusion detection and prevention systems (idps). *NIST Special Publication*, *800*(2007), 94.

Seth, S., Chahal, K. K., & Singh, G. (2021). A novel ensemble framework for an intelligent intrusion detection system. *IEEE Access*, *9*, 138451–138467.

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization.. In *The 4th international conference on information systems security and privacy* ICISSP, (pp. 108–116).

Shwartz-Ziv, R., & Armon, A. (2022). Tabular data: Deep learning is not all you need. *Information Fusion*, *81*, 84–90.

Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. In *2010 IEEE symposium on security and privacy* (pp. 305–316). IEEE.

Tama, B. A., Comuzzi, M., & Rhee, K.-H. (2019). TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system. *IEEE Access*, *7*, 94497–94507.

Tama, B. A., & Lim, S. (2021). Ensemble learning for intrusion detection systems: A systematic mapping study and cross-benchmark evaluation. *Computer Science Review*, *39*, Article 100357.

Tama, B. A., Nkenyereye, L., Islam, S. R., & Kwak, K.-S. (2020). An enhanced anomaly detection in web traffic using a stack of classifier ensemble. *IEEE Access*, *8*, 24120–24134.

Tama, B. A., & Rhee, K.-H. (2019). An in-depth experimental study of anomaly detection using gradient boosted machine. *Neural Computing and Applications*, *31*(4), 955–965.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE symposium on computational intelligence for security and defense applications* (pp. 1–6). IEEE.

Thaseen, I. S., Chitturi, A. K., Al-Turjman, F., Shankar, A., Ghalib, M. R., & Abhishek, K. (2020). An intelligent ensemble of long-short-term memory with genetic algorithm for network anomaly identification. *Transactions on Emerging Telecommunications Technologies*, Article e4149.

Tian, Q., Han, D., Hsieh, M.-Y., Li, K.-C., & Castiglione, A. (2021). A two-stage intrusion detection approach for software-defined IoT networks. *Soft Computing*, *25*(16), 10935–10951.

Van der Laan, M. J., Polley, E. C., & Hubbard, A. E. (2007). Super learner. *Statistical Applications in Genetics and Molecular Biology*, *6*(1).

Wang, Z., Fok, K. W., & Thing, V. L. (2022). Machine learning for encrypted malicious traffic detection: Approaches, datasets and comparative study. *Computers & Security*, *113*, Article 102542.

Wang, Z., Liu, J., & Sun, L. (2022). EFS-DNN: An ensemble feature selection-based deep learning approach to network intrusion detection system. *Security and Communication Networks*, *2022*.

Wei, W., Chen, S., Lin, Q., Ji, J., & Chen, J. (2020). A multi-objective immune algorithm for intrusion feature selection. *Applied Soft Computing*, *95*, Article 106522.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*(2), 241–259.

Wu, C., & Li, W. (2021). Enhancing intrusion detection with feature selection and neural network. *International Journal of Intelligent Systems*, *36*(7), 3087–3105.

Yang, J., Sheng, Y., & Wang, J. (2020). A GBDT-paralleled quadratic ensemble learning for intrusion detection system. *IEEE Access*, *8*, 175467–175482.

Yang, Y., Zheng, K., Wu, B., Yang, Y., & Wang, X. (2020). Network intrusion detection based on supervised adversarial variational auto-encoder with regularization. *IEEE Access*, *8*, 42169–42184.

Zhang, H., Huang, L., Wu, C. Q., & Li, Z. (2020). An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Computer Networks*, *177*, Article 107315.

Zhang, H., Li, J.-L., Liu, X.-M., & Dong, C. (2021). Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection. *Future Generation Computer Systems*, *122*, 130–143.

Zhang, H., Li, Y., Lv, Z., Sangaiah, A. K., & Huang, T. (2020). A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. *IEEE/CAA Journal of Automatica Sinica*, *7*(3), 790–799.

Zhang, Y., & Liu, Q. (2022). On IoT intrusion detection based on data augmentation for enhancing learning on unbalanced samples. *Future Generation Computer Systems*, *133*, 213–227.

Zhong, Y., Chen, W., Wang, Z., Chen, Y., Wang, K., Li, Y., et al. (2020). HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning. *Computer Networks*, *169*, Article 107049.

Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. CRC Press.

Zhou, Z.-H. (2021). Ensemble learning. In *Machine learning* (pp. 181–210). Springer.

Zhou, Y., Cheng, G., Jiang, S., & Dai, M. (2020). Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Computer Networks*, *174*, Article 107247.

Volume 296PA, 15 January 2026

ISSN 0957-4174

Expert Systems with Applications

An International Journal

Editor-in-Chief
Ling Wang

EXPERT SYSTEMS WITH APPLICATIONS

An International Journal

EDITORIAL ADVISORY BOARD

Expert Systems with Applications

Supports open access

Volume 213, Part B

1 March 2023

Previous vol/issue Next vol/issue

Expert Systems with Applications

SeisDeepNET: An extension of Deeplabv3+ for full waveform inversion problem

Vahid Honarbakhsh, Hamid Reza Siahkoohi, Mansoor Rezghi, Hamid Sabeti

Article 118848

View PDF Article preview

Research article Full text access

FSOLAP: A fuzzy logic-based spatial OLAP framework for effective predictive analytics

Sinan Keskin, Adnan Yazıcı

Article 118961

View PDF Article preview

Research article *Open access*

A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier

Sunanda Das, Md. Samir Imtiaz, Nieb Hasan Neom, Nazmul Siddique, Hui Wang

Article 118914

View PDF Article preview

Research article Full text access

A military fleet mix problem for high-valued defense assets: A simulation-based optimization approach

Ismail M. Ali, Hasan Hüseyin Turan, Sondoss Elsawah

Article 118964

View PDF Article preview

Research article Full text access

Hybrid sentiment analysis with textual and interactive information

Jiahui Wen, Anwen Huang, Mingyang Zhong, Jingwei Ma, Youcai Wei

Article 118960

View PDF Article preview

Research article Full text access

Benchmarking state-of-the-art imbalanced data learning approaches for credit scoring

Cuiqing Jiang, Wang Lu, Zhao Wang, Yong Ding

Article 118878

View PDF Article preview

Research article *Open access*

A compression strategy for an efficient TSP-based microaggregation

Armando Maya-López, Antoni Martínez-Ballesté, Fran Casino

Article 118980

View PDF Article preview

Expert Systems with Applications

Development of a cyber-physical-system perspective based simulation platform for optimizing connected automated vehicles dedicated lanes

Xiangmo Zhao, Ying Gao, Shaojie Jin, Zhigang Xu, ... Peng Liu

Article 118972

View PDF Article preview

Research article Full text access

Product diffusion in dynamic online social networks: A multi-agent simulation based on gravity theory

Xiaochao Wei, Haobo Gong, Lin Song

Article 119008

View PDF Article preview

Research article Full text access

A fast and efficient discrete evolutionary algorithm for the uncapacitated facility location problem

Fazhan Zhang, Yichao He, Haibin Ouyang, Wenben Li

Article 118978

View PDF Article preview

Research article Full text access

A two-stage estimation method based on Conceptors-aided unsupervised clustering and convolutional neural network classification for the estimation of the degradation level of industrial equipment

Mingjing Xu, Piero Baraldi, Zhe Yang, Enrico Zio

Article 118962

View PDF Article preview

Review article Full text access

A systematic review of capability and maturity innovation assessment models: Opportunities and challenges

M. Giménez-Medina, J.G. Enríquez, F.J. Domínguez-Mayo

Article 118968

View PDF Article preview

Research article Full text access

Sizing and shape optimization of truss employing a hybrid constraint-handling technique and manta ray foraging optimization

Hongyou Cao, Wen Sun, Yupeng Chen, Fan Kong, Liuyang Feng

Article 118999

View PDF Article preview

Research article Full text access

Collaboration graph for feature set partitioning in data classification

Khalil Taheri, Hadi Moradi, Mostafa Tavassolipour

Expert Systems with Applications

View PDF Article preview

Research article Full text access

A novel two-phase clustering-based under-sampling method for imbalanced classification problems

A. Farshidvard, F. Hooshmand, S.A. MirHassani

Article 119003

View PDF Article preview

Research article Full text access

mmGesture: Semi-supervised gesture recognition system using mmWave radar

Baiju Yan, Peng Wang, Lidong Du, Xianxiang Chen, ... Yirong Wu

Article 119042

View PDF Article preview

Research article Full text access

Efficient graph neural architecture search using Monte Carlo Tree search and prediction network

TianJin Deng, Jia Wu

Article 118916

View PDF Article preview

Research article Full text access

Object localization and edge refinement network for salient object detection

Zhaojian Yao, Luping Wang

Article 118973

View PDF Article preview

Research article Full text access

Prototype contrastive learning for point-supervised temporal action detection

Ping Li, Jiachen Cao, Xingchao Ye

Article 118965

View PDF Article preview

Research article Full text access

Forecasting and trading credit default swap indices using a deep learning model integrating Merton and LSTMs

Weifang Mao, Huiming Zhu, Hao Wu, Yijie Lu, Haidong Wang

Article 119012

View PDF Article preview

Research article Full text access

Batch sizing control of a flow shop based on the entropy-function theorems

Zhifeng Zhang, Janet David, Jun Liu

Expert Systems with Applications

View PDF Article preview

Research article Full text access

Interval-enhanced Graph Transformer solution for session-based recommendation

Huanwen Wang, Yawen Zeng, Jianguo Chen, Ning Han, Hao Chen

Article 118970

View PDF Article preview

Research article Full text access

A graph structure feature-based framework for the pattern recognition of the operational states of integrated energy systems

Li Zhang, Huai Su, Enrico Zio, Luxin Jiang, ... Jinjun Zhang

Article 119039

View PDF Article preview

Research article Full text access

Semantic similarity information discrimination for video captioning

Sen Du, Hong Zhu, Ge Xiong, Guangfeng Lin, ... Nan Xing

Article 118985

View PDF Article preview

Research article Full text access

Dynamic analysis and community recognition of stock price based on a complex network perspective

Yingrui Zhou, Zengqiang Chen, Zhongxin Liu

Article 118944

View PDF Article preview

Research article Full text access

Critical success factors and architecture of innovation services models in data industry

Tsung-Yi Chen, Hsiu-Fang Chang

Article 119014

View PDF Article preview

Research article Full text access

Optimized hyperbolic tangent function-based contrast-enhanced mammograms for breast mass detection

Romesh Laishram, Rinku Rabidas

Article 118994

View PDF Article preview

Research article Full text access

A defense method against backdoor attacks on neural networks

Expert Systems with Applications

Article 118903

View PDF Article preview

Research article *Open access*

Optimal scheduling of electric vehicle charging operations considering real-time traffic condition and travel distance

Yisheng An, Yuxin Gao, Naiqi Wu, Jiawei Zhu, ... Jinhui Yang

Article 118941

View PDF Article preview

Review article *Open access*

A survey on Sign Language machine translation

Adrián Núñez-Marcos, Olatz Perez-de-Viñaspre, Gorka Labaka

Article 118993

View PDF Article preview

Research article Full text access

Improved sine algorithm for global optimization

Yanbin Luo, Weimin Dai, Yen-Wu Ti

Article 118831

View PDF Article preview

Research article Full text access

Training generalizable quantized deep neural nets

Charles Hernandez, Bijan Taslimi, Hung Yi Lee, Hongcheng Liu, Panos M. Pardalos

Article 118736

View PDF Article preview

Review article Full text access

AI-based ICD coding and classification approaches using discharge summaries: A systematic literature review

Rajvir Kaur, Jeewani Anupama Ginige, Oliver Obst

Article 118997

View PDF Article preview

Research article Full text access

Online continual learning via the knowledge invariant and spread-out properties

Ya-nan Han, Jian-wei Liu

Article 119004

View PDF Article preview

Research article Full text access

PANC: Prototype Augmented Neighbor Constraint instance completion in knowledge graphs

Expert Systems with Applications

Feng Zhou, Qun Zhang, Yuan Zhu, Tian Li
Article 119020

View PDF Article preview

Research article Full text access

A weighted distance-based approach with boosted decision trees for label ranking

Alessandro Albano, Mariangela Sciandra, Antonella Plaia
Article 119000

View PDF Article preview

Research article *Open access*

Multimodal detection of epilepsy with deep neural networks

Loukas Ilias, Dimitris Askounis, John Psarras
Article 119010

View PDF Article preview

Research article Full text access

Learning to automatically spectate games for Esports using object detection mechanism

Ho-Taek Joo, Sung-Ha Lee, Cheong-mok Bae, Kyung-Joong Kim
Article 118979

View PDF Article preview

Research article Full text access

Multiple attentional aggregation network for handwritten Dongba character recognition

Yanlong Luo, Yiwen Sun, Xiaojun Bi
Article 118865

View PDF Article preview

Research article Full text access

Development of an EEG artefact detection algorithm and its application in grading neonatal hypoxic-ischemic encephalopathy

Mark E. O'Sullivan, Gordon Lightbody, Sean R. Mathieson, William P. Marnane, ... John M. O'Toole
Article 118917

View PDF Article preview

Review article Full text access

A survey of smart home energy conservation techniques

Muhammad Zaman Fakhar, Emre Yalcin, Alper Bilge
Article 118974

View PDF Article preview

Expert Systems with Applications

EffViT-COVID: A dual-path network for COVID-19 percentage estimation

Joohi Chauhan, Jatin Bedi

Article 118939

View PDF Article preview

Research article Full text access

The multi-depot general share-a-ride problem

Vincent F. Yu, Mareg Marye Zegeye, Sisay Geremew Gebeyehu, Putu A.Y. Indrakarna, Panca Jodiawan

Article 119044

View PDF Article preview

Research article Full text access

Improving the performance of stochastic local search for maximum vertex weight clique problem using programming by optimization

Yi Chu, Chuan Luo, Holger H. Hoos, Haihang You

Article 118913

View PDF Article preview

Research article Full text access

MACR: Multi-information Augmented Conversational Recommender

Chengyang Zhang, Xianying Huang, Jiahao An

Article 118981

View PDF Article preview

Research article Full text access

Personalized Dynamic Attention Multi-task Learning model for document retrieval and query generation

Jun Zeng, Yang Yu, Junhao Wen, Wenying Jiang, Luxi Cheng

Article 119026

View PDF Article preview

Research article Full text access

Exploring indirect entity relations for knowledge graph enhanced recommender system

Zhonghai He, Bei Hui, Shengming Zhang, Chunjing Xiao, ... Fan Zhou

Article 118984

View PDF Article preview

Research article Full text access

Identifying firm-specific technology opportunities in a supply chain: Link prediction analysis in multilayer networks

Yingwen Wu, Yangjian Ji, Fu Gu

Article 119053

View PDF Article preview

Expert Systems with Applications

An adaptive quadratic interpolation and rounding mechanism sine cosine algorithm with application to constrained engineering optimization problems

Xiao Yang, Rui Wang, Dong Zhao, Fanhua Yu, ... Huiling Chen

Article 119041

View PDF Article preview

Research article Full text access

A hybrid transient search naked mole-rat optimizer for image segmentation using multilevel thresholding

Supreet Singh, Nitin Mittal, Anand Nayyar, Urvinder Singh, Simrandeep Singh

Article 119021

View PDF Article preview

Research article *Open access*

Three-dimensional reconstruction using SFM for actual pedestrian classification

Francisco Gomez-Donoso, Julio Castano-Amoros, Felix Escalona, Miguel Cazorla

Article 119006

View PDF Article preview

Research article Full text access

Multi-view and Multi-level network for fault diagnosis accommodating feature transferability

Na Lu, Zhiyan Cui, Huiyang Hu, Tao Yin

Article 119057

View PDF Article preview

Research article Full text access

CSwin-PNet: A CNN-Swin Transformer combined pyramid network for breast lesion segmentation in ultrasound images

Haonan Yang, Dapeng Yang

Article 119024

View PDF Article preview

Research article Full text access

Boosted sooty tern optimization algorithm for global optimization and feature selection

Essam H. Houssein, Diego Oliva, Emre Çelik, Marwa M. Emam, Rania M. Ghoniem

Article 119015

View PDF Article preview

Research article Full text access

Multi-scale deep multi-view subspace clustering with self-weighting fusion and structure preserving

Jiao Wang, Bin Wu, Zhenwen Ren, Hongying Zhang, Yunhui Zhou

Article 119031

View PDF Article preview

Expert Systems with Applications

Parallel multiple watermarking using adaptive Inter-Block correlation
Xingrun Wang, Xiaochen Yuan, Mianjie Li, Ying Sun, ... Jianqing Li
Article 119011

View PDF Article preview

Research article Full text access
Protein encoder: An autoencoder-based ensemble feature selection scheme to predict protein secondary structure
Uzma, Usama Manzoor, Zahid Halim
Article 119081

View PDF Article preview

Research article Full text access
A secure image encryption algorithm based on hyper-chaotic and bit-level permutation
Deyun Wei, Mingjie Jiang, Yang Deng
Article 119074

View PDF Article preview

Research article Full text access
Wind turbine fault detection based on deep residual networks
Jiayang Liu, Xiaosun Wang, Shijing Wu, Liang Wan, Fuqi Xie
Article 119102

View PDF Article preview

Research article Full text access
Deep neural network-based multi-stakeholder recommendation system exploiting multi-criteria ratings for preference learning
Rahul Shrivastava, Dilip Singh Sisodia, Naresh Kumar Nagwani
Article 119071

View PDF Article preview

Research article Full text access
Wavelet transform based deep residual neural network and ReLU based Extreme Learning Machine for skin lesion classification
Fayadh Alenezi, Ammar Armghan, Kemal Polat
Article 119064

View PDF Article preview

Research article Full text access
Vehicle vibro-acoustical comfort optimization using a multi-objective interval analysis method
Haibo Huang, Xiaorong Huang, Weiping Ding, Mingliang Yang, ... Jian Pang
Article 119001

Expert Systems with Applications

Research article Full text access

2D hyperchaotic system based on Schaffer function for image encryption

Uğur Erkan, Abdurrahim Toktas, Qiang Lai

Article 119076

View PDF Article preview

Research article Full text access

Prediction of pipe failures in water supply networks for longer time periods through multi-label classification

Alicia Robles-Velasco, Pablo Cortés, Jesús Muñuzuri, Bernard De Baets

Article 119050

View PDF Article preview

Research article Full text access

A hybrid optimization algorithm and its application in flight trajectory prediction

Xuxu Zhong, Zhisheng You, Peng Cheng

Article 119082

View PDF Article preview

Research article Full text access

A collaboration-based hybrid GWO-SCA optimizer for engineering optimization problems

Yuchen Duan, Xiaobing Yu

Article 119017

View PDF Article preview

Research article Full text access

Superiority-comparison-based transformation, consensus, and ranking methods for heterogeneous multi-attribute group decision-making

Weiwei Li, Pingtao Yi, Lingyu Li

Article 119018

View PDF Article preview

Research article Full text access

Data synthesis with dual-stage sample grouping for electromyography signals

Donghee Lee, Wonseok Yang, Gyoungryul Cho, Dayoung You, Woochul Nam

Article 119059

View PDF Article preview

Research article Full text access

Missing-edge aware knowledge graph inductive inference through dual graph learning and traversing

Yuxuan Zhang, Yuanxiang Li, Yini Zhang, Yilin Wang, ... Jianhua Luo

Expert Systems with Applications

View PDF Article preview

Research article Full text access

Imbalance example-dependent cost classification: A Bayesian based method

Javier Mediavilla-Relaño, Marcelino Lázaro, Aníbal R. Figueiras-Vidal

Article 118909

View PDF Article preview

Research article Full text access

Learning-based systems for assessing hazard places of contagious diseases and diagnosing patient possibility

Mansoor Davoodi, Mohsen Ghaffari

Article 119043

View PDF Article preview

Research article *Open access*

LSTM-CNN model of drowsiness detection from multiple consciousness states acquired by EEG

Chungho Lee, Jinung An

Article 119032

View PDF Article preview

Research article *Open access*

Towards understanding the importance of time-series features in automated algorithm performance prediction

Gašper Petelin, Gjorgjina Cenikj, Tome Eftimov

Article 119023

View PDF Article preview

Research article Full text access

Privacy-Preserving and Traceable Federated Learning for data sharing in industrial IoT applications

Junbao Chen, Jingfeng Xue, Yong Wang, Lu Huang, ... Zhixiong Zhou

Article 119036

View PDF Article preview

Research article Full text access

Swin transformer based vehicle detection in undisciplined traffic environment

Prashant Deshmukh, G.S.R. Satyanarayana, Sudhan Majhi, Upendra Kumar Sahoo, Santos Kumar Das

Article 118992

View PDF Article preview

Research article *Open access*

How different network disturbances affect route choice of public transport passengers. A descriptive study based on tracking

Alessio Daniele Marra, Francesco Corman

Expert Systems with Applications

View PDF Article preview

Research article Full text access

A Kalman filter-based prediction strategy for multiobjective multitasking optimization

Qianlong Dang, Jiawei Yuan

Article 119025

View PDF Article preview

Research article Full text access

Trust-aware location recommendation in location-based social networks: A graph-based approach

Deniz Canturk, Pinar Karagoz, Sang-Wook Kim, Ismail Hakki Toroslu

Article 119048

View PDF Article preview

Research article Full text access

Autonomous path planning with obstacle avoidance for smart assistive systems

Charis Ntakolia, Serafeim Moustakidis, Athanasios Siouras

Article 119049

View PDF Article preview

Research article Full text access

Open-world link prediction via type-constraint embedding and hybrid attention for knowledge reuse of AI chip design

Xiaobo Jiang, Kun He, Yongru Chen

Article 118936

View PDF Article preview

Research article Full text access

An NLP-guided ontology development and refinement approach to represent and query visual information

Ashish Singh Patel, Giovanni Merlino, Antonio Puliafito, Ranjana Vyas, ... Vivek Tiwari

Article 118998

View PDF Article preview

Research article Full text access

Proving unfairness of decision making systems without model access

Yehezkel S. Resheff, Yair Horesh, Moni Shahar

Article 118987

View PDF Article preview

Research article Full text access

Boosted crow search algorithm for handling multi-threshold image problems with application to X-ray images of COVID-19

Songwei Zhao, Pengjun Wang, Ali Asghar Heidari, Xuehua Zhao, Huiling Chen

Expert Systems with Applications

View PDF Article preview

Page 1 of 2

Previous vol/issue Next vol/issue

About this publication

ISSN: 0957-4174

Expert Systems with Applications

COUNTRY

United Kingdom

Universities and research institutions in United Kingdom

Media Ranking in United Kingdom

SUBJECT AREA AND CATEGORY

Computer Science
- Artificial Intelligence
- Computer Science Applications

Engineering
- Engineering (miscellaneous)

PUBLISHER

Elsevier Ltd

SJR 2024

1.854 Q1

H-INDEX

290

PUBLICATION TYPE

Journals

ISSN

09574174

COVERAGE

1990-2025

INFORMATION

Homepage

How to publish in this journal

SCOPE

Expert Systems With Applications is a refereed international journal whose focus is on exchanging information relating to expert and intelligent systems applied in industry, government, and universities worldwide. The thrust of the journal is to publish papers dealing with the design, development, testing, implementation, and/or management of expert and intelligent systems, and also to provide practical guidelines in the development and management of these systems. The journal will publish papers in expert and intelligent systems technology and application in the areas of, but not limited to: finance, accounting, engineering, marketing, auditing, law, procurement and contracting, project management, risk assessment, information management, information retrieval, crisis management, stock trading, strategic management, network management, telecommunications, space education, intelligent front ends, intelligent database management systems, medicine, chemistry, human resources management, human capital, business, production management, archaeology, economics, energy, and defense. Papers in multi-agent systems, knowledge management, neural networks, knowledge discovery, data and text mining, multimedia mining, and genetic algorithms will also be published in the journal.

Quartiles

FIND SIMILAR JOURNALS

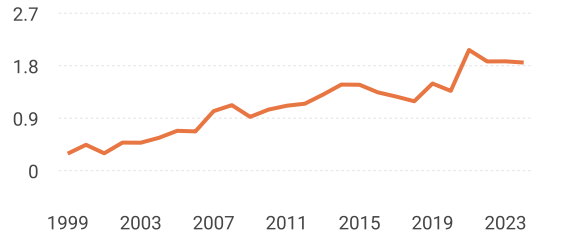
1
International Journal of Computational Intelligence
CHE
64%
similarity

2
Evolving Systems
DEU
64%
similarity

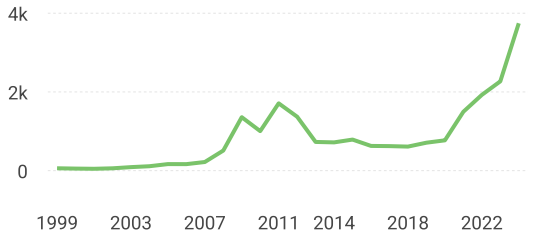
3
Artificial Intelligence Review
NLD
63%
similarity

4
Intelligent S Applications
NLD
6
si

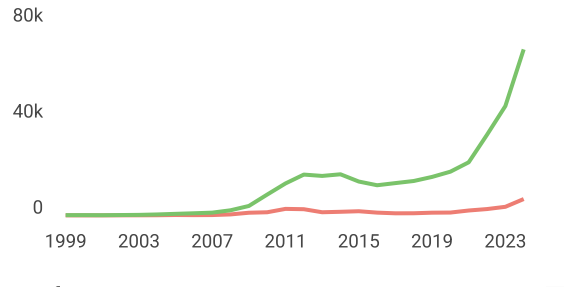
SJR



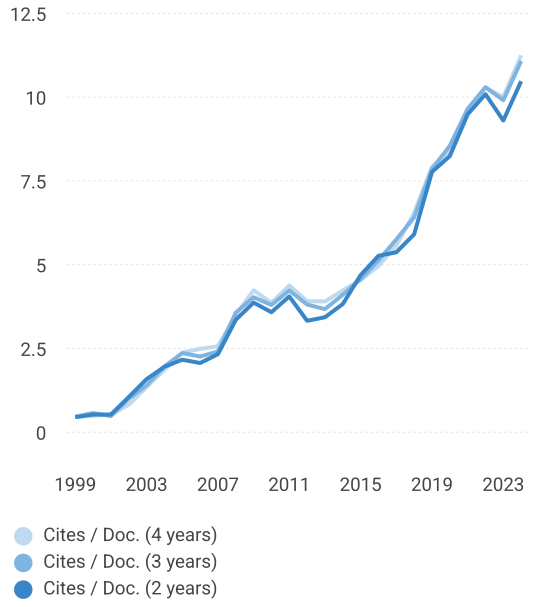
Total Documents



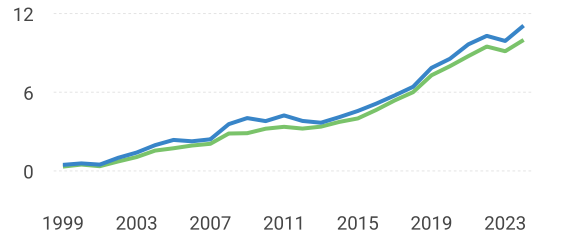
Total Cites Self-Cites



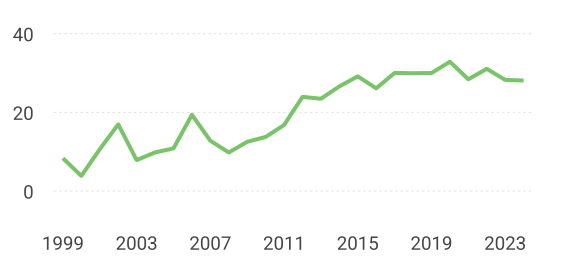
Citations per document



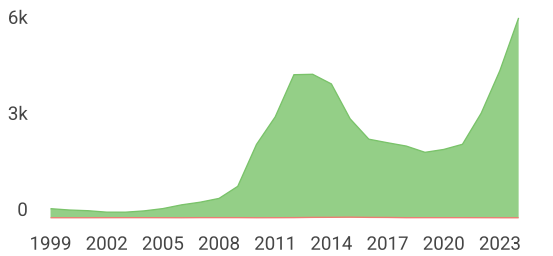
External Cites per Doc Cites per Doc



% International Collaboration

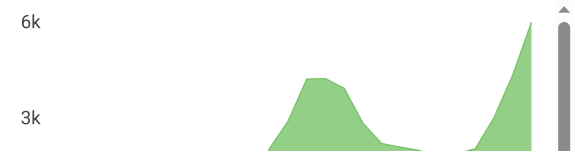


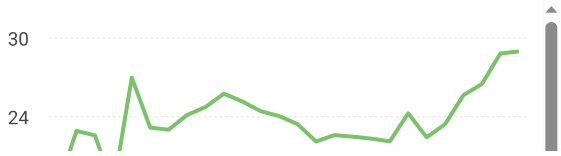
Citable documents Non-citable documents



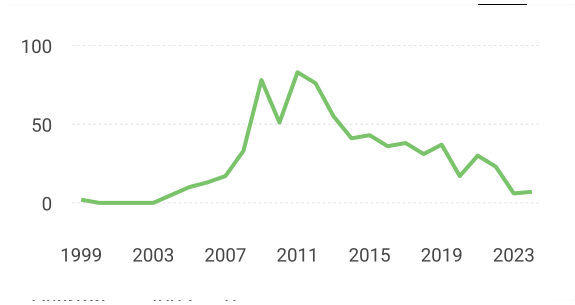
Cited documents Uncited documents

% Female Authors

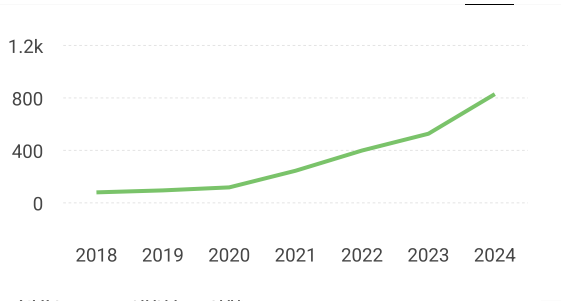




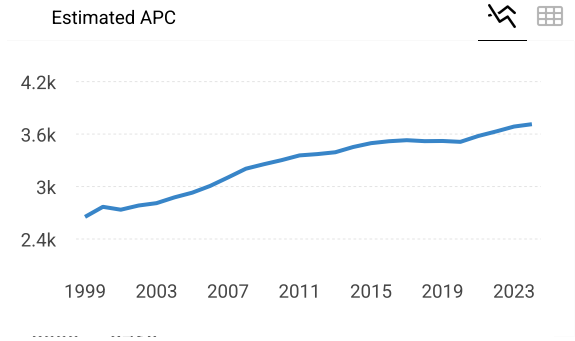
Documents cited by public policy (Overton)



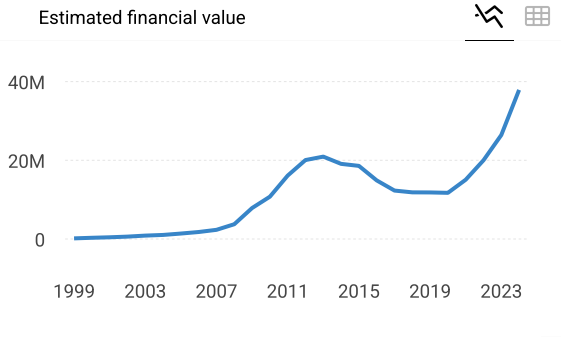
Documents related to SDGs (UN)



Estimated APC



Estimated financial value



← Show this widget in your own website

Just copy the code below and paste within your html code:

```
<a href="https://www.scimag
```

SCImago Graphica

Explore, visually communicate and make sense of data with our **new data visualization tool**.

Metrics based on Scopus® data as of March 2025

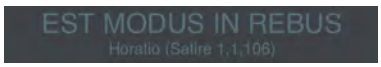
Loading comments...

Developed by: SCImago

Powered by: Scopus

Ads

Follow us on @ScimagoJR



Source details

Expert Systems with Applications

Years currently covered by Scopus: from 1990 to 2026

Publisher: Elsevier

ISSN: 0957-4174

Subject area: Engineering: General Engineering | Computer Science: Computer Science Applications | Computer Science: Artificial Intelligence

Source type: Journal

CiteScore 2024
15.0

SJR 2024
1.854

SNIP 2024
2.548

View all documents | Set document alert | Save to source list

CiteScore | CiteScore rank & trend | Scopus content coverage

CiteScore 2024

15.0 = 141,273 Citations 2021 - 2024 / 9,435 Documents 2021 - 2024

Calculated on 05 May, 2025

CiteScoreTracker 2025

13.8 = 154,496 Citations to date / 11,202 Documents to date

Last updated on 05 September, 2025 • Updated monthly

CiteScore rank 2024

Category	Rank	Percentile
Engineering General Engineering	#10/344	97th
Computer Science Computer Science Applications	#52/947	94th

View CiteScore methodology | CiteScore FAQ | Add CiteScore to your site

About Scopus

What is Scopus

Content coverage

Scopus blog

Scopus API

Privacy matters

Language

日本語版を表示する

查看简体中文版本

查看繁體中文版本

Просмотр версии на русском языке

Customer Service

Help

Tutorials

Contact us