

New hybrid ensemble method for anomaly detection in data science

Amina Mohamed Elmahalwy, Hayam M. Mousa, Khalid M. Amin

Department of Information Technology, Faculty of Computers and Information, Menofia University, Menofia Governorate, Egypt

Article Info

Article history:

Received Jul 9, 2022

Revised Sep 12, 2022

Accepted Oct 1, 2022

Keywords:

Isolation forest

Isolation forest-k-means

Ensemble learning

vote method

ABSTRACT

Anomaly detection is a significant research area in data science. Anomaly detection is used to find unusual points or uncommon events in data streams. It is gaining popularity not only in the business world but also in different of other fields, such as cyber security, fraud detection for financial systems, and healthcare. Detecting anomalies could be useful to find new knowledge in the data. This study aims to build an effective model to protect the data from these anomalies. We propose a new hyper ensemble machine learning method that combines the predictions from two methodologies the outcomes of isolation forest-k-means and random forest using a voting majority. Several available datasets, including KDD Cup-99, Credit Card, Wisconsin Prognosis Breast Cancer (WPBC), Forest Cover, and Pima, were used to evaluate the proposed method. The experimental results exhibit that our proposed model gives the highest realization in terms of receiver operating characteristic performance, accuracy, precision, and recall. Our approach is more efficient in detecting anomalies than other approaches. The highest accuracy rate achieved is 99.9%, compared to accuracy without a voting method, which achieves 97%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Amina Mohamed Elmahalwy

Department of Information Technology, Faculty of Computers and Information, Menofia University

Menofia Governorate 6131567, Egypt

Email: amina.elmahalawi1@ci.menofia.edu.eg

1. INTRODUCTION

Data science is an important area of research that is represented by a collection of measurements and observations. It has been interpreted into a form that computers can handle. Data science controls how we measure, route, and record performance measures to streamline businesses and improve decision-making that aided in improving our quality of life. Data science is becoming increasingly important to organizations. One study predicts that by 2023, the global market for data science will reach \$115 billion [1]. In areas of public interest like health-related research, fraud detection, financial market analysis, power consideration, ecological preservation, and others, data science is widely used. This study focuses on the difficulties with data analysis, specifically with data quality [1]. Data quality problems contain uncertainty troubles, noise that contains errors or outliers, inconsistencies that contain inconsistencies in symbols or names, and incompleteness that contains lost values. The idea of this study highlights the anomaly detection problem as one of the most critical challenges in the area of data management in data science.

Finding anomalous points, uncommon events, irregular behaviors, or outliers in a data set is the process of anomaly detection. Significant differences exist between these anomalies and the rest of the data. Anomalies frequently indicate issues such as equipment failures, technical mistakes, structural flaws, bank frauds, intrusion attempts, or health issues [2]. Investigating anomalies helps you clarify the situation, rule out plausible causes, improve the quality of your data, and optimize datasets. A large range of practical issues can be solved with ease via data anomaly detection [2].

Anomaly detection techniques rely on machine learning methods. It can be used to learn the features of a system from observed data. These methods help to improve the progress of detection and enable the detection and classify anomalies in data sets effectively. In our previous research [3], we talked about this point in detail, explaining the most important points in each method. These methods include statistical, nearest neighbor, clustering, subspace, ensemble-based, and other approaches [3].

Most current anomaly detection methods are based on a model that gets the data and starts creating a profile of what a normal data point should look like. Then it identifies points that do not fit these criteria. Next, computing the anomaly score for every data point sample is possible to be anomalous or not. The distance and density of data points are the most common forms used. Many different methods, such as the local outlier factor (LOF) [4], histogram-based outlier score (HBOS) [5], and cluster-based local outlier factor (CBLOF) [6]. The disadvantage of these methods is that they try to find perfect normal data points instead of focusing on improving methods for anomalies. Finally, our previous research concludes that isolation forest executes better than most other outlier detection methods across different datasets, based on receiver operating characteristic (ROC) performance and precision [3]. In this research, we propose a hybrid ensemble model that combines the isolation forest and k-means (IForest-KMeans) method with the random forest classifier to detect the anomalies activities in the data. The fundamental contributions of this research are i) introducing comprehensive research of previous anomaly detection algorithms, ii) proposing an ensemble machine learning model for detecting anomalies is proposed, and iii) accomplishing a comparative study between our proposed model and the most recent ones.

The following is a summary of the remainder of this study. The related work is summarized in Section 2. We present our proposed model in section 3. Section 4 provides experimental work. The results and discussion are presented in Section 5. Section 6 concludes with the conclusion and future work.

2. RELATED WORK

Different approaches were used to improve anomaly detection systems in the early years. The majority of previous research relies on supervised learning techniques, which required normal and abnormal labeled training data to train the models [7]–[9]. For supervised methods, different classification-based machine learning methods such as support vector machine [9], [10], naive Bayes [11], random forest [12], k-nearest neighbor [13], decision tree [14], as well as an anomaly or intrusion detection is carried out using ensemble learning techniques that incorporate multiple classifiers [15]. Some of the work also combined a variety of learning strategies [15]. For unsupervised learning [16]–[19], different machine learning techniques such as local outlier factor, HBOS, and CBLOF are employed.

Liu *et al.* [20] proposed a method for detecting anomalies based on a tree-structured named the isolation forest. IForest has many advantages compared with other machine learning approaches. It is independent of the distance calculation that is required by various distance–or density-based measures to identify anomalies. Moreover, it has a linear complexity over time and requires little memory [21]. IForest also could raise to handle high-dimensional issues in large datasets. These reasons make the isolation forest the best choice to be utilized for anomaly detection in big data scenarios.

Gao *et al.* [22] compared three methods: density-based, classification-based, and isolation-based are contrasted. In addition, a modified strategy based on isolation forest is suggested. The fundamental concept is that the k-means method divides data points into various groups. Then, the anomaly scores of the data points are calculated in each group using the isolation forest approach. Experimental work shows that the improved approach is greater than the classification and density methods in detecting anomaly points.

Feng *et al.* [23] integrated the support vector machine technique with the unsupervised self-organized ant colony network architecture and proposed combining support vectors with ant colony (CSVAC) model. It is used in network intrusion detection. Regardless of the performance of supervised learning techniques, they cannot be applied in various scenarios. For unsupervised learning methods, different clustering-based machine learning methods [23] have been used for anomaly detection, which can be partitioned into multiple classes, such as partitioning-based methods, distance-based methods, or density-based methods. However, these methods have various disadvantages. For instance, partitioning techniques are crucial to the cluster's width and necessitate repeated testing to determine the ideal width [23]. In large datasets, this experiment iteration turns into a very time-consuming process.

Laskar *et al.* [24] structured an intrusion detection system to maintain computer networks from cyberattacks. Proposed a novel unsupervised machine learning method that integrates the k-means algorithm with an isolation forest approach. Experimental results show that the proposed method is efficient in detecting anomalies in data anomalies.

Our goal in this study is to identify anomalies in the data. We present a new model for anomaly detection in various applications by first providing an isolation forest, merging it with k-means [24], and then

employing a random forest classifier. On a different dataset in an industrial setting, we also assess our proposed model for the detection of anomalies.

3. THE PROPOSED METHOD

Traditional machine learning methods still have some problems. It only judges whether the sample is abnormal. It is hard to locate and define the threshold of outliers. For this problem, we proposed the following framework, as shown in Figure 1. Firstly, the isolation forest approach is applied to define the anomaly score of each point. These scores are saved as a new attribute in the data set and fed into the next stage. Secondly, the k-means model is adopted, which partitions the anomaly scores into K clusters. Meaningful class labels are assigned to each cluster. Third, the random forest classifier is applied to the data to predict the labels and then uses voting to determine the final predicted label as normal or abnormal.

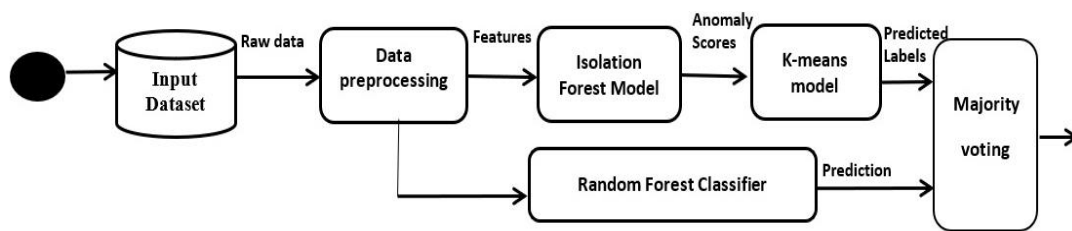


Figure 1. Proposed model framework

3.1. Framework component

3.1.1. Data preprocessing step

The first step is data preprocessing, which is the main step in data analysis. To extract information from a dataset and transform it into a usable form. Datasets are deployed to manage in an intelligible format. In preprocessing, the dataset is split where 70% is training data, and 30% is testing data.

3.1.2. Isolation forest method

IForest is an unsupervised method used in detecting anomaly data points. A data point is isolated when it is separated from the other data points. It can isolate every data point sample in a dataset. Since the anomalies are few and different, they are isolated near the tree root. As depicted in Figure 2, the normal points are isolated at the deeper end of the tree.

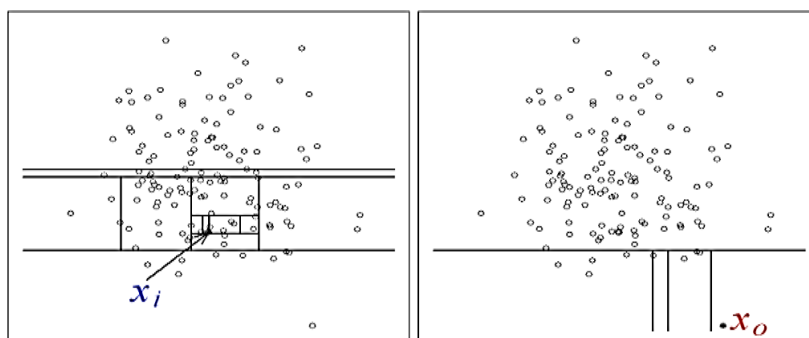


Figure 2. Anomalies (X_o) in the right panel are isolated faster than normal data (X_i) in the left panel

For a specific set of training data points, the isolation forest approach builds a group of trees. The points in the dataset that have short average path lengths on these trees are anomalies. The number of isolated trees in the forest must first be determined in order to construct the isolation forest.

The following steps are then used to create each isolation tree.

- 1) From the training dataset, randomly sample n instances.
- 2) Choose a feature to split randomly.

- 3) Randomly select a split value from a constant distribution passing from the minimum value to the maximum value of the feature chosen in step 2.
- 4) Steps 2 and 3 are reiterated sequentially until all n instances from the random sample selected are isolated in the leaf nodes of the isolation tree. In this track, the anomalies will demand fewer random splits to be isolated in leaf nodes. This results in a shorter expected path length from the root node to the leaf node.
- 5) Compute the anomaly value using (1) [16].

$$c(m) = \left\{ \begin{array}{ll} 2H(m-1) - \frac{2^{(m-1)}}{n} & \text{for } m > 2 \\ 1 & \text{for } m = 2 \\ 0 & \text{other wise} \end{array} \right\} \quad (1)$$

In (1), n is the testing data size, and m is the size of the sample set. Both m and n can be of equal size when the same data set is used for training. H is the harmonic number and can be calculated by $H(i) = \ln(i) + \gamma$, where $\gamma = 0.5772136649$ (the Euler–Mascheroni constant [16]). As $c(m)$ is the average of $h(x)$ given m , we can use this to normalize $h(x)$.

The anomaly scores of a data point x can be defined with the (2) [16].

$$s = 2^{\frac{-E(h(x))}{c(m)}} \quad (2)$$

$E(h(x))$ is the average of $h(x)$ from a set of i trees. With this formula, we can conclude three points.

- When $E(h(x))$ approaches $c(m)$, then $s(x; m) = s = 2^{\frac{-c(m)}{c(m)}} c(m)$. Resulting in s approaching 0.5.
- when $E(h(x))$ approaches 0, meaning it is close to the root, it results in s approaching 1.
- When $E(h(x))$ approaches $m - 1$, the end of the tree, it results in s approaching with these previous three points in mind; we can infer the following three as well.

If the data point has an anomaly score of s close to 1, it is very likely to be an anomaly. If the data point has an anomaly score of s smaller than 0.5, it is very likely to be an inlier. If all the data points return an anomaly score of around 0.5, then it is very likely that it is normal. After applying the isolation forest and determining the anomaly score for each point, the anomaly score is added to the dataset as a new attribute. k-means clustering is adapted to the new data to determine the normal and abnormal points.

3.1.3. k-means clustering algorithm

k-means is a well-known unsupervised learning method used to solve clustering problems. The technique uses a simple strategy to categorize a dataset into a predetermined number of clusters. First, every point should be placed in the cluster whose mean has the smallest squared Euclidean distance. Next, new means (centroids) of the observations in the new clusters are updated and calculated [23].

The output of IForest (anomaly score) is fed to a k-means algorithm as the second layer. This classifies the results from the forest into clusters of normal/anomaly classes. The k-means method aims to cluster n data points into K groups so that each data point can only belong to one of them. In k-means, k has a predetermined value. The technique attempts to divide the data points into k distinct clusters with the goal of maintaining the most comparable data points in the same cluster while also making sure the spacing between the data points in various clusters is as much as possible. The following is how k-means operates: i) firstly, k data points are chosen at random. Then each data point is then assigned to one of the k clusters so that each cluster only includes one data point; ii) for each cluster, the arithmetic mean (also known as the cluster's centroid) of each cluster's data points is determined; iii) subsequently, the squared distance between each data point and the cluster centroids is calculated. Each data point is assigned to the closest cluster based on the value of the determined distance. When the total squared distance between the data points and the cluster's centroid reaches the lowest; iv) steps 2-4 are repeated until the centroids do not change.

3.1.4. Random forest classifier

Random forest [25] is a supervised machine learning method that was proposed by Breiman in 2001. It is difficult to overfit and effective at reducing noise. Additionally, it is simple to implement, performs quickly during training, and does not affect abnormal points. It has been extensively utilized in a wide range of fields for classification and regression tasks due to its superior performance in addition to its straightforward structure. From the original data, a random forest uses a random sampling method to create multiple sub-training sets and test sets. The working mechanism of the random forest classifier is as follows: i) randomly choose k features from a total of m features, ii) determine the node using the best split point among k features, iii) divide the node into daughter nodes using the best split, iv) repeat steps 1 through 3 until the desired number of nodes is reached, and v) create a forest by re-performing steps 1 through 4 n times to produce n trees.

The advantages and disadvantages of each method utilized in the proposed framework model are presented in Table 1.

Table 1. Advantages and disadvantages of the used methods

Algorithm	Type	Advantages	Disadvantage
Isolation Forest	Unsupervised	1. It is not based on any distance density or model. 2. Low CPU time and memory consumption. 3. Efficient in anomaly detection.	1. No clear threshold for decision 2. Parameters and t should depend on dataset dimensions and size.
k-Means	Unsupervised	1. Easy to understand and change 2. Low time complexity 3. Capable of handling a huge dataset	1. Critical to the centroid initialization 2. Critical to outliers datapoints 3. Only detect spherical shape cluster 4. Only work with numerical data
Random Forest Classifier	Supervised	1. It produces highly accurate predictions. 2. Over fitting can be avoided with a large amount of data.	1. Complexity while computing large memory size is needed.

3.1.5. Voting ensembles

A voting ensemble is also called majority voting. It is an ensemble machine learning model that incorporates the results of various other models to produce predictions. In comparison to using just one model in the ensemble, the majority voting strategy improves model performance. It is utilized for regression or classification. Calculating the average of the model's predictions is the process of regression. The label with the most votes is projected for classification once the predictions for each label have been added up. Hard voting and soft voting are the two strategies for predicting the classification of the majority of votes.

In a hard vote, the predictions for each class label are added together, and the class label with the highest votes is predicted. Soft voting includes estimating the class label with the highest probability by adding the anticipated probabilities for each class label. When you have two or more models that excel at predictive modeling tasks, you should use a voting ensemble. Most of the predictions made by the ensemble of models must be accurate. We used the hard classification voting ensemble in our model. When the voting ensemble's models can forecast class labels, that is when it is more appropriate. It is not guaranteed that the voting ensemble will perform better than any individual model included in the ensemble.

In majority voting, various methods produce predictions for each instance in the testing data. The prediction that receives the most votes and more than half of the votes is the one that is chosen as the correct answer for each occurrence. The proposed model uses the majority voting method. The majority vote-based ensemble classifier method increases the accuracy by combining the advantages of each individual approach. The proposed model's pseudocode is shown in Figure 3 which combines the predictions from two methodologies the outcomes of IForest-KMeans [24] and random forest using a voting majority.

```

Algorithm 1 Proposed Hyper Ensemble Model
-----
Input: dataset
Output: Normal / Abnormal data point
Type: Training or Evaluation
IForest-KMeans (Data):
  Dp ← Preprocess Data(D)
  if Type = Training then
    ModelIForest ← Isolation Forest: Train (Dp)
    DAS ← getAnomalyScore (ModelIForest)
    ModelKMeans ← KMeans: Train (DAS)
    ModelIForestKMeans ← [ModelIForest, ModelKMeans]
    ModelIForestKMeans: Save (Path) /* Saving the trained models in the local disk */
  else
    DResult1 ← ModelIForestKMeans:Predict(Dp)
  end if
Random Forest (Data):
  Dp ← Preprocess Data(D)
  if Type = Training then
    ModelRF ← Random Forest: Train (Dp)
    DAS ← getAnomalyScore (ModelRF)
    ModelRF: Save (Path) /* Saving the trained models in the local disk */
  else
    DResult2 ← ModelRF: Predict (Dp)
For the evaluation Phase
  DResult1 ← ModelIForestKMeans:Predict(Dp)
  DResult2 ← ModelRF: Predict (Dp)
  Voting Classifier ← (ModelIForestKMeans, ModelRF)
  Voting Classifier ← (DResult1, DResult2)
end for

```

Figure 3. Algorithm for proposed hyper ensemble model

4. EXPERIMENTAL WORK

4.1. Datasets and parameter setting

To evaluate the effectiveness of our model, five test datasets are used, which are KDD Cup-99, Credit Card, Wisconsin Prognosis Breast Cancer (WPBC), Forest Cover, and Pima. These datasets are different scales, dimensions, and fields described in Table 2. The data sets used to come from a bigger data set: in “KDDCUP99”, from the UCI machine learning repository [9]. The KDD Cup-99 is also used for network intrusion detection systems. The WPBC dataset records data on breast cancer cases. The Credit Card was a European–holding transaction. Forest Cover includes information on tree species, shadow coverage, distance to nearby landmarks (roads, etcetera), soil type, and terrain of the area, and Pima, which describes the medical records. We chose these datasets because they have been frequently used in recent years to test various anomaly detection techniques.

Table 2. Experimental datasets

Dataset name	Number of records	Number of attributes	Anomaly rate
KDD Cup-99	703067	4	0.5%
Credit Card	284807	28	0.2%
WPBC	683	9	2.72%
Forest Cover	286048	10	0.9%
Pima	768	8	35%

The experiment was executed using a laptop with an Intel Core™ i7-7700HQ CPU running at 2.80 GHz, 16 GB of memory, and a 64-bit operating system. The operating system is Windows 10 Professional. The python programming language was used to create the code. In the preprocessing phase, the dataset is split into 70% training data and 30% as testing data.

4.2. Evaluation metric

A key component of creating a powerful machine learning model is model evaluation. A common metric is utilized to calculate the performance of the proposed system in order to assess it. The following metrics are used.

4.2.1. ROC curve

The ROC curve is a popular performance evaluation tool. It is plotted against the false positive rate (FPR) with the true positive rate (TPR), where FPR is on the x-axis, and TPR is on the y-axis [18]. TPR is defined as in (3), and FPR is defined as in (4) [18].

$$TPR = \frac{TP}{TP+FN} \quad (3)$$

$$FPR = \frac{FP}{FP+TN} \quad (4)$$

4.2.2. Accuracy

It represents how accurately an anomaly detection system works by measuring the percentage of normal and anomaly samples that are identified correctly. How the accuracy metric is computed is described in (5).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

4.2.3. Recall (true positive rate)

It displays the percentage of positive data points that are actually interpreted positively when compared to all positive data points. This metric is shown in (6). So, it clarifies the percentage of correctly classified anomaly points with respect to all anomaly points.

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

4.2.4. Precision (positive predicted value)

It represents the percentage of relevant samples that are identified in the prediction, as shown in (7). Thus, it reflects how many anomaly points are actually anomalies from the ones that are classified as an anomaly.

$$precision = \frac{TP}{TP+FP} \quad (7)$$

5. RESULTS AND DISCUSSION

As shown in Figures 4(a) to 4(e), we investigate the effectiveness of using the area under the ROC curve (AUC) for our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively. We evaluate them on five different data sets. A good test result should be one in which the curve is closer to the upper left corner. It is evident from Figure 4 that the AUC for the proposed ensemble model ROC curve is higher than that for the isolation forest ROC curve and IForest-KMeans ROC curve in various datasets. Therefore, we can say that the proposed ensemble model did a better result in detecting anomalies in the dataset.

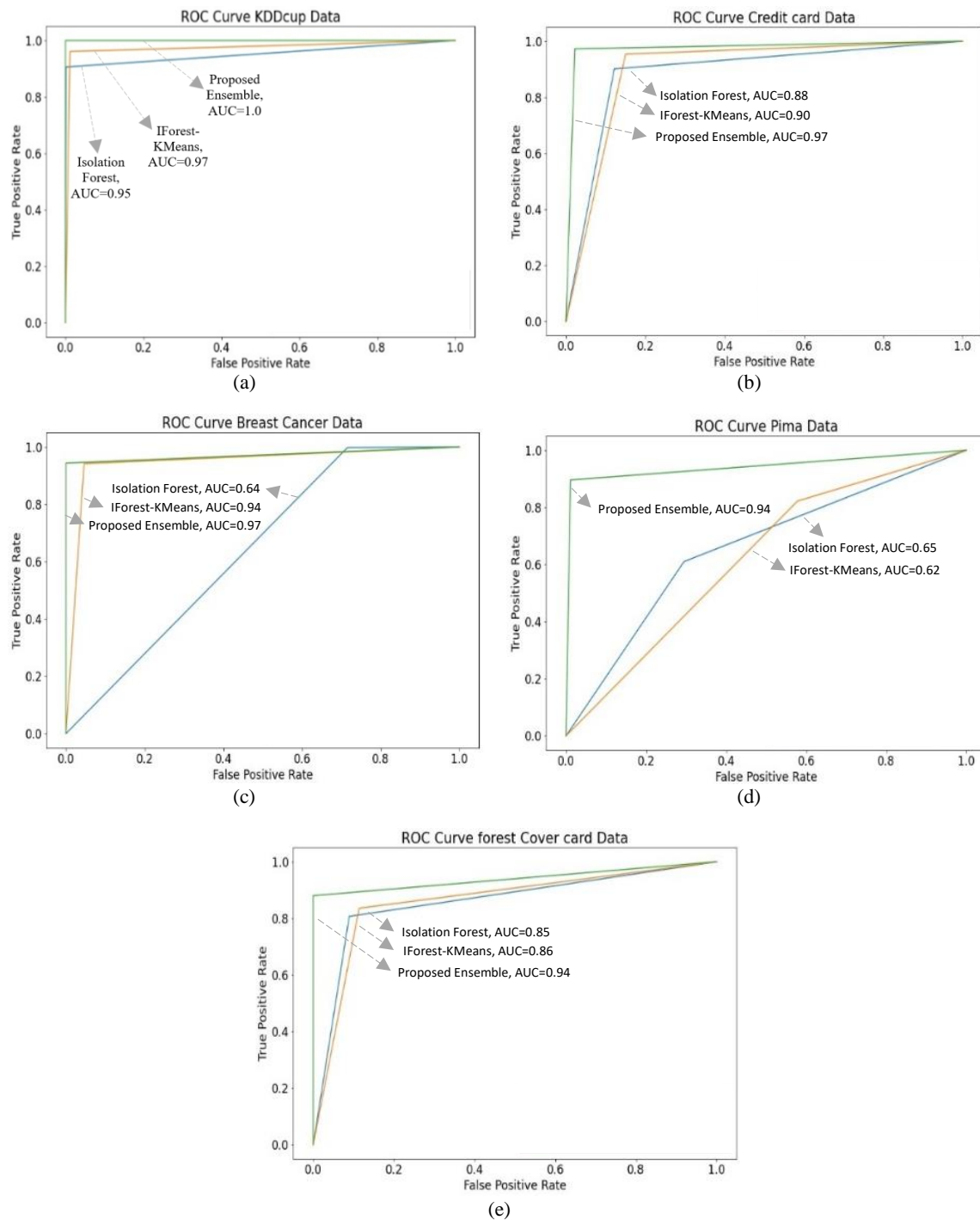


Figure 4. Comparison of ROC and AUC of different models with different datasets; (a) KDDCUP99 dataset, (b) Credit Card dataset, (c) WPBC dataset, (d) Pima dataset, and (e) Forest Cover dataset

5.1. Accuracy

A comparison of the accuracy can be seen in Figure 5 between our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively on each of the datasets (KDD Cup-99, Credit Card, WPBC, Forest Cover, and Pima). For the KDD Cup-99 dataset, it can be observed that the accuracy of isolation forest (95.22%), IForest-KMeans (97.47%), and proposed ensemble method (99.7%). For the Credit Card dataset, it can be observed that the accuracy of isolation forest (88.97%), IForest-KMeans (90.20%), and proposed ensemble method (97.54%). For the WPBC dataset, it can be observed that the accuracy of isolation forest (64.11%), IForest-KMeans (94.88%), and proposed ensemble method (97.07%). For the Forest Cover dataset, it can be observed that the accuracy of isolation forest (85.89%), IForest-KMeans (86.11%), and proposed ensemble method (94%). For the Pima dataset, it can be observed that the accuracy of isolation forest (65.70%), IForest-KMeans (62.18%), and proposed ensemble method (94.24%). Conforming to the comparison of the outcomes in various scale datasets, it is found that the accuracy of the proposed ensemble algorithm outperforms the isolation forest algorithm and the IForest-KMeans in large and small data sets, which is increased by 2%~4% in the KDD Cup-99, 2%~7% in the Credit Card, 20%~3% in the WPBC, 2%~7% in the Forest Cover, and 2%~7% in the Pima respectively.

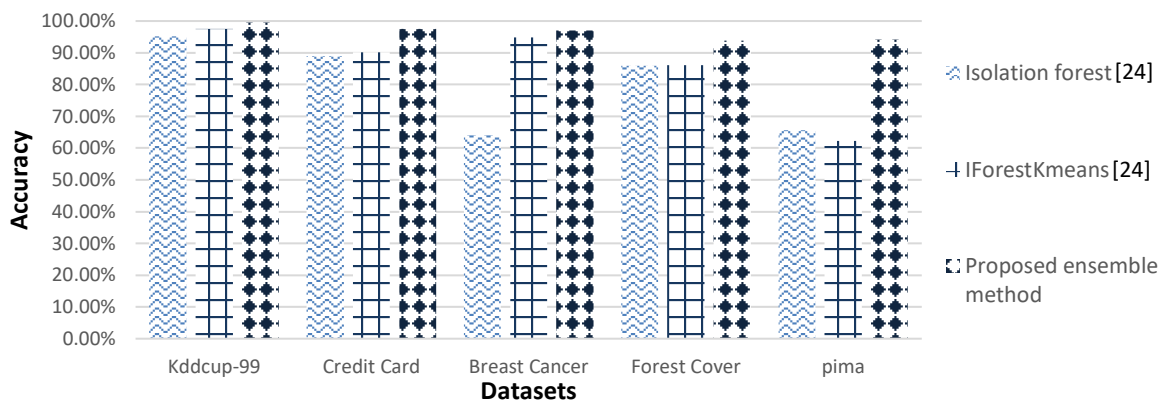


Figure 5. Accuracy between our proposed model and existing approaches in different datasets

5.3. Recall

As shown in Figure 6 the recall of our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively. By comparing each algorithm’s Recall values, it is evident that our proposed method performs well on the KDD Cup-99, Credit Card, Forest Cover, and Pima datasets, and represents a significant advancement compared to the isolation forest and IForest-KMeans algorithms. The performance on the WPBC dataset is a little worse than the original isolation forest algorithm and IForest-KMeans algorithm. Based on the comparison of the outcomes from various scale datasets, It has been discovered that the recall of the proposed ensemble algorithm is better than the isolation forest algorithm and better than the K-mean-isolation forest in large and small data sets, which is increased by 10%~4% in the KDD Cup-99, 7%~2% in the Credit Card, 8%~4% in the Forest Cover, 40%~20% in the Pima a respectively.

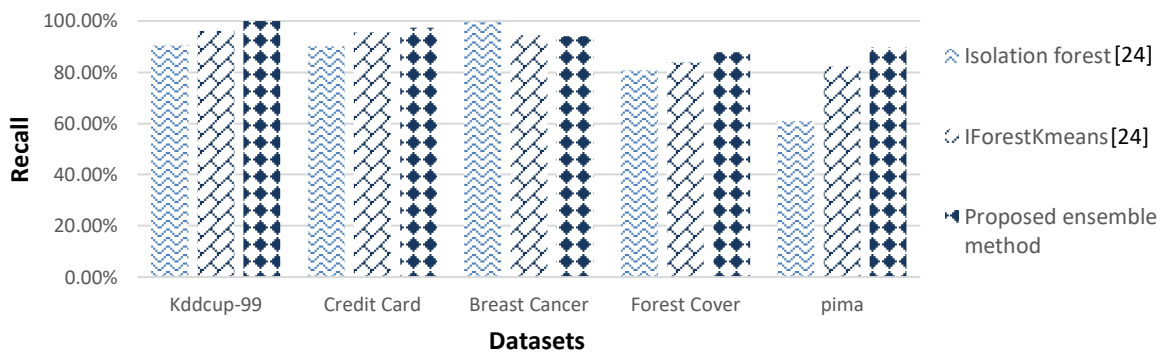


Figure 6. Recall between our proposed model and existing approaches in different datasets

5.4. Precision

Figure 7 depicts the precision of our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively on different scale datasets. It turns out that the precision of the proposed ensemble algorithm is better than the isolation forest algorithm and better than the KMeans-IForest in both large and small data sets, which is increased by 3%~1% in the KDD Cup-99 and 1%~1% in the Credit Card, respectively. From 3% to 18% with WPBC, from 1%~1% in the Forest Cover and 99 and 10%~20% in the Pima dataset.

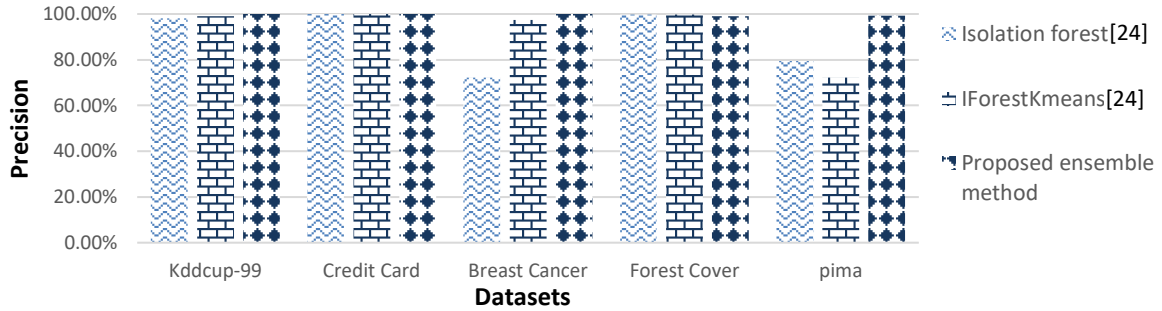


Figure 7. Comparison of our proposed model's precision performance against that of existing methods across several datasets

5.5. Time execution

It represents the amount of time in minutes taken required by a task to complete its execution. As shown in Table 3, it can be observed the time of our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively on each of the datasets (KDD Cup-99, Credit Card, WPBC, Forest Cover, and Pima). For the KDD Cup-99 dataset, it can be observed that the time execution of isolation forest (0.8 m), IForest-KMeans (0.95 m), and proposed ensemble method (0.94 m). For the Credit Card dataset, it can be observed that the time execution of isolation forest (0.37 m), IForest-KMeans (0.81 m), and the proposed ensemble method (1.3 m). It can be seen from the WPBC dataset that the time execution of isolation forest (0.0026 m), IForest-KMeans (0.006 m), and proposed ensemble method (0.0032 m). For the Forest Cover dataset, it can be observed that the time execution of isolation forest (0.26 m), IForest-KMeans (0.63), and proposed ensemble method (0.37 m). For the Pima dataset, it can be observed that the time execution of isolation forest (0.0026), IForest-KMeans (0.0059), and proposed ensemble method (0.0036 m). The proposed algorithm's execution time is evidently competitive with both KDD Cup-99, WPBC, Forest Cover, and Pima datasets. However, it consumes more time than the Credit Card dataset. Therefore, we conclude that the proposed ensemble method is more appropriate for small and medium-sized datasets.

Table 3. Time execution between our proposed model and existing approaches for different datasets

Datasets	Isolation forest [24]	IForest-KMeans [24]	Proposed ensemble method
KDD Cup-99	0.8	0.95	0.94
Credit Card	0.37	0.81	1.3
WPBC	0.0026	0.006	0.0032
Forest Cover	0.26	0.63	0.37
Pima	0.0026	0.0059	0.0036

6. CONCLUSION AND FUTURE WORK

In this research, we suggest a model that can detect anomalies in different types of datasets. First, we address the issues of the different techniques used, such as the isolation forest algorithm, k-means approach, and random forest classifier. Then we proposed an ensemble machine learning model by combining IForest-k-means with a random forest classifier using a voting majority. We compare our proposed method (ensemble method) and the other employed methods isolation forest and IForest-KMeans respectively on different scale datasets in terms of ROC, recall, precision, and time execution. Finally, our proposed model proved that it is efficient for the detection of anomalies in data in different applications. In the future, we will evaluate our proposed model in more use cases. We'll attempt to assess the model in real-time data in a real environment. The proposed hybrid model combines three algorithms; it has a greater computing cost and is more




complicated. Therefore, by concentrating on the number of trees and clusters employed, the sensitivity of the cluster structure, and the clustering quality, we will also try to reduce its complexity.

REFERENCES

- [1] A. Thierer and A. O'Sullivan, "Projecting the growth and economic impact of the internet of things," *Mercatus Center George Mason University*, 2015.
- [2] M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, Jan. 2016, doi: 10.1016/j.jnca.2015.11.016.
- [3] A. Elmahalawy, H. Mousa, and K. Amin, "A comparative study for outlier detection strategies Based on traditional machine learning for IoT data analysis," *IJCI. International Journal of Computers and Information*, Sep. 2021, doi: 10.21608/ijci.2021.91957.1059.
- [4] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," *ACM SIGMOD Record*, vol. 29, no. 2, pp. 93–104, Jun. 2000, doi: 10.1145/335191.335388.
- [5] M. Goldstein and A. Dengel, "Histogram-based outlier score (HBOS): A fast unsupervised anomaly detection algorithm," in *Conference: KI-2012: Poster and Demo Track*, 2012, pp. 1–5.
- [6] Z. He, X. Xu, and S. Deng, "Discovering cluster-based local outliers," *Pattern Recognition Letters*, vol. 24, no. 9–10, pp. 1641–1650, Jun. 2003, doi: 10.1016/S0167-8655(03)00003-5.
- [7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, Jul. 2009, doi: 10.1145/1541880.1541882.
- [8] S. Otoum, B. Kantarci, and H. Mouftah, "A comparative study of AI-based intrusion detection techniques in critical infrastructures," *ACM Transactions on Internet Technology*, vol. 21, no. 4, pp. 1–22, Jul. 2021, doi: 10.1145/3406093.
- [9] V. A. Sotiris, P. W. Tse, and M. G. Pecht, "Anomaly detection through a Bayesian support vector machine," *IEEE Transactions on Reliability*, vol. 59, no. 2, pp. 277–286, Jun. 2010, doi: 10.1109/TR.2010.2048740.
- [10] Y. Liu, A. An, and X. Huang, "Boosting prediction accuracy on imbalanced datasets with SVM ensembles," in *AKDD 2006: Advances in Knowledge Discovery and Data Mining*, 2006, pp. 107–118, doi: 10.1007/11731139_15.
- [11] S. Mukherjee and N. Sharma, "Intrusion detection using naive Bayes classifier with feature reduction," *Procedia Technology*, vol. 4, pp. 119–128, 2012, doi: 10.1016/j.protcy.2012.05.017.
- [12] N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Computer Science*, vol. 89, pp. 213–217, 2016, doi: 10.1016/j.procs.2016.06.047.
- [13] Y. Liao and V. R. Vemuri, "Use of K-nearest neighbor classifier for intrusion detection," *Computers & Security*, vol. 21, no. 5, pp. 439–448, Oct. 2002, doi: 10.1016/S0167-4048(02)00514-X.
- [14] M. Aloqaily, S. Otoum, I. Al Ridhawi, and Y. Jararweh, "An intrusion detection system for connected vehicles in smart cities," *Ad Hoc Networks*, vol. 90, Jul. 2019, doi: 10.1016/j.adhoc.2019.02.001.
- [15] S. Otoum, B. Kantarci, and H. T. Mouftah, "A novel ensemble method for advanced intrusion detection in wireless sensor networks," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, 2020, pp. 1–6, doi: 10.1109/ICC40277.2020.9149413.
- [16] H. Choi, M. Kim, G. Lee, and W. Kim, "Unsupervised learning approach for network intrusion detection system using autoencoders," *The Journal of Supercomputing*, vol. 75, no. 9, pp. 5597–5621, Sep. 2019, doi: 10.1007/s11227-019-02805-w.
- [17] P. Fránti and S. Sieranoja, "How much can k-means be improved by using better initialization and repeats?," *Pattern Recognition*, vol. 93, pp. 95–112, Sep. 2019, doi: 10.1016/j.patcog.2019.04.014.
- [18] S. Otoum, B. Kantarci, and H. Mouftah, "Adaptively supervised and intrusion-aware data aggregation for wireless sensor clusters in critical infrastructures," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6, doi: 10.1109/ICC.2018.8422401.
- [19] W. Feng, Q. Zhang, G. Hu, and J. X. Huang, "Mining network data for intrusion detection through combining SVMs with ant colony networks," *Future Generation Computer Systems*, vol. 37, pp. 127–140, Jul. 2014, doi: 10.1016/j.future.2013.06.027.
- [20] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, Dec. 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.
- [21] Z. Cheng, C. Zou, and J. Dong, "Outlier detection using isolation forest and local outlier factor," in *Proceedings of the Conference on Research in Adaptive and Convergent Systems*, Sep. 2019, pp. 161–168, doi: 10.1145/3338840.3355641.
- [22] R. Gao, T. Zhang, S. Sun, and Z. Liu, "Research and improvement of isolation forest in detection of local anomaly points," *Journal of Physics: Conference Series*, vol. 1237, no. 5, Jun. 2019, doi: 10.1088/1742-6596/1237/5/052023.
- [23] Y. Feng, J. Zhong, C. Ye, and Z. Wu, "Clustering based on self-organizing snt colony networks with supplication to intrusion detection," in *Sixth International Conference on Intelligent Systems Design and Applications*, Oct. 2006, vol. 2, pp. 1077–1080, doi: 10.1109/ISDA.2006.253761.
- [24] M. T. R. Laskar *et al.*, "Extending isolation forest for anomaly detection in big data via K-means," *ACM Transactions on Cyber-Physical Systems*, vol. 5, no. 4, pp. 1–26, Oct. 2021, doi: 10.1145/3460976.
- [25] J. D. Malley, K. G. Malley, and S. Pajevic, "Random forests – trees everywhere," in *Statistical Learning for Biomedical Data*, Cambridge University Press, 2011, pp. 137–154, doi: 10.1017/CBO9780511975820.008.




BIOGRAPHIES OF AUTHORS






Amina Mohamed Elmahalawy    received the M.Sc. degree from the Information Technology Department, Faculty of Computers and Information, Menoufia University, Egypt, in 2015. She is currently pursuing a Ph.D. degree in the Information Technology Department, Faculty of Computers and Information, Menoufia University. She has been a demonstrator and an assistant lecturer with the Faculty of Computers and Information, Menoufia University. Her research interests include machine learning and outlier detection in data science. She can be contacted at eng_amina_86@yahoo.com.

New Hybrid Ensemble method for anomaly detection in data science (Amina Mohamed Elmahalwy)



Hayam M. Mousa    graduated from Menoufia University, Faculty of Computers and Information in 2006. She got an M.Sc. degree from the faculty in 2011. She started a joint supervision scholarship between Menoufia University, Egypt, and University of Lyon, France in 2014. She has finished her PhD. in 2019. She works as a lecturer at the Faculty of computers and information, Menoufia University. She is a director of the Bioinformatics Program at the faculty. She can be contacted at hayam910@gmail.com.



Khalid M. Amin    is a professor and vice dean of the faculty of information and computers for graduate studies and research, Menoufia University, Egypt. He received his Ph.D., 2006 in electronics from the Faculty of Engineering of Ain Shams University. He has published more than 40 papers in international journals and conferences. Currently, he is the associate editor of the international journal of information and computers. His research interests include document image processing, medical image segmentation, and biomedical signal processing. He can be contacted at kh.amin.0.0@gmail.com.