International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-12 Issue-5, April 2023

A Weighted Ensemble Learning Approach Enhanced by the Cuckoo Search Algorithm for Credit Card Fraud Detection

Mohammed Tayebi

Abstract: ***The prevention and identification of unauthorised credit card transactions play a crucial role in combating credit card fraud, as such transactions can lead to significant financial losses for both cardholders and financial institutions. This article introduces a novel ensemble learning approach that utilizes a weighted average technique involving a set of n decision trees (DT). Furthermore, an intelligent feature selection algorithm that implements a recursive feature elimination method is employed. The optimal weights are determined using the Cuckoo Search algorithm. By combining the predicted model probability with its associated weight, the ensemble learning model achieves enhanced overall predictions. The evaluation process involves multiple ensemble learning-based decision trees, including Random Forest (RF), AdaBoost (AD), Gradient Boosting Tree Classifier (GB), and Extra Trees (ET). A credit card transaction dataset is randomly undersampled to evaluate the performance of the proposed solution. Through extensive experiments conducted on real data, the results highlight the superior predictive capabilities of the ensemble learning approach, demonstrating improvements in accuracy (ACC), precision (PRE), recall score (REC), and F-measure (F).***

Keywords: Credit card, Ensemble learning, Cuckoo Search, Fraud detection, Dimensionality reduction

I. INTRODUCTION

The security of online transactions, particularly credit card transactions, is of utmost importance in today's digital landscape. With the growing prevalence of credit card usage for online purchases, the risk of fraud has escalated [1]. Fraudsters employ various techniques to unlawfully obtain sensitive information, resulting in significant financial losses for both individuals and businesses. Fortunately, multiple strategies are available to counter credit card fraud [2]. Notably, machine learning classifiers have emerged as practical tools for categorizing credit card transactions into legitimate or fraudulent [3]. To further improve fraud detection, several approaches and techniques have been successfully implemented, including hyperparameter optimisation [4]–[5], feature selection [6], and ensemble learning [7], among others [8].

Manuscript received on 08 March 2023 | Revised Manuscript received on 05 April 2023 | Manuscript Accepted on 15 April 2023 | Manuscript published on 30 April 2023.
*Correspondence Author(s)
Tayebi Mohammed, Faculty of Sciences and Techniques, Hassan First University of Settat, Settat, Morocco. Email ID: moha28tabi@gmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

Ensemble learning approaches aim to combine multiple predictions to create a more robust and accurate model for fraud detection. The underlying principle is that the collaboration of various models can yield superior results compared to an individual model operating alone. The ensemble models can be trained on the same or different datasets and with the same or different hyperparameters. One significant advantage of ensemble learning is its ability to reduce prediction variance and enhance the stability of results [9]. Moreover, by combining the predictions of multiple models, the ensemble learning technique can improve the overall accuracy of the projections, thereby mitigating the risk of false-positive or false-negative outcomes. Boosting, a type of ensemble learn, specifically focuses on improving the performance of weaker models within the ensemble. On the other hand, stacking is a more sophisticated form of ensemble learning that involves training multiple models using the predictions of different models [10] [11].

This paper presents an innovative ensemble learning approach that integrates the cuckoo search algorithm to enhance prediction accuracy and robustness. The proposed model utilises an intelligent technique to combine predictions from multiple decision tree classifiers, thereby strengthening the overall prediction performance. The cuckoo search algorithm is employed to optimise the weights assigned to each base learner, thereby strengthening the ensemble's effectiveness. By synergising these two techniques, our model aims to outperform traditional ensemble learning approaches. Specifically, each decision tree is trained on a subset of the training data using a unique set of hyperparameters. The final prediction of the model is determined through a weighted combination of the predictions of each tree. The weights are assigned based on the performance of each tree on the validation set, leveraging an optimization process facilitated by the cuckoo search algorithm. This approach prioritizes more accurate trees, assigning them greater importance within the ensemble. This paper is organised as follows: In Section 1, an introduction to credit card fraud transactions is provided. Section 2 presents a comprehensive overview of significant research and developments in the field of credit card fraud transactions. Additionally, Section 3 presents the background information relevant to the proposed solution. The methodology and materials used In this study, the methods are outlined in Section 4. Specifically, Section 5 provides an in-depth examination of the proposed solution. The results obtained from the

© Copyright: All rights reserved.

experiments conducted are discussed in Section 6. Finally, Section 7 concludes this paper, summarizing the key findings and contributions of this research.

II. RELATED WORK

In recent years, the proliferation of online transactions has significantly increased the risk of credit card fraud. Fraudulent activities not only result in financial losses for both cardholders and financial institutions but also erode trust in the electronic payment ecosystem [12][13]. To combat this growing threat, extensive research and development efforts have been undertaken to devise effective techniques for detecting and preventing credit card fraud. This section provides an overview of significant work and advancements in the field of credit card fraud transaction analysis, highlighting key contributions a n d approaches employed by researchers and practitioners [14]. By examining the existing literature, we aim to establish a foundation for our proposed solution and identify gaps that warrant further investigation [15]. Through a comprehensive review of related work, we strive to contribute to ongoing efforts in mitigating credit card fraud and enhancing the security of online transactions [16].

The initial research focuses on data preprocessing and feature engineering.

[17] [18] [19] highlights the importance of preparing the data before applying any detection algorithms. By carefully selecting and engineering relevant features, researchers aim to improve the performance and accuracy of fraud detection models. This step is crucial for identifying patterns and anomalies in credit card transactions that may indicate fraudulent behaviour. In recent years, there has been a shift towards exploring the architecture of fraud detection systems [20][21]. The introduction of novel neural network (NN) designs tailored explicitly for fraud detection reflects the recognition of the power of deep learning in handling complex and nonlinear patterns in transaction data. These NN architectures leverage the capability of deep learning models to automatically learn hierarchical representations, thereby enhancing the accuracy and efficiency of fraud detection algorithms.

Decision tree (DT) algorithms have emerged as popular choices for fraud detection [22][23][24][25]. The simplicity and interpretability of DT models make them particularly valuable in understanding and explaining fraud detection decisions. DTs can easily be interpreted as a set of rules, which can be beneficial in augmenting existing rules derived from domain expertise. Researchers have also explored novel DT algorithms that allow for online learn- ing [26], enabling continuous adaptation to evolving fraud patterns and concept drift.

Beyond DTs and NNs, other machine-learning techniques have been investigated for credit card fraud detection. Evolutionary computing techniques [27][28] offer a unique approach by mimicking biological evolution to optimise fraud detection models. Bayesian networks [29][30] provide a probabilistic framework that captures dependencies and uncertainties in credit card transactions, aiding in the identification of fraudulent patterns. Support vector machines (SVMs) [31], on the other hand, excel at separating different classes of transactions by constructing effective decision boundaries. These diverse approaches reflect the ongoing efforts to address the challenges associated with detecting credit card fraud. Each technique offers unique strengths and considerations, and researchers have explored their application to enhance detection accuracy, efficiency, and adaptability. By leveraging these advancements, researchers and practitioners can develop more robust and effective fraud detection systems that mitigate the risks associated with fraudulent credit card transactions.

Fraudsters often engage in cooperative and organised activities, forming gangs to carry out sophisticated and complex fraudulent schemes [32]. Furthermore, fraudulent activities can involve multiple transactions, with the participation of various cardholders and/or merchants [33]. This creates a complex network of interactions underlying each fraud case. While some research has explored anomaly detection in graphs for fraud detection [34], this work focuses on supervised learning approaches for anomaly detection in graphs. In another study, the authors emphasise feature engineering through the application of social network analysis (SNA). Social network analysis utilises graph theory to examine social structures, offering a range of network metrics and algorithms that can be leveraged to derive informative features summarising the network topology [35]. These features serve as inputs to train machine learning classifiers for fraud detection. For instance, in previous work [36], geonet features, as well as metrics derived from shortest paths and strongly connected components, were utilized to construct a fraud detection model. Another study [37] focused on a heterogeneous network composed of vertices representing users, reviews, and products. Metapath analysis was employed to characterise the network, with the aim of spam detection. In the domain of auction fraud, researchers observed that fraudsters tend to establish stronger connections with their neighbours [38]. Consequently, they utilized the weighted degree centrality measure as an indicator of fraudulent behavior. In one study [39], a semi-supervised algorithm for fraud detection was developed using a Markov random field. The belief propagation algorithm was adapted to incorporate transaction information and observed fraud labels, optimizing the Markov random field. Another popular propagation algorithm is Personalised PageRank (PPR), which calculates a score for each node indicating its proximity to a set of source nodes. In the case of fraud detection, confirmed fraud transactions serve as the source nodes. For instance, in the domain of insurance fraud detection, the Birank algorithm—a variation of PageRank designed for bipartite graphs—was employed [40]. Similarly, PageRank was utilised in another study to detect social insurance fraud [41].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Personalized PageRank was specifically applied to credit card fraud detection [42].

To enhance the effectiveness of the credit card fraud detection technique proposed in [43], several improvements were introduced. Firstly, a regularized Commute time kernel was incorporated to improve the treatment of hub nodes in the network [44]. Secondly, a feedback loop was introduced to integrate the insights and results from fraud investigators into the PageRank model [44]. Lastly, the free energy distance measure was employed [45] further to enhance the accuracy of the fraud detection system. These algorithms, such as the adaptation of belief propagation, Personalised PageRank, and the improvements made in [44], showcase the significance of propagation-based approaches in detecting fraud within complex networks. By leveraging the relationships and connectivity patterns in the network, these algorithms contribute to the identification of suspicious nodes and transactions, ultimately aiding in the prevention and mitigation of fraudulent activities.

In contrast to existing solutions for fraud detection, this paper introduces a novel approach that leverages the power of multiple classifiers through a weighted ensemble learning strategy. The objective is to construct a robust classifier that can effectively identify fraudulent transactions. To achieve this, an optimisation process utilising the cuckoo search algorithm is implemented to determine the optimal weights to be assigned to each classifier. By combining multiple classifiers, the proposed strategy aims to capitalise on their diverse strengths and mitigate their weaknesses. Each classifier contributes its unique perspective and expertise to the ensemble, leading to more accurate and reliable predictions. The ensemble learning approach offers several advantages, including improved performance, enhanced prediction capabilities, and increased resilience to noise and uncertainty in the data. The optimisation process plays a crucial role in fine-tuning the weights assigned to each classifier. The cuckoo search algorithm, renowned for its ability to find optimal solutions in complex optimisation problems, is utilised to search for the optimal weight configuration.

III. BACKGROUND

A. Cuckoo Search Algorithm

The Cuckoo Search algorithm is a nature-inspired optimisation algorithm that mimics the breeding behaviour of the cuckoo bird. It was first proposed by Xin-She Yang and Suash Deb in 2009 and has gained popularity due to its simplicity and effectiveness in solving optimization problems [45]. One of the key features of the Cuckoo Search algorithm is the utilization of Levy flights, which enables efficient exploration of the search space [46]. Let's dive into the mathematical formulation of the Cuckoo Search algorithm. Suppose we have an optimisation problem with an objective function f(x), where x is the vector of decision variables. The goal is to find the optimal solution x^* that minimizes or maximizes the objective function [47].

1. Initialization: Set the population size N, the maximum number of iterations T, and the range for each decision variable. Generate an initial population of N cuckoo solutions randomly within the defined range.
2. Levy Flight Generation: Levy flights are random steps inspired by the behavior of Levy flights observed in nature. Levy flights are generated using a power-law distribution called the Levy distribution, which has heavy tails that enable long-distance jumps. The step size of each cuckoo bird is determined by sampling from the Levy distribution.
3. Evaluate Fitness: Evaluate the fitness of each cuckoo solution in the population using the objective function $f(x)$.
4. Update Population: Sort the cuckoo solutions based on their fitness values, with better solutions having higher fitness. Keep the best solutions in the population and discard the rest. Generate new cuckoo solutions by performing Levy flights and randomly replacing the discarded solutions. Ensure that the new solutions are within the defined range of each decision variable.
5. Abandon Nesting: Some cuckoo species lay their eggs in the nests of other birds, leading to the abandonment of their nests. In the Cuckoo Search algorithm, a fraction of the worst solutions is abandoned and replaced by new solutions. This process introduces diversity in the population and aids in escaping local optima.
6. Iteration and Termination: Repeat steps 2-5 for a specified number of iterations or until a termination criterion is met (e.g., convergence).

The algorithm's convergence behaviour depends on several factors, including the population size, the Levy flight step size, and the exploration-exploitation trade-off. Adjusting these parameters can affect the algorithm's performance and convergence speed [48].

B. Feature selection

Feature selection is a process in machine learning that involves selecting a subset of relevant features from a larger set of available features in a dataset. The goal of feature selection is to enhance model performance by reducing data dimensionality and eliminating irrelevant or redundant features. By selecting the most informative features, feature selection helps mitigate the curse of dimensionality, improves computational efficiency, prevents overfitting, and enhances model interpretability. It also facilitates a deeper understanding of the underlying relationships between features and the target variable, thereby aiding in feature engineering and leveraging domain knowledge. Overall, feature selection is a crucial step in the machine learning pipeline that contributes to improved model performance and more effective decision-making.

In this paper, we propose an algorithm that leverages the power of Recursive Feature Elimination (RFE) in conjunction with a Decision Tree Classifier to conduct feature selection. By iteratively eliminating less essential features based on the ranking or importance provided by the decision tree, our algorithm selects a subset of features that is optimal for the given dataset. We determine the

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

optimal number of features and assess their accuracy score on a validation set. The feature selection process leads to reduced model complexity, improved generalisation, and enhanced interpretability. Additionally, it offers valuable insights into the relationships.

Between features and the target variable, facilitating feature engineering and enhancing domain understanding. Through our algorithm, we efficiently identify the most informative features, ultimately boosting model accuracy and enabling better decision-making.

For more details, the mathematical formulation of the algorithm using Recursive Feature Elimination (RFE) with a Decision Tree classifier for feature selection can be presented as follows (Alg. 1):

Algorithm 1 Recursive Feature Elimination (RFE) with a Decision Tree

```
Input: X_train, X_val, y_train, y_val are the training and validation sets
C = Decision Tree Classifier () be the decision tree classifier
N = range (1, X train. Shape [1]+1) is the range of the number of features to select.
best accuracy = 0
best num features = 0
best selected features = None
for num features in N do
    rfe=RFE (C, num_features_to_select=num_features)
    ref. fit (X train, y train)
    X train selected, X_val_selected=ref. Transform (X train), ref. Transform (X val),
    acc=accuracy_score (C. predict_(y val, C. predict (X val))
    if acc > best_acc then
        best acc=acc
        best num features=num features best selected features = rfe. support end if

end for

Return best selected features.
```

IV. RESEARCH METHODOLOGY

A. Dataset

The dataset used in this study consists of credit card transactions conducted by individuals residing in the United States. The dataset comprises a total of 10,000,000 transactions, each described by nine distinct features [49]. Among these transactions, only 5.96% are identified as fraudulent. Those features encompass both numerical and categorical variables, while the target variable signifies two classes: everyday transactions and abnormal transactions. The categorical features are transformed into numerical representations using the one-hot encoding technique, which facilitates analysis and interpretation. The primary objective of this study is to develop a machine-learning model capable of accurately identifying.

Fraudulent transactions based on the available features. The dataset is commonly split into training, validation, and test sets to facilitate the training, evaluation, and testing of the model's performance. Through the use of this dataset, researchers and practitioners in the field of fraud detection have developed more robust algorithms for detecting credit card fraud. Additionally, the features in the dataset are scaled using the MinMaxScaler to ensure consistency in their ranges.

B. Evaluation Metrics

In the realm of machine learning, evaluation metrics play a crucial role in assessing a model's performance and evaluating its efficacy in addressing a specific problem. The selection of appropriate evaluation metrics depends on factors such as the type of problem, data characteristics, and the model's objectives. In our study, we encountered a binary classification problem, specifically focused on detecting fraudulent transactions within a randomly undersampled dataset. For such classification tasks, several commonly employed metrics prove instrumental in evaluating model performance. The first of these is accuracy (ACC), which quantifies the overall correctness of the model's predictions by measuring the proportion of correctly classified instances. Precision (PRE) measures the accuracy of optimistic predictions, evaluating the model's ability to identify actual positive cases among the predicted positives correctly. Recall (REC), also known as sensitivity or actual positive rate, gauges the model's capability to identify actual positive instances from the entire set of positives. Additionally, the f-measure (F), which combines precision and recalls into a single metric, provides a balanced evaluation of the model's performance by

© Copyright: All rights reserved.

considering both false positives and false negatives. By employing these evaluation metrics, we can comprehensively assess the model's performance in accurately identifying fraudulent transactions. These metrics enable us to measure the model's overall accuracy, precision in identifying fraudulent cases, recall in capturing all instances of fraud, and a balanced F-measure that considers both precision and recall simultaneously. Such a comprehensive evaluation enables us to assess the model's effectiveness in detecting fraudulent activities and make informed decisions about its utility and potential deployment in real-world scenarios.

Table 1: The used evaluation metrics

Name	*Formula*
ACC	$\frac{tn+tp}{tn+fp+tp+fn}$
PRE	$\frac{tp}{fp+tp}$
REC	$\frac{tp}{tp}$
F	$\frac{tn+tp \quad 2\times Precision\times Recall}{Recall+Precision}$

V. PROPOSED SOLUTION

This study introduces a novel approach in the realm of ensemble learning, presenting a weighted ensemble model that combines a set of decision tree classifiers. In this approach, each decision tree is trained independently, utilizing a unique subset of randomly selected data points from the training set along with distinct hyperparameters. The primary objective is to construct a unified model that outperforms individual trees in terms of prediction accuracy. The foundation of this proposed ensemble model lies in assigning appropriate weights to each decision tree, reflecting their respective contributions and significance in the final prediction. The determination of these weights is achieved through an optimisation process that employs the cuckoo search algorithm. The optimisation process is conducted using a validation set, enabling the identification of optimal weights that maximise the overall performance of the ensemble. To elaborate further, let us consider a set of decision trees $T1, T2, ..., Tn$, where each tree corresponds to a weight $w_i \in [-1, 1]$. These weights signify the importance assigned to each tree in the ensemble. By adjusting these weights, the ensemble model can effectively highlight the trees that exhibit stronger predictive capabilities, while minimising the impact of weaker ones. Through the utilization of cuckoo search optimization, the weights are iteratively adjusted to enhance the ensemble's overall predictive accuracy. The prediction of the ensemble E for a new instance x_i is calculated as follows:

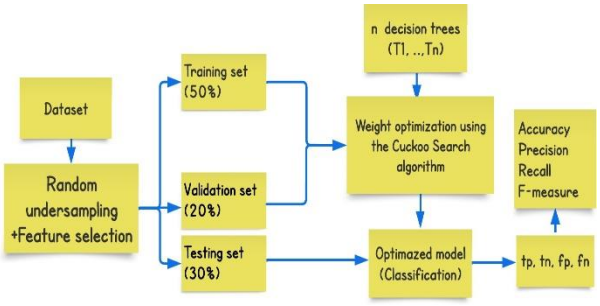
$$E(x_i) = \Phi\left(\sum_{j=1} w_j P_{T_j}(x_i)\right) \quad (1)$$

By incorporating this weighted ensemble learning approach, this work aims to enhance the predictive capabilities of the model, surpassing the performance of individual decision trees. The utilisation of cuckoo search optimisation ensures the determination of optimal weights, thereby enabling the construction of a powerful and accurate ensemble model capable of making robust predictions across various domains and applications. To enhance the performance of our model, we have implemented a weighted optimisation process utilising the Cuckoo search algorithm. This process focuses on determining the optimal weights assigned to each prediction made by the model, denoted as P_{T_j}, given a sample x_i. The optimization process can be modelled as follows

$$w = \arg\min_{w_j} \Im(E(x_i), y_i) \quad (2)$$

These weights play a crucial role in determining the significance and contribution of each prediction to the outcome. By employing the Cuckoo search algorithm, we can efficiently search for the optimal set of weights that maximize the model's predictive accuracy. The final ensemble learning is then.
formulated as follows.

$$E(x_i) = \Phi\left(\sum_{j=1}^{n} w_j P_{T_j}(x_i)\right) \quad (3)$$



[Fig.1: The Proposed Framework]

Table 2: Information About Random Hyperparameters Selected for Each Decision Tree Classifier

Name	Range
max_depth	⊏ 3, 20 ⊐
min_samples_leaf	⊏ 3, 20 ⊐
min_samples_split	⊏ 3, 20 ⊐
Criterion	⊏ *Gini, entropy* ⊐
max_features	⊏ *None, Sqrt, log2* ⊐

In our ensemble learning model, multiple decision tree classifiers are combined to make predictions. Each decision tree classifier can be customized by selecting different hyperparameters. The table contains information about the random hyperparameters selected for each decision tree classifier in the ensemble. Here's a description of the hyperparameters and their ranges:

- Max Depth: This hyperparameter determines the maximum depth of the decision tree. It limits the number of levels in the tree. The range specified is from 3 to 20, meaning that the randomly selected value for each decision tree classifier will fall within this range.
- Min Samples Leaf: This hyperparameter defines the minimum number of samples required to be at a leaf node. It controls the complexity of the tree and helps avoid fitting. The range given is from 3 to 20, indicating that the randomly chosen

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

value for each classifier will be within this range.

- Min_Samples_Split: This hyperparameter specifies the minimum number of samples required to split an internal node. It also contributes to controlling. The complexity of the tree and preventing overfitting. The range provided is from 3 to 20, meaning that each decision tree classifier will have a randomly selected value within this range.
- Criterion: This hyperparameter determines the function used to measure the quality of a split in the decision tree. The options given are" G i n i " and" entropy" (represented as" gini" and" entropy" in the table, respectively). The criterion is randomly chosen for each decision tree classifier in the ensemble.
- Max_Features: This hyperparameter represents the number of features to consider when looking for the best split. The options provided are" None,"" Sqrt," and" Log2" (represented as" None"," Sqrt", and " Log2" in the table, respectively). The randomly selected value for each classifier can be one of these options.

By randomly selecting hyperparameters within the specified ranges, the ensemble learning model ensures diversity among the decision tree classifiers, leading to improved performance and more robust predictions.

The proposed solution follows a well-defined pipeline, as illustrated in Figure 1. The pipeline consists of several steps designed to address the problem at hand effectively. In the first step, a random undersampling method is applied to balance the dataset, ensuring a representative distribution of both normal and abnormal instances. This step helps mitigate the issue of class imbalance, a common problem in fraud detection scenarios. Subsequently, several dimensionality reduction methods are employed to transform the dataset from its original 51 features into a lower-dimensional space. The trans- formed dataset is then divided into three distinct subsets: the training set (50%), the validation set (20%), and the testing set (30%). This division enables the model to be trained on a representative portion of the data, validated to fine-tune its performance, and tested to assess its effectiveness on unseen data.

To optimise the performance of the ensemble learning model, an optimisation process is conducted using the cuckoo search algorithm. This process involves the construction of 20 decision tree classifiers, each utilising a different set of hyperparameters. The weights assigned to each tree in the ensemble are then optimised using the cuckoo search algorithm, with the training and validation set serving as the basis for this optimisation. The outcome of this optimisation process is an optimised-weighted ensemble learning model. This model, which incorporates the contributions of individual decision trees and their respective weights, exhibits enhanced predictive capabilities. Finally, this optimised ensemble learning model is utilised to classify the testing set, enabling accurate predictions on previously unseen data. The main steps of our algorithm are presented in Algorithm 2.

Algorithm 2 Ensemble Learning with Decision Trees and Cuckoo Search

```
Input: Training dataset (X_train, y_train), validation dataset (X_val, y_val), number of decision trees n_trees
Initialize an empty list to store the predictions from each decision tree:
predicted probs = []
Randomly initialize the weights for the ensemble:
weights = random initialization of weights
for i = 1 to n_trees do
    Randomly select a subset of the training set:
    subset indices = randomly select subset size indices from X_train 9: Randomly select
hyperparameters for the decision tree:
    hyperparameters = randomly select hyperparameters for the decision tree model
    Train a decision tree on the selected subset with the selected hyperparameters.
    DT= train DT (X_train[subset indices], y_train[subset indices], hyperpa- rameters)
end for

Perform weighted optimization using the Cuckoo Search algorithm based on the validation set:
weights = cuckoo_search (X_val, y_val, n_trees, weights)
return weights
```

VI. RESULTS AND ANALYSIS

Extensive experiments were conducted to evaluate the performance of our proposed solution in comparison to several well-established ensemble learning models, namely decision trees (DT), adaptive boosting (AD), random forests (RF), gradient boosting (GB), and extra trees (ET). These comparisons were carried out using a dedicated test set, providing a reliable and unbiased assessment of the models' performance.

The decision tree (DT) algorithm is widely used due to its simplicity and interpretability. However, it may suffer from overfitting, resulting in

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

suboptimal generalization on unseen data. Adaptive boosting (AD) and gradient boosting (GB) are ensemble methods that combine weak learners, such as decision trees, to create a more robust and accurate predictive model. Random forest (RF) is another ensemble technique that leverages multiple decision trees by aggregating their predictions. Extra trees (ET) further enhance the random forest approach by introducing additional randomness during the tree-building process. While these ensemble learning models have proven effective in various applications, they also possess certain limitations. One notable drawback is their lack of consideration for the specific contribution and importance of each base learner (e.g., decision tree). This can lead to suboptimal utilisation of the diverse strengths and weaknesses exhibited by different.
base learners. Additionally, these models may not fully leverage the potential of feature combinations and the optimisation of weights assigned to each base learner.

In contrast, our proposed solution incorporates a weighted ensemble learning approach based on decision tree classifiers. By assigning optimal weights to each decision tree, our model can effectively combine their predictions, taking into account their performance and relevance. This approach enables a more refined and precise integration of the base learners' contributions, leading to improved accuracy and robustness in detecting fraudulent transactions. The comparative evaluation conducted on the dedicated test set provides valuable insights into the strengths and weaknesses of different ensemble learning models, including their performance in credit card fraud detection scenarios. By demonstrating the superiority of our proposed weighted ensemble learning model based on decision trees, we highlight the advantages of incorporating specific optimisation techniques to enhance the predictive capabilities of the ensemble.

Table-III: Performance Evaluation Metrics on the Validation Set: Findings from the Experiment

Model	ACC	PRE	REC	F
This work	0.8812	0.8631	0.9060	0.8840
DT	0.8511	0.8506	0.8517	0.8512
RF	0.8685	0.8604	0.8796	0.8699
AD	0.8777	0.8807	0.8735	0.8771
GB	0.8814	0.8680	0.8995	0.8835
ET	0.8632	0.8584	0.8698	0.8640

Table 3 presents the evaluation metrics for different models on the validation set, including" This work" (which refers to the proposed solution) as well as Decision Tree (DT), Random Forest (RF), AdaBoost (AD), and Extra Trees (ET). The evaluation metrics considered are accuracy, precision, recall, and F1-score.

Starting with the proposed solution, it achieved an accuracy of 0.8812, indicating that around 88.12% of the transactions were classified correctly. The precision score of 0.9060 suggests that approximately 90.60% of the transactions classified as fraudulent were true positives. The recall score of 0.8840 indicates that around 88.40% of the fraudulent transactions were detected, and the F1-score of 0.8631 represents a good balance between precision and recall. Moving on to the individual models, the Decision Tree (DT) model achieved lower scores across all evaluation metrics compared to the proposed solution.

Its performance was moderate, with accuracy, precision, recall, and F1-score ranging around 0.85.

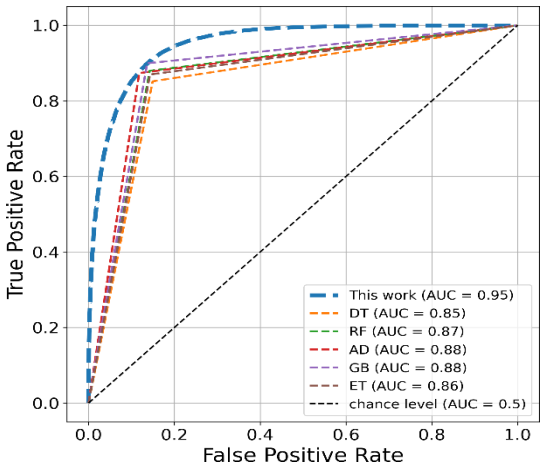
The Random Forest (RF) model outperformed the Decision Tree model, demonstrating higher scores for accuracy, precision, recall, and F1-score. The RF model's scores were relatively better, ranging from 0.87 to 0.88 for most metrics.

The AdaBoost (AD) model demonstrated competitive performance, with consistent scores across all evaluation metrics. Its accuracy, precision, recall, and F1-score were in the range of 0.87-0.88, comparable to or slightly better than those of the RF model.

The Gradient Boosting (GB) model performed exceptionally well, achieving the highest scores for accuracy, recall, and F1-score among all models. It exhibited scores of around 0.88-0.89, indicating its superior performance in correctly classifying fraudulent transactions.

The Extra Trees (ET) model obtained moderate scores, performing slightly better than the Decision Tree model but not as well as the RF or GB models. Its scores were in the range of 0.86-0.87, similar to the RF model.

Overall, the results highlight the effectiveness of the proposed solution in comparison to the individual models. The proposed solution (referred to as" This work") showcased competitive or superior performance, especially in terms of accuracy, precision, recall, and F1-score, when compared to the other models. This suggests that the proposed solution offers an improved approach to detecting fraudulent transactions. However, other factors, such as computational efficiency, interpretability, and scalability, should also be considered when evaluating the feasibility and suitability of the proposed solution for real-world implementation.

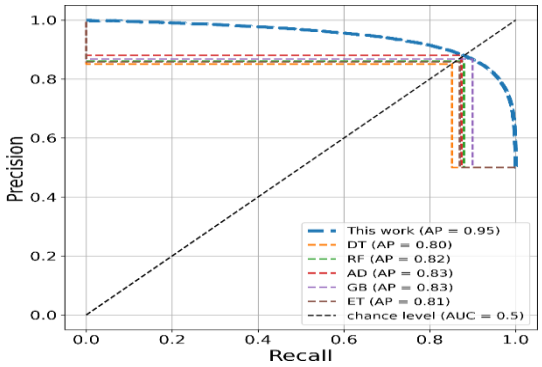


[Fig.2: The Comparison of Ensemble Learning Models on the Validation Set: Evaluating Performance through ROC-AUC Curve Analysis]

Figure 2 displays the ROC curve of our implemented models, with the blue line representing

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

our model's exceptional out-performance compared to other ensemble learning models. Our proposed solution achieves the highest AUC score of 0.95 in the experiments, positioning its curve in the top right corner of the graph. This indicates the superior classification accuracy and predictive power of our model, as it effectively distinguishes between positive and negative instances.



[Fig.3: Comparative Analysis of Ensemble Learning Models on the Validation Set: Examining Performance Through Precision-Recall Curve]

Figure 3 presents the Precision-Recall curve, where the blue line represents our model's superior performance compared to other ensemble learning models. Positioned in the top corner, our proposed solution demonstrates exceptional precision and recall rates. This indicates its ability to accurately identify positive instances while minimising false positives and false negatives. Additionally, our model achieves the highest AUC score of 0.95, further validating its outstanding classification accuracy. Table 4 presents the evaluation metrics for different models on the testing set, including" This work" (which refers to the proposed solution) as well as Decision Tree (DT), Random Forest (RF), AdaBoost (AD), and Extra Trees (ET). The evaluation metrics considered are accuracy, precision, recall, and F1-score.

Starting with the proposed solution, it achieved an accuracy of 0.8818, indicating that approximately 88.18% of the transactions in the testing set were classified correctly. The precision score of 0.9046 suggests that around 90.46%

Table-IV: Performance Evaluation Metrics on the Testing Set: Findings from the Experiment

Model	ACC	PRE	REC	F
This work	0.8818	0.8655	0.9046	0.8846
DT	0.8516	0.8528	0.8505	0.8516
RF	0.8686	0.8627	0.8774	0.8700
AD	0.8783	0.8834	0.8720	0.8777
GB	0.8820	0.8704	0.8983	0.8841
ET	0.8637	0.8599	0.8696	0.8647

The transactions classified as fraudulent where the true positives were. The recall score of 0.8846 indicates that approximately 88.46% of the fraudulent transactions were detected, and the F1-score of 0.8655 represents a good balance between precision and recall.

Comparing the proposed solution to the individual models, we observe that the Decision Tree (DT) model achieves similar or slightly lower scores across all evaluation metrics. Its performance was moderate, with accuracy, precision, recall, and F1-score ranging around 0.85.

The Random Forest (RF) model outperformed the Decision Tree model, achieving higher scores for accuracy, precision, recall, and F1-score. The RF model's scores ranged from 0.87 to 0.88 for most metrics, indicating improved performance compared to the DT model.

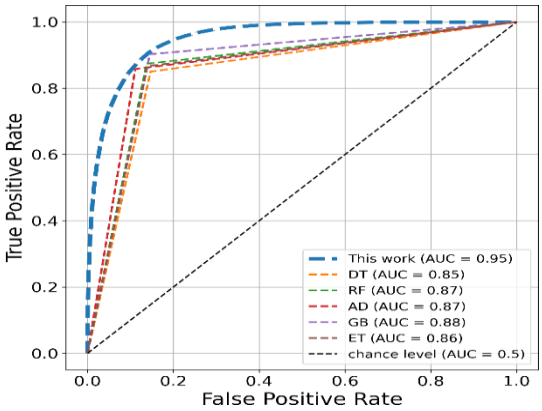
The AdaBoost (AD) model demonstrated competitive performance, with scores comparable to or slightly better than those of the RF model. Its accuracy, precision, recall, and F1-score were in the range of 0.87-0.88, showcasing its effectiveness in fraud detection.

The Gradient Boosting (GB) model performed exceptionally well, achieving the highest scores for accuracy, recall, and F1-score among all models. Its scores were in the range of 0.88-0.89, demonstrating its superior performance in correctly classifying fraudulent transactions.

The Extra Trees (ET) model obtained similar scores to the Decision Tree model, with moderate performance. Its scores ranged from 0.86 to 0.87, indicating performance comparable to that of the DT model.

Overall, the proposed solution (referred to as" This work") showcased competitive or superior performance, particularly in terms of accuracy, precision, recall, and F1-score, when compared to the individual models. This suggests that the proposed solution offers an improved approach to detecting fraudulent transactions. However, as mentioned earlier, other factors, such as computational efficiency, interpretability, and scalability, should also be considered when evaluating the feasibility and suitability of the proposed solution for real-world implementation.

The findings from the testing set analysis reveal the superior performance of our implemented models, as indicated by both the ROC and Precision-Recall curves. The ROC curve illustrates the trade-off between sensitivity and specificity, while the Precision-Recall curve showcases the trade-off between precision and recall.



[Fig.4: The Comparison of Ensemble Learning Models on the Testing Set: Evaluating- Performance through ROC-AUC Curve Analysis]

In terms of the ROC curve, our model's performance surpasses that

Retrieval Number: 100.1/ijitee.D105314040325
DOI: 10.35940/ijitee.D1053.12050423
Journal Website: www.ijitee.org

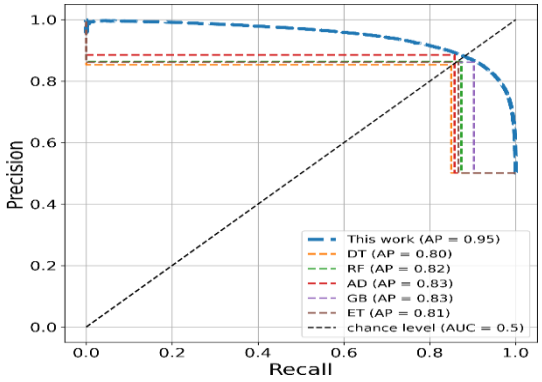
Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

of other ensemble learning models, as indicated by the blue line, which depicts its outperformance. With an impressive AUC score of 0.95, our model achieves the highest score among all the experiments, demonstrating its exceptional cost-benefit analysis. Assessing the economic impact of implementing the proposed solution in real-world credit card fraud prevention systems would provide valuable insights, effectively distinguishing between the two classes while minimising misclassifications.

Similarly, the Precision-Recall curve 5 confirms the superior performance of our model. Once again, the blue line representing our model's curve is situated in the top corner, indicating its exceptional precision and recall rates. Our model demonstrates a high level of precision in accurately identifying positive instances while maintaining a strong recall rate, capturing a substantial proportion of actual positive instances. This highlights its ability to provide accurate and reliable predictions with minimal false positives and false negatives.

VII. CONCLUSION AND FUTURE WORK

In conclusion, this paper introduces a novel ensemble learning approach for preventing and detecting unauthorised credit card transactions, addressing the critical issue of credit card fraud and its associated financial losses. The proposed approach combines a weighted average technique.



[Fig.5: Comparative Analysis of Ensemble Learning Models on the Testing Set: Examining Performance through Precision-Recall Curve]

With a set of decision trees, where optimal weights are determined using the Cuckoo Search algorithm. By leveraging the ensemble model's probability and associated weight, enhanced predictions are achieved. The evaluation process involves comparing ensemble learning-based decision trees, including Random Forest, AdaBoost, Gradient Boosting Tree Classifier, and Extra Trees, on a randomly undersampled dataset of credit card transactions. Extensive experiments on real data demonstrate the superior predictive capabilities of the ensemble learning approach, leading to improved accuracy, precision, recall scores, and F-measure. For future work, several avenues can be explored. Firstly, the proposed ensemble learning approach can be further evaluated and validated on larger and more diverse credit card transaction datasets to ensure its generalizability. Additionally, incorporating additional features or exploring alternative feature engineering techniques may enhance the model's performance. It would also be valuable to compare the proposed approach with other state-of-the-art fraud detection methods, such as anomaly detection algorithms or deep learning models, to determine the most effective strategy for preventing credit card fraud. Moreover, integrating real-time data streaming and adaptive learning techniques could enhance the model's responsiveness to emerging fraud patterns and enable proactive fraud detection. Finally, conducting a cost-benefit analysis to assess the economic impact of implementing the proposed solution in real-world credit card fraud prevention systems would provide valuable insights into its practical viability. Overall, these potential future research Directions aim to refine further and optimise the ensemble learning approach for more effective credit card fraud prevention and detection.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted with objectivity and without any external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical approval or consent to participate with supporting documentation.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Esenogho, Ebenezer, et al." A neural network ensemble with feature engineering for improved credit card fraud detection." IEEE Access 10 (2022): 16400-16407. http://dx.doi.org/10.1109/ACCESS.2022.3148298
2. Alfaiz, Noor Saleh, and Suliman Mohamed Fati." Enhanced credit card fraud detection model using machine learning." Electronics 11.4 (2022): 662. https://doi.org/10.3390/electronics11040662
3. Xie, Yu, et al." Learning transactional behavioral representations for credit card fraud detection." IEEE Transactions on Neural Networks and Learning Systems (2022). https://doi.org/10.1109/tnnls.2022.3208967
4. Tayebi, Mohammed, and Said El Kafhali." Performance analysis of meta-heuristics-based hyperparameters optimisation for fraud transactions detection." Evolutionary Intelligence (2022): 1-19. http://dx.doi.org/10.1007/s12065-022-00764-5
5. El Kafhali, Said, and Mohammed Tayebi." Generative Adversarial Neural Networks based Oversampling Technique for Imbalanced Credit Card Dataset." 2022 6th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI). IEEE, 2022.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

http://dx.doi.org/10.1109/SLAAI-ICAI56923.2022.10002630
6. El Kafhali, Said, and Mohammed Tayebi. "XGBoost based solutions for detecting fraudulent credit card transactions." 2022 International Conference on Advanced Creative Networks and Intelligent Systems (ICACNIS). IEEE, 2022. https://doi.org/10.1109/ICACNIS57039.2022.10054965
7. rabhakaran, N., and R. Nedunchelian. "Oppositional Cat Swarm Optimization-Based Feature Selection Approach for Credit Card Fraud Detection." Computational Intelligence and Neuroscience 2023 (2023). https://doi.org/10.1155/2023/2693022
8. Afriyie, Jonathan Kwaku, et al. "A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions." Decision Analytics Journal 6 (2023): 100163. http://dx.doi.org/10.1016/j.dajour.2023.100163
9. Tayebi, Mohammed, and Said El Kafhali. "Credit Card Fraud Detection Based on Hyperparameters Optimization Using the Differential Evolu- tion." International Journal of Information Security and Privacy (IJISP) 16.1 (2022): 1-21. http://dx.doi.org/10.4018/IJISP.314156
10. Baabdullah, Tahani, et al. "An Ensemble-Based Machine Learning for Predicting Fraud of Credit Card Transactions." Intelligent Computing: Proceedings of the 2022 Computing Conference, Volume 2. Cham: Springer International Publishing, 2022. https://par.nsf.gov/servlets/purl/10341852
11. Ahmad, Hadeel, et al. "Class balancing framework for credit card fraud detection based on clustering and similarity-based selection (SBS)." International Journal of Information Technology 15.1 (2023): 325-333. https://doi.org/10.1007/s41870-022-00987-w
12. Shahana, T., Vilvanathan Lavanya, and Aamir Rashid Bhat. "State of the art in financial statement fraud detection: A systematic review." Technological Forecasting and Social Change 192 (2023): 122527. https://doi.org/10.1016/j.techfore.2023.122527
13. Fu, Kang, et al. "Credit card fraud detection using convolutional neural networks." Neural Information Processing: 23rd International Conference, ICONIP 2016, Kyoto, Japan, October 16–21, 2016, Proceedings, Part III 23. Springer International Publishing, 2016. http://dx.doi.org/10.1007/978-3-319-46675-0_53
14. Li, Zhenchuan, Guanjun Liu, and Changjun Jiang. "Deep representation learning with complete centre loss for credit card fraud detection." IEEE Transactions on Computational Social Systems 7.2 (2020): 569-579. https://doi.org/10.1109/TCSS.2020.2970805
15. Zhang, Xinwei, et al. "HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture." Information Sciences 557 (2021): 302-316. https://doi.org/10.1016/j.ins.2019.05.023
16. Cao, Ruihao, et al. "Two-level attention model of representation learning for fraud detection." IEEE Transactions on Computational Social Systems 8.6 (2021): 1291-1301. http://dx.doi.org/10.1109/TCSS.2021.3074175
17. Dorronsoro, Jose R., et al. "Neural fraud detection in credit card operations." IEEE Transactions on Neural Networks, 8(4), 1997, pp. 827-834. https://doi.org/10.1109/72.595879
18. Chou, Tsung-Nan. "A novel prediction model for credit card risk management." Second International Conference on Innovative Computing, Information and Control (ICICIC 2007). IEEE, 2007. https://doi.org/10.1109/ICICIC.2007.68
19. Guo, Tao, and Gui-Yang Li. "Neural data mining for credit card fraud detection." 2008 International Conference on Machine Learning and Cybernetics. Vol. 7. IEEE, 2008. https://doi.org/10.1109/ICMLC.2008.4621035
20. Lee, Chae Chang. "A data mining approach using transaction patterns for card fraud detection." arXiv preprint arXiv:1306.5547 (2013). https://doi.org/10.48550/arXiv.1306.5547
21. Zakaryazad, Ashkan, and Ekrem Duman. "A profit-driven Artificial Neural Network (ANN) with applications to fraud detection and direct marketing." Neurocomputing 175 (2016): 121-131. https://doi.org/10.1016/j.neucom.2015.10.042
22. Stolfo, Salvatore, et al. "Credit card fraud detection using meta-learning: Issues and initial results." AAAI-97 Workshop on Fraud Detection and Risk Management. 1997. https://www.researchgate.net/publication/2282588_Credit_Card_Fraud_Detection_Using_Meta-Learning_Issues_and_Initial_Results
23. Dal Pozzolo, Andrea, et al. "Credit card fraud detection: a realistic modelling and a novel learning strategy." IEEE transactions on neural networks and learning systems 29.8 (2017): 3784-3797. https://doi.org/10.1109/TNNLS.2017.2736643
24. Brause, Rüdiger, T. Langsdorf, and Michael Hepp. "Neural data mining for credit card fraud detection." Proceedings 11th International Conference on Tools with Artificial Intelligence. IEEE, 1999. http://dx.doi.org/10.1109/ICMLC.2008.4621035
25. Minegishi, Tatsuya, and Ayahiko Niimi. "Proposal of credit card fraud -ulent use detection by online-type decision tree construction and verification of generality." International Journal for Information Security Research (IJISR) 1.4 (2011): 229-235. https://infonomics-society.org/wp-content/uploads/ijisr/volume-3-2013/Proposal-of-Credit-Card-Fraudulent-Use-Detection-by-Online-type-Decision-Tree-Construction-and-Verification-of-Generality.pdf
26. Minegishi, Tatsuya, et al. "Extension of decision tree algorithm for stream data mining using real data." Proceedings: Fifth International Workshop on Computational Intelligence Applications. Vol. 2009. No. 1. IEEE SMC Hiroshima Chapter, 2009. https://www.researchgate.net/publication/41023106_Extension_of_Decision_Tree_Algorithm_for_Stream_Data_Mining_Using_Real_Data
27. Gadi, Manoel Fernando Alonso, Xidi Wang, and Alair Pereira do Lago. "Credit Card Fraud Detection with Artificial Immune System." ICARIS. Vol. 8. 2008. https://doi.org/10.1007/978-3-540-85072-4_11
28. Bhusari, V., and S. Patil. "Application of hidden Markov model in credit card fraud detection." International journal of distributed and parallel systems 2.6 (2011): 203. http://dx.doi.org/10.5121/ijdps.2011.2618
29. Maes, Sam, et al. "Credit card fraud detection using Bayesian and neural networks." Proceedings of the 1st International Naiso Congress on Neuro-Fuzzy Technologies. Vol. 261. 2002. https://www.researchgate.net/publication/2524707_Credit_Card_Fraud_Detection_Using_Bayesian_and_Neural_Networks
30. Bahnsen, Alejandro Correa, et al. "Cost sensitive credit card fraud detection using Bayes minimum risk." 2013 12th International Conference on onmachine learning and Applications. Vol. 1. IEEE, 2013. https://albahnsen.github.io/files/Cost%20Sensitive%20Credit%20Card%20Fraud%20Detection%20using%20Bayes%20Minimum%20Risk%20-%20Publish.pdf
31. Dheepa, V., and R. Dhanapal. "Behavior based credit card fraud detection using support vector machines." ICTACT Journal on Soft Computing 2.4 (2012): 391-397. https://pdfs.semanticscholar.org/43ee/79be92dbb853a811874a7ac193ad9c803cea.pdf
32. Reiss Jr, Albert J. "Co-offending and criminal careers." Crime and justice 10 (1988): 117-170. https://www.jstor.org/stable/1147404
33. Chen, Cen, et al. "InfDetect: A large-scale graph-based fraud detection system for E-commerce insurance." 2019 IEEE International Conference on Big Data (Big Data). IEEE, 2019. https://doi.org/10.48550/arXiv.2003.02833
34. Pourhabibi, Tahereh, et al. "Fraud detection: A systematic literature review of graph-based anomaly detection approaches." Decision Support Systems 133 (2020): 113303. https://doi.org/10.1016/j.dss.2020.113303
35. Newman, Mark. Networks. Oxford University Press, 2018. https://doi.org/10.1093/acprof:oso/9780199206650.001.0001
36. Molloy, Ian, et al. "Graph analytics for real-time scoring of cross-channel transactional fraud." Financial Cryptography and Data Security: 20th International Conference, FC 2016, Christ Church, Barbados, February 22–26, 2016, Revised Selected Papers 20. Springer Berlin Heidelberg, 2017. https://fc16.ifca.ai/preproceedings/02_Molloy.pdf
37. Shehnepoor, Saeedreza, et al. "NetSpam: A network-based spam detection framework for reviews in online social media." IEEE Transactions on Information Forensics and Security 12.7 (2017): 1585-1595. http://dx.doi.org/10.1109/TIFS.2017.2675361
38. Bangcharoensap, Phiradet, et al. " Two-step graph-based semi-supervised learning for online auction fraud detection." Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part III 15. Springer International Publishing, 2015. https://doi.org/10.1007/978-3-319-23461-8_11
39. Li, Yuan, Yiheng Sun, and Noshir Contractor. "Graph mining assisted semi-supervised learning for fraudulent cash-out detection." Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

and Mining 2017. 2017. https://ieeexplore.ieee.org/document/9069093
40. Mar´ıa, et al. ”Social network analytics for supervised fraud detection in insurance.” Risk Analysis 42.8 (2022): 1872-1890. https://doi.org/10.1111/risa.13693
41. Van Vlasselaer, V´eronique, et al. ”APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions.” Decision Support Systems 75 (2015): 38-48. http://dx.doi.org/10.1016/j.dss.2015.04.013
42. Van Vlasselaer, V´eronique, et al. ”Gotcha! network-based fraud detection for social security fraud.” Management Science 63.9 (2017): 3090-3110. https://doi.org/10.1287/mnsc.2016.2489
43. Tiwari, Pooja, et al. ”Credit card fraud detection using machine learning: a study.” arXiv preprint arXiv:2108.10005 (2021). https://doi.org/10.48550/arXiv.2108.10005
44. Lebichot, Bertrand, et al. ”A graph-based, semi-supervised, credit card fraud detection system.” Complex Networks Their Applications V: Proceedings of the 5th International Workshop on Complex Networks and their Applications (COMPLEX NETWORKS 2016). Springer Interna- tional Publishing, 2017. http://dx.doi.org/10.1007/978-3-319-50901-3_57
45. Cuong-Le, Thanh, et al. ”A novel version of Cuckoo search algorithm for solving optimization problems.” Expert Systems with Applications 186 (2021): 115669. https://doi.org/10.1016/j.eswa.2021.115669
46. Yang, Xin-She, and Suash Deb. ”Cuckoo search via L´evy flights.” 2009 World Congress on Nature-Inspired Biologically Inspired Computing (NaBIC). IEEE, 2009. https://doi.org/10.1109/NABIC.2009.5393690
47. Jamil, Momin, and Hans-Jürgen Zepernick. ”L´evy flights and global opti- mization.” Swarm intelligence and bio-inspired computation. Elsevier, 2013. 49-72. https://doi.org/10.1016/B978-0-12-405163-8.00003-X
48. Ling, Ying, Yongquan Zhou, and Qifang Luo. ”L´evy flight trajectory- based whale optimization algorithm for global optimization.” IEEE access 5 (2017): 6168-6186. https://doi.org/10.1109/ACCESS.2017.2695498
49. “Index of /datasets.” [Online]. packages.revolutionanalytics.com. (accessed May 16, 2022). http://packages.revolutionanalytics.com/datasets/

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)/ journal and/or the editor(s). The Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP) and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

© Copyright: All rights reserved.