future internet

MDPI

Article

Improved Classification of Blockchain Transactions Using Feature Engineering and Ensemble Learning

Chandrashekar Jatoth [1,†], Rishabh Jain [2,†], Ugo Fiore [3,*,†] and Subrahmanyam Chatharasupalli [4,†]

[1] Department of Information Technology, National Institute of Technology Raipur, Raipur 492010, India; chandrashekar.jatoth@gmail.com
[2] Department of Computer Science & Engineering, National Institute of Technology Hamirpur, Hamirpur 177005, India; jainrishabh2397@gmail.com
[3] Department of Management and Quantitative Studies, Parthenope University, 80132 Napoli, Italy
[4] Union Public Service Commission, New Delhi 110069, India; subrahmanyamch1981@gmail.com
* Correspondence: ugo.fiore@uniparthenope.it
† These authors contributed equally to this work.

Abstract: Although the blockchain technology is gaining a widespread adoption across multiple sectors, its most popular application is in cryptocurrency. The decentralized and anonymous nature of transactions in a cryptocurrency blockchain has attracted a multitude of participants, and now significant amounts of money are being exchanged by the day. This raises the need of analyzing the blockchain to discover information related to the nature of participants in transactions. This study focuses on the identification for risky and non-risky blocks in a blockchain. In this paper, the proposed approach is to use ensemble learning with or without feature selection using correlation-based feature selection. Ensemble learning yielded good results in the experiments, but class-wise analysis reveals that ensemble learning with feature selection improves even further. After training Machine Learning classifiers on the dataset, we observe an improvement in accuracy of 2–3% and in F-score of 7–8%.

Keywords: machine learning; artificial intelligence; ensemble learning; blockchain; performance metrics

Citation: Jatoth, C.; Jain, R.; Fiore, U.; Chatharasupalli, S. Improved Classification of Blockchain Transactions Using Feature Engineering and Ensemble Learning. *Future Internet* **2022**, *14*, 16. https://doi.org/10.3390/fi14010016

Academic Editor: Maumita Bhattacharya

Received: 29 November 2021
Accepted: 27 December 2021
Published: 28 December 2021

1. Introduction

The Blockchain is a distributed database, also known as a digital ledger. Blockchains are designed to be decentralized, meaning there is no centralized authority (e.g., a government, bank, or corporation) that can control it. Due to various factors, such as high compatibility with financial systems and the ability to support smart contracts [1], blockchain technology in banking and financial services is expected to expand rapidly around the world. The banking and financial services industry has recognized the value of blockchain technology, which helps customers in safe transactions. All the data on a blockchain are recorded accurately and can serve as a history that is available for all applications [2]. A blockchain is a constantly developing computerized record in pieces known as blocks which are connected and verified utilizing cryptographic hash functions. Data blocks are stored in a direct chain and each block in the chain contains information (for example, transactions in bitcoins), is time-stamped and cryptographically hashed. Blocks in the blockchain comprise block headers, block identifiers and Merkle trees (referring to the assembly of transactions). Blockchain is a constantly evolving technology. Due to its innovative characteristics, fresh applications are usually created incessantly over its framework. Blockchains are, in fact, an immutable collection of records that are cryptographically linked for auditing purposes [3]. It is similar to an accounting ledger system, but there is no single entity responsible for verifying the records. As illustrated in Figure 1, a blockchain is established by connecting valid blocks: the previous block's hash is contained in the current block. As a result, blockchains are identifiable and impervious to modification [4,5]. Older blocks cannot be altered, as their hash would change if they were altered in any

way. The main focus on linking hashes in subsequent blocks is necessary to re-validate the network of the blockchain. Each person inside the network has a copy of the blockchain; as a result, any alterations can be cross-verified by other users [6].

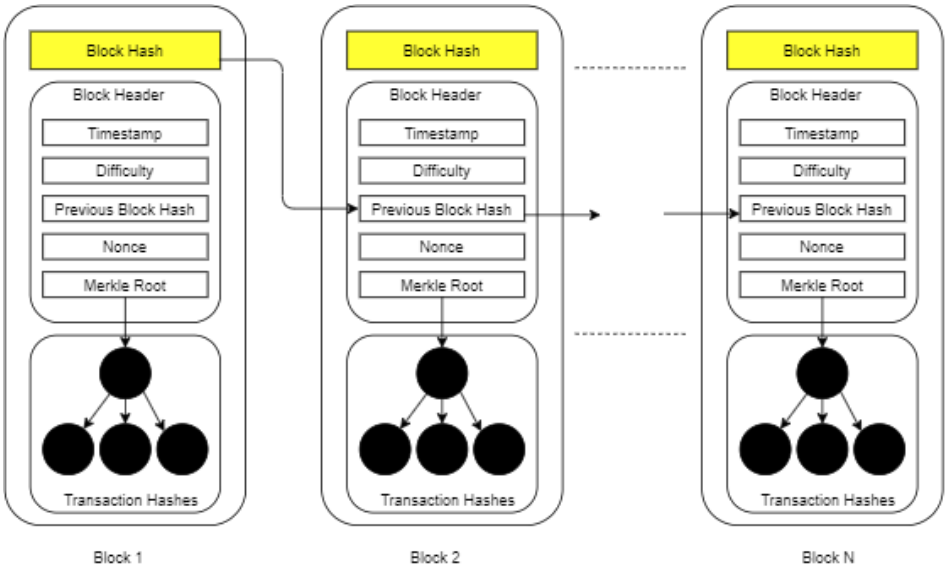

Figure 1. Structure of Blockchain.

These replicas of blockchain are revised whenever a new block is added. The new block is then visible to everyone, based on the administrator-assigned permissions [7,8].

As transactions are written out on the blockchain, the entire transaction history is shared with every peer in the network. With blockchain, participants can cooperate legitimately and can make transactions over the web without the need for a trusted third party. Blockchain is extensively used in the Internet of Things [9,10] and it also is at the basis of trust mechanism that can detect internal attacks [11]. Many researchers have proposed methods to improve security using machine learning algorithms [3,12–16]. However, said models did not address the identification of risky and non-risky blocks in the blockchain that may effect the network after any transactions.

To address the transaction classification problem, this study has concentrated on ensemble learning coupled with feature selection. This paper's contributions are outlined below:

- An extensive literature review of Blockchain blocks analysis approaches with or without machine learning;
- Feature extraction during preprocessing;
- Improved feature selection;
- Application of ensemble base classifier in a stack-based and boosting approach.

The performance of the proposed approach performance is compared to current work by different researchers using assessment measures, such as accuracy, precision, recall, and F-measure.

The remaining portion of the paper is structured as follows: Section 2 refers to previously published writing that is relevant to our methodology. After a brief recapitulation (Section 3) of the essential intuitions about the techniques used, Section 4 describes the approach used in this paper, comprising features and the suggested feature selection strategy. The following Section 5 explains the details of the experiments and their outcomes. Finally, Section 6 summarizes the proposed strategy and outlines suggestions for further study and experimentation.

2. Related Work

In this section, we provide a short outline of past work on AI (ML) procedures and blockchain.

Abu-Elezz et al. [7] investigated and categorized the advantages and drawbacks of executing blockchain innovation in a well-being administration. Salah et al. [12] provided a comprehensive description of blockchain technology powered by artificial intelligence. The experts examined the tabulate, writing, and condensing the emerging blockchain applications, platforms, and conventions with an emphasis on the AI sector. Tanwar et al. [13] proposed a clear study on machine learning adoption to make BT-based smart applications more adaptable to attacks. Numerous traditional machine learning methods, such as clustering, bagging, SVM as well as deep learning algorithms including LSTM and CNN, can then be used to analyze attacks on a blockchain-based system. Sgantzos and Grigg [17] presented a model viewpoint for how an AI becomes a troubling mechanical paragon by utilizing blockchain technology, which originates in the universe of deep learning. Since a long time ago, data scientists have continued to maintain the essence of a dataset for AI use by an AI entity. However, researchers have not yet defined how they intend to use AI in blockchain.

Hassani et al. [14] conducted the most comprehensive survey of the impact of blockchain technology on the banking industry to date by highlighting the opportunities and challenges from a banker's perspective. Mermer et al. [4] provided an overview of blockchain, a relative innovation that has the potential to change the way society communicates and trades. Zheng et al. [3] conducted an in-depth analysis of blockchain technology. This paper defines blockchains, incorporates common blockchain consensus algorithms, reviews blockchain applications, and describes technical hurdles and recent advancements in overcoming them. Marwala and Xing [18] provided an overview of how AI can be used to produce bug-free smart contracts to reach the aims of blockchain 2.0, emphasizing that blockchain integration can be added or enhanced through numerous AI techniques. However, the author stated that the combination of blockchain and AI is scheduled to open up a plethora of possibilities. Gatteschi et al. [19] showed that as AI advances from science fiction to become the frontier of world-changing technologies, there seems to be an urgent need for deliberate advancement and application of AI in order to see its true impact in the on-coming advent of new frameworks, specifically Industry 4.0. Puthal et al. [8] provided a study of blockchain technology for the non-vulnerable and precise recognition of security crosswise over-dispersed gatherings. It guarantees an innocuous dispersed structure that will encourage distribution, trading, and the introduction of data from both internal and external customers.

As a central theme of their paper, Conley [20] were formulating an efficient token that required consideration of certain aspects of money theory, financial-economic arrangements, and game theory. Samaniego and Deters [21] used the authorization-based blockchain protocol to deliver numerous smart technology things. Michalski et al. [16] proposed a set of features that quantify the behavior of nodes in the network by using supervised machine learning algorithms to find out whether the character of nodes can be revealed based on these features. Seebacher and Schüritz [22] were able to assess the impact of an inventory network, the key ordered the written work of an audit of peer-reviewed articles. Hall and Pesenti [23] showed that the task of "AI readiness" may extend the current pockets of technology development and expertise throughout the public sector. Ogiela and Ogiela [24] outlined and studied cognitive schemes for the security of shared data in the cloud, methods that can greatly benefit from the adoption of a blockchain to guarantee an immutable history. Wu et al. [15] centered around the discovery of the addresses associated with mixing services, which is a significant assignment for anti-money laundering in Bitcoin. However, the above methods did not address the identification of risky and non-risky blocks in the blockchain that may effect the network after any transactions. The main objective of this paper is to classify the risky and non-risky block in a blockchain network using ensemble learning by training different classifiers on datasets.

3. Preliminaries

Our primary work is mainly focused on classification of risky and non-risky blocks in the blockchain network using ensemble learning by training different classifiers on a dataset.

Ensemble learning is an approach for solving a difficult problem that involves deliberately merging multiple machine learning models into a single predictive model. Ensemble methods, in general, are used to increase a model's overall performance accuracy by combining several separate models, also known as base learners, to predict the outcomes, in spite of using a single model. We utilize this strategy in our daily lives as well—for example, we seek the advice of several experts before making decisions, we read various product evaluations before purchasing a product, and a panel of judges confer among themselves to determine a winner. Ensemble learning is often divided into four categories:

1. Boosting;
2. Stacking;
3. Bagging;
4. Cascading.

We will discuss the first two techniques that help us in generating our classification model.

3.1. Ensemble Boosting

Boosting is a technique for transforming weak base learners into strong ones [25]. Boosting includes iteratively training the poor learners, with each model attempting to address the former model's shortcomings. This is accomplished by training a weak model on the entire training dataset, then developing a second model to correct the first model's faults. Then, a third model is created that attempts to fix the faults produced by the previous two models, and so on. Iteratively, models are added until the last model has corrected all of the previous models' flaws.

When the models are introduced at each stage, some weights are assigned to the model based on the preceding model's accuracy. Following the addition of a weak classifier, the weights are re-adjusted. The points that were wrongly categorized are given larger weights, while the points that were correctly categorized are given lower weights. As a result of this method, the next classifier will concentrate on the prior model's errors. By adopting a high-bias, low-variance model and significantly decreasing the bias, boosting minimizes the generalization error. Boosting allows us to deal with classification and regression models simultaneously. Boosting involves fitting a weak learner iteratively, aggregating it to the ensemble model, and updating the training dataset to better account for the current ensemble model's strengths and weaknesses when fitting the next base model.

3.2. Ensemble Stacking

Stacking is an ensemble learning strategy for combining the expectations of various classification models into a single meta-classifier [25]. Individual models are trained autonomously on the whole training dataset and adjusted to improve accuracy. The meta-classifier, which takes care of either the class labels anticipated by the fundamental models or the expected probabilities for each class label, is the last model. From that point forward, the meta-classifier learns to utilize the yields of the fundamental models. Stacking is the way toward preparing another model dependent on the forecasts delivered by earlier models.

This procedure is carried out in a sequential manner. This means that numerous models are trained and fine-tuned at stage 1. In step 2, the expected probabilities from each model from stage 1 are transmitted into all of the models. Stage-2 models are fine-tuned, and relevant outputs are passed on to stage-3 models, and so on. Depending on how many layers of stacking one wants to employ, this technique may be repeated several times. The last layer comprises a solitary model that joins the outputs of previous levels to create the final output. This single model at the end of a stacking pipeline is known as the

meta-classifier. Stacking is a method of instructing a meta-model to produce outputs based on the outcomes of some base layer flexible learners.

4. Proposed Work

The flowchart in Figure 2 presents the structure of the proposed approach which helps in achieving the objectives and improve the automated classification of blocks in impactful/risky and non-impactful/non-risky blocks. The flowchart involves the following steps:

1. The first step involves the extraction of features from the dataset and labeling.
2. Feature extraction and selection: initially, we analyze the document's binary presence or absence of a feature to build an feature vector. If it appears in a document, it will receive a score of 1, otherwise it will receive a score of 0;
3. The second step applies the machine learning approach by an ensemble of classifiers. Classifier are divided into parametric and non-parametric. Parameters models make explicit assumptions and non-parametric ones do not define strong assumptions.
4. In ensemble learning, we combine k-Nearest Neighbor (KNN) and Bayesian approaches for parametric (parametric approaches in which predefined variables and it is not depended or effected by dataset behavior). We also combine Bayesian networks, decision trees, and logistic Bayesian approach for non-parametric (non-parametric approaches in which we have not predefined variables and it is not depended or effected by dataset behavior).
5. In the third step, learning is done based on the classifier model and it helps in improving the classifier model.
6. In the fourth step, analysis is done on the basis of evaluation parameters precision, recall, F-score and accuracy.

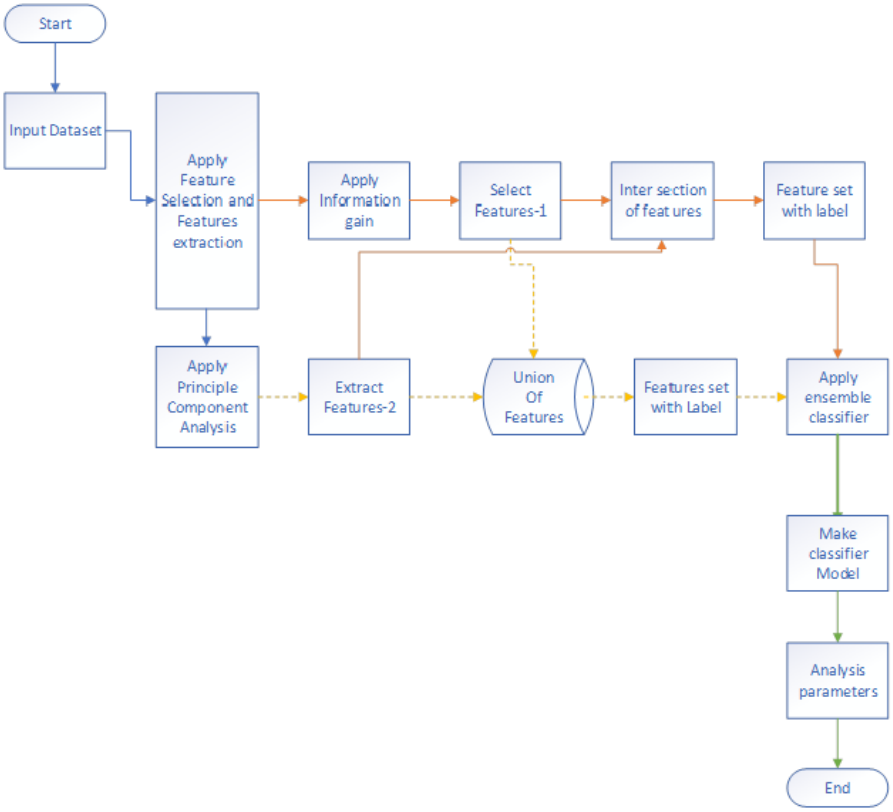

Figure 2. Proposed Flowchart.

4.1. Dataset

We used a dataset from Kaggle. The Elliptic company released this dataset in order to arouse the interest of the academic and crypto communities in developing a more secure

financial system based on cryptocurrency [26]. This dataset is built using the transaction graph from the Bitcoin blockchain. A transaction is represented by a node in the graph which is a directed acyclic graph (DAG), and each edge may be thought of as a flow of Bitcoins from one transaction to the next. Except from edge cases (in which numerous outcomes of a transaction are later spent as inputs with the same transaction), a node's in-degree displays the proportion of input data of a transaction, while its out-degree indicates the set of monies that have been spent, and the nodes without any incoming edges are coinbase transactions. Each node is labeled as "licit" or "illicit" if the related transaction has been made by an entity classified as licit (trades, wallet suppliers, diggers, monetary specialist co-ops, and so on) or illicit (tricks, malware, psychological oppressor associations, ransomware, and so on). The graph consists of 203,769 nodes and 234,355 edges. Class 1 (illicit) is assigned to 2% (4545) of the nodes. A total of 42,019 nodes (21%) are classified as (licit) class 2. The leftover transactions are unlabeled as to whether they are legal or illegal. Each node has 166 different features. Each node has a time step connected with it, which represents the amount of time it took for a transaction to be broadcast to the Bitcoin network. The time steps, which range from 1 to 49, are uniformly spaced with a two-week break between them.

The initial 94 features out of 166 features exchange explicit data, for example, the time steps referenced over, the quantity of sources of input/output, exchange expense, yield volume, and accumulated figures such as normal BTC received (spent) by the information input/output and a normal number of approaching (active) exchanges related with the data input/output. The extra 72 features are obtained by one-skipping backward/forward from the center using exchange data—giving the greatest, least, standard deviation, and connection coefficients of neighbor exchanges for a piece of comparative information data (number of data sources/yields, exchange charge, and so on). So, our main task is to classify these nodes as "licit" or "illicit" by implementing supervised learning models on this dataset which helps in classification of nodes coming from unknown sources and increasing the security of the network. M. Weber et al. [26] developed a model for the analysis of illicit activity over this dataset using ML techniques. We performed this analysis with the ensemble learning to achieve better results.

4.2. Feature Selection

Feature selection is one the three main tenets of practical financial fraud detection [27]. The feature selection method (FSM) is a basic undertaking for improving classification accuracy [28]. In general, FSMs are statistically represented by the feature-class category connection. The feature set has the most influence on the classifier's performance; however, on the off chance that the feature selection measure is effective, the least powerful classifier may likewise give high accuracy by planning. When utilizing a feature selection methodology, the key concept is that the data contain some features that are either redundant or irrelevant, and hence may be deleted without causing significant information loss. Correlation-based feature selection reduces the interdependence between features.

4.3. Feature Extraction

Feature extraction generates new features based on the functions of the original features, whereas feature selection only returns a subset of the features. Feature extraction is the process of decreasing the amount of resources needed to describe a huge amount of data. One of the primary issues in completing complicated data analysis is the large number of variables involved. An analysis with a high number of variables usually necessitates a lot of memory and processing resources.

4.4. Combination of Feature Selection Methods

As each feature selection strategy utilizes particular rules to separate a feature subset [29], diverse feature subsets are applied. We utilized the measurable strategy UNION to choose all features and INTERSECTION to choose just regular features to integrate these

separate feature sub-lists. In our paper, we have considered both UNION and INTERSECTION to obtain all top-ranked including common selected features.

Let F $\{f_1, f_2, \ldots, f_n\}$ be the principal feature set. $F_{SUB}1$ $\{f_{11}, f_{12}, \ldots, f_{IG}\}$ and $F_{SUB}2$ $\{f_{21}, f_{22}, \ldots, f_{PCA}\}$ are two feature subsets selected by Information Gain (IG) and Principal Component Analysis (PCA). All of the features in these subsets must be sorted based on their weight or score.

With UNION, we simply combined entirety of the features from the feature selection technique to obtain a feature sub-list

$$F_{SUB}3 = F_{SUB}1 \cup F_{SUB}2. \tag{1}$$

We have just chosen common features in every one of the three feature subsets to obtain a feature sub-list with INTERSECTION

$$F_{SUB}4 = F_{SUB}1 \cap F_{SUB}2. \tag{2}$$

Following that, we used an updated UNION technique to catch all of the top-ranked characteristics as well as common features. Because the characteristics have effectively been arranged, the most elevated and least scoring features have been assigned to their right positions.

As a result, we tested the UNION and INTERSECTION methods on the top-ranked ($T1$) and lowest ranked ($L1$) results. We used the improved UNION method to apply union to the top $T1\%$ of features and intersection to the remainder $L2\%$

$$F_{SUB}5 = \{T1\%\{F_{SUB}1\} \cup T1\%\{F_{SUB}2\}\} \cup \{L1\%\{F_{SUB}1\} \cap L1\%\{F_{SUB}2\}\}. \tag{3}$$

These merged feature subsets will be used to train supervised classifiers and compare their performance to feature subsets created using independent feature selection methods.

4.5. Proposed Algorithm

The input dataset first goes through feature selection and extraction techniques. Features were already labeled while gone through preprocessing. The dataset is trained over ensemble classifiers for classification of risky and non-risky blocks. At last, we analyze evaluation parameters. For our categorization model, we suggest Algorithm 1.

Algorithm 1 Proposed algorithm for Classification Model

Input: Dataset
Output: Classified Blocks
1. Begin
2. $N \leftarrow No.of Instances$
3. while *N greater than 0* **do**
  Start Pre-process (N_i)
  CFS(N_s)
  $N_s \leftarrow (features, label)$
  End
end
4. $Features \leftarrow \{x_1, x_2, \ldots., x_n\}$
5. $Label \leftarrow \{l_1, l_2, \ldots., l_n\}$
6.$T_s \leftarrow (features, label)$
7.$Train \leftarrow Train(kNN, SVM, DT)$
8. $Test \leftarrow Train(T_s)$
9. Analyze Evaluation Parameters
10. End

4.6. Expected Outcomes

After the implementation of the proposed approach, the following outcomes will be achieved:

- Improve the classification of the risky and non-risky block in an efficient way;
- Improve the automatic process of the financial aspect of the organization;
- Improve the time computation.

5. Results and Analysis

Four performance metrics are used in this paragraph to measure the performance of the method mentioned below.

1. *Accuracy*: Refers to the degree to which a prediction is identical to the real or agreed outcome. It can be computed using Equation (4)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (4)$$

In Equation (4), TP denotes correctly classified positive samples, TN represents correctly classified negative samples, FP denotes incorrectly classified negative samples and FN denotes incorrectly classified positive samples.

2. *Precision*: It specifies the true negatives that can be correctly classified. It can be computed using Equation (5)

$$Precision = \frac{TP}{TP + FP}. \quad (5)$$

3. *Recall*: This metric indicates the proportion of actual positives who are classified as such. Recall is computed using Equation (6)

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

4. *F-measure*: It is the harmonic mean of precision and recall

$$F\text{-}measure = \frac{2TP}{2TP + FP + FN}. \quad (7)$$

5.1. Hyper-Parameters

The variables that determine how a model is structured and trained are referred to as hyper-parameters. Their values should be calibrated accurately because they affect the performance to a large degree. Table 1 lists the values of the hyper-parameters that have been selected, by trial and error, in the experiments.

Table 1. Hyper-Parameters.

Classifier	Name of Hyper-Parameter
SVM	Kernel = rbf
KNN	k = 7
Decision Tree	MaxDepth = 4
Logisitic	Random State = 1
Random Forest	n-estimators = 50, max-depth = 100, random-state = 15
MLP	No. of units (165, 50), Activation = Sigmoid, Optimizer = Adam, Epochs = 10, Layers = 3

5.2. Results

The results for different metrics as discussed above are presented in Table 2. We compared the performance of several classifiers with the proposed ensemble-based approach with or without feature selection. SVM, KNN, logistic, and their ensemble classifiers were used in the experiment, along with a boosting and stacking approach. Different classifiers, such as SVM, KNN, logistic, and decision tree, are used in Ensemble learning, coordinated

using the sampling and the stacking method. In Table 2, CFS indicates the results with feature selection.

It is clear by this analysis that Ensemble learning improves significantly the performance metrics, compared to single classifiers. In Figure 3, we compare the results in graphical form for an easier analysis.

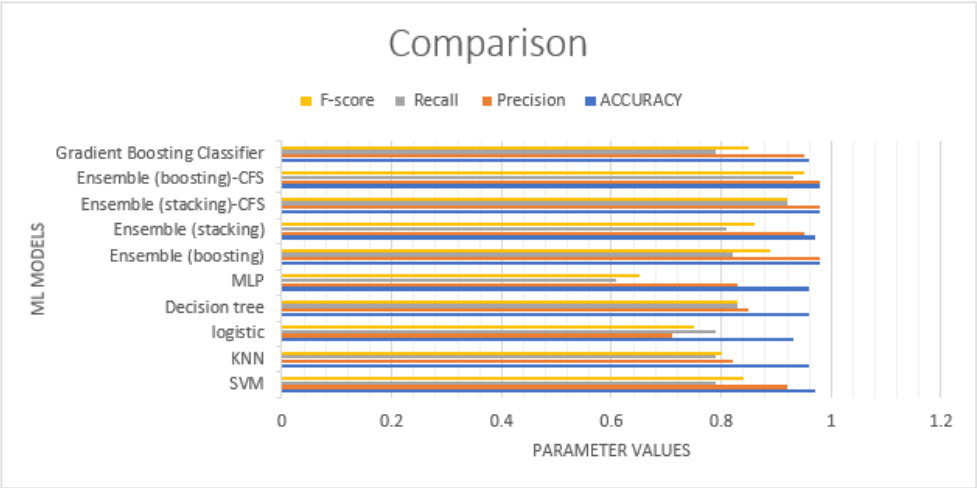

Figure 3. Comparison of proposed and existing approaches on Different Performance Metrics.

For pattern recognition, a classifier makes its decisions based on what is has learned from training data, adapting its behavior to the structure of data it is able to identify. However, in some cases, different hints may arise from different aspects of the data, and these hints may be conflicting. In such situations, the classification model must make a choice. Ensemble pattern recognition improves classification metrics by using multiple methods of pattern recognition at the same time. Ensemble learning improves parameters with explicit and implicit weighting through domain specialization, resulting in better classification performance metrics. This is especially useful when multiple classifiers capture different patterns in the structure of data. Based on Figure 3, we observe that the Ensemble (Boosting-CFS) approach exhibits a better performance compared to other approaches. In general, models with CFS perform better than their counterparts without CFS. The Ensemble (Stacking-CFS), in fact, is second best.

Following a cumulative performance analysis of both classes of risky and non-risky blocks of the proposed solution, without feature selection in Table 3 and with feature selection in Table 4, it is clear that both types of blocks are expected in equilibrium.

Table 2. Classification Report.

Approaches	Accuracy	Precision	Recall	F-Measure
SVM	0.97	0.92	0.79	0.84
KNN	0.96	0.82	0.79	0.80
Logistic	0.93	0.71	0.79	0.75
Decision Tree	0.96	0.85	0.83	0.83
MLP	0.96	0.83	0.61	0.65
Ensemble (Boosting)	**0.98**	**0.98**	0.82	0.89
Ensemble (Stacking)	0.97	0.95	0.81	0.86
Ensemble (Stacking-CFS)	**0.98**	**0.98**	0.92	0.92
Ensemble (Boosting-CFS)	**0.98**	**0.98**	**0.93**	**0.95**
Gradient Boosting Classifier	0.96	0.95	0.79	0.85

Figures 4 and 5 show the results for different evaluation parameters, for the Ensemble Boosting and Stacking models. It can be observed that feature selection influences positively the predictive power. Results, in general suggest that the data have a very complex structure,

more evidently so for the non-risky records which cover a wider variety of situations, and that classifiers are unable to capture all this complexity when working in isolation.

Table 3. Class-wise comparison of Ensemble classifier by Boosting and Stacking without Feature Selection.

	Ensemble (Boosting)			Ensemble (Stacking)		
Class	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
Risk	0.97	0.98	0.98	0.97	0.99	0.98
Not Risk	0.67	0.59	0.63	0.86	0.59	0.70

Table 4. Class-wise comparison of Ensemble classifier by Boosting and Stacking with Feature Selection.

	Ensemble (Boosting)			Ensemble (Stacking)		
Class	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
Risk	0.97	1.00	0.99	0.97	1.00	0.98
Not Risk	0.98	0.62	0.76	0.95	0.62	0.75

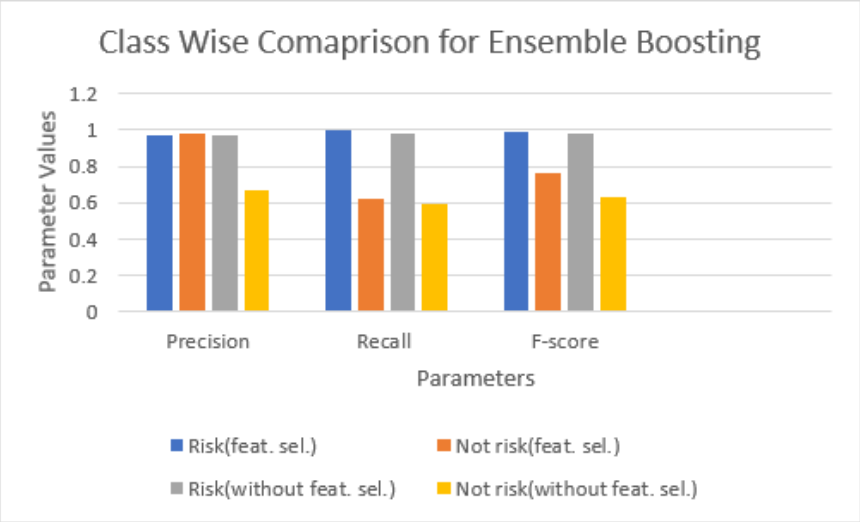

Figure 4. Comparison of performance metrics with or without Feature Selection for Boosting Process.

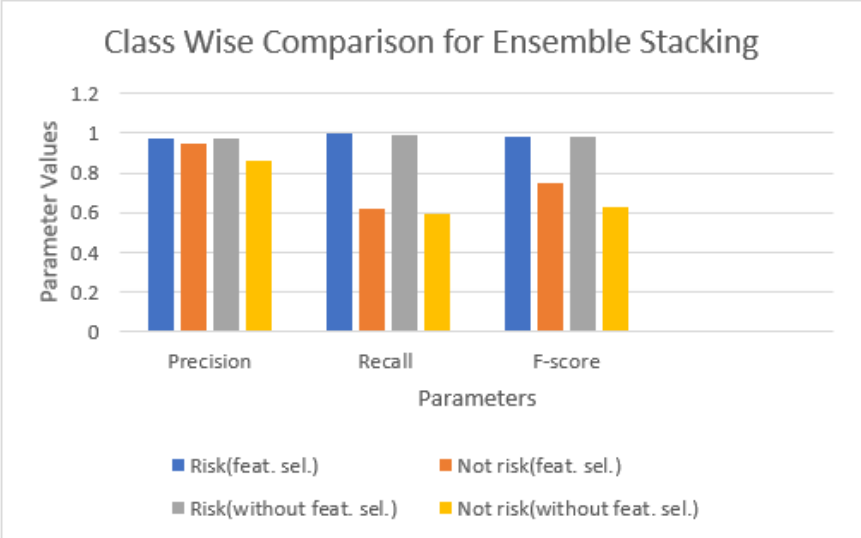

Figure 5. Comparison of performance metrics with or without Feature Selection for Stacking Process.

6. Conclusions and Future Work

The blockchain is getting progressively perceived and acknowledged as a valuable technology by money-related organizations. Financial firms and institutions are investigating the benefits possibly achievable with blockchain usage and advancement. Firms want to know who their customers are and how they can improve their services to serve them better. Knowing who they are trading with is one of the most critical tasks for financial companies today. Banks and other financial institutions focus on the authorization or financial process because they provide a sense of security [30] that protects their business from fraud and illegal financial transactions. Traditional financial practices are very strict, resource-intensive, and time-consuming. Organizations often need days to complete financial processes. This process is not only time-consuming but also creates costs. So, the improvement of detection of risky blocks is challenging and worth studying.

We developed a classification model which helps in the detection of risky blocks in a blockchain. To cope with the complexity of the data, which is difficult to address with a single classifier, we applied Ensemble learning. Experiments showed an improvement over competing models in Accuracy (2–3%), Precision (5–6%), Recall (8–9%), and F-score (7–8%).

Future work will be centered on developing an effective incremental learning strategy, to improve work on real-time data coming as a stream, and on enhancing the ensemble of classifiers to capture the structure of the problem even better.

Author Contributions: Conceptualization, C.J., R.J., U.F. and S.C.; methodology, C.J. and R.J.; validation, C.J., U.F. and S.C.; investigation, R.J.; resources, S.C.; data curation, R.J.; writing—original draft preparation, C.J. and R.J.; writing—review and editing, C.J. and U.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not Applicable, the study does not report any data.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory
ML	Machine Learning
MLP	Multi-Layer Perceptron
SVM	Support Vector Machine

References

1. Sestrem Ochôa, I.; Augusto Silva, L.; De Mello, G.; Garcia, N.M.; de Paz Santana, J.F.; Quietinho Leithardt, V.R. A cost analysis of implementing a blockchain architecture in a smart grid scenario using sidechains. *Sensors* **2020**, *20*, 843. [CrossRef] [PubMed]
2. Moosavi, J.; Naeni, L.M.; Fathollahi-Fard, A.M.; Fiore, U. Blockchain in supply chain management: A review, bibliometric, and network analysis. *Environ. Sci. Pollut. Res.* **2021**. [CrossRef] [PubMed]
3. Zheng, Z.; Xie, S.; Dai, H.N.; Chen, X.; Wang, H. Blockchain challenges and opportunities: A survey. *Int. J. Web Grid Serv.* **2018**, *14*, 352–375. [CrossRef]
4. Mermer, G.B.; Zeydan, E.; Arslan, S.S. An overview of blockchain technologies: Principles, opportunities and challenges. In Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2–5 May 2018; pp. 1–4.
5. Bahga, A.; Madisetti, V. Blockchain Platform for Industrial Internet of Things. *J. Softw. Eng. Appl.* **2016**, *9*, 533–546. [CrossRef]
6. Golosova, J.; Romanovs, A. The advantages and disadvantages of the blockchain technology. In Proceedings of the 2018 6th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), Vilnius, Lithuania, 8–10 November 2018; pp. 1–6.
7. Abu-Elezz, I.; Hassan, A.; Nazeemudeen, A.; Househ, M.; Abd-Alrazaq, A. The benefits and threats of blockchain technology in healthcare: A scoping review. *Int. J. Med. Inform.* **2020**, *142*, 104246. [CrossRef] [PubMed]

8. Puthal, D.; Malik, N.; Mohanty, S.P.; Kougianos, E.; Yang, C. The blockchain as a decentralized security framework [future directions]. *IEEE Consum. Electron. Mag.* **2018**, *7*, 18–21. [CrossRef]
9. Tariq, N.; Asim, M.; Al-Obeidat, F.; Zubair Farooqi, M.; Baker, T.; Hammoudeh, M.; Ghafir, I. The security of big data in fog-enabled IoT applications including blockchain: A survey. *Sensors* **2019**, *19*, 1788. [CrossRef] [PubMed]
10. Viel, F.; Augusto Silva, L.; Leithardt, V.R.Q.; De Paz Santana, J.F.; Celeste Ghizoni Teive, R.; Albenes Zeferino, C. An Efficient Interface for the Integration of IoT Devices with Smart Grids. *Sensors* **2020**, *20*, 2849. [CrossRef] [PubMed]
11. Tariq, N.; Asim, M.; Khan, F.A.; Baker, T.; Khalid, U.; Derhab, A. A blockchain-based multi-mobile code-driven trust mechanism for detecting internal attacks in internet of things. *Sensors* **2021**, *21*, 23. [CrossRef]
12. Salah, K.; Rehman, M.H.U.; Nizamuddin, N.; Al-Fuqaha, A. Blockchain for AI: Review and open research challenges. *IEEE Access* **2019**, *7*, 10127–10149. [CrossRef]
13. Tanwar, S.; Bhatia, Q.; Patel, P.; Kumari, A.; Singh, P.K.; Hong, W.C. Machine learning adoption in blockchain-based smart applications: The challenges, and a way forward. *IEEE Access* **2019**, *8*, 474–488. [CrossRef]
14. Hassani, H.; Huang, X.; Silva, E. Banking with blockchain-ed big data. *J. Manag. Anal.* **2018**, *5*, 256–275. [CrossRef]
15. Wu, J.; Liu, J.; Chen, W.; Huang, H.; Zheng, Z.; Zhang, Y. Detecting Mixing Services via Mining Bitcoin Transaction Network with Hybrid Motifs. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, 1–13. [CrossRef]
16. Michalski, R.; Dziubałtowska, D.; Macek, P. Revealing the Character of Nodes in a Blockchain with Supervised Learning. *IEEE Access* **2020**, *8*, 109639–109647. [CrossRef]
17. Sgantzos, K.; Grigg, I. Artificial intelligence implementations on the blockchain. Use cases and future applications. *Future Internet* **2019**, *11*, 170. [CrossRef]
18. Marwala, T.; Xing, B. Blockchain and artificial intelligence. *arXiv* **2018**, arXiv:1802.04451.
19. Gatteschi, V.; Lamberti, F.; Demartini, C.; Pranteda, C.; Santamaría, V. Blockchain and smart contracts for insurance: Is the technology mature enough? *Future Internet* **2018**, *10*, 20. [CrossRef]
20. Conley, J.P. Blockchain and the economics of crypto-tokens and initial coin offerings. In *Vanderbilt University Department of Economics Working Papers*; Vanderbilt University Department of Economics: Nashville, TN, USA, 2017.
21. Samaniego, M.; Deters, R. Internet of smart things-IOsT: Using blockchain and clips to make things autonomous. In Proceedings of the 2017 IEEE International Conference on Cognitive Computing (ICCC), Honolulu, HI, USA, 25–30 June 2017; pp. 9–16.
22. Seebacher, S.; Schüritz, R. Blockchain technology as an enabler of service systems: A structured literature review. In *International Conference on Exploring Services Science*; Springer: Berlin/Heidelberg, Germany 2017; pp. 12–23.
23. Hall, W.; Pesenti, J. Growing the artificial intelligence industry in the UK. In *Department for Digital, Culture, Media & Sport and Department for Business, Energy & Industrial Strategy. Part of the Industrial Strategy UK and the Commonwealth*; 2017. Available online: https://www.d-long.com/eWebEditor/uploadfile/20171019203827815166683.pdf (accessed on 28 November 2021).
24. Ogiela, L.; Ogiela, M.R. Cognitive security paradigm for cloud computing applications. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5316. [CrossRef]
25. Paul, S. Ensemble Learning—Bagging, Boosting, Stacking and Cascading Classifiers in Machine Learning Using SKLEARN and MLEXTEND libraries. 2018.
26. Weber, M.; Domeniconi, G.; Chen, J.; Weidele, D.K.I.; Bellei, C.; Robinson, T.; Leiserson, C.E. Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics. In Proceedings of the KDD'19 Workshop on Anomaly Detection in Finance, Anchorage, AK, USA, 5 August 2019.
27. West, J.; Bhattacharya, M. Mining financial statement fraud: An analysis of some experimental issues. In Proceedings of the 2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA), Auckland, New Zealand, 15–17 June 2015; pp. 461–466.
28. Ghosh, M.; Sanyal, G. An ensemble approach to stabilize the features for multi-domain sentiment analysis using supervised machine learning. *J. Big Data* **2018**, *5*, 1–25. [CrossRef]
29. West, J.; Bhattacharya, M. An investigation on experimental issues in financial fraud mining. In Proceedings of the 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), Hefei, China, 5–7 June 2016; pp. 1796–1801.
30. Ogiela, U. Cognitive cryptography for data security in cloud computing. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5557. [CrossRef]