https://doi.org/10.31891/csit-2025-4-9
UDC 004.8

KOCHURA Vitaly, LOKTIKOVA Tamara, KUSHNIR Nadia
Zhytomyr Polytechnic State University

ANALYTICAL WEB SERVICE FOR IDENTIFYING SUSPICIOUS HIGH-RISK DIGITAL ASSET TRANSACTIONS

The rapid growth of digital asset transactions significantly complicates their analysis and monitoring. Although blockchain provides transparency of operations, the high level of pseudonymity among participants creates substantial challenges in identifying the nature of interactions and detecting potentially undesirable activity. This increases the demand for modern tools capable of automatically processing large volumes of data, grouping addresses into clusters, and evaluating their behavior based on aggregated transactional characteristics. The development of a web service for digital asset transaction analytics that automatically forms address clusters and classifies them using machine learning models is proposed. The system provides an informative and interpretable data presentation, enabling users to assess the nature of activity associated with a given address. The methodology is based on heuristics for clustering blockchain addresses in networks utilizing the UTXO model, as well as on the calculation of structural and behavioral characteristics of the formed clusters. Random Forest, Extra Trees, and XGBoost models were used for classification, which were trained on a labeled subset of the scientific dataset. The technical implementation of the web service is based on the Symfony framework for the server side, the React library for the client side, MySQL and PostgreSQL DBMS for data storage, the Python programming language for machine learning, as well as Docker and Nginx tools for deployment and stability. The results of the study demonstrate that machine learning models can effectively classify clusters of digital asset addresses according to their behavioral characteristics, and the integration of algorithms into the web service provides automatic generation of analytical reports. The scientific novelty lies in combining heuristics for address clustering with machine learning models, which allows evaluating the behavior of clusters and identifying potential risk. The practical significance of the developed web service is defined by the possibility of its application for rapid preliminary checks of cryptocurrency addresses, detecting relationships between them, and assessing potential interaction risks, making it valuable for organizations and individual users alike.

Keywords: web service, digital assets, blockchain, transaction, clustering, machine learning, Symfony, React.

КОЧУРА Віталій, ЛОКТІКОВА Тамара, КУШНІР Надія
Державний університет «Житомирська політехніка»

АНАЛІТИЧНИЙ ВЕБСЕРВІС ДЛЯ ІДЕНТИФІКАЦІЇ ПІДОЗРІЛИХ ТРАНЗАКЦІЙ ЦИФРОВИХ АКТИВІВ З ПІДВИЩЕНИМ РИЗИКОМ

Стрімке зростання операцій із цифровими активами ускладнює їхні аналіз і контроль. Хоча блокчейн забезпечує прозорість транзакцій, але високий рівень псевдонімності учасників створює значні труднощі у визначенні характеру взаємодій та виявленні потенційно небажаної активності. Це викликає потребу в сучасних інструментах, здатних автоматично обробляти великі обсяги даних, об'єднувати адреси у кластери та оцінювати їхню поведінку на основі узагальнених транзакційних характеристик. Пропонується розробка вебсервісу для аналітики транзакцій цифрових активів, який автоматично формує кластери адрес і виконує їхню класифікацію за допомогою моделей машинного навчання. Система забезпечує інформативне, інтерпретоване подання даних і надає користувачеві можливість оцінювати природу активності певної адреси. Методика побудована на застосуванні евристик для кластеризації блокчейн-адрес у мережах з UTXO-моделлю, а також на обчисленні структурних і поведінкових характеристик сформованих кластерів. Для класифікації було використано моделі Random Forest, Extra Trees та XGBoost, навчання яких проводилося на маркованій підмножині наукового датасету. Технічна реалізація вебсервісу здійснена на основі фреймворку Symfony для серверної частини, бібліотеки React для клієнтської частини, СУБД MySQL і PostgreSQL для роботи з даними, мови програмування Python для машинного навчання, а також інструментів Docker і Nginx для розгортання та забезпечення стабільності роботи. Результати дослідження показали, що моделі машинного навчання здатні ефективно класифікувати кластери адрес цифрових активів за їхніми поведінковими ознаками, а інтеграція алгоритмів у вебсервіс забезпечує автоматичне формування аналітичних звітів. Наукова новизна полягає у поєднанні евристичних підходів до кластеризації адрес із моделями машинного навчання, що дозволяє оцінювати поведінку кластерів та визначати їхню потенційну ризиковість. Практична значимість розробленого вебсервісу визначається можливістю його застосування для оперативної перевірки криптовалютних адрес, виявлення зв'язків між ними та оцінювання потенційних ризиків взаємодії, що є корисним як для організацій, так і для звичайних користувачів.

Ключові слова: вебсервіс, цифрові активи, блокчейн, транзакція, кластеризація, машинне навчання, Symfony, React.

Received / Стаття надійшла до редакції 26.11.2025
Accepted / Прийнята до друку 20.12.2025

Introduction

The rapid growth in the volume of digital asset transactions and the widespread adoption of cryptocurrencies significantly complicate the processes of their analysis and monitoring. Blockchain technology, which underpins most digital currencies and ensures data transparency, simultaneously introduces a high degree of pseudonymity among its users [1]. This characteristic enables direct peer-to-peer transactions but makes it more difficult to detect risky or potentially undesirable transactional interactions.

Particular attention is required for blockchain networks that operate under the UTXO model, most notably Bitcoin [2]. In such networks, asset flows are represented as sequences of unspent transaction outputs that reflect changes in an address's state but do not store information about account balances or the identity of the address owner. As a result, large volumes of low-level transactional data must undergo preprocessing, aggregation, and subsequent structuring to form a generalized representation of interactions.

In blockchain networks, individual addresses may participate in joint transactions, engage in repeated operations, or exhibit stable patterns of interaction. Such interactions form groups of addresses that act in a coordinated manner and reflect the activity of a single user or a single entity within the network. These groups are commonly referred to as clusters [3]. The analysis of such clusters makes it possible to gain insights into the movement of digital assets, identify recurring interaction patterns, and detect potential indicators of atypical behavior [4].

One of the major challenges is assessing the risk associated with clusters based on transactional data. Traditional financial monitoring techniques, designed for centralized systems, are not sufficiently effective in decentralized blockchain environments. Under these conditions, the use of machine learning methods becomes necessary, as they are capable of analyzing complex transactional relationships, uncovering hidden behavioural patterns, and automating the process of determining cluster risk based on their extracted characteristics.

These factors underscore the relevance of developing a web service that provides comprehensive blockchain analytics – from retrieving and structuring transactional data to forming clusters and subsequently classifying them using modern machine learning algorithms. This approach not only enhances the transparency of digital asset flows but also enables the creation of a tool for identifying potentially risky clusters, which may be valuable in the context of anti-money laundering efforts, financial monitoring, and risk assessment within the digital ecosystem.

Related works

With the increasing complexity of transactional relationships in blockchain networks, the need for specialized tools capable of automating data collection, processing, and analysis has become more pronounced [5]. Today, several leading solutions offer various approaches to examining digital assets, visualizing transactions, and assessing the potential risk of address behaviour. An examination of such solutions makes it possible to identify their advantages and limitations and outline the key requirements for developing an analytical instrument of one's own.

One of the most well-known solutions in the field of blockchain analytics is Elliptic [6]. The platform provides tools for constructing transactional graphs, detecting relationships between addresses, and performing clustering based on behavioural characteristics [7]. A key advantage of Elliptic is its use of machine learning algorithms, which enables the identification of complex money-laundering schemes and other atypical patterns in digital asset movements. The platform is designed for deep, comprehensive analysis; however, its accessibility is limited – it is a commercial, closed service primarily oriented toward the corporate sector.

Another widely used tool is Crystal Intelligence (Crystal Blockchain) [8]. This service offers detailed visualization of transactions, construction of asset-movement graphs, and detection of linked addresses. It is characterized by an intuitive interface and extensive graphical representations, which allow analysts to trace complex transfer chains efficiently. Despite its powerful analytical capabilities, Crystal Intelligence is also a commercial product that requires a paid subscription to access its full functionality.

For basic open-access address checking, simpler tools exist, such as Check Crypto Address [9]. The service enables quick validation of a cryptocurrency address and provides limited insights into its activity. However, the functionality of such platforms is significantly restricted: they do not perform clustering, offer no comprehensive analysis of transaction history, and do not apply intelligent risk-assessment methods. These instruments are useful only for preliminary checks, not for in-depth behavioural analysis of digital assets [10].

A comparison of existing solutions shows that while modern tools offer substantial functionality, they also possess notable limitations. The review and analysis of existing solutions helped identify core requirements for the proposed service: the ability to perform address clustering, compute behavioural characteristics of clusters, support machine learning-based classification, and provide a clear and intuitive interface. These criteria form the foundation for developing a solution that combines extensive analytical capabilities for digital asset transactions with accessibility and ease of use.

Purpose

The purpose of this paper is to examine the development process of a web service for digital asset transaction analytics, designed to automatically form address clusters in blockchain networks and assess their risk level using machine learning methods.

Materials and methods

In the modern digital environment, digital assets play a significant role as units of value that exist exclusively in electronic form and can be transferred between users without the involvement of traditional financial

institutions. The most widespread type of such assets is cryptocurrencies, whose operation is based on cryptographic mechanisms and decentralized protocols. A fundamental characteristic is the absence of a centralized issuer, and one of the most important features is the ability to conduct peer-to-peer transactions directly between participants.

The first and most well-known cryptocurrency is Bitcoin, which demonstrated the practical implementation of the concept of decentralized electronic money. Its operation is based on blockchain technology – a distributed ledger that stores the complete history of transactions and ensures transparency and immutability of data. Each block in the chain contains a set of confirmed transactions and is linked to the previous one through cryptographic hashing, which guarantees data integrity and resistance to tampering.

Users interact with the blockchain network through cryptocurrency wallets and addresses, which act as public identifiers. The flow of transactions between such addresses forms a complex network of interconnections that can be used for analytical purposes [11]. At the same time, addresses do not contain information about real owners, which creates pseudonymity and complicates direct tracking of interactions between participants.

In several blockchain networks, particularly Bitcoin, transactions operate according to the UTXO (Unspent Transaction Output) model. In this model, there is no concept of an account or balance – instead, the state of an address is defined by a set of unspent outputs from previous transactions. Each UTXO can be used as an input in a new transaction, while the outputs produced by that transaction form new UTXOs. This structure provides high transparency of asset flows, allowing their origin to be traced precisely, but it also generates a large volume of low-level data requiring additional processing and interpretation.

Within transaction activity, it is often possible to observe cases where multiple addresses participate in joint operations, exhibit recurring behavioral patterns, or form stable logical relationships. Such groups of addresses are considered clusters, and they represent an important object of analysis because they enable a shift from examining individual transactions to studying behavioral models of network participants. The origin of clusters may vary: they can belong to regular users, services, exchanges, or even suspicious platforms [12]. For this reason, clustering is a crucial stage in the context of risk analysis and the detection of atypical activity.

One of the most important and crucial stages of processing transaction data is address clustering – the process of grouping public keys into sets that, with high probability, belong to the same user or entity within the network. Since the blockchain does not contain information about the real owners of addresses, clustering is performed using specific heuristic rules that reduce the level of pseudonymity and provide a generalized structure of interactions.

In the developed web service, such heuristic rules form the basis of the clustering process and are applied automatically during transaction data processing. They make it possible to construct the structure of a cluster, upon which its aggregated transactional characteristics are calculated. These characteristics are subsequently used for classification with machine learning models.

One of the most well-known and widely used heuristics in networks that implement the UTXO model is the Multi-Input Heuristic (also known as Common-Input Ownership or Multi-Input Address Clustering), whose schematic representation is shown in Figure 1 [13, 14]. Its principle is based on the assumption that if a transaction includes multiple input addresses, all of these addresses are highly likely to belong to the same user or organization. This follows from the fact that creating a transaction requires access to the private keys of all input UTXOs, meaning the corresponding addresses must be controlled by a single entity. Due to its simplicity and high practical efficiency, this heuristic serves as a fundamental mechanism in most address clustering systems.

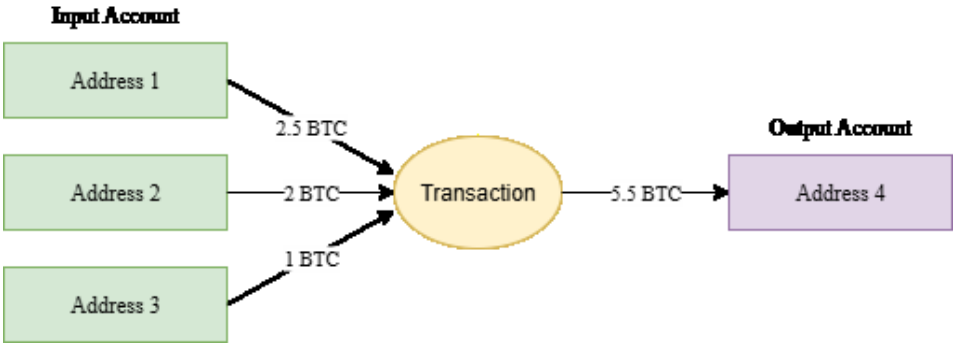

Fig. 1. Multi-Input Heuristic (Common-Input-Ownership Heuristic/Multi-Input Address Clustering) schematic diagram

Another important heuristic is the Change Address Heuristic, which aims to identify the address that receives the «change» in a transaction [15, 16]. In most modern cryptocurrency wallets, any amount exceeding the value of the payment is returned to the sender and, for enhanced privacy, is typically transferred not to an already used address but to a newly generated one. If a transaction contains one output associated with a known address and another that is newly created, the latter can, with high probability, be identified as the change address. Using this heuristic, a cluster can be expanded with additional addresses that are also controlled by the same user. A schematic illustration of this identification process is presented in Figure 2.

To further improve identification accuracy, the Force Merge of Inputs Heuristic is used [17]. It analyzes situations in which an address does not have sufficient funds to complete a payment. In such cases, the wallet is forced to combine multiple UTXOs belonging to different addresses it controls, creating a transaction with several inputs. By analyzing the total value of inputs and the structure of outputs, it becomes possible to detect the change address and refine the address ownership relationships. This heuristic relies on the mathematical relationship between the sum of input UTXOs, the required payment amount, and the remaining funds: if the difference between the total input value and the required transfer amount is smaller than the amount received by the recipient, then the smaller output is, with high probability, the change address.

Although these heuristics are essential tools in the clustering process, they are probabilistic in nature and cannot guarantee absolute accuracy. Errors may occur in cases of intentional obfuscation of links, the use of privacy-enhancing systems, mixers, or complex fund distribution schemes.

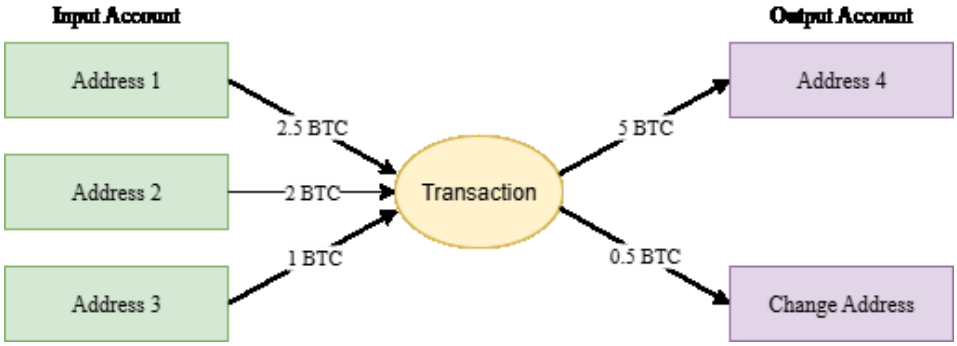

Fig. 2. Change Address Heuristic schematic diagram

After the clustering stage, there arises a need for methods capable of interpreting the obtained groups of addresses based on their aggregated characteristics and determining their membership in specific behavioral categories. Since blockchain transaction data contains a large number of parameters and exhibits complex interdependencies, it is appropriate to apply machine learning algorithms capable of effectively handling high-dimensional structured features.

Among the algorithms that demonstrate high effectiveness in structured data classification tasks, particular attention should be given to the Random Forest, Extra Trees, and XGBoost methods.

Random Forest is a classical ensemble method composed of a large number of independent decision trees [18]. Each tree generates its own prediction based on a random subset of features, while the final prediction is determined by aggregating the results of all trees. This approach reduces the risk of overfitting, increases model robustness to noise, and enables stable performance on datasets with imbalanced or highly variable distributions. Owing to these properties, Random Forest is an effective tool for analyzing the behavioral characteristics of digital asset clusters.

Extra Trees (Extremely Randomized Trees) employs a strategy in which randomness plays a central role in building decision trees [19]. In contrast to classical algorithms that select split points based on an optimal criterion, Extra Trees generates split thresholds at random, reducing the model's dependence on local data properties and significantly accelerating training. Despite the increased randomness, the large number of trees ensures accurate and consistent results. This method is well suited for transactional data, which often exhibit complex distributions and high dimensionality.

XGBoost belongs to the family of gradient boosting methods and operates as an ensemble of decision trees constructed sequentially, with each subsequent tree correcting the errors of its predecessors [20]. This mechanism enables the model to capture complex patterns in the data while maintaining high computational efficiency. XGBoost also incorporates regularization, which reduces the risk of overfitting, and handles missing values and large feature sets effectively. Due to its flexibility and strong predictive power, XGBoost is one of the most suitable methods for classifying digital asset clusters and assessing their potential risk.

Thus, heuristic approaches make it possible to form address clusters, while machine learning models provide the means for their further classification based on behavioral characteristics. Together, these techniques enable comprehensive analysis of interactions within the network and the detection of anomalous behavioral patterns.

An important stage in the development of the described solution is its technical implementation, which directly depends on the choice of programming languages, frameworks, and supporting tools that ensure the reliable operation of the system. In this context, the following section examines the technologies employed in the development of the proposed service.

PHP was chosen as the primary programming language for implementing the server-side component. For many years, it has remained one of the leading tools in web development. Its popularity is reinforced by a large developer community, extensive learning resources, and long-term ecosystem support. These factors make PHP relatively easy to learn while still enabling the development of scalable and efficient solutions. An additional

advantage is the availability of numerous frameworks that significantly simplify and accelerate the software development process.

Among such frameworks, Symfony holds a particularly important place due to its modular architecture and extensive collection of ready-to-use components [21]. This structure allows developers to handle standard tasks without duplicating code, thereby increasing the efficiency of the development process. Furthermore, Symfony provides a high level of security and stability and supports a flexible configuration system, making it adaptable to the specific requirements of the project.

To ensure efficient request processing, the API Platform framework was used. It offers a comprehensive set of modern tools for rapid API development in accordance with current standards. Thanks to its seamless integration with Symfony, it automatically generates CRUD operations and significantly simplifies service deployment, reducing development time and increasing overall productivity.

Modules responsible for data processing and analysis using artificial intelligence methods were implemented in Python. This language is considered an industry standard in machine learning due to its clear syntax, extensive library ecosystem, and strong community support. Python enabled the project to combine the development and testing of machine learning models with their integration into the overall system architecture.

MySQL was selected as the primary data management system. It is known for its ease of use, accessibility for beginners, and wide popularity among developers. Moreover, MySQL is considered one of the most reliable and secure relational database management systems, and it is successfully used in many well-known web applications.

In addition to MySQL, PostgreSQL was also employed. Its use was dictated by the format of the dataset, which was provided as a database dump specifically for this DBMS. PostgreSQL enabled the correct and complete restoration of data structures containing large tables with clusters and their characteristics. Due to its high performance and optimization for handling large volumes of information, it became an effective tool during the stages of loading, exploring, and preprocessing the data required for further analysis and model training.

Nginx was chosen as the web server due to its high performance and ability to handle a large number of simultaneous connections efficiently. Its stability, built-in caching mechanisms, and support for load balancing have made it one of the most widely adopted solutions for modern web applications.

The client-side component was developed using the React library, which is one of the most widely used technologies in the field of frontend development [22]. It enables the creation of dynamic and interactive user interfaces with less code compared to classical JavaScript. The core advantage of React lies in its component-based architecture: the interface is divided into independent and reusable elements, simplifying both maintenance and further development of the application.

The deployment of the application was carried out using Docker, a platform that provides containerization of system components and convenient management of them. Containers isolate services along with all their dependencies, ensuring stability and predictable behavior across different environments. This approach greatly simplified the scalability of the system and increased the efficiency of the development workflow.

The coherent combination of these technologies established a reliable foundation for the operation of the system. The use of mature programming languages, modern frameworks, flexible database engines, and containerization tools made it possible to build a scalable, secure, and efficient software solution. Each technology plays a clearly defined role – from implementing business logic and performing data analytics to managing user interaction and optimizing server infrastructure. As a result, the system achieved an optimal balance between performance, stability, and ease of further development.

Dataset

The training of the models was performed using an open scientific dataset of the Bitcoin transaction network, published by a research group in Scientific Data (Nature Portfolio) [23]. The dataset contains a complete temporal graph of digital asset transfers constructed from real Bitcoin transactions collected over many years. In total, it includes approximately 252 million nodes (clusters), making it the largest publicly available source for analyzing the structure and behavior of participants in a blockchain network.

A separate subset of this dataset contains 34,000 labeled clusters, each assigned a specific category (label) describing the type of activity associated with the entity that controls the cluster. This portion of the data was used for supervised learning and for evaluating the potential risk of clusters based on their transactional characteristics. The key attributes of the clusters used in the analysis are summarized in Table 1, which provides descriptions of the structural and behavioral parameters that form the input features for the machine learning models.

The labeled clusters in the dataset belong to various categories that reflect the behavioral patterns of participants in the Bitcoin network. Among them are the following:

– Individual.
– Mining: individual or entity that validates and confirms transactions on the Bitcoin network.
– Exchange: online platform that facilitates the buying, selling, and trading of cryptocurrencies and fiat currencies.
– Marketplace: online platform where users can buy and sell goods or services using bitcoin as the primary form of payment.

– Gambling: online platform where users can wager and play casino games, sports betting, and participate in lotteries using Bitcoin.

– Bet: address created by a gambling service specifically for receiving funds related to a unique bet.

– Faucet: promotional tool that rewards users with small amounts of bitcoin for completing tasks or viewing advertisements.

– Mixer: service that enhances the privacy and anonymity of transactions by making it more difficult to trace transactions on the blockchain.

– Ponzi: a financial scheme promising high returns to investors by using funds from new investors to pay returns to earlier investors.

– Ransomware: malicious software that encrypts files on a victim's computer, demanding payment to decrypt and restore access.

– Bridge: protocol that facilitates the exchange of assets between Bitcoin and different blockchain networks (e.g. Ethereum).

Table 1

Attributes of the nodes (clusters)

Column Name	Description
alias	Identifier of the node
label	Label describing the type of entity represented by the node
total_transaction_in	Total count of value transfers received by the node
total_transaction_out	Total count of value transfers initiated by the node
first_transaction_in	Block index of the first transaction received by the node
last_transaction_in	Block index of the last transaction received by the node
first_transaction_out	Block index of the first transaction sent by the node
last_transaction_out	Block index of the last transaction sent by the node
min_sent	Minimum amount sent by the node during a transaction
max_sent	Maximum amount sent by the node during a transaction
total_sent	Total amount sent by the node during all transactions
min_received	Minimum amount received by the node during a transaction
max_received	Maximum amount received by the node during a transaction
total_received	Total amount received by the node during all transactions
cluster_size	Number of addresses represented by the node
cluster_num_edges	Number of transactions between the addresses represented by the node
cluster_num_cc	Number of connected components in the transaction graph of the addresses represented by the node
cluster_num_nodes_in_cc	Number of non-isolated addresses in the cluster

Results

The analysis of transactional clusters formed on the basis of heuristic rules requires assessing how accurately the behavioral type of each cluster can be determined from its aggregated characteristics. To this end, a comprehensive evaluation of machine learning models was conducted using the labeled subset of clusters from the dataset described above. The models were trained on a unified set of structural and behavioral features that represent the activity of each cluster within the Bitcoin network. The results obtained made it possible to compare different algorithms and identify the most effective approach for classifying digital asset clusters.

After completing the training process, the main performance metrics of the models were obtained. Their aggregated values are presented in Table 2, which allows comparing accuracy, recall, precision, and F1-scores for each algorithm.

Table 2

Performance comparison of the trained models

Model	Accuracy	Macro Precision	Macro Recall	Macro F1-score
Random Forest	0.9222	0.7342	0.5538	0.6159
Extra Trees	0.9199	0.7541	0.5864	0.6484
XGBoost	0.9199	0.7053	0.6402	0.6635

All three models demonstrate high accuracy; however, XGBoost provides the best balance between precision and recall. This model achieved the highest Macro F1-score and showed superior ability to recognize less frequent cluster categories, which is particularly important when working with imbalanced data.

To further evaluate the performance of XGBoost, a confusion matrix was constructed, enabling a visual examination of correct and incorrect classifications across different cluster categories (Figure 3). The matrix shows that the model recognizes rare classes more effectively and reduces the number of cross-category misclassifications compared to the other algorithms.

An additional feature importance analysis was performed to determine which cluster parameters have the greatest influence on the model's decision-making. The results of this analysis are presented in Figure 4.

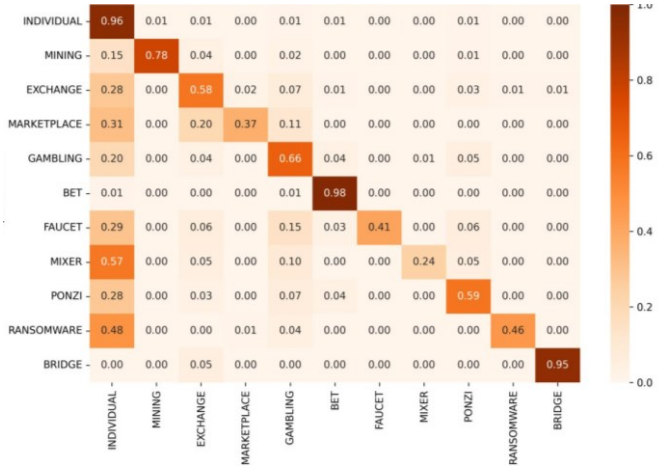

Fig. 3. Confusion matrix of the XGBoost model

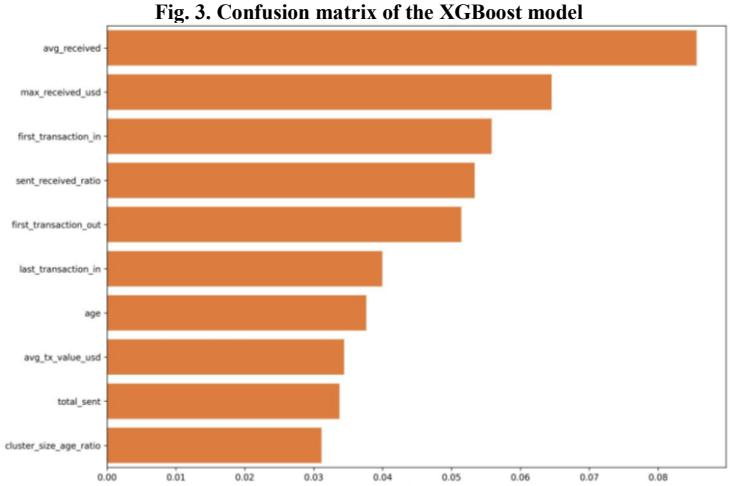

Fig. 4. Feature importance of the XGBoost model

Thus, the developed web service automatically performs the retrieval and loading of information about the address entered by the user, carries out clustering based on heuristic rules, generates aggregated transactional characteristics of the cluster, and passes them to the analytical module for behavioral classification. The main page of the web service is presented in Figure 5. As shown, the proposed approach covers all stages – a from processing low-level blockchain data to obtaining a prediction about the potential category of the cluster and the address associated with it.

On the client side, a report page is generated, containing information on the cluster's transactional indicators, temporal activity, detected address relationships, and the predicted category. Figure 6 shows the generated report page for the analyzed address. This ensures a clear and intuitive presentation of complex data, allowing the user to quickly understand the behavioral characteristics of the desired address.

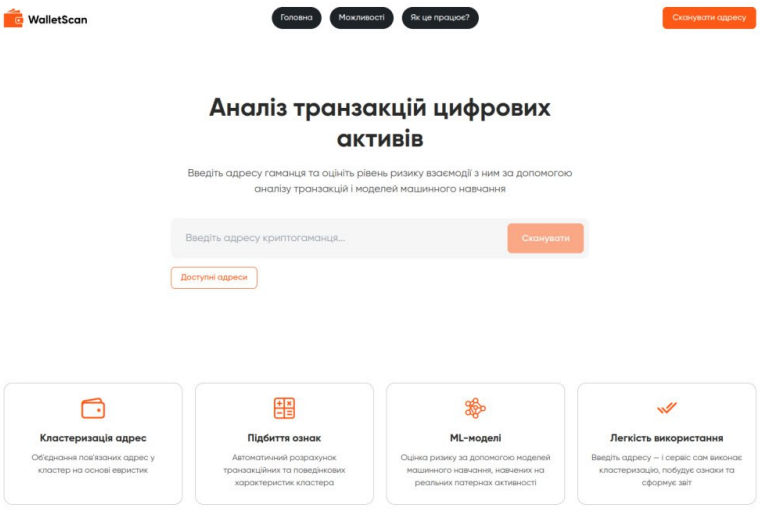

Fig. 5. Main page of the analytical web service

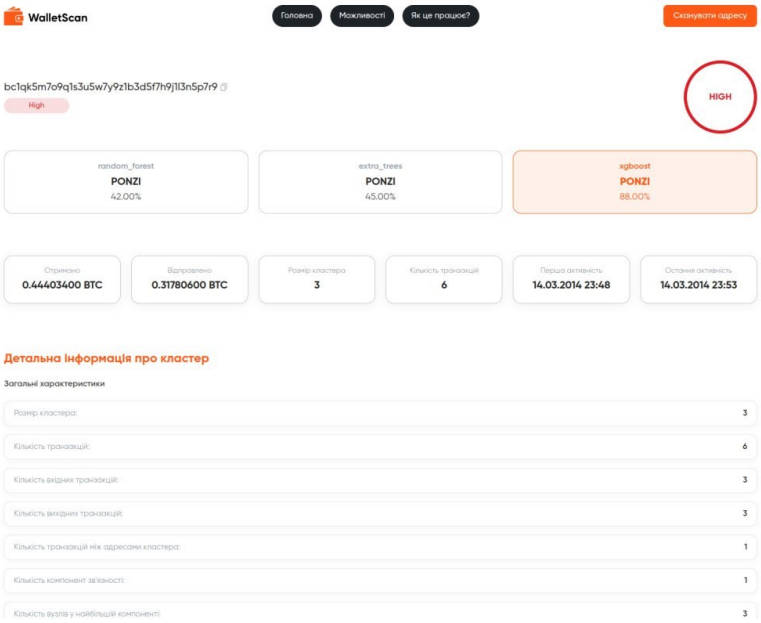

Fig. 6. Generated report page for the analyzed address

Author Contributions

Conceptualization: Vitaly Kochura, Tamara Loktikova;
Methodology: Vitaly Kochura, Tamara Loktikova;
Software: Vitaly Kochura;
Validation: Vitaly Kochura, Tamara Loktikova, Nadia Kushnir;
Formal analysis: Vitaly Kochura, Tamara Loktikova;
Investigation: Vitaly Kochura, Nadia Kushnir;
Resources: Vitaly Kochura;
Data curation: Vitaly Kochura;
Writing – original draft preparation: Vitaly Kochura;
Writing – review and editing: Tamara Loktikova, Nadia Kushnir;
Visualization: Vitaly Kochura;
Supervision: Tamara Loktikova;
Project administration: Tamara Loktikova;
Funding acquisition: Nadia Kushnir.

All authors have read and agreed to the published version of the manuscript.

Declaration on the use of generative artificial intelligence tools

During the preparation of this scientific work, the authors used generative artificial intelligence tools, including ChatGPT, Gemini, and DeepL Translate, exclusively for supportive purposes.

These tools were not used for generating scientific content, research results, or conclusions, but solely for language editing, improving stylistic clarity, paraphrasing individual formulations, grammar and spelling checks, and enhancing the overall readability of the text.

After using each of these tools, the authors carefully reviewed and edited the corresponding parts of the manuscript and take full responsibility for the content, accuracy, and scientific integrity of the published work.

Conclusions

As a result of the conducted research, an analytical web service was developed that enables automated blockchain data processing, address clustering, and subsequent classification using machine learning models. The proposed methodology combines heuristic clustering rules for UTXO-based networks with algorithmic analysis of structural and behavioral cluster characteristics, enabling the creation of a tool for assessing their potential risk. Experimental results confirmed the effectiveness of the applied models, particularly XGBoost, which demonstrated the best classification performance.

Promising directions for further research include extending clustering methods through the integration of additional heuristics, advancing visualization tools, and adapting the system for other cryptocurrencies that use the UTXO model. Collectively, these enhancements will contribute to improving the accuracy, scalability, and practical value of the proposed service.

References

1. What Is Blockchain and How Does It Work? *Binance Academy*. URL: https://academy.binance.com/en/articles/what-is-blockchain-and-how-does-it-work (date of access: 25.11.2025).
2. What is a Bitcoin unspent transaction output (UTXO)? *Kraken*. URL: https://www.kraken.com/learn/what-is-bitcoin-unspent-transaction-output-utxo (date of access: 25.11.2025).
3. What Is Transaction Clustering in Crypto? Address Analysis. *Nansen AI*. URL: https://www.nansen.ai/post/what-is-transaction-clustering-in-crypto-address-analysis (date of access: 25.11.2025).
4. Wang M., Ichijo H., Xiao B. Cryptocurrency Address Clustering and Labeling. *arXiv*. 2020. DOI: https://doi.org/10.48550/arXiv.2003.13399.
5. Meiklejohn S., Pomarole M., Jordan G., Levchenko K., McCoy D., Voelker G. M., Savage S. A Fistful of Bitcoins: Characterizing Payments Among Men with No Names. *IMC '13: Proceedings of the 2013 conference on Internet measurement conference*. 2013. P. 127–140. DOI: http://dx.doi.org/10.1145/2504730.2504747.
6. Elliptic. URL: https://www.elliptic.co/ (date of access: 25.11.2025).
7. Androulaki E., Karame G. O., Roeschlin M., Scherer T., Capkun S. Evaluating User Privacy in Bitcoin. *Financial Cryptography and Data Security*. 2013. Vol. 7859. P. 34–51. DOI: https://doi.org/10.1007/978-3-642-39884-1_4.
8. Crystal Blockchain. URL: https://crystalintelligence.com/ (date of access: 25.11.2025).
9. CheckCryptoAddress. URL: https://checkcryptoaddress.com/ (date of access: 25.11.2025).
10. Spagnuolo M., Maggi F., Zanero S. Bitlodine: Extracting Intelligence from the Bitcoin Network. *Financial Cryptography and Data Security*. 2014. Vol. 8437. P. 457–468. DOI: https://doi.org/10.1007/978-3-662-45472-5_29.
11. Liu X. F., Jiang X.-J., Liu S.-H., Tse C. K. Knowledge Discovery in Cryptocurrency Transactions: A Survey. *IEEE Access*. 2021. Vol. 9. P. 37229–37254. DOI: https://doi.org/10.1109/ACCESS.2021.3062652.
12. Yin H. H. S., Langenheldt K., Harlev M., Mukkamala R. R., Vatrapu R. Regulating Cryptocurrencies: A Supervised Machine Learning Approach to De-Anonymizing the Bitcoin Blockchain. *Journal of Management Information Systems*. 2019. Vol. 36, № 1. P. 37–73. DOI: https://doi.org/10.1080/07421222.2018.1550550.
13. Zhang Y., Wang J., Luo J. Heuristic-Based Address Clustering in Bitcoin. *IEEE Access*. 2020. Vol. 8. P. 210581–210589. DOI: https://doi.org/10.1109/ACCESS.2020.3039570.
14. He X., He K., Lin S., Yang J., Mao H. Bitcoin address clustering method based on multiple heuristic conditions. *IET Blockchain*. 2022. Vol. 2, № 2. P. 44–56. DOI: https://doi.org/10.1049/blc2.12014.
15. Caringella M., Violante F., De Lucci F., Galantucci S., Costantini M. BACH: A Tool for Analyzing Blockchain Transactions Using Address Clustering Heuristics. *Information*. 2024. Vol. 15, № 10. P. 589. DOI: https://doi.org/10.3390/info15100589.
16. Introduction to Bitcoin Heuristics. *CryptoQuant Team*. URL: https://medium.com/cryptoquant/introduction-to-bitcoin-heuristics-487c298fb95b (date of access: 25.11.2025).
17. Schnoering H., Porthaux P., Vazirgiannis M. Assessing the Efficacy of Heuristic-Based Address Clustering for Bitcoin. *arXiv*. 2024. DOI: https://doi.org/10.48550/arXiv.2403.00523.
18. Understanding the Random Forest Algorithm – A Comprehensive Guide. *Data Science Dojo*. URL: https://datasciencedojo.com/blog/random-forest-algorithm/ (date of access: 25.11.2025).
19. Extra Trees, Explained: A Visual Guide with Code Examples. *Medium*. URL: https://medium.com/@samybaladram/extra-trees-explained-a-visual-guide-with-code-examples-4c2967cedc75 (date of access: 25.11.2025).
20. What is XGBoost? *IBM*. URL: https://www.ibm.com/think/topics/xgboost (date of access: 25.11.2025).
21. Symfony Documentation. URL: https://symfony.com/ (date of access: 25.11.2025).
22. React Documentation. URL: https://react.dev/ (date of access: 25.11.2025).
23. Schnoering H., Vazirgiannis M. Bitcoin research with a transaction graph dataset. *Scientific Data*. 2025. Vol. 12, Article 404. DOI: https://doi.org/10.1038/s41597-025-04684-8

Vitaly Kochura **Віталій Кочура**	master's student, Zhytomyr Polytechnic State University, Zhytomyr, Ukraine, e-mail: kochuravitaly@gmail.com. https://orcid.org/0009-0002-9598-7380	магістрант, Державний університет «Житомирська політехніка», м. Житомир, Україна.
Tamara Loktikova **Тамара Локтікова**	senior lecturer, Department of Software Engineering, Zhytomyr Polytechnic State University, Zhytomyr, Ukraine, e-mail: dfikt_ltn@ztu.edu.ua. https://orcid.org/0000-0002-3525-0179	старший викладач кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», м. Житомир, Україна.
Nadia Kushnir **Надія Кушнір**	senior lecturer, Department of Software Engineering, Zhytomyr Polytechnic State University, Zhytomyr, Ukraine, e-mail: kipz_kno@ztu.edu.ua. https://orcid.org/0000-0002-0797-3687	старший викладач кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», м. Житомир, Україна.