Published by
NIGERIAN SOCIETY OF PHYSICAL SCIENCES
Available online @ https://journal.nsps.org.ng/index.php/jnsps

J. Nig. Soc. Phys. Sci. 7 (2025) 2177

Journal of the Nigerian Society of Physical Sciences

A machine learning sentiment classification of factors that shape trust in smart contracts

Unyime Ufok Ibekwe[a,*], Uche M. Mbanaso[b], Nwojo Agwu Nnanna[a], Umar Adam Ibrahim[c]

[a]*Department of Computer Science, Nile University of Nigeria, Abuja, Nigeria.*
[b]*Centre for Cyberspace Studies, Nasarawa State University, Keffi, Nigeria*
[c]*Software Engineering and Information Technology Department, Nile University of Nigeria, Abuja, Nigeria.*

Abstract

Smart contracts have attracted significant attention within the blockchain ecosystem due to their ability to automate agreements when specific pre-defined conditions are met. However, concerns about the reliability of smart contracts persist due to potential vulnerabilities and unexpected outcomes. This study seeks to examine the perception of various stakeholders in the blockchain community, including developers, regulators, investors, researchers, auditors, and enthusiasts to understand the factors that influence trust in smart contracts. Data was gathered from 213 respondents through a survey administered across two blockchain communities. The responses were analyzed to identify key factors shaping trust in smart contracts within the blockchain space. The study identified five critical factors that significantly affect trust perceptions: Perceived Security Measures (PSM), Perceived Design Practices and Developer's Reputation (PDR), User Experience (UX), Perceived Social and Psychological Influence (PSP), and Perceived Regulatory Compliance and Continuous Improvement (PRC). Additionally, machine learning algorithms namely Support Vector Machine, Decision Tree, Logistic Regression, and Naive Bayes were applied on open-ended responses to conduct sentiment analysis, providing deeper insights into the perceptions of trust in blockchain smart contracts. The results revealed that Logistic Regression classifier outperformed the other models in analyzing trust levels in smart contracts.

DOI:10.46481/jnsps.2025.2177

Keywords: Blockchain, Cybersecurity, Machine learning, Smart contracts, Sentiment analysis, Trust

Article History :
Received: 06 June 2024
Received in revised form: 23 October 2024
Accepted for publication: 24 October 2024

Available online: 03 November 2024

Communicated by: O. Akande

1. Introduction

The blockchain network is a decentralized system made up of peer-to-peer nodes [1]. It is often referred to as a trustless technology due to its decentralized, transparent, and immutable characteristics. This foundational framework enables trustless interactions, contributing to its widespread adoption across various sectors such as finance, healthcare, Decentralized Finance (DeFi), supply chain management, education, and more. A key element of most blockchain networks are smart contracts, self-executing codes with the terms of the agreement embedded within them. Smart contracts automate and streamline processes thereby eliminating the need for intermediaries [2], enhancing efficiency, reducing costs, and promoting transparency.

Despite their potential to transform multiple industries, concerns persist regarding the reliability of smart contracts and

*Corresponding author: Tel.: +234-701-378-0159.
Email address: keenconsole@gmail.com (Unyime Ufok Ibekwe)

the overall user experience with this technology. These concerns stem from the vulnerabilities and potential for exploitation that can result in financial losses, legal issues, and reputational damage. A notable example is the 2016 Decentralized Autonomous Organization (DAO) hack, where a hacker exploited a reentrancy bug in the DAO's crowdfunding smart contract, recursively invoking its payout function [3] and draining over 60 million US dollars worth of cryptocurrency [4]. Unfortunately, such incidents are not rare, as security flaws in smart contracts are frequently uncovered and exploited, raising significant doubts about their trustworthiness.

Though security, trust, privacy, transparency, consensus, immutability, and decentralization [5] are core attributes of blockchain technology, how these features are applied, accepted and how they influence user perceptions of trust in smart contracts remain underexplored. It is crucial to understand how users perceive these characteristics and their implications, as well as how various perceptual factors shape trust in smart contracts.

Therefore, this study aims to conduct a comprehensive investigation into the diverse factors influencing trust in smart contracts within the blockchain ecosystem. By utilizing a combination of survey methods and machine learning techniques, the research seeks to explore the complex dynamics that affect trust. The study engages a wide range of stakeholders from two blockchain communities, including developers, cryptocurrency traders, researchers, auditors, and enthusiasts to identify the underlying determinants shaping perceptions of trust in smart contracts. Data was collected from 213 respondents across two blockchain communities in Nigeria, and statistical analysis was applied to evaluate their perceptions of trust.

Additionally, four machine learning classifiers: Logistic Regression, Decision Tree, Naive Bayes, and Support Vector Machine (SVM) were employed for sentiment analysis on the open-ended survey responses to gain deeper insights into perceptions of trust in blockchain smart contracts. The selection of these classifiers was influenced by several important factors, including their diverse approaches, the balance between performance and interpretability, their effectiveness with small to medium datasets, and their ability to accommodate different data characteristics. These classifiers align with established research practices in sentiment analysis, allowing the researchers to capture a wide range of sentiments expressed in the survey and offer a comprehensive understanding of the factors influencing trust in smart contracts within the blockchain ecosystem.

The findings of this research have significant implications for developers, users, regulators, and policymakers. They provide valuable insights for enhancing the reliability and security of smart contract implementations, thereby promoting their broader adoption. Furthermore, this study contributes to the ongoing dialogue about trust and accountability in blockchain technologies, facilitating informed decision-making and supporting the advancement of trust mechanisms within decentralized systems. The outcomes will guide the development of strategies and best practices aimed at improving the trustworthiness of smart contracts, encouraging their wider acceptance, and fostering the growth of blockchain applications across various industries.

2. Theoretical background

2.1. Overview of blockchain smart contracts

The decentralized and transparent nature of blockchain technology has revolutionized the way transactions and agreements are carried out. Functioning as a distributed ledger made up of interconnected blocks, each active node in the blockchain network holds a full copy of all transactions, ensuring both transparency and security. While blockchain was initially developed to facilitate financial transactions within the Bitcoin network [6], it has since gained widespread popularity and is now adopted across numerous sectors, including finance, healthcare, education, supply chain management, record keeping, identity management [7] and Central Bank Digital Currencies (CBDCs).

A key feature of most blockchain platforms are smart contracts, which are self-executing codes designed to automatically perform tasks once predefined conditions are met. Unlike traditional agreements that depend on trusted third parties and arbitration, smart contracts effectively remove the need for intermediaries. The concept of smart contract was first introduced by Nick Szabo in the early 1990s [8], to facilitate transactions and agreements without intermediaries. The aim was to reduce costs and increase transparency. By leveraging the immutability and decentralization of blockchain technology, smart contracts provide trust and transparency in fulfilling contractual obligations, enabling the automatic execution of terms between untrusted parties.

While smart contracts hold great promise, they are still vulnerable to security risks. They are especially vulnerable to attacks because once deployed on the blockchain, they cannot be modified [9]. This characteristic stems from the immutability and decentralized nature of blockchain technology. Vulnerabilities in the smart contract code can lead to unforeseen consequences that could damage the reputation of the blockchain platform, potentially resulting in substantial financial losses and reduced user trust. These challenges have sparked considerable interest in the community, prompting researchers to investigate ways to enhance the security and reliability of smart contracts.

For instance, Ref. [10] identified common programming errors and real-world attacks on Ethereum smart contracts, providing recommendations to mitigate these risks. Another study, [11] introduced a symbolic execution tool called OYENTE to identify vulnerabilities in smart contracts. Additionally, Ref. [12] proposed a transaction-based approach using Long Short-Term Memory (LSTM) to classify and detect vulnerabilities in smart contracts. Similarly, researchers in Ref. [13] developed a machine learning-based model to detect various types of smart contract vulnerabilities, such as front-running, bad randomness, reentrancy, arithmetic errors, access control issues, denial of service, and unchecked low-level calls. Collectively, these efforts underscore the importance of implementing robust security measures and practices to ensure the reliability of smart contracts.

2.2. Trust

The concept of trust is complex and can vary depending on the parties involved, the factors at play, and the specific context [14]. Trust is both dynamic and subjective, incorporating behavioral intentions and cognitive aspects. It plays a crucial role in social and economic interactions marked by dependence and uncertainty [15]. Trust is essential for facilitating successful transactions, whether in physical or digital environments.

In the field of Information Systems (IS) research, trust is understood as a collection of user perceptions regarding the attributes of technology [16]. It reflects the confidence and assurance individuals and organizations have in the reliability, authenticity and security of digital systems, platforms, and services. In any trust-based relationship, a distinction exists between the trustor, who grants trust, and the trustee, who receives it. Recent IS research regarding trust in technology, views the technological artifact as the trustee and the user as the trustor [17]. Trust becomes particularly important in environments where risk and uncertainty prevail, as the trustor accepts vulnerability to potential negative consequences stemming from the trustee's actions [18]. In this setting, the trustee must actively cultivate trust in the system. Trust in an Information Technology (IT) system is shaped by several key factors, including a thorough understanding of the trustee's operations, transparency in the processes used to generate system output, the accuracy of provided information, the system's reliability, and clear communication about system activities [17].

2.3. Trust in blockchain technology

Trust in the execution of blockchain smart contracts encompasses users' confidence in their reliability, security, and accuracy. It reflects the belief that smart contracts will function as intended, without errors, vulnerabilities, or malicious interference. However, like any software, smart contracts have demonstrated vulnerabilities and security flaws [19]. Notably, even well-known Ethereum smart contracts have been found to contain vulnerabilities. The consequences of such flaws are especially severe because smart contracts directly manage valuable assets and funds. Thus, threats to these contracts, can severely undermine the trust that individuals and institutions place in these digital platforms and services.

One of the primary advantages of the blockchain technology is its potential to foster trust. Often described as a means to create trust-free systems, blockchain technology aims to eliminate trust concerns by ensuring the validity of transactions [20]. As a result, smart contracts that utilize blockchain technology are expected to exhibit essential attributes such as trustlessness, decentralization, distributed technology, immutability, security, privacy, and a consensus mechanism. These components are vital for establishing a reliable and trustworthy delivery platform. Nonetheless, the actual implementation of these attributes has frequently fallen short of expectations.

The formation of trust among end users in blockchain-based platforms is hindered by several factors, which in turn limits the realization of the technology's potential benefits, impede its acceptance and usage. These factors include lack of experience and understanding of the technology, concerns about privacy, security heuristics, fairness, transparency, legal and financial accountability [21]. According to Ref. [22], perceived security and privacy are significant factors influencing trust in blockchain technology. Additionally, Ref. [23] acknowledged that a lack of understanding of the technology and technical competence can hinder trust in blockchain systems. Furthermore, Ref. [24] underscored how the complexity of blockchain technology, combined with the potential for financial losses, serves as a barrier to building trust.

While there has been substantial research regarding smart contract attacks and vulnerability detection techniques, a notable gap exists in the literature concerning the perception of trust in these contracts. Many studies have investigated various dimensions of trust in blockchain technology, but limited attention has been directed toward trust in smart contracts, which are a critical component of most blockchain platforms. Given that the level of trust in smart contracts is closely linked to users' perceptions and sentiments about the technology, this study aims to explore the factors that influence users' trust in smart contracts within the blockchain ecosystem, thereby addressing this significant gap.

3. Research methodology

To gain deeper insight into the factors influencing users' trust in smart contracts, this study utilizes an electronically distributed survey. The survey includes 38 items, organized into three sections: demographic details of the respondents, perceptions of smart contract trustworthiness measured on a five-point Likert scale ranging from 1 (strongly disagree) to 5 (strongly agree) and open-ended questions addressing trust in smart contracts. The items used to capture these theoretical constructs were developed based on a comprehensive review of existing literature.

3.1. Data collection

The study's population comprises a diverse group of stakeholders within the blockchain ecosystem, all possessing some degree of knowledge about blockchain and smart contracts. Data was gathered from 213 participants, including developers, traders, investors, regulators, researchers, auditors, and enthusiasts drawn from two distinct blockchain communities.

3.2. Data analysis

To analyze the survey data and identify patterns, we used Python's data analytics capabilities. We performed factor analysis on Likert-scale responses concerning user perceptions of smart contract trustworthiness. This statistical method reduces complex datasets to uncover the underlying dimensions that explain relationships among multiple variables [25].

To ensure the suitability of the dataset for factor analysis, two tests of sampling adequacy were conducted. The first test was Bartlett's test of sphericity, which examines the null hypothesis of a linear dependence between variables. The second test was the Kaiser-Meyer-Olkin (KMO) test, which measures the proportion of variance within the sample

dataset [26]. Bartlett's test yielded an extremely low value of 3.6317572870745657e-214 confirming the presence of linear dependence among the variables and justifying the analysis. Furthermore, the KMO test yielded a value of 0.80733, affirming the likelihood of satisfactory results from the factor analysis.

Using Python's 'factor analyzer' module, we employed a factor loading cutoff of 0.5 to group related information and then determined the optimal number of factors using a scree plot [27] as shown in Figure 1, thereby retaining only factors with eigenvalues exceeding 1.0. A heatmap as shown in Figure 2 visually represents these factors. Varimax rotation was then applied to obtain factor loadings, which were subsequently used to analyze overall trends in smart contract trust.

4. Factor analysis results

4.1. Demographic variables

Table 1 displays the results of our demographic data analysis.

4.2. Factor analysis

Using Python's 'factor analyzer' library, we performed a factor analysis of 20 variables, reducing them to five key factors influencing trust in smart contracts. Variables with loading values of 0.5 or higher were considered to have a significant impact on users' trust in smart contracts as detailed in Table 2. The top three factors impacting trust are: Perceived Security Measures (PSM1, loading 0.778699), Perceived Social and Psychological Influence (PSP1, loading 0.77836), and Perceived Design Practices and Developer's Reputation (PDR1, loading 0.751088). The other two factors are User Experience (UX1, loading 0.714375) and Perceived Regulatory Compliance and Continuous Improvement (PRC1, 0.63114).

4.2.1. Perceived security measures

Our survey found that a significant majority of respondents, 74 percent strongly agree that utilizing secure and reputable oracle boosts the trustworthiness of smart contracts, with another 16 percent also in agreement. Similarly, 95 percent of respondents recognized that comprehensive security audits improve the reliability of smart contracts. These audits are essential for identifying and mitigating vulnerabilities within systems and networks that could be exploited by unauthorized users or malicious actors [28].

Further analysis indicated a strong correlation in the perception of trust among various user groups, including developers, investors, enthusiasts, casual users, and auditors. This implies that all demographic groups view the implementation of security measures as a fundamental aspect of building trust in smart contracts. These results align with Ref. [29], which highlights the essential role of security in shaping practitioners' perceptions of smart contracts.

Given these insights, it is vital to engage third-party organizations to conduct thorough security audits of both smart contracts and external oracles prior to their deployment in production. Such actions will enhance user confidence by ensuring

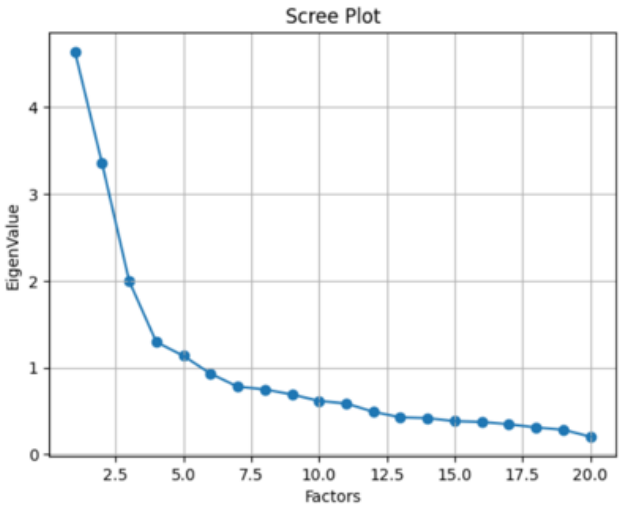

Figure 1: Scree plot diagram for factors analysis.

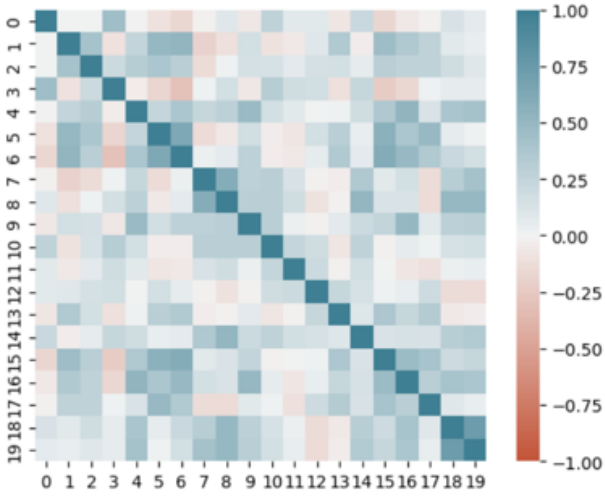

Figure 2: Heatmap for correlation between factors.

that the smart contracts have undergone independent assessments for potential vulnerabilities and risks.

4.2.2. Perceived design practices and developer's reputation

Our findings underscore the important role that a developer's reputation plays in establishing trust in smart contracts. A significant majority of respondents (22 percent strongly agree and 39 percent agree) believed that the credibility and reputation of the developer or project team directly impact the perceived trustworthiness of a smart contract. Additionally, a notable portion (16 percent strongly agree and 32 percent agree) recognized the risk of developers introducing malicious code, whether intentionally or unintentionally. Overall, these responses clearly indicate that a developer's reputation, whether positive or negative, is a key factor in shaping trust in smart contracts.

Users are more likely to trust smart contracts created by reputable and reliable developers or organizations. To build

Table 1: Descriptive information.

Variable	Description	Frequency (n)	Percentage (%)
Gender	Male	184	86
	Female	25	12
	Prefer not to say	4	2
Age	18 - 24	9	4
	25 - 34	78	37
	35 - 44	100	47
	45 - 54	24	11
	55 - 64	2	1
Educational Background	High School	2	1
	Bachelor's degree	90	42
	Master's degree	108	51
	Doctorate	13	6
How familiar are you with smart contracts?	Extremely familiar	11	5
	Very familiar	80	38
	Moderately familiar	68	32
	Somewhat familiar	46	22
	Not at all familiar	8	3
How familiar are you with blockchain technology?	Extremely familiar	23	11
	Very familiar	78	37
	Moderately familiar	78	37
	Somewhat familiar	33	15
	Not at all familiar	1	0
In what way are you involved in the blockchain /smart contract ecosystem?	Investor/ Cryptotraders	74	35
	Casual user (In IOT, Health, Education, Supply Chain, etc.)	44	21
	Enthusiast (Researcher, Policy Maker, Influencer, etc.)	43	20
	Developer	38	18
	Auditor	14	6
How frequently do you engage with smart contracts?	Daily	32	15
	Weekly	71	33
	Monthly	48	23
	Not too frequently	56	26
	Never	6	3

and maintain this trust, developers and organizations should prioritize transparency, uphold ethical standards, and maintain open communication with users. This includes honoring commitments, seeking developer certifications, quickly addressing any issues, and actively engaging with the community to gather feedback.

4.2.3. User experience

User experience plays a critical role in shaping trust in smart contracts. A significant portion of respondents (25 percent strongly agree and 37 percent agree) believed that code complexity influences their level of trust, while 64 percent agree that a user-friendly interface and ease of use are key factors in fostering trust. Interestingly, age did not significantly affect these perceptions, as no notable differences in trust were observed across different age groups. This suggests that devel-

oping intuitive, user-friendly interfaces can greatly enhance the user experience, promoting broader adoption and higher satisfaction.

Moreover, a significant majority (39 percent strongly agree and 45 percent agree) indicated that their awareness and understanding of smart contract functionalities directly impact their trust. This finding aligns with Ref. [30], which emphasizes that trust in smart contracts is closely tied to users' ability to comprehend, explain, and validate a contract's semantics. Thus, educating and raising awareness about smart contract technology is crucial.

Enhancing user experience through intuitive interfaces, alongside providing accessible resources, tutorials, workshops, and educational programs, will empower users to make informed decisions and strengthen trust in smart contract technology.

4.2.4. Perceived social and psychological influence

Social and psychological factors play a significant role in shaping trust in smart contracts. A notable 69 percent of respondents indicated that their past experiences affect their level of trust in these contracts. Additionally, 67 percent highlighted the importance of positive user feedback and ratings in influencing their trust. These findings emphasize the crucial impact of positive endorsements on trust perceptions.

Trust in smart contracts is also influenced by social norms, peer recommendations, and psychological factors like perceived risk and the trustworthiness of other parties. Users' risk perceptions, shaped by their past experiences, directly affect the trust they place in smart contracts. Negative experiences can lead to skepticism, while positive experiences foster trust. To mitigate the effects of negative past experiences and build trust, developers should focus on resolving issues, maintaining open communication, and actively seeking and responding to user feedback. By engaging with users and incorporating their feedback, developers can make improvements and effectively communicate these changes, demonstrating their commitment to addressing concerns and enhancing trust in smart contracts.

4.2.5. Perceived regulatory compliance and continuous improvement

Regulatory compliance and adherence to legal standards play a key role in building trust in smart contracts, with 32 percent of respondents strongly agreeing and 44 percent agreeing with this perspective. This involves accountability to stakeholders, adherence to legal requirements, creation of robust security policies, clearly defining roles and responsibilities within the ecosystem, and establishing standards and frameworks for smart contract development and best practices [28]. Integrating smart contracts into legal frameworks ensures that agreements are legally enforceable.

Furthermore, 56 percent of respondents viewed regular updates and maintenance as essential to maintaining trust in smart contracts. Overall, trust in smart contracts is enhanced by legal compliance and consistent updates, as well as standardized frameworks for development, deployment, and auditing. By implementing these measures, trust in smart contracts can be cultivated and reinforced.

5. Sentiment analysis process

User feedback and comments often provide an authentic insight into their opinions and are crucial for assessing services or products [31]. Sentiment analysis involves the extraction and interpretation of these expressed views to better understand users' emotions and attitudes [32]. This study employs sentiment analysis of user comments to further investigate trust in smart contracts. Four machine learning algorithms were employed for this process. Figure 3 outlines the steps, which includes data collection, preprocessing, sentiment extraction, feature extraction, modeling the machine learning classifiers (MLCs), and evaluating the models.

Python was chosen as the programming language for this study due to its extensive library support and user-friendliness. It is highly esteemed in data science and machine learning, primarily because of its robust library ecosystem. Python offers a variety of tools for tackling different machine learning tasks. For this research, the Scikit-learn package was utilized, as it provides a comprehensive range of supervised machine learning algorithms [33], including classification methods like Support Vector Machines (SVM), Logistic Regression, Decision Trees, and Naive Bayes.

5.1. Data collection

The dataset used for the sentiment analysis was derived from the responses to the open-ended questions in the survey. The survey consists of three sections: demographic information, Likert-scale questions concerning the factors influencing trust in smart contracts, and open-ended questions regarding users' perceived trust in blockchain smart contracts. Initially, demographic data and Likert-scale responses were extracted to perform factor analysis, aimed at identifying key factors influencing trust. Subsequently, the text from the open-ended responses was collected to create the dataset for sentiment analysis. In total, 213 responses were gathered, totaling over 10,500 words, all focused on the trustworthiness of smart contracts.

5.2. Data preprocessing

Given that machines cannot inherently comprehend written natural language, the extracted text was preprocessed to ensure data quality and consistency before training the machine learning models. Preprocessing is a vital step in preparing text for classification, as it eliminates noisy characters and words that could adversely affect the results. The assumption is that reducing noise in the text enhances the performance of the classifier. In this study, several preprocessing steps were implemented, including removing unnecessary white spaces, converting text to lowercase to eliminate case sensitivity, removing punctuation and special characters, eliminating stop words, and tokenizing the text into individual words or tokens.

Table 2: Factor analysis result.

Factors	Variables	Factor loading
Perceived Design Practices and Reputation of Developers	[PDR1] The type of programming language for developing smart contracts can impact their reliability, including security, correctness, and robustness, which in turn can influence trust levels.	0.751088
	[PDR2] The inclusion of a dispute resolution mechanism within a smart contract enhances its trustworthiness.	0.695496
	[PDR3] The distribution of authority and control across multiple parties in a smart contract design enhances the reliability of smart contracts.	0.623492
	[PDR4] The trustworthiness of a smart contract can be influenced by the reputation and credibility of the developers or project team associated with it.	0.601301
	[PDR5] There is a possibility that the development team may wittingly or unwittingly introduce malicious code into a smart contract.	0.552374
User Experience	[UX1] The complexity of a smart contract code affects my perception of its trustworthiness.	0.714375
	[UX2] The user-friendliness and ease of use of a smart contract influence its trustworthiness.	0.688086
	[UX3] The level of awareness and comprehension regarding the functioning of a smart contract directly impacts the trust placed in its functionality.	0.577537
Perceived Social and Psychological Influence	[PSP1] I am willing to invest more trust in a smart contract if I have prior positive experiences with similar contracts.	0.77836
	[PSP2] Positive feedback and ratings provided by other users significantly contribute to my perception of the trustworthiness of a smart contract.	0.697256
Perceived Regulatory Compliance and Continuous Improvement	[PRC1] Ensuring regular updates and maintenance of a smart contract contributes to its trustworthiness.	0.63114
	[PRC2] Ensuring legal compliance and adherence to regulatory standards during the development process enhances the trustworthiness of a smart contract.	0.617014
Perceived Security Measures	[PSM1] The use of secure and reputable oracles for fetching external data enhances the trustworthiness of smart contracts.	0.778699
	[PSM2] Smart contracts that have undergone comprehensive security audits are more trustworthy.	0.523572

5.3. Exploratory data analysis

Exploratory data analysis was performed on the preprocessed data to identify patterns.

5.3.1. Sentiment extraction

The exploratory analysis categorized comments as positive, negative, or neutral to determine sentiment scores. The results presented in Figure 4 revealed that 81.7 percent of the comments conveyed positive sentiment, 11.7 percent were negative, and 6.6 percent were neutral regarding the reliability of smart contracts. Overall, there was a predominance of positive sentiment towards smart contracts.

5.3.2. Word cloud

The Python Word Cloud library was employed to identify the most frequently used words in the dataset, providing valuable insights during the feature extraction phase. The frequency count of these terms plays a key role in supporting the analysis. The generated word cloud, created from the entire dataset, emphasizes the most commonly occurring words across all sections. Notable terms like ”contract audit,” ”user awareness,”

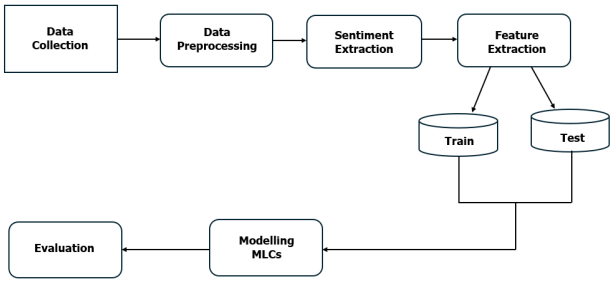

Figure 3: Sentiment analysis process.

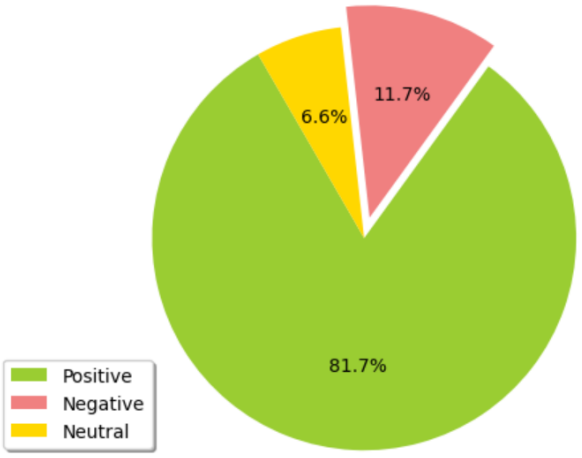

Figure 4: Distribution of sentiment categories.

Figure 5: Most commonly used words among respondents.

and "security control" were among the most frequently mentioned, as shown in Figure 5.

5.3.3. Data re-sampling

The exploratory data analysis revealed a notable class imbalance within the dataset. Class imbalance occurs when classes are distributed unevenly, which can negatively impact the performance of machine learning models [34]. To tackle this issue, various techniques can be employed, such as over-sampling and under-sampling. Over-sampling increases the number of samples in the minority class, while under-sampling decreases the number of samples in the majority class to create a more balanced dataset.

In this study, we opted not to use under-sampling due to the small size of the minority classes; further reduction would adversely affect classification performance. Instead, we employed the Synthetic Minority Oversampling Technique (SMOTE) to oversample the minority class, resulting in a balanced dataset. This technique enhances the representation of the minority class to better align with the majority class, thereby addressing the class imbalance issue. Figure 6 illustrates the dataset classes before and after the re-sampling process.

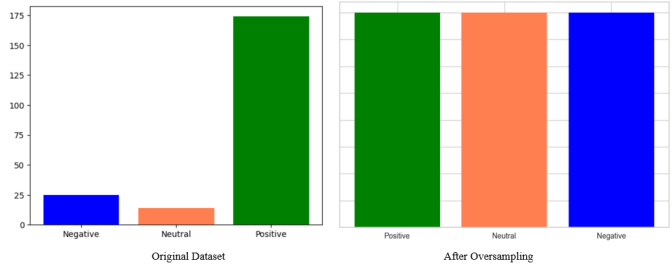

Figure 6: Dataset before and after re-sampling.

5.4. Feature extraction

The dataset was divided into an 80:20 ratio, with 80 percent used for training and the remaining 20 percent set aside for testing. This distribution enables a substantial amount of data to be utilized for model training while still ensuring enough data for evaluation. Both the training and testing datasets were vectorized using Term Frequency-Inverse Document Frequency (TF-IDF) to transform the text into numerical features. By applying these techniques, irrelevant words are eliminated, and the text is converted into numerical representations.

5.5. Model training

Four machine learning classifiers: Logistic regression (LR), Naive Bayes (NB), Support Vector Machine (SVM), and Decision Tree (DT) were used to train the dataset.

5.5.1. Logistic regression

A logistic regression model is a supervised learning algorithm designed for classification tasks. It is typically used when the dependent or target variable is categorical. This model predicts the probability of a categorical outcome by employing a logistic function [35].

5.5.2. Support vector machine

A Support Vector Machine (SVM) model is a supervised machine learning algorithm used for both classification and regression tasks. It works by finding the optimal hyperplane that separates data points belonging to different classes in a high-dimensional space [36]. The goal of SVM is to maximize the margin between classes, creating a robust decision boundary.

5.5.3. Naive Bayes

Naive Bayes (NB) is founded on Bayes' theorem, which determines the probability of a class's presence based on a specific text or input. The Naive Bayes classifier assumes that the value of each predictor independently affects the class, regardless of the values of other predictors [37]. There are several variations of Naive Bayes, including MultinomialNB, BernoulliNB, CategoricalNB, ComplementNB, and GaussianNB. In this study, we used MultinomialNB, as it is particularly effective for text classification tasks.

5.5.4. Decision tree

The decision tree (DT) algorithm is structured like a tree, where each internal node represents a feature or attribute. Each branch signifies a decision made based on a specific feature, and each leaf node indicates the associated outcome or prediction. The decision tree model includes three types of nodes: root node, internal node, and leaf node, which are essential components for the analysis performed by the decision tree [38]. It functions by recursively dividing the data into subsets according to the features that most effectively distinguish the target variable.

5.6. Model evaluation

Model performance was evaluated using accuracy, recall, precision, and F1-score as detailed in equations (1), (2), (3), and (4), respectively. In these metrics, True Positive is represented as TP, False Positive as FP, False Negative as FN, and True Negative as TN. Furthermore, we evaluated the models' performance using the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC). The ROC curve describes the classification model's performance across different classification thresholds by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR), which are calculated using equations (5) and (6), respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}. \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FP}. \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{3}$$

$$F1 - \text{Score} = \frac{2 \times (\text{Precision} \times \text{recall})}{\text{recall} + \text{precision}}. \tag{4}$$

$$TPR = \frac{TP}{(TP + FN)}. \tag{5}$$

$$FPR = \frac{FP}{FP + TN}. \tag{6}$$

6. Experimental results and discussion

Tables 3, 4, 5, and 6 present the performance of Logistic Regression, SVM, Naive Bayes and Decision Tree classifiers on both balanced and imbalanced datasets. Although the imbalanced dataset achieved higher accuracy, a deeper analysis of precision, recall, and F1-scores reveals a bias toward the positive class, leading to poorer performance for the negative and neutral classes. This highlights the limitations of using accuracy alone, especially with imbalanced data.

In contrast, the balanced dataset produced more consistent precision, recall, and F1 scores across all classes, as shown in Table 7. While SVM and Naive Bayes achieved higher accuracy (0.77 and 0.84 respectively), Logistic Regression demonstrated superior performance as measured by AUC, which is less sensitive to class imbalance than accuracy alone. This is

Table 3: LR classification report.

	Imbalanced data		Balanced data	
	Recall	F1-score	Recall	F1-score
Negative	0.00	0.00	0.00	0.00
Neutral	0.00	0.00	0.33	0.33
Positive	1.00	0.91	0.86	0.86
Accuracy	0.84		0.74	

Table 4: SVM classification report.

	Imbalanced data		Balanced data	
	Recall	F1-score	Recall	F1-score
Negative	0.00	0.00	0.00	0.00
Neutral	0.00	0.00	0.33	0.50
Positive	1.00	0.94	0.89	0.88
Accuracy	0.89		0.77	

Table 5: NB classification report.

	Imbalanced Data		Balanced Data	
	Recall	F1-score	Recall	F1-score
Negative	0.00	0.00	0.00	0.00
Neutral	0.00	0.00	0.00	0.00
Positive	1.00	0.91	1.00	0.99
Accuracy	0.84		0.84	

Table 6: DT classification report.

	Imbalanced Data		Balanced Data	
	Recall	F1-score	Recall	F1-score
Negative	0.00	0.00	0.50	0.25
Neutral	0.00	0.00	0.67	0.40
Positive	0.92	0.88	0.61	0.73
Accuracy	0.77		0.60	

Table 7: Result of balanced dataset.

	Accuracy	Precision	Recall	F1-score	AUC
SVM	0.77	0.79	0.77	0.77	0.75
LR	0.74	0.76	0.74	0.74	0.83
NB	0.84	0.70	0.84	0.84	0.68
DT	0.60	0.88	0.60	0.77	0.62

because the AUC considers the trade-off between true positive and false positive rates and is less influenced by class imbalance.

Additionally, Logistic Regression is recognized for its strong performance on smaller datasets, which may apply to the data in our study. While it recorded a lower accuracy compared to Naive Bayes and SVM, its AUC score of 0.83 indicates a robust ability to differentiate between classes. This suggests that Logistic Regression effectively captures the nuances of trust sentiments in smart contracts. Our findings are consistent with existing literature that underscores the advantages of Logistic Regression in scenarios where interpretability and performance on imbalanced datasets are essential. For instance, studies like [39] have reported similar outcomes, demonstrating Logistic Regression's effectiveness in text-based sentiment classification tasks, particularly with relatively small or imbalanced datasets.

7. Conclusion and future research direction

This study examined trust in smart contracts among 213 members of two blockchain communities, identifying five key trust drivers: Perceived Security Measures (PSM), Perceived Design Practices and Developer's Reputation (PDR), User Experience (UX), Perceived Social and Psychological Influence (PSP), and Perceived Regulatory Compliance and Continuous Improvement (PRC). These findings highlight the complex nature of trust, emphasizing the importance of both technical and social factors in fostering confidence in smart contracts.

A central aspect of this research was the application of machine learning algorithms for sentiment analysis, which offered deeper insights into stakeholder perceptions. Notably, the Logistic Regression classifier was found to be satisfactory in classifying and predicting trust levels in smart contracts. This analytical approach not only enhanced our understanding of the factors influencing trust but also showcased the potential of machine learning techniques in processing qualitative data related to sentiment and perceptions.

The significance of this study lies in its contribution to the growing body of knowledge on trust within blockchain technologies, particularly regarding smart contracts. By identifying the key factors influencing trust and providing a method for sentiment analysis, this research offers practical insights for developers, regulators, and other stakeholders seeking to enhance the reliability and adoption of smart contracts. The findings can also inform the design of more secure and user-friendly smart contract systems, as well as regulatory frameworks aimed at strengthening trust in blockchain technologies.

Nevertheless, our study has its limitations. Although machine learning algorithms demonstrate significant predictive capabilities, their effectiveness is constrained by the quality and quantity of the input data. Therefore, this research should be viewed as a foundational step for further investigation into the multifaceted factors that affect trust in smart contracts within the blockchain ecosystem.

Looking ahead, future research could expand the dataset to include a broader range of stakeholders and investigate trust dynamics across different blockchain platforms. Additionally, employing more advanced machine learning models or deep learning techniques may yield deeper insights into sentiment analysis, potentially improving the accuracy of trust predictions. Further studies could also explore how evolving regulatory environments and technological advancements impact trust in smart contracts, ensuring that the findings remain relevant in the context of ongoing innovations in the blockchain space.

Ultimately, understanding and addressing trust in smart contracts, supported by sentiment analysis, will be crucial for their successful integration into mainstream applications. This could unlock opportunities for innovation and efficiency across multiple sectors.

Data availability

The data associated with this work can be obtained from https://osf.io/wtv5f/?view_only=9bf75a95462a4f3293c1f9d29240e8e9.

References

[1] A. E. Ibor, E. B. Edim & A. A. Ojugo, "Secure health information system with blockchain technology", Journal of the Nigerian Society of Physical Sciences **5** (2023) 992.https://doi.org/10.46481/jnsps.2023.992.

[2] S. Siddamsetti & M. Srivenkatesh, "Efficient fraud detection in ethereum blockchain through machine learning and deep learning approaches", International Journal on Recent and Innovation Trends in Computing and Communication **11** (2023) 71 . https://doi.org/10.17762/ijritcc.v11i11s.8072.

[3] C. F. Torres, M. Baden, R. Norvill, B. B. F. Pontiveros, H. Jonker & S. Mauw, *ÆGIS: shielding vulnerable smart contracts against attacks*, Proceedings of the 15th ACM Asia Conference on Computer and Communications Security, 2020, pp. 584-597. https://doi.org/10.1145/3320269.3384756.

[4] W. Wang, J. Song, G. Xu, Y. Li, H. Wang & C. Su, "ContractWard: automated vulnerability detection models for ethereum smart contracts", IEEE Trans Netw Sci Eng **8** (2021) 1133. https://doi.org/10.1109/TNSE.2020.2968505.

[5] D. Kundu, "Blockchain and trust in a smart city", Environment and urbanization Asia **10** (2019) 31. https://doi.org/10.1177/0975425319832392.

[6] S. Amani, M. Bortin, M. Bégel & M. Staples, *Towards verifying Ethereum smart contract bytecode in Isabelle/HOL*, CPP 2018 - Proceedings of the 7th ACM SIGPLAN International Conference on Certified Programs and Proofs, Co-located with POPL, Association for Computing Machinery, Inc, 2018, pp. 66-77. https://doi.org/10.1145/3167084.

[7] T. Abdellatif & K. L. Brousmiche, *Formal verification of smart contracts based on users and blockchain behaviours models*, 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), IEEE, February 2018, pp. 1-5. https://doi.org/10.1109/NTMS.2018.8328737.

[8] S. Qian, H. Ning, Y. He & M. Chen, "Multi-Label vulnerability detection of smart contracts based on bi-lstm and attention mechanism", Electronics (Switzerland) **11** (2022) 3260. https://doi.org/10.3390/electronics11193260.

[9] C. F. Torres, J. Schütte & R. State, *Osiris: hunting for integer bugs in ethereum smart contracts*, 34th ACM International Conference Proceeding Series, Association for Computing Machinery, 2018, pp. 664–676. https://doi.org/10.1145/3274694.3274737.

[10] N. Atzei, M. Bartoletti & T. Cimoli, *A survey of attacks on Ethereum smart contracts*, Principles of Security and Trust: 6th International Conference, POST 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, 2017, pp. 164-186. https://doi.org/10.1007/978-3-662-54455-6_8.

[11] L. Luu, D. H. Chu, H. Olickel, P. Saxena & A. Hobor, *Making smart contracts smarter*, Proceedings of the ACM Conference on Computer and Communications Security, Association for Computing Machinery, 2016, pp. 254-269. https://doi.org/10.1145/2976749.2978309.

[12] T. Hu, X. Liu, T. Chen, X. Zhang, X. Huang, W. Niu, J. Lu, K. Zhou & Y. Liu , "Transaction-based classification and detection approach for Ethereum smart contract", Information Processing and Management **58** (2021) 102462. https://doi.org/10.1016/j.ipm.2020.102462.

[13] Y. Xu, G. Hu, L. You & C. Cao, "A Novel machine learning-based analysis model for smart contract vulnerability", Security and Communication Networks **2021** (2021) 5798033. https://doi.org/10.1155/2021/5798033.

[14] N. B. Truong & G. M. Lee, "A reputation and knowledge-based trust service platform for trustworthy social internet of things". [Online]. Accessed: 9 March 2024. http://researchonline.ljmu.ac.uk/id/eprint/2599.

[15] T. S. H. Teo & J. Liu, "Consumer trust in e-commerce in the United States, Singapore and China", Omega (Westport) **35** (2007) 22. https://doi.org/10.1016/j.omega.2005.02.001.

[16] D. Pienta, H. Sun & J. B. Thatcher, "Habitual and misplaced trust: the role of the dark side of trust between individual users and cybersecurity systems research-in-progress". [Online]. Accessed: 23 February 2024. https://aisel.aisnet.org/icis2016/ISSecurity/Presentations/11.

[17] M. Söllner, A. Hoffmann, H. Hoffmann, A. Wacker & J. M. Leimeister, "Understanding the formation of trust", in *Socio-Technical Design of Ubiquitous Computing Systems*, Springer International Publishing, 2014. https://doi.org/10.1007/978-3-319-05044-7_3.

[18] C. Loebbecke, L. Lueneborg & D. Niederle , "Blockchain Technology Impacting the Role of Trust in Transactions: Reflections in the Case of Trading Diamonds", Research-in-Progress Papers **68** (2018) 1. https://aisel.aisnet.org/ecis2018_rip/68.

[19] K. Werbach, "Trust, but verify", Technology Law Journal **33** (2018) 487. https://doi.org/10.15779/Z38H41JM9N.

[20] M. Fleischmann & B. S. Ivens, "Exploring the role of trust in blockchain adoption: an inductive approach", 2019. [Online]. http://hdl.handle.net/10125/60120.

[21] D. Shin & W. T. Bianco, "In blockchain we trust: does blockchain itself generate trust?", Social Science Quarterly **101** (2020) 2522.https://doi.org/10.1111/ssqu.12917.

[22] D. D. H. Shin, "Blockchain: The emerging technology of digital trust", Telematics and Informatics **45** (2019) 101278. https://doi.org/10.1016/j.tele.2019.101278.

[23] V. Sadhya, H. Sadhya, R. Hirschheim & E. Watson, "Exploring technology trust in bitcoin: The blockchain exemplar", Research papers **5** (2018) 1. https://core.ac.uk/download/pdf/301378498.pdf.

[24] N. Ostern, "Do you trust a trust-free technology? Toward a trust framework model for blockchain technology", 2018. http://tubiblio.ulb.tu-darmstadt.de/110885.

[25] M. Tavakol & A. Wetzel, "Factor analysis: a means for theory and instrument development in support of construct validity", International Journal of medical education **11** (2020) 245. https://doi.org/10.5116%2Fijme.5f96.0f4a.

[26] N. Shrestha, "Factor analysis as a tool for survey analysis", American Journal of Applied Mathematics and Statistics **9** (2011) 4. https://doi.org/10.12691/ajams-9-1-2.

[27] R. D. Ledesma, P. Valero-Mora & G. Macbeth, "The scree test and the number of factors: a dynamic graphics approach", The Spanish Journal of psychology **18** (2015) E11. https://doi.org/10.1017/sjp.2015.13.

[28] A. Alshammari, "A novel security framework to mitigate and avoid unexpected security threats in saudi arabia", Engineering, Technology and Applied Science Research **13** (2023) 11445. https://doi.org/10.48084/etasr.6091.

[29] Z. Wan, X. Xia, D. Lo, J. Chen, X. Luo & X. Yang, *Smart contract security: a practitioners' perspective*, IEEE/ACM 43rd International Conference on Software Engineering (ICSE), 2021, pp. 1410-1422. https://doi.org/10.1109/ICSE43902.2021.00127.

[30] F. Al Khalil, T. Butler, L. O'Brien & M. Ceci, *Trust in smart contracts is a process, as well*, Financial Cryptography and Data Security: FC 2017 International Workshops, Malta, 2017, Revised Selected Papers 21, Springer International Publishing, pp. 510-519. https://doi.org/10.1007/978-3-319-70278-0_32.

[31] M. R. R. Rana, A. Nawaz, T. Ali, A. M. El-Sherbeeny & W. Ali, "A BiLSTM-CF and BiGRU-based Deep sentiment analysis model to explore customer reviews for effective recommendations", Engineering, Technology and Applied Science Research **13** (2023) 11739. https://doi.org/10.48084/etasr.6278.

[32] W. M. S. Yafooz, E. A. Hezzam & W. Alromema, "Arabic sentiment analysis on chewing khat leaves using machine learning and ensemble methods", Engineering, Technology and Applied Science Research **11** (2021) 6845. https://doi.org/10.48084/etasr.4026.

[33] J. Hao & T. K. Ho, "Machine learning made easy: a review of scikit-learn package in python programming language", Journal of Educational and Behavioral Statistics **44** (2019) 348. https://doi.org/10.3102/1076998619832248.

[34] H. Alamoudi, N. Aljojo, A. Munshi, A. Alghoson, A. Banjar, A. Tashkandi, A. Al-Tirawi & I. Alsaleh, "Arabic sentiment analysis for student evaluation using machine learning and the arabert transformer", Engineering, Technology and Applied Science Research **13** (2023) 11945. https://doi.org/10.48084/etasr.6347.

[35] S. Swamy & S. Hegde. "Exploring sentiment analysis in kannada language: a comprehensive study on COVID-19 data using machine learning and ensemble algorithms", International Journal of Intelligent Systems and Applications in Engineering **12** (2024) 21. https://www.ijisae.org/.

[36] T. Jain, V. K. Verma, A. K. Sharma, B. Saini, N. P. Bhavika, H. Mahdin, M. Ahmad, R. Darman, S. C. Haw, S. M. Shaharudin & M. S. Arshad, "Sentiment analysis on COVID-19 vaccine tweets using machine learning and deep learning algorithms", International Journal of Advanced Computer Science and Applications (IJACSA) **14** (2023) 5. http://dx.doi.org/10.14569/IJACSA.2023.0140504.

[37] M. Dandotiya, D. K. Bandil, K. Sankhla, P. Yadav & M. Kumari, "An implementation of computerized valuation of descriptive answers: a machine learning approach", International Journal on Recent and Innovation Trends in Computing and Communication **11** (2023) 40. https://www.ijritcc.org/.

[38] P. Parisha, G. K. Srivastava & S. Kumar, "Automated heart syndrome forecast model exploiting machine learning approaches", International Journal on Recent and Innovation Trends in Computing and Communication **11** (2023) 319. https://doi.org/10.17762/ijritcc.v11i11s.8158.

[39] M. A. Kausar, S. O. Fageeri & A. Soosaimanickam, "Sentiment classification based on machine learning approaches in Amazon product reviews", Engineering, Technology and Applied Science Research **13** (2023) 10849. https://doi.org/10.48084/etasr.5854.