Scientific Research Publishing

Journal of Information Security, 2025, 16(1), 184-196
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

Information Security, Ethics, and Integrity in LLM Agent Interaction

Ying-Jung Chen[1], Vijay K. Madisetti[2]

[1]College of Computing, Georgia Institute of Technology, Atlanta, GA, USA
[2]School of Cybersecurity & Privacy, Georgia Institute of Technology, Atlanta, GA, USA
Email: yingjungcd@gmail.com, vkm@gatech.edu

How to cite this paper: Chen, Y.-J. and Madisetti, V.K. (2025) Information Security, Ethics, and Integrity in LLM Agent Interaction. *Journal of Information Security*, **16**, 184-196.
https://doi.org/10.4236/jis.2025.161010

Received: December 17, 2024
Accepted: January 23, 2025
Published: January 26, 2025

Abstract

This study addresses security and ethical challenges in LLM-based Multi-Agent Systems, as exemplified in a blockchain fraud detection case study. Leveraging blockchain's secure architecture, the framework involves specialized LLM Agents—ContractMining, Investigative, Ethics, and PerformanceMonitor, coordinated by a ManagerAgent. Baseline LLM models achieved 30% accuracy with a threshold method and 94% accuracy with a random-forest method. The Claude 3.5-powered LLM system reached an accuracy of 92%. Ethical evaluations revealed biases, highlighting the need for fairness-focused refinements. Our approach aims to develop trustworthy and reliable networks of agents capable of functioning even in adversarial environments. To our knowledge, no existing systems employ ethical LLM agents specifically designed to detect fraud, making this a novel contribution. Future work will focus on refining ethical frameworks, scaling the system, and benchmarking it against traditional methods to establish a robust, adaptable, and ethically grounded solution for blockchain fraud detection.

Keywords

Multi LLM Agents Systems, Blockchain, Cooperative Interactions, Fraud Detection, Ethics and Safety

1. Introduction

As LLM multi-agent systems become more advanced, ensuring they remain aligned with human values is crucial. However, agent-agent dependencies in LLM multi-agent systems can lead to various security issues Figure 1. These include unintended information leakage, where sensitive data might be shared inappropriately between agents. The interconnected nature of these systems can result in

cascading failures and rapid propagation of attacks. To mitigate these risks, robust security measures and careful design of agent interactions are essential, including encryption, authentication, anomaly detection, and careful design of agent interactions and permissions. Typically, agents autonomously gather, process, and analyze large amounts of data in research, offering valuable insights without human intervention. They are also effective in real-time applications like financial trading, making split-second decisions based on market conditions. However, little has been done to examine agents' interaction framework design [1], which can potentially lead to data leakage, or misinformation. This will not only cause a security and privacy risk but also violate the ethical rules since humans tend to over-reliance and overly trust the output from LLM models [2]. The threat among agents of interaction can be categorized into two types: competitive interaction threats and cooperative interaction threats. Agent-to-agent competitive behavior may lead to untrustworthy interactions and ethical concerns, as seen with Meta's AI Cicero learning deception. Balancing effective competition and ethical behavior in AI agents remains a significant challenge requiring further research [1]. On the other hand, cooperative multi-LLM agent systems boost efficiency [3] but face threats like secret collusion leading to bias [4], amplification of errors or hallucinations, rapid spread of misinformation, and attacks exploiting agent connectivity. However, they also enhance security by defending against jailbreak attacks (as demonstrated by AutoDefense) and improving software security and accountability by detecting and controlling irreversible code execution [5]. For this reason, this study selected the cooperative framework with ethics rules to examine fraud detection.

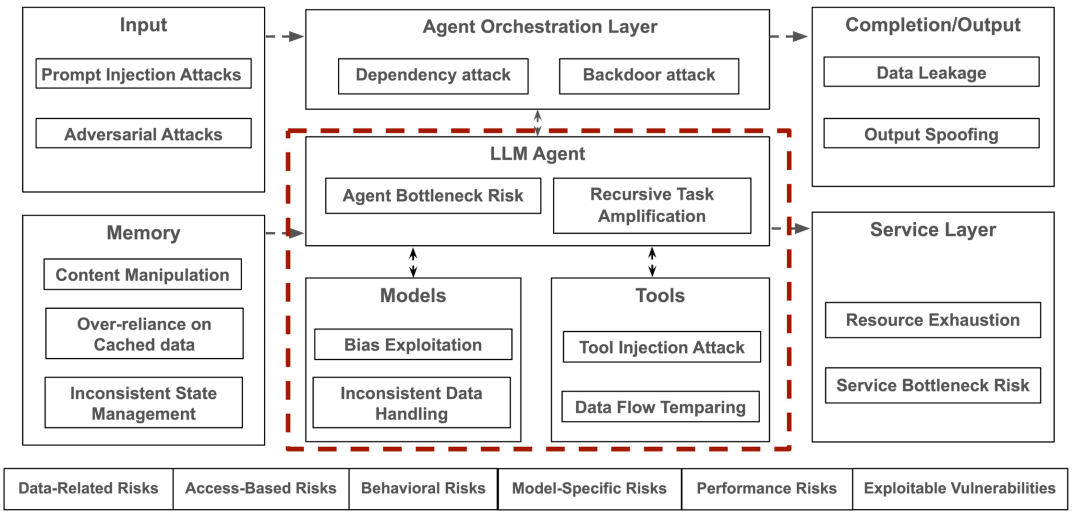

Figure 1. Security, risks, and vulnerabilities within LLM multi-agent systems (red dash line represents this study focus area).

Blockchain technology, with its decentralized structure, distributed ledgers, hash encryption, and consensus protocols, offers a solution to these issues by providing transparency, immutability, and secure verification mechanisms that

enhance trust, information integrity, and authority within multi-LLM agent systems. Smart contracts in blockchain techniques can automate processes, ensuring transaction authenticity and removing the need for a central authority. While integrating blockchain technology with multi-LLM agent systems can enhance trust, it also raises ethical concerns, particularly regarding privacy, the immutability of records, and the need for transparency and explainability in smart contracts [6] [7].

Thus, the main goals for this study are: 1) Investigate fraud within the cooperative multi-LLM agent systems, and 2) Design a series of ethics rules to avoid bias and collusion from the multi-LLM agent systems output based on smart contracts in blockchain datasets. To accomplish these goals, we will explore how cooperative agent interactions raise ethical concerns, such as bias. In addition, we will develop a series of ethical rules that enhance fairness. This proposed solution will balance efficiency and ethical behavior while addressing risks like fraud and data leakage among LLM agents.

2. Related Work

This work relates to three main areas of related research work, which we discuss in this section: 1) identify fraud based on blockchain technology, 2) multi-LLM agent system, and 3) ethical rules for reducing bias and collusion within multi-LLM agent systems.

2.1. Fraud Detection in Blockchain

Fraud detection in blockchain technology is an emerging research area, especially with the growing adoption of smart contracts on platforms like Ethereum. Blockchain's immutability poses challenges, as vulnerabilities in smart contracts cannot be updated. Notable studies include [8], who used graph neural networks to detect vulnerabilities in contract code, and [9], who developed a Heterogeneous Graph Transformer Network (S_HGTNs) to analyze transaction and code data for improved anomaly detection. Additionally, [10] proposed Deep Blockchain-Enabled Collaborative Anomaly Detection (DBC-CAD) to address IoT security challenges through blockchain and deep learning.

Recent advancements in multi-LLM agent systems often lack secure and decentralized transaction recording, a gap blockchain can address. However, abstract concepts like asset ownership remain challenging for blockchain alone. Integrating blockchain with multi-LLM agents and artificial institutions enhances transaction reliability by combining decentralized trust with consistent concept management [11]. To date, no such integration exists for using smart contracts and multi-LLM agents in fraud detection, which is a primary goal of this study.

2.2. Multi-LLM Agent System

However, recent advancements include the development of autonomous AI agents for fraud detection, such as: 1) Network agents: Monitor and respond to

potential threats in real-time. 2) Knowledge agents: Serve as repositories of information and assist in decision-making. 3) Root Cause Analysis (RCA) agents: Investigate causes of fraudulent activity surges. These autonomous agents can work together in an agent network, providing a more comprehensive and cooperative approach to fraud detection and risk management [12]. In addition, [13] utilizes a collaborative network of LLM based AI agents, each specialized in distinct tasks such as data conversion, expert analysis, cross-checking, and report generation to enhance anomaly detection in financial markets, specifically focusing on automating the validation process for anomaly alerts. Although these multi-LLM agent systems have been built in either a traditional competitive framework or a modern cooperative interaction framework, little has been examined regarding ethical behavior in the interaction process. For this reason, we will build a multi-LLM agent system under the modern cooperative framework for fraud detection cases, but with extra safety and ethics considerations by applying performance and ethics metrics.

2.3. Ethical Rules in Multi-Agent System

Multi-LLM agent systems can be prone to increase users' vulnerability to misinformation, as people may trust them too readily and rely on them as accurate sources. Additionally, these systems might deliver biased or partial information to align with user expectations, potentially reinforcing ideologies and undermining political debate. They may also degrade societal trust in shared knowledge by spreading large amounts of plausible but low-quality information. These issues can occur when AI agents are designed to prioritize their users' interests. A key issue is within the cooperative multi-LLM agent systems, where the best outcomes occur when all AI agents cooperate. However, individual AI agents may choose to act selfishly, benefiting their users while others cooperate, leading to worse overall outcomes [14].

Thus, it is critical to implement ethical rules, such as fairness and bias metrics, as well as transparency. Previous studies have addressed model framework fairness and how bias can emerge from data used to train AI systems, as well as in their outputs. For example, work by [15] on facial recognition audits points out challenges in mitigating bias in real-world applications. Also, [16] indicated that transparency is a critical factor in ensuring accountability in AI systems. Their work explores how a lack of transparency can erode public trust and emphasizes the need for AI systems to provide clear and interpretable outputs. Finally, [14] discussed how AI behavior impacts society, particularly in unintended harm or misuse scenarios. According to these approaches, this study will implement fairness, bias, and transparency within multi-LLM agent systems.

3. Data and Methods

This study presents a new form of agent interaction architecture to manage the risk of fraud and ethics based on the blockchain database. The first set of data used

in this project was collected in a manner like [17] gathered their data. This work generates a labeled dataset of malicious contracts by fetching transaction data from Etherscan.io using its API, computing detailed address-specific features (e.g., transaction counts, Ether balances, contract interactions), and creating balanced datasets with specified proportions of hacked and non-hacked addresses. We first designed a sample agent, including the sampling function, to collect the malicious contract dataset. The dataset used in this study includes key features such as the number of transactions (sent and received), Ether values (min, max, average), contract creation interactions, unique sender and receiver addresses, and time intervals between transactions. Our system is designed to utilize these common features for generalizability, ensuring it can be applied to a real-time Ethereum-based database.

To test the scalability of the proposed LLM agent framework, we leverage the second set of data: **Malicious Smart Contract Dataset**, available on Hugging Face, which provides a comprehensive resource for analyzing and identifying malicious behavior in blockchain smart contracts. This dataset includes labeled data categorized into benign and malicious contracts based on patterns of fraudulent activities, vulnerabilities, and exploitation techniques. It serves as a valuable tool for researchers and developers focused on blockchain security, enabling the development of machine learning models to detect and mitigate risks associated with smart contract vulnerabilities. By integrating this dataset with Forta's predictive machine learning models, which analyze vast amounts of on-chain data to detect anomalous patterns and predict attacks before exploitation, we aim to enhance blockchain security. This combination of advanced ML techniques and robust datasets fosters real-time risk mitigation, safeguarding decentralized applications and preventing losses, thereby advancing the trust and scalability of blockchain ecosystems.

Here, we first use a couple of baseline models (threshold based and random forest model) for detecting suspicious activity in smart contracts involving monitoring new wallet connections and flagging unusual behavior.

The threshold-based model implemented here detects suspicious activities in smart contracts by leveraging an entity-link approach. Specifically, it monitors wallet connection activities using a key metric, such as the average transaction value received. A predefined threshold, such as 0.8 in this case, is set as a cutoff point for flagging suspicious behavior. Wallets exceeding this threshold are labeled as suspicious. This approach is simple, interpretable, and computationally efficient, making it suitable for preliminary anomaly detection. However, its reliance on a static threshold may result in a lack of flexibility when facing dynamic or evolving behavior patterns. In addition, the random forest model leverages historical data and multiple decision trees to classify wallet activities as suspicious or benign. Features like wallet connection patterns and transaction details are used to train the model. Each tree independently evaluates these features, and their outputs are aggregated for final predictions. As implemented, this model demon-

strates strong robustness to overfitting, adaptability to complex behaviors, and improved detection accuracy compared to simpler threshold-based methods, making it suitable for identifying sophisticated fraudulent activities.

Second, we use a multi-LLM agent cooperative system to identify fraud and/or misinformation contracts. In general, this process involves identifying fraudulent contracts within Ethereum transactions. This information is then processed and stored in a format that can be used for evaluation from the output of both baseline and LLM based multi-agent system framework. The overall workflow is shown in Figure 2:

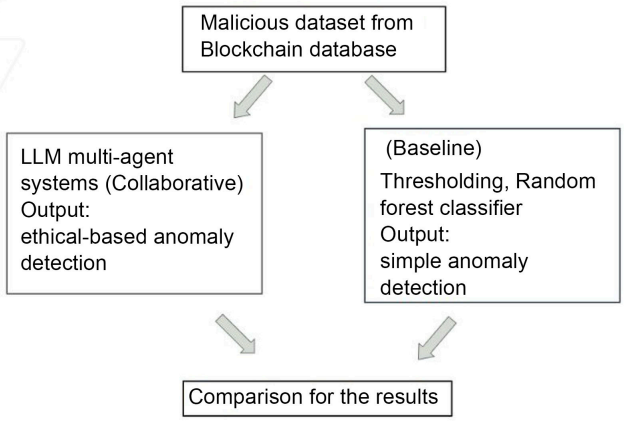

Figure 2. Overall workflow for this study.

In this study, the first baseline model is a threshold-based approach that uses the average received value as a feature to classify Ethereum addresses as hacked or non-hacked. The dataset is split into training and testing sets using stratified sampling to ensure a balanced representation. The model's performance is evaluated using metrics such as the classification report, providing insights into its accuracy and overall effectiveness. Additionally, the second baseline model employs a more advanced machine learning approach with a Random Forest Classifier to detect fraudulent Ethereum addresses. It splits the dataset into training and testing sets, trains the model, and evaluates its performance using metrics such as accuracy, precision, recall, and F1 score. It also includes an optional feature importance analysis to identify the most significant contributors to the classification process, offering valuable insights into the factors driving the model's predictions.

We propose a multi-LLM agent collaborative system where agents represent different parties interacting with smart contracts (e.g., contract mining agents, fraud detection agents, and investigative agents) to detect and mitigate fraudulent activities. The LLM model was applied to test the framework: OpenAI GPT 4o (https://platform.openai.com/docs/models#gpt-4o), and Anthropic Claude 3.5 sonnet. This collaborative interaction among agents is driven by rewards tied to contract mining, fraud detection, and evasion. The system can be implemented as a hierarchical process within the CrewAI LLM agentic framework (https://www.crewai.com/). The design is described below and illustrated in Fig-

ure 3:

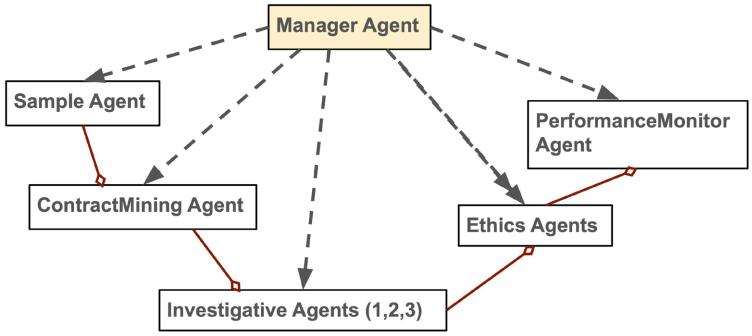

Figure 3. Multi-LLM agents (hierarchical process).

The system integrates a hierarchical process to perform tasks such as contract mining, fraud detection, ethical evaluation, and performance monitoring, ensuring an efficient and adaptive approach to fraud detection in blockchain ecosystems. The key agents include:

1) ContractMining Agent: This role involves mining fraudulent contracts from blockchain explorers like Etherscan.io in real-time, alongside retrieving normal contract data to create a combined dataset. Responsibilities include managing the fraud-to-normal contract ratio to enable dynamic dataset difficulty adjustments and challenging investigative agents to adapt by balancing the proportions of fraudulent and normal contracts effectively.

2) Investigative Agents (1, 2, and 3): Employ various detection algorithms (A, B, and C) to identify fraudulent contracts. This role involves using diverse detection algorithms, such as random forest classifiers and isolation forest models, to identify fraudulent contracts with precision. Key responsibilities include early and accurate fraud detection, adapting dynamically to dataset changes to minimize false negatives, and effectively identifying rare fraud instances. Success relies on exploring and leveraging dataset properties to refine detection accuracy while maintaining a balanced approach.

3) Ethics Agent: This role ensures fairness and mitigates bias in fraud detection processes, fostering ethical and equitable outcomes. Responsibilities include monitoring the outcomes of Investigative Agents' algorithms (A, B, and C) to identify disparities in detection rates across contract types or groups. Fairness is quantified using metrics such as demographic parity, also called statistical parity or group fairness [18]

$$P\left(\text{Flagged as Fraudulent} \middle| \text{Group A}\right) = P\left(\text{Flagged as Fraudulent} \middle| \text{Group B}\right) \quad (1)$$

and equalized odds: this requires that the prediction outcomes be conditionally independent of protected groups given the actual outcome. This metric ensures fairness across groups for true positive and false positive rates [19]

$$P\left(\text{True Positive} \middle| \text{Fraudulent}\right) = P\left(\text{True Positive} \middle| \text{Non Fraudulent}\right) \quad (2).$$

Bias is measured through false positive rates and false negative rates across contract categories [20].

$$\text{FPR} = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}} \tag{3}$$

$$\text{FNR} = \frac{\text{False Negatives}}{\text{False Negatives} + \text{True Positives}} \tag{4}$$

Bias mitigation strategies involve identifying sources of algorithmic bias, such as over-representation of certain protocols flagged as fraudulent, and recommending adjustments to detection thresholds or reweighting training datasets to ensure equitable treatment. Ethical oversight prevents unfair targeting or disproportionate flagging of specific groups, promoting balanced detection strategies that maximize fairness and accuracy. The role also generates and reports fairness and bias metrics, ensuring transparency and guiding agents and stakeholders in refining detection processes.

1) PerformanceMonitor Agent: This role involves tracking and evaluating the performance of Investigative Agents to drive continuous improvement. Responsibilities include recording key performance metrics such as accuracy, precision, recall, and efficacy, and providing feedback to the ContractMining Agent to dynamically adjust dataset difficulty. This creates a feedback loop to refine detection models and enhance overall effectiveness.

2) Manager Agent: This role focuses on coordinating workflows and fostering seamless collaboration between agents. Responsibilities include overseeing the hierarchical system to align agent activities with overarching goals, while facilitating real-time communication and data sharing to ensure adaptive and efficient fraud detection processes.

This adaptive and collaborative system leverages real-time data, dynamic feedback loops, and ethical oversight to enhance fraud detection while ensuring fairness and reducing biases in blockchain ecosystems.

4. Empirical Evaluation

The results for identifying malicious contracts from the 1st dataset and 2nd dataset using Baseline Models 1 and 2 show accuracies of (0.3, 0.5) and (0.94, 0.5), respectively. Baseline Model 2 (random forest classifier) also highlighted that the top features in Ether transaction dynamics are **Total Ether Received** and **Time Difference Between First and Last Transactions**. Key factors include diversity (**Unique Received From Addresses**), timing patterns (**Avg Min Between Transactions**), and value indicators (**Max Value Received** and **Total Ether Balance**). These features effectively capture activity, diversity, and value retention.

Furthermore, the multi-LLM agent system with GPT-4 achieves from 1st and 2nd datasets providing an accuracy range of (0.8 - 0.9), (0.78 - 0.9) among Investigate Agents 1 - 3, while the system with Claude 3.5 achieves a higher accuracy range of (0.88 - 0.92), (0.78 - 0.9). A comparison of accuracy values for Baseline Models 1 and 2, as well as the two multi-LLM agent frameworks, is shown in Table

1. In our comparison, the Random Forest model (Baseline 2) demonstrated the highest accuracy values in the 1st dataset, which only collected 1000 samples, showcasing its capability to effectively classify suspicious activities. However, there is evidence to suggest that the model may have overfitted the dataset, potentially limiting its generalizability to new or unseen data. In contrast, the multi-LLM agent framework, utilizing Claude 3.5, achieved slightly lower accuracy values in the range of 0.88 to 0.92 for the 1st dataset with 1000 samples, and in the range of 0.78 to 0.9 for the 2nd dataset with 139,600 samples. Despite this, it provided invaluable insights into model fairness and bias, aspects that are critical for ensuring ethical and reliable decision-making in real-world applications. Ultimately, the multi-LLM agent framework with Claude 3.5 emerged as the optimal solution in this study, balancing strong predictive performance with enhanced interpretability and fairness analysis.

Table 1. The results of model accuracy values from two sets of smart contract datasets.

	Baseline 1	Baseline 2	GPT 4o	Claude 3.5
1st dataset	0.3	0.94	0.8 - 0.9	0.88 - 0.92
2nd dataset	0.5	0.5	0.78 - 0.9	0.78 - 0.9

The summary from the multi-LLM agent system with Claude 3.5 for the 1st dataset and the 2nd dataset are as follows:

1) 1st dataset:

A dataset of 1000 contracts was generated using the **ContractMinerTool**, consisting of **995 normal contracts (99.5%)** and **5 fraud contracts (0.5%)**. The dataset includes key features such as **Address**, **Flag**, **Total Ether Received**, and **Total Ether Balance**, which are stored in **CSV format** to facilitate analysis. For fraud detection, **Algorithm A** successfully identified all 5 fraud contracts with **100% accuracy**, effectively covering the expected cases. However, **Algorithm B** expanded on this by detecting **6 fraud contracts**, adding one additional contract (ID: 7890) not flagged by Algorithm A. Taking it further, **Algorithm C** identified **7 fraud contracts**, incorporating two new IDs (7890 and 9012) beyond those flagged by Algorithm A.

The **EthicsCheckerTool** highlighted critical fairness and bias issues, including incomplete data provided by **Specialist B**, biases in **Algorithm A** (which tended to favor larger contract values), and **Algorithm C** (which exhibited sector-specific bias). To address these concerns, recommendations include improving data completeness, standardizing reporting processes to ensure consistency, and refining detection criteria to reduce bias and promote fairness across all algorithms.

Performance evaluation revealed strengths and limitations across agents. **Agent A** achieved **92% accuracy** and **85% efficiency**, excelling in detecting complex fraud cases but demonstrating a noticeable bias toward large contract values. **Agent B** offered a more **balanced approach** with **88% accuracy** and **78% efficiency** but underperformed compared to Agent A. Meanwhile, **Agent C** demon-

strated strong capabilities in **time-series analysis**, achieving **90% accuracy** and **82% efficiency**, but exhibited sector-specific biases that require further mitigation.

To enhance both **performance** and **fairness**, a set of actionable recommendations has been proposed. First, conduct **cross-training** among agents to share their unique strengths, such as Agent A's expertise in complex fraud detection and Agent C's proficiency in time-series analysis. Second, refine all algorithms to mitigate observed biases and further improve accuracy. Third, implement a **hybrid detection system** that integrates the strengths of all agents to create a robust and unbiased solution capable of addressing diverse fraud scenarios. Lastly, establish a framework for **regular audits** and **performance reviews**, complemented by additional training sessions focused on ethical considerations, to ensure fairness, accountability, and continuous improvement across the detection system.

2) 2nd dataset

A dataset of 139,600 contracts was collected and used in the ContractMiner-Tool, comprising 139,451 normal contracts (99.9%) and 149 fraud contracts (0.1%). The dataset includes attributes like Contract Address, Creation Date, Creator Address, Balance, Transaction Count, Code Size, and a fraud indicator. Three algorithms (A, B, and C) were applied to detect fraudulent contracts in the dataset. **Algorithm A** detected 149 fraudulent contracts, achieving a fraud detection rate of 0.107%, which was significantly lower than the expected 10%. Similarly, **Algorithm B** identified the same number of fraudulent contracts as Algorithm A, maintaining consistency. **Algorithm C** also confirmed similar results, with a detection rate of 0.1068%.

This consistency across algorithms highlights their accuracy and reliability, with minor discrepancies likely due to rounding or subtle refinements in detection methods. The detected fraud rate closely matches the expected 0.1%, demonstrating strong alignment with dataset expectations. However, the dataset's low fraud rate reflects real-world challenges in detecting subtle fraud patterns, suggesting the need for further validation, such as reviewing flagged and non-flagged contracts, and refining algorithms to enhance detection rates. These findings provide a robust starting point for fraud analysis while underscoring the importance of continuously improving detection tools to address evolving fraud strategies.

The **EthicsCheckerTool** was utilized to assess the fairness of the algorithms. **Algorithm A** exhibited moderate bias, with higher false positive rates for minority groups, despite achieving an accuracy of 85%. **Algorithm B** demonstrated reduced bias, maintaining consistent false positive rates across demographic groups, with an accuracy of 82%. **Algorithm C** emerged as the least biased, with consistent accuracy and fairness across groups, though its overall accuracy was slightly lower at 80%. Bias in false positive rates and accuracy variations revealed disparities likely caused by the underrepresentation of certain demographics in the training data. These disparities highlight the importance of ensuring diverse and representative datasets to improve algorithmic fairness.

The performance agents are responsible for evaluating the implementation of tasks based on these algorithms. Agent A achieved an accuracy of 85% but required significant improvements in time management, demonstrating consistent accuracy but lacking efficiency. Agent B showed commendable effort but needed enhancements in both accuracy (78%) and efficiency, performing below its counterparts. Agent C outperformed the others, excelling in both metrics with a 90% accuracy rate and high efficiency. To address these performance variations, recommendations included implementing mentoring programs led by Agent C to share best practices, providing time management training for the other agents to boost efficiency, and engaging in continuous algorithm refinement to enhance overall accuracy and performance.

In this study, the optimal multi-LLM agent fraud detection system has demonstrated significant potential, leveraging diverse algorithms and specialized agents to identify fraudulent contracts with high accuracy. The inclusion of fairness evaluations and performance monitoring has provided critical insights into both strengths and areas for improvement. While the system achieves robust results, several challenges must be addressed to enhance its reliability, fairness, and adaptability. The fraud detection system excelled, with Algorithm C showing the highest sensitivity and Agents A, B, and C demonstrating strengths in complexity, balance, and time-series analysis, respectively. Improvements are needed to address biases, enhance transparency, and optimize efficiency. A hybrid model combining agent strengths, along with continuous monitoring and training, is recommended to ensure robust, ethical, and adaptive detection.

Path Forward: By addressing these areas of improvement, the system can evolve into a more fair, accurate, and adaptable fraud detection framework. This holistic approach will enhance trust in the system, making it a reliable tool for high-stakes applications in blockchain ecosystems and beyond. Continuous collaboration among agents and a commitment to ethical principles will ensure the system remains aligned with evolving challenges in fraud detection.

5. Summary and Future Work

This work explores a novel framework leveraging multi-LLM agent networks capable of cooperation to address fraud detection in blockchain systems. By introducing roles such as manager agents, ethical agents, investigative agents, contract mining agents, and fraud detective agents, the framework showcases how agents can function effectively even when some are malicious. A test case demonstrates the system's potential: fraud detective agents recognize malicious contracts from the blockchain datasets, while ethical agents assess fairness and bias in these transactions. This approach aims to establish a trustworthy and reliable multi-LLM agent network that surpasses traditional non-agent methods, even in adversarial scenarios. The proposed framework is in its early stages, with no prior systems of ethical LLM agents for fraud detection to serve as a reference. Furthermore, the effectiveness of agent cooperation and the robustness of the system in highly ad-

versarial settings require further validation. The potential computational overhead of managing large networks of agents remains a challenge.

Future efforts will focus on implementing and testing the framework to evaluate its efficacy in detecting and mitigating fraud on blockchain platforms. In conclusion, this work includes comparative studies against baseline non-agent approaches to quantify the system's advantages. Additionally, refining the interaction between ethical advisors and malicious agents will enhance robustness, paving the way for broader applications in fraud detection and beyond.

Acknowledgements

The author would like to thank Seyoung Song for helping collect and process the blockchain data.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Deng, Z., Guo, Y., Han, C., Ma, W., Xiong, J., Wen, S. and Xiang, Y. (2024) AI Agents under Threat: A Survey of Key Security Challenges and Future Pathways. arXiv: 2406.02630.

[2] Humphreys, D., Koay, A., Desmond, D. and Mealy, E. (2024) AI Hype as a Cyber Security Risk: The Moral Responsibility of Implementing Generative AI in Business. *AI and Ethics*, **4**, 791-804. https://doi.org/10.1007/s43681-024-00443-4

[3] Wang, J., Hong, Y., Wang, J., Xu, J., Tang, Y., Han, Q., *et al.* (2022) Cooperative and Competitive Multi-Agent Systems: From Optimization to Games. *IEEE/CAA Journal of Automatica Sinica*, **9**, 763-783. https://doi.org/10.1109/jas.2022.105506

[4] Motwani, S.R., Baranchuk, M., Hammond, L. and de Witt, C.S. (2023) A Perfect Collusion Benchmark: How Can AI Agents Be Prevented from Colluding with Information-Theoretic Undetectability? *Multi-Agent Security Workshop Neu-rIPS'*23.

[5] Zeng, Y., Wu, Y., Zhang, X., Wang, H. and Wu, Q. (2024) Autodefense: Multi-Agent LLM Defense against Jailbreak Attacks. arXiv: 2403.04783.

[6] Calvaresi, D., Calbimonte, J., Dubovitskaya, A., Mattioli, V., Piguet, J. and Schumacher, M. (2019) The Good, the Bad, and the Ethical Implications of Bridging Blockchain and Multi-Agent Systems. *Information*, **10**, Article 363. https://doi.org/10.3390/info10120363

[7] Woodward, C.R. (2022) Analysis of Integrating Blockchain Technologies into Multi-Agent Systems. arXiv: 2212.12313.

[8] Zhuang, Y., Liu, Z., Qian, P., Liu, Q., Wang, X. and He, Q. (2020) Smart Contract Vulnerability Detection Using Graph Neural Network. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, Yokohama, 11-17 July 2020, 3283-3290. https://doi.org/10.24963/ijcai.2020/454

[9] Liu, L., Tsai, W., Bhuiyan, M.Z.A., Peng, H. and Liu, M. (2022) Blockchain-Enabled Fraud Discovery through Abnormal Smart Contract Detection on Ethereum. *Future Generation Computer Systems*, **128**, 158-166. https://doi.org/10.1016/j.future.2021.08.023

[10] Ravuri, A., Sendil, M.S., Rani, M., Srikanth, A., Sharath, M.N., Sudarsa, D., *et al.*

(2024) Blockchain-Enabled Collaborative Anomaly Detection for IoT Security. *MATEC Web of Conferences*, **392**, Article ID: 01141. https://doi.org/10.1051/matecconf/202439201141

[11] Papi, F.G., Hübner, J.F. and de Brito, M. (2022) A Blockchain Integration to Support Transactions of Assets in Multi-Agent Systems. *Engineering Applications of Artificial Intelligence*, **107**, Article ID: 104534. https://doi.org/10.1016/j.engappai.2021.104534

[12] Bajaj, S. (2023) Autonomous AI Agents in Fraud and Risk Management. https://oscilar.com/blog/autonomous-ai-agents

[13] Park, T. (2024) Enhancing Anomaly Detection in Financial Markets with an LLM-based Multi-Agent Framework. arXiv: 2403.19735.

[14] Gabriel, I., Manzini, A., Keeling, G., Hendricks, L.A., Rieser, V., Iqbal, H., Manyika, J., *et al.* (2024) The Ethics of Advanced AI Assistants. arXiv: 2404.16244.

[15] Raji, I.D., Xu, P., Honigsberg, C. and Ho, D. (2022) Outsider Oversight: Designing a Third Party Audit Ecosystem for AI Governance. *Proceedings of the* 2022 *AAAI/ACM Conference on AI, Ethics, and Society*, Oxford, 19-21 May 2021, 557-571. https://doi.org/10.1145/3514094.3534181

[16] Murikah, W., Nthenge, J.K. and Musyoka, F.M. (2024) Bias and Ethics of AI Systems Applied in Auditing—A Systematic Review. *Scientific African*, **25**, e02281. https://doi.org/10.1016/j.sciaf.2024.e02281

[17] Jung, E., Le Tilly, M., Gehani, A. and Ge, Y. (2019) Data Mining-Based Ethereum Fraud Detection. 2019 *IEEE International Conference on Blockchain* (*Blockchain*), Atlanta, 14-17 July 2019, 266-273. https://doi.org/10.1109/blockchain.2019.00042

[18] Hardt, M., Price, E. and Srebro, N. (2016) Equality of Opportunity in Supervised Learning. *Proceedings of the* 30*th International Conference on Neural Information Processing Systems*, 3323-3331.

[19] Kleinberg, J., Mullainathan, S. and Raghavan, M. (2016) Inherent Trade-Offs in the Fair Determination of Risk Scores. arXiv: 1609.05807.

[20] Barocas, S., Hardt, M. and Narayanan, A. (2023) Fairness and Machine Learning: Limitations and Opportunities. MIT Press.