Truman: A Large Language Model-based Multi-agent Simulator for Synthetic Money Laundering Data Generation

Extended Abstract

Dattatray Vishnu Kute
The University of New South Wales
Sydney, Australia
d.kute@unsw.edu.au

Zihao Xu
The University of New South Wales
Sydney, Australia
zhltroin@gmail.com

Yuekang Li
The University of New South Wales
Sydney, Australia
yuekang.li@unsw.edu.au

Fethi Rabhi
The University of New South Wales
Sydney, Australia
rabhi@acm.org

ABSTRACT

Money laundering (ML) facilitates the cross-border movement of illicit funds, enabling organized crime by disguising the origins of illegal money. Financial institutions face significant challenges in combating it, primarily due to barriers in adopting advanced technologies such as machine learning, caused by restricted access to sensitive transaction data. Existing synthetic datasets often lack critical customer information and realism, reducing their utility for ML detection. This study presents Truman, an innovative data generator that leverages Large Language Model (LLM) based agents to create realistic financial transaction data, incorporating simulation of ML patterns. Expert validation confirms the dataset's quality and applicability for anti-money laundering research.

KEYWORDS

Money Laundering; Financial Transaction Data; Synthetic Data Generator; Multi-agent Simulator; Financial Crime; LLM; OpenAI

ACM Reference Format:
Dattatray Vishnu Kute, Zihao Xu, Yuekang Li, and Fethi Rabhi. 2025. Truman: A Large Language Model-based Multi-agent Simulator for Synthetic Money Laundering Data Generation: Extended Abstract. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 3 pages.

1 INTRODUCTION

Money laundering (ML) has been a transnational crime and global issue for decades. The United Nations Office on Drugs and Crime (UNODC) estimates that 2% to 5% of global Gross Domestic Product (GDP) is laundered annually, amounting to over a trillion dollars [15]. This poses a substantial threat to social security and economic prosperity of nations. In response, governments have implemented Anti-Money Laundering (AML) and Counter-Terrorism Financing (CTF) policies, requiring regulated entities involved in financial services to ensure compliance by detecting and reporting suspicious transactional behavior of customers. Financial institutions (FIs) often face significant penalties for non-compliance due to inadequate AML controls. In 2022, global FIs incurred over $8 billion in AML-related fines, raising the total AML fines since the 2007-2008 financial crisis to $56.1 billion [5]. In 2024, the historic $3.1 billion settlement by TD Bank highlights the increasing enforcement of AML regulations worldwide [13].

To combat ML, FIs have setup AML programs and AML controls that rely on rule-based systems to monitor and identify suspicious transactions. However, these systems suffer from high false positive rates, estimated between 70% to 99% [10]. A major challenge in developing and training machine learning models is the limited availability of labeled financial transaction data with ML transactions because such data is highly sensitive and protected [2]. This restriction constrains the research in the AML domain, despite the suitability of machine learning for analyzing transaction flows to detect illicit activities.

To address the issue of a lack of data for training models, several attempts have been made in the past to create synthetic ML transaction data [1, 9, 14], however they suffer from inadequate attributes, semantic incorrectness and data quality issues. To fill these gaps, this research introduces a synthetic financial transaction data generator, **Truman** [1]. Truman aims to produce financial transactions and ML transactions data for a bank and its customers.

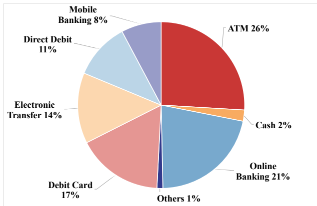
2 DATA GENERATOR DESIGN

This section covers the design of Truman as shown in Figure 1, consisting of two components: financial transaction data generation and the simulation of ML patterns to generate ML transactions. Truman is implemented by extending the AgentVerse [4] framework.

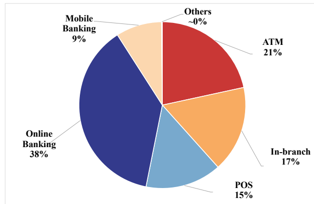
Figure 1.a shows the workflow used to generate the financial transaction data. It involves four steps. First step generates the user profile containing attributes such as age, gender, profession and address etc. Subsequent data generation relies on these demographic patterns and characteristics. The Retrieval-Augmented Generation (RAG) technique [8] facilitates this functionality and is used in prompt design to extract relevant information from generated data.

[1]The name Truman is inspired by the movie "The Truman Show", where the protagonist believes he is living a real life, but is actually part of a television show. Similarly, the agents in Truman's generated data think they are real persons performing transactions.

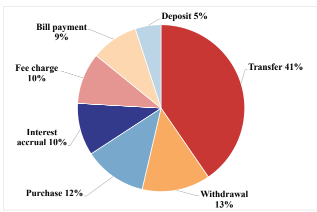
Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA.



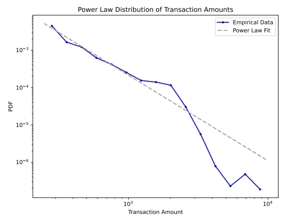
(a) Transaction Method



(b) Transaction Origin



(c) Transaction Type



(d) Transaction Amount

Figure 2: Data Distribution

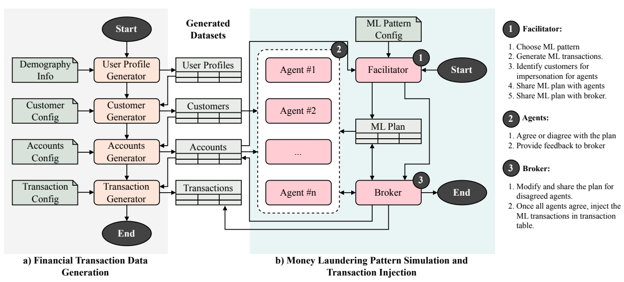


Figure 1: Overview of Truman's Design

Table 1: A Sample Dataset Summary

S. No.	*Data Entity*	*Records*	*Remarks*
1.	Customers	50	Retail customers
2.	Saving accounts	50	1 per customer
3.	Legitimate transactions	33041	40-80 txn per month
4.	ML transactions	648	10 instances per pattern.

The second step generates the customer profile using user data and attributes like enrollment date and risk status. The third step creates a savings account, and the fourth step generates its monthly transactions. To fully simulate real-world scenarios, a single LLM agent is used to impersonate a user and generate the transactions iteratively for each customer. This flexibility is a key advantage of using LLM agents compared to previous studies [1]. To ensure adherence to the output format, we utilize few-shot prompts [3] to guide the LLM's output, with feedback from the Truman parser to report errors and enhance the prompt with error information.

Figure 1.b shows the simulation of ML patterns. The following foundational ML patterns [14] are simulated - Fan In, Fan Out, Scatter-Gather, Gather-Scatter, Simple Cycle, Random Cycle, Bipartite, Stack and Peel Chain pattern. The multi-agent simulation comprises a facilitator, a broker, and multiple agents that helps to reduce the task complexity by having a distinct responsibilities for each role. Assigning excessive instructions to a single role complicates task comprehension and correct execution. The facilitator comprehends the plan using the chain-of-thought [16] technique.

The quality of prompts plays a crucial role in determining the final quality of generated data. Our prompt design philosophy is based on three key principles: conciseness, relevance with high information density, and clear directionality. Truman utilizes the capabilities of the OpenAI GPT-4 Turbo APIs [12] to implement these principles effectively. Detailed explanation of Truman framework along with the patterns and prompts can be found here [7].

3 DATASET

The data generation using Truman is a two step run process. In first step, the base data containing customers, accounts and regular transactions are generated as per the input configuration parameters. The second step runs the ML simulation framework and generates the transactions as per the ML pattern and other configuration parameters. Table 1 shows the sample dataset summary and figure 2 shows the data distribution. It is observed that the generated data approximately follows the power-law distribution (alpha 2.29), aligning with real-world distribution [6].

3.1 Key Findings

The LLM model generates new values for some attributes beyond those in the prompt, which are mostly accurate but occasionally include invalid data. The LLMs tend to generate multiples of 5 and 10s for transaction amount, even though the prompt examples show otherwise. In ML simulation, if there is feedback from too many agents, the Broker cannot comprehend and synthesize them to generate complex ML patterns (example Stack pattern). Likewise, if the instructions contain too many goals, LLMs fail to complete multiple tasks as expected unless clear guidelines are given. In addition to the prompts, empirical rules based on observations of the generator output are used to improve the data quality. It is recommended that LLM instructions be explicitly aligned with detailed rules or examples rather than allowing the LLM to determine how to utilize range of values. This is because LLMs may lack genuine reasoning capabilities and instead replicate reasoning steps based on the training data as demonstrated in [11].

4 LIMITATIONS AND FUTURE WORK

A key limitation is that LLMs often create biased data despite instructions for randomization, so we manually randomized attributes before input; full automation would be ideal. The Truman is LLM-agnostic, with output quality and cost dependent on the capability and API pricing of the LLM used. Future work includes modeling transactions for additional account types; simulating transactions for small, medium, and large business customers; developing machine learning models to detect injected ML transactions as a data validation method; and simulating transactions associated with other financial crimes, such as fraud and cybercrime.

ACKNOWLEDGMENTS

This research was supported by the Australian Government Research Training Program Scholarship.

REFERENCES

[1] Erik Altman, Jovan Blanuša, Luc Von Niederhäusern, Béni Egressy, Andreea Anghel, and Kubilay Atasu. 2024. Realistic synthetic financial transactions for anti-money laundering models. *Advances in Neural Information Processing Systems* 36 (2024).

[2] Samuel A Assefa, Danial Dervovic, Mahmoud Mahfouz, Robert E Tillman, Prashant Reddy, and Manuela Veloso. 2020. Generating synthetic data in finance: opportunities, challenges and pitfalls. In *Proceedings of the First ACM International Conference on AI in Finance*. Association for Computing Machinery, 1–8.

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL]

[4] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848* (2023). https://arxiv.org/abs/2308.10848

[5] Fenergo. 2023. *Global AML Fines Research Report*. Report. Fenergo. https://www.fenergo.com/aml-fines-report

[6] Xavier Gabaix, Parameswaran Gopikrishnan, Vasiliki Plerou, and H Eugene Stanley. 2003. A theory of power-law distributions in financial market fluctuations. *Nature* 423, 6937 (2003), 267–270.

[7] Dattatray Vishnu Kute, Zihao Xu, Yuekang Li, and Fethi Rabhi. 2025. Truman framework. https://sites.google.com/view/truman-antiml/home Accessed: 2025-02-02.

[8] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv:2005.11401 [cs.CL]

[9] Maryam Mahootiha. 2021. Money laundering data: InPlusLab Anti-Money Laundering DataDatasets. https://www.kaggle.com/datasets/maryam1212/money-laundering-data.

[10] McKinsey. 2019. Risk Transforming approaches to AML and financial crime. https://www.mckinsey.com/~/media/McKinsey/BusinessFunctions/Risk/OurInsights/TransformingapproachestoAMLandfinancialcrime/Transforming-approaches-to-AML-and-financialcrime-vF.pdf

[11] Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2024. GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models. arXiv:2410.05229 [cs.LG] https://arxiv.org/abs/2410.05229

[12] OpenAI. 2025. GPT-4 Turbo and GPT-4 Models Documentation. https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4. Accessed: 2025-01-27.

[13] Financial Services Perspectives. 2024. *TD Bank's Historic $3.1B Money Laundering Settlement: A Warning to All Financial Institutions*. Bradley. https://www.financialservicesperspectives.com/2024/10/td-banks-historic-3-1b-money-laundering-settlement-a-warning-to-all-financial-institutions/ Accessed: 2025-01-27.

[14] Toyotaro Suzumura and Hiroki Kanezashi. 2021. Anti-Money Laundering Datasets: InPlusLab Anti-Money Laundering DataDatasets. http://github.com/IBM/AMLSim/.

[15] UNODC. 2011. *Estimating illicit financial flows resulting from drug trafficking and other transnational organized crimes*. Technical Report. UNODC.

[16] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. arXiv:2201.11903 [cs.CL]