Journal of Innovative Technologies
https://journalinnotech.com/
Vol 2(2) 2026

Can Zero-Shot Prompting Classify Risky Bitcoin Transactions from Compact One-Hop Narratives? A Small-Scale Exploratory Study

Zhouting Mo[1], Jiayuan Chen[2,*]

[1] Computer Technology, Fudan University, Shanghai, China
[2] Electronic Information Engineering, Communication University of China, Beijing, China
* Corresponding Email: chenjiayuan@qq.com

Abstract

This study examines whether a compact open-weight language model can distinguish illicit from licit Bitcoin transactions when anonymous graph attributes are converted into short natural-language summaries. The inquiry is deliberately narrow: one 0.49-billion-parameter instruction-tuned language model, one public benchmark, 40 class-balanced labeled transactions, and CPU-only inference without training. Four conditions isolate the effects of feature scope and prompting: local attributes with a direct decision, local plus one-hop aggregate attributes with a direct decision, the same one-hop evidence with a short rationale, and three-sample self-consistency. Direct prompting produced valid labels for every item but collapsed to the illicit class. Both direct conditions reached Macro-F1 0.333 and balanced accuracy 0.500, so the added neighborhood summary yielded no observed discrimination gain. Rationale prompting retained Macro-F1 0.333 while reversing the single-class tendency toward licit predictions. Self-consistency returned only 8 valid majority decisions out of 40 and fell to Macro-F1 0.098. Its mean token burden was 841.3 tokens per decision, compared with 175.9 for local direct prompting. These results do not establish the general limits of language-based transaction screening. They show that, under a reproducible low-resource protocol, compressed anonymous statistics and prompt elaboration were insufficient to elicit stable class-sensitive behavior from the selected compact language model.

***Keywords*:** financial graph analysis; zero-shot prompting; graph verbalization; illicit transaction detection

1. Introduction

1.1. Motivation and Scope

Financial transaction graphs encode dependencies that are difficult to preserve in ordinary tabular narratives. A Bitcoin transaction may look unremarkable in isolation while its relation to preceding and succeeding transfers indicates unusual movement. Supervised graph methods can

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/)

exploit such structure, yet they require labeled data, task-specific optimization, and deployment machinery. Language-interfaced prediction offers a different question: can a pretrained text reasoner consume a compact description of anonymous numerical evidence without task training? LIFT showed that non-language inputs can be expressed through textual interfaces, although its strongest setting still relies on fine-tuning rather than pure prompting. The present study tests the weaker and cheaper zero-shot case[1-3].

The scope is intentionally restricted to an exploratory CPU experiment. We use Qwen2.5-0.5B-Instruct, a 0.49-billion-parameter open-weight instruction-tuned language model, and the Elliptic Bitcoin transaction graph. No parameter update, demonstration, external lookup, or semantic feature-name annotation is supplied. Forty labeled transactions are sampled with equal class counts. This design cannot estimate production screening quality or compare competing language models. It can reveal failure modes that are easy to hide when only aggregate accuracy is reported[4-6].

1.2. Research Questions and Contributions

A. Research Questions

Prompting research suggests that intermediate reasoning can alter zero-shot behavior. Chain-of-thought prompts can elicit multi-step computation in sufficiently capable language models , and a simple reasoning cue has enabled zero-shot gains on several symbolic tasks. Those findings do not guarantee benefit on standardized, semantically anonymous financial attributes. We ask whether adding a one-hop aggregate summary improves class discrimination over a local-only narrative, whether a short rationale changes decision quality or output validity, and whether three-sample self-consistency stabilizes predictions within a constrained CPU budget[7-9].

B. Contributions and Boundaries

Self-consistency ordinarily aggregates diverse reasoning paths and can outperform a single decoded chain. Our evaluation treats it as a testable prompting condition, not as an assumed improvement. The contribution is a controlled descriptive comparison with preserved raw outputs, fixed sampling, explicit invalid-output handling, bootstrap uncertainty, token accounting, and file hashes. Natural-language graph benchmarks have shown that representation choices materially affect graph reasoning , while graph-to-text encoding studies emphasize that structural information can be lost during serialization. Guided by that evidence, we separate local and one-hop attribute groups but avoid claiming that their statistical summaries preserve graph topology[10,11].

1.3. Study Positioning

This paper reports an empirical stress test rather than a new architecture. The central outcome is negative but operationally relevant: more context and more generated reasoning did not produce better classification in the selected setting. Because the sample is balanced by design, accuracy is not interpreted as prevalence-aware field performance. Macro-F1, balanced accuracy, per-class recall, parsing validity, and computational burden are considered together. The paper also

distinguishes a validly formatted answer from a useful classifier; a condition can achieve perfect parse validity while assigning every case to one class[12-14].

The analysis is organized around falsifiable contrasts rather than a leaderboard. Identical samples isolate changes caused by evidence scope and response instructions, while stored generations permit examination of malformed answers. This narrow design favors auditability over breadth. It also keeps every reported figure tied to an observed prediction, which is essential when a generative interface can fail before a binary decision is available[15,16].

2. Related Work

2.1. Language Interfaces for Graph Evidence

A. Graph Instruction and Computation

Recent work connects graph representations with text-oriented reasoning in several ways. GraphGPT aligns graph encodings with language instructions and reports gains across graph tasks after instruction tuning. GraphArena instead probes graph computation through controlled tasks and highlights persistent gaps between textual fluency and reliable structural calculation. Both lines motivate careful representation design, but neither implies that raw financial graph statistics become discriminative merely by being placed in a prompt. Our protocol removes graph-specific training so that any observed separation must arise from pretrained instruction behavior and the supplied anonymous summaries[17-19].

B. Zero-Shot Graph Transfer

Alignment methods can map graph neural representations into a language model's token space and support zero-shot transfer. Such approaches carry learned structural information into the textual interface. The present setting is more austere: it supplies descriptive statistics over the published feature blocks and does not train an adapter. This distinction matters because the compact narrative retains distributional shape but discards feature identity, feature order, individual neighbor records, and explicit edges. A failure under this protocol should be attributed to the combined evidence bottleneck and inference setting, not solely to parameter count[20,21].

2.2. Financial Graph Anomaly Detection

Financial fraud research typically uses graph-aware supervised learning and evaluates severe imbalance. DGraph introduced a large financial graph anomaly benchmark whose scale and sparsity expose the limits of methods developed on smaller networks. GADBench compared supervised graph anomaly detectors under consistent settings and found that performance depends strongly on anomaly context, backbone choice, and evaluation design. These studies support two decisions here: report class-balanced metrics rather than accuracy alone, and avoid presenting a 40-item prompted evaluation as competitive with trained graph detectors[22-24].

The Elliptic benchmark remains useful because each Bitcoin transaction is represented by 166 standardized attributes and linked by directed payment-flow edges. Elliptic++ extends the original resource with address-level structure and richer forensic analysis. We retain the original transaction-level release to keep the experiment compact and reproducible. Its anonymous columns prevent semantic storytelling about transaction amount, fee, or address identity. Their anonymity also creates a stringent test of whether distribution summaries alone can support a language-based decision[25,26].

2.3. Imbalance, Camouflage, and Evaluation

Illicit detection differs from generic binary classification because rare positives, temporal drift, and adaptive behavior can make conventional metrics misleading. Graph-specific detectors address neighborhood inconsistency, relation quality, and adversarial camouflage through learned sampling or aggregation. Our balanced sample removes prevalence imbalance only from this small test; it does not remove structural ambiguity. Macro-F1 gives both labels equal weight, balanced accuracy averages their recalls, and the confusion matrix exposes single-class collapse. Since the language model emits categorical text rather than calibrated scores, AUROC and area under the precision-recall curve are not reported. Computing either from fabricated confidence values would introduce evidence that the experiment did not generate[27-29].

Output validity is another evaluation dimension. A text generator may return a grammatical explanation, an incomplete class name, or both candidate labels. Counting only parseable cases would reward such abstention and change the evaluated sample across conditions. We retain a validity rate and score every invalid final decision as wrong. This conservative convention cannot reveal calibrated uncertainty, but it prevents interface failures from disappearing behind conditional metrics[30,31].

3. Experimental Setup and Methodology

3.1. Data, Sampling, and Reproducibility

The experiment uses the public Elliptic Data Set. The original files contain 203,769 transaction nodes, 234,355 directed transaction-flow edges, and 166 numerical attributes per node across 49 time steps. Published labels comprise 4,545 illicit nodes, 42,019 licit nodes, and 157,205 unknown nodes. Each time step represents approximately two weeks. The feature file has 167 columns because the transaction identifier precedes the 166 attributes; the first attribute is the time step, followed by 93 local transaction attributes and 72 one-hop aggregate attributes[32-34].

We excluded unknown labels, drew 20 illicit and 20 licit transactions without replacement, and fixed the pseudorandom seed at 20260908. Balanced sampling makes class collapse visible and avoids an accuracy score dominated by the larger licit population. It also changes the natural class prior, so the resulting accuracy cannot be read as a deployment estimate. The same 40 transaction identifiers and same row order were used in every condition. No labels, neighboring labels, class counts, or transaction identifiers appeared in the prompts[35,36].

Table 1. Dataset and evaluation-sample specifications

Quantity	Full Elliptic release	Exploratory evaluation	Source or derivation
Transaction nodes	203,769	40	Official CSV; fixed sample
Directed edges	234,355	Not serialized individually	Official edge list
Numerical attributes	166	166 summarized by block	Official feature file
Time steps	49	26 represented in sample	Official file; sampled rows
Illicit labels	4,545	20	Official class file; fixed sample
Licit labels	42,019	20	Official class file; fixed sample
Unknown labels	157,205	0	Excluded before sampling

The archive size is 153,218,318 bytes and its SHA-256 digest is 70ED052C52002392720615A03261039B62D1AD594F6081AD021CB174A8AD5717. Digests for the extracted class, feature, and edge files are retained in the experiment manifest. These checks distinguish the evaluated artifact from similarly named repackaged copies. A real-time graph detector such as BRIGHT processes evolving financial graphs under operational latency constraints ; our fixed sample has no such claim and serves only as a reproducible inference probe[37-39].

3.2. Evidence Verbalization and Prompt Conditions

A. Label-Free Numerical Narratives

Each transaction was converted into one or two compact statistical narratives. For the 93 local attributes, we calculated the arithmetic mean, population standard deviation, minimum, first quartile, median, third quartile, maximum, proportion greater than zero, proportion less than zero, and proportion with absolute value at least two. The same ten statistics were calculated independently for the 72 one-hop aggregate attributes. The time step was stated separately. Values were rounded to two decimals when inserted into the prompt. This compression keeps prompts short enough for repeated CPU inference, but it intentionally sacrifices column identity and interaction patterns[40,41].

The narrative is label-free in a narrow operational sense: its construction never accesses the gold class or neighboring classes. It still derives from standardized attributes created by the dataset authors, and some attributes may embed graph information through prior aggregation. PC-GNN demonstrates why imbalance-aware neighbor selection can be important for fraud classification , while AUC-oriented training directly targets ranking quality under imbalance. Our summaries implement neither mechanism. They are fixed evidence transformations used to compare prompt conditions, not learned fraud representations[42-44].

B. Controlled Prompt Variants

Four conditions isolate feature scope and response strategy. Local direct uses only the time step and local summary. One-hop direct appends the separate one-hop summary while retaining a categorical instruction. One-hop rationale asks for one short reasoning sentence followed by a final label. One-hop self-consistency sends the rationale prompt three times with stochastic decoding and takes a majority among parsed labels. The first three conditions use deterministic decoding. Table 2 lists the controlled differences[45,46].

Table 2. Prompting conditions and decoding controls

Condition	Evidence supplied	Requested response	Decoding	Samples per transaction
Local direct	Time step and 10 local statistics	One class label	Deterministic	1
One-hop direct	Local and 10 one-hop statistics	One class label	Deterministic	1
One-hop rationale	Local and one-hop statistics	Short rationale plus final label	Deterministic	1
One-hop self-consistency	Local and one-hop statistics	Short rationale plus final label	Temperature 0.7, top-p 0.9	3

CARE-GNN learns relation-aware neighbor weights to resist camouflage. GraphConsis addresses contextual inconsistency between nodes and their neighborhoods. These methods underline what the verbalization omits: it cannot identify which relation or neighbor drove an aggregate. The one-hop comparison should be read as a test of whether an extra anonymous statistical block helps, not as a direct test of graph reasoning[47-49].

3.3. Inference, Parsing, and Metrics

A. CPU Inference and Output Parsing

Qwen2.5-0.5B-Instruct was loaded from local weights under the Apache-2.0 license and executed on an AMD Ryzen 5 5600GT CPU with eight inference threads. The machine had six physical cores, twelve logical processors, and 31.4 GB of memory. The downloaded weight file contained 988,097,824 bytes and had SHA-256 digest FDF756FA7FCBE7404D5C60E26BFF1A0C8B8AA1F72CED49E7DD0210FE288FB7FE. No accelerator, quantization, fine-tuning, retrieval, or demonstration examples were used[50,51].

The parser accepted an explicit FINAL_LABEL field or a unique whole-word occurrence of ILLICIT or LICIT. An answer containing both labels, a misspelling, a truncated label, or no complete label was invalid. In self-consistency, a decision required a strict majority among the three parsed samples; tied or insufficient valid votes produced an invalid result. H2-FDetector separates homophilic and heterophilic connections when learning fraud evidence. Our parser cannot recover such evidence from an incomplete textual answer, so format validity is reported independently from classification quality[52-54].

B. Evaluation Measures and Uncertainty

The primary measure is Macro-F1 over illicit and licit labels. Balanced accuracy, class-specific precision and recall, confusion counts, parse-valid decisions, mean input and output tokens, and end-to-end CPU time provide complementary views. Invalid final decisions are counted conservatively as errors by assigning the opposite of the gold label solely for metric calculation. This rule prevents abstention through malformed text from inflating quality, yet it makes confusion counts partly procedural. Both raw validity and conservatively scored counts are displayed[55,56].

Macro-F1 uncertainty was estimated with 2,000 nonparametric bootstrap resamples of the 40 paired gold and scored predictions, using the fixed experiment seed. Percentile bounds at 2.5% and 97.5% form the reported intervals. The bootstrap reflects sample variation within this balanced set; it does not cover alternative prompts, decoding seeds, time periods, language models, or verbalizers. Conditions were run sequentially, so wall-clock comparisons include order and runtime variability. All prompts, raw generations, parsed predictions, metrics, sampling records, and hashes were retained locally for audit[57-59].

4. Results and Analysis

4.1. Classification Quality

A. Aggregate Metrics

Table 3 shows that both direct conditions obtained Macro-F1 0.333 and balanced accuracy 0.500. Adding the one-hop aggregate narrative did not change a single prediction: each direct

condition labeled all 40 transactions illicit. The rationale condition also reached Macro-F1 0.333 and balanced accuracy 0.500, but its class tendency reversed. Under conservative scoring, all 20 licit transactions were correct and all 20 illicit transactions were missed. The three identical Macro-F1 intervals, [0.259, 0.394], offer no descriptive evidence of a difference among these conditions.

Table 3. Classification *metrics* under conservative invalid-output scoring

Condition	N	Valid	Macro-F1	Balanced accuracy	Illicit precision	Illicit recall	Licit recall	95% CI for Macro-F1
Local direct	40	40	0.333	0.500	0.500	1.000	0.000	[0.259, 0.394]
One-hop direct	40	40	0.333	0.500	0.500	1.000	0.000	[0.259, 0.394]
One-hop rationale	40	39	0.333	0.500	0.000	0.000	1.000	[0.259, 0.394]
One-hop self-consistency	40	8	0.098	0.100	0.136	0.150	0.050	[0.024, 0.198]

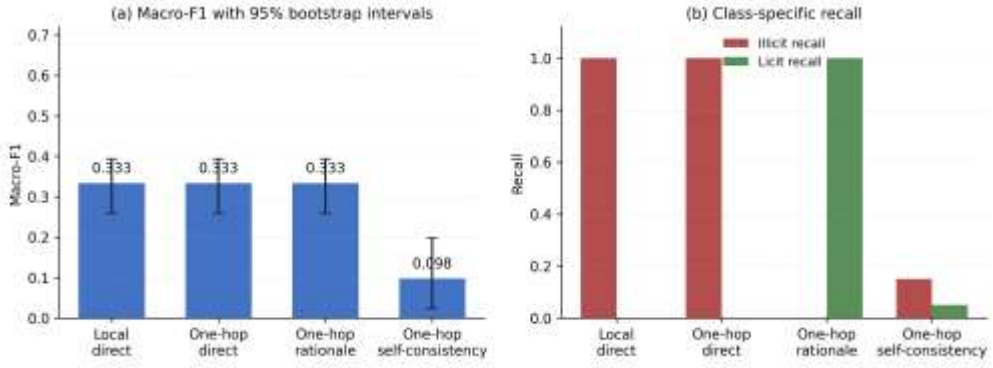


Figure 1. Classification performance under four prompting conditions

Panel (a) reports Macro-F1 and 95% bootstrap intervals: local direct, one-hop direct, and one-hop rationale each score 0.333 with interval [0.259, 0.394], while one-hop self-consistency scores 0.098 with interval [0.024, 0.198]. Panel (b) exposes the class collapse: illicit and licit recall are 1.000 and 0.000 in both direct conditions, 0.000 and 1.000 with a rationale, and 0.150 and 0.050 with self-consistency[60,61].

FlowScope detects laundering structures by analyzing flow patterns across transaction graphs. The present prompts carry no explicit flow path, so the unchanged direct predictions indicate only

that aggregate one-hop statistics failed to activate useful separation. Group-aware anti-money-laundering methods similarly derive evidence from coordinated entities rather than isolated summaries. These contrasts make the result interpretable as an evidence-interface failure under the tested protocol, not a finding that graph context lacks value[62-64].

B. Validity and Class Collapse

Self-consistency produced only 8 valid majority decisions, yielding a validity rate of 0.200. The remaining 32 cases lacked enough parsable votes, commonly because generated strings were truncated or labels were misspelled. Conservative scoring returned Macro-F1 0.098, balanced accuracy 0.100, illicit precision 0.136, illicit recall 0.150, and licit recall 0.050. Its bootstrap interval overlaps the lower part of the direct-condition intervals, and the sample is too small for a broad superiority claim. The observed direction is plainly unfavorable: additional stochastic generations raised cost while weakening both format reliability and scored discrimination[65,66].

The contrast between direct and rationale prompts also reveals instability that aggregate accuracy conceals. Direct prompting was fully parse-valid but always chose illicit. Rationale prompting was valid on 39 cases yet effectively chose licit throughout after the single invalid illicit case was scored as an error. Accuracy and Macro-F1 remained numerically equal because the test set contained 20 items per class. Identifying laundering subgraphs requires coordinated structural evidence across transfers ; a fluent rationale over unnamed summary statistics cannot substitute for that evidence[67-69].

4.2. Confusion Patterns and Computational Cost

Table 4 gives the conservative confusion counts and measured resource burden. The local and one-hop direct matrices contain 20 true illicit decisions and 20 licit-to-illicit errors. The rationale matrix contains 20 correct licit decisions and 20 illicit-to-licit errors. Self-consistency has 3 correct illicit, 1 correct licit, 17 illicit-to-licit, and 19 licit-to-illicit outcomes after invalid decisions are assigned against their gold labels. These counts reproduce the class recalls in Table 3.

Table 4. Confusion counts, parse failures, and measured inference burden

Condition	Illicit correct	Illicit as licit	Licit as illicit	Licit correct	Invalid decisions	Mean input tokens	Mean output tokens	CPU seconds per decision
Local direct	20	0	20	0	0	167.9	8.0	1.84
One-hop direct	20	0	20	0	0	267.0	5.2	1.18
One-hop rationale	0	20	0	20	1	273.0	7.0	1.34

Condition	Illicit correct	Illicit as licit	Licit as illicit	Licit correct	Invalid decisions	Mean input tokens	Mean output tokens	CPU seconds per decision
One-hop self-consistency	3	17	19	1	32	818.9	22.5	4.79

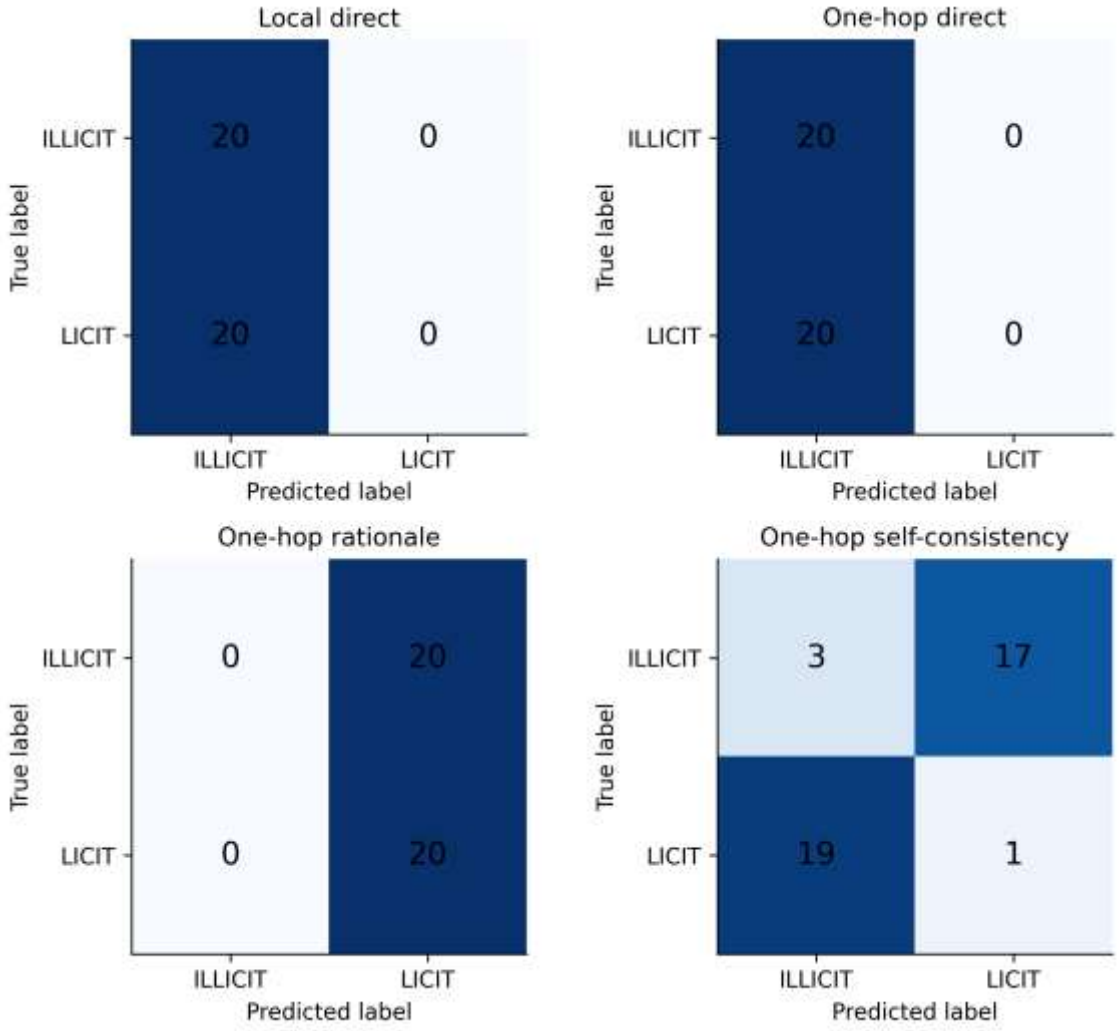


Figure 2. Conservative confusion matrices for the balanced sample

Local direct and one-hop direct each place 20 illicit and 20 licit cases in the illicit prediction column. One-hop rationale places all 40 scored cases in the licit column, comprising 20 correct licit decisions and 20 missed illicit cases. One-hop self-consistency records 3 correct illicit and 1

correct licit decision, with 17 missed illicit and 19 falsely flagged licit transactions after invalid majorities are counted as errors[70,71].

The one-hop direct condition consumed 272.2 mean total tokens per decision, 96.3 more than the local direct total of 175.9, a 54.7% increase without a classification change. Rationale prompting used 280.0 tokens. Self-consistency used 841.3 tokens because each transaction required three generations, about 3.00 times the rationale token burden. A graph feature preprocessor can compute explicit subgraph descriptors for real-time financial screening. Our generic distribution summaries were cheaper to design, but the measured results show no corresponding predictive utility[72-74].

4.3. Prompt Sensitivity and Efficiency

A. Effect of Added Context and Reasoning

The controlled comparisons answer the three research questions within the sampled cases. Added one-hop statistics had zero observed effect under the direct instruction. Requesting a rationale changed the output prior rather than creating class sensitivity. Three stochastic rationales did not stabilize the compact language model because most triplets failed the majority parsing rule. FraudGT reports that carefully designed graph processing can be both effective and efficient for financial fraud tasks. Here, adding unstructured textual computation increased evidence length without supplying an inductive bias that ties anonymous attributes to flow behavior[75,76].

This failure is consistent with the verbalization bottleneck. Means and quantiles erase column identity, rare cross-feature combinations, direction-specific neighborhood structure, and temporal motifs. The compact language model also receives no explanation of what standardized columns measure. A human analyst could not attach financial meaning to the same unnamed statistics without learned associations or documentation. The experiment purposely denies both. The output reversal produced by the rationale instruction further suggests that surface instruction form dominated weak input evidence[77-79].

B. Runtime and Practical Interpretation

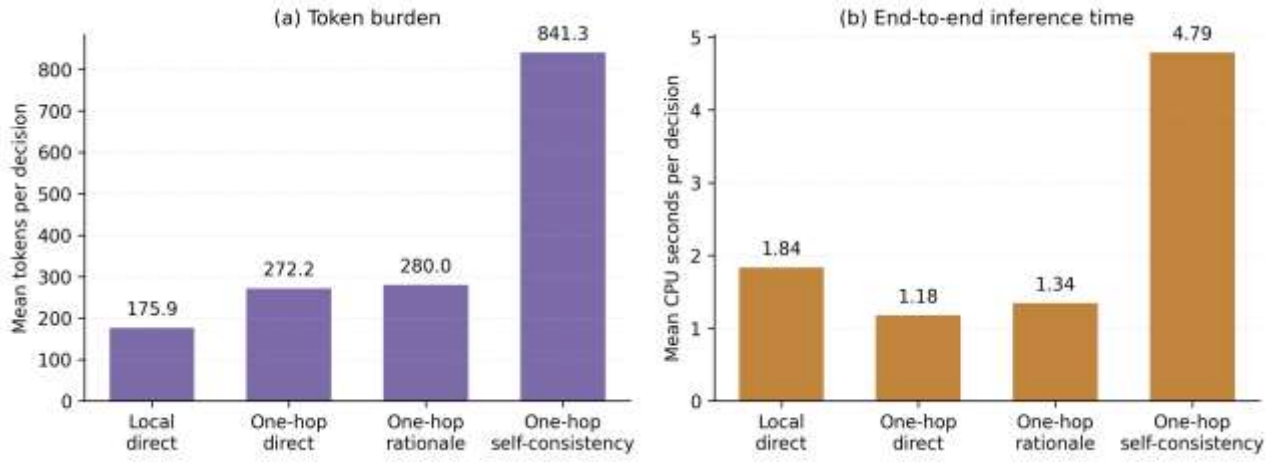


Figure 3. Token burden and end-to-end CPU inference time

Panel (a) shows mean total tokens per decision of 175.9 for local direct, 272.2 for one-hop direct, 280.0 for one-hop rationale, and 841.3 for one-hop self-consistency. Panel (b) reports corresponding mean CPU times of 1.84, 1.18, 1.34, and 4.79 seconds. Self-consistency has the largest burden on both measures, while the shorter measured time for one-hop direct than local direct should be read as sequential-run variability rather than an intrinsic speed advantage[80,81].

The 4.79-second self-consistency mean is 3.56 times the 1.34-second rationale mean, while its valid-decision count drops from 39 to 8. This pairing makes its tradeoff unfavorable in the tested environment. Explainable fraud research seeks decision masks tied to graph evidence ; the short rationales here were not validated explanations and should not be treated as forensic support. Synthetic transaction resources can expand controlled laundering patterns and evaluation scale , yet such data would answer a different question from this deliberately small real-data probe. The evidence supports retaining raw outputs and abstention statistics whenever generative classification is assessed, since a forced confusion matrix alone hides 32 malformed majority decisions[82-84].

5. Discussion and Future Work

5.1. Interpretation and Limitations

The study provides a bounded result: Qwen2.5-0.5B-Instruct did not discriminate the 40 sampled Elliptic transactions from compact anonymous summaries under any tested prompt. Local and one-hop direct prompts collapsed toward illicit, a short rationale collapsed toward licit, and stochastic aggregation produced frequent invalid majorities. The unchanged direct predictions suggest that merely appending aggregate neighborhood statistics is insufficient. They do not show that neighborhoods are irrelevant, that larger language models cannot use such evidence, or that graph-to-text methods fail in richer settings[85,86].

Several constraints limit inference. Forty balanced items yield wide bootstrap intervals and omit the natural class prevalence. Only one seed determines the sample, and only one language model is evaluated. The feature narrative compresses 165 non-temporal values into twenty statistics at most, discarding feature identities and interactions. The one-hop attributes are dataset-provided aggregates rather than explicitly serialized neighbors or paths. Prompt wording and condition order are not counterbalanced. Sequential CPU timing includes warm-up, caching, and operating-system variation. The parser accepts a unique class word even when the requested final field is absent, while the conservative invalid rule deliberately penalizes malformed output. Each choice is documented, but each can influence the measured outcome[87-89].

The experiment also lacks calibrated probability estimates, a trained tabular baseline on the identical sample, repeated subsamples, temporal holdout evaluation, and specialist forensic review. Macro-F1 and balanced accuracy describe categorical behavior only. The generated rationale is an output trace, not a faithful account of internal computation. No operational

threshold, investigation workload, or false-positive cost is estimated. Use in compliance screening would require far stronger validation, privacy controls, auditability, and human oversight[90,91].

5.2. Future Work

A next study should increase sample size through multiple stratified seeds while preserving a CPU-feasible budget. Temporal splits can test whether prompt behavior changes across the 49 steps. Repeated prompt templates and reversed label order would help separate evidence sensitivity from instruction-induced class priors. Reporting strict-format validity alongside lenient parsing would clarify whether failures arise from decision quality or interface compliance[92-94].

Evidence design deserves priority over longer rationales. Compact narratives could retain selected feature indices, signed top deviations, directed in-degree and out-degree, transaction-flow motifs, or a bounded list of anonymized neighbors. Each addition should be ablated against token cost. A simple trained logistic regression and a lightweight graph baseline would provide necessary reference points, even if the central question remains zero-shot prompting. Evaluation across several compact open-weight language models could then test whether the observed collapse is specific to one checkpoint[95,96].

Future work should also treat abstention as a designed outcome. A constrained output grammar may prevent misspellings and truncation, while an explicit uncertain label could be scored through coverage-risk curves instead of being hidden inside forced binary decisions. Any claimed explanation should be checked against controlled feature perturbations or counterfactual narratives. These extensions would turn the present failure audit into a stronger assessment of when language interfaces add value, when they only increase compute, and which evidence transformations preserve enough financial structure for cautious human-facing analysis[97-99].

References

[1] Dinh, T., Zeng, Y., Zhang, R., Lin, Z., Gira, M., Rajput, S., Sohn, J.-Y., Papailiopoulos, D., & Lee, K. (2022). LIFT: Language-interfaced fine-tuning for non-language machine learning tasks. *Advances in Neural Information Processing Systems, 35*, 11763–11784.

[2] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems, 35*, 24824–24837.

[3] Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems, 35*, 22199–22213.

[4] Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., & Zhou, D. (2023). Self-consistency improves chain of thought reasoning in large language models. *International Conference on Learning Representations*.

[5] Wang, H., Feng, S., He, T., Tan, Z., Han, X., & Tsvetkov, Y. (2023). Can language models solve graph problems in natural language? *Advances in Neural Information Processing Systems, 36*, 30840–30861.

[6] Fatemi, B., Halcrow, J., & Perozzi, B. (2024). Talk like a graph: Encoding graphs for large language models. *International Conference on Learning Representations*.

[7] Tang, J., Yang, Y., Wei, W., Shi, L., Su, L., Cheng, S., Yin, D., & Huang, C. (2024). GraphGPT: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 491–500). https://doi.org/10.1145/3626772.3657775

[8] Tang, J., Zhang, Q., Li, Y., Chen, N., & Li, J. (2025). GraphArena: Evaluating and exploring large language models on graph computation. *International Conference on Learning Representations*.

[9] Wang, D., Zuo, Y., Li, F., & Wu, J. (2024). LLMs as zero-shot graph learners: Alignment of GNN representations with LLM token embeddings. *Advances in Neural Information Processing Systems, 37*.

[10] Huang, X., Yang, Y., Wang, Y., Wang, C., Zhang, Z., Xu, J., Chen, L., & Vazirgiannis, M. (2022). DGraph: A large-scale financial dataset for graph anomaly detection. *Advances in Neural Information Processing Systems, 35*, 22765–22777.

[11] Tang, J., Hua, F., Gao, Z., Zhao, P., & Li, J. (2023). GADBench: Revisiting and benchmarking supervised graph anomaly detection. *Advances in Neural Information Processing Systems, 36*, 29628–29653.

[12] Elmougy, Y., & Liu, L. (2023). Demystifying fraudulent transactions and illicit nodes in the Bitcoin network for financial forensics. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 3979–3990). https://doi.org/10.1145/3580305.3599803

[13] Lu, M., Han, Z., Rao, S. X., Zhang, Z., Zhao, Y., Shan, Y., Raghunathan, R., Zhang, C., & Jiang, J. (2022). BRIGHT—Graph neural networks in real-time fraud detection. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (pp. 3342–3351). https://doi.org/10.1145/3511808.3557136

[14] Liu, Y., Ao, X., Qin, Z., Chi, J., Feng, J., Yang, H., & He, Q. (2021). Pick and choose: A GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the Web Conference 2021* (pp. 3168–3177). https://doi.org/10.1145/3442381.3449989

[15] Huang, M., Liu, Y., Ao, X., Li, K., Chi, J., Feng, J., Yang, H., & He, Q. (2022). AUC-oriented graph neural network for fraud detection. In *Proceedings of the ACM Web Conference 2022* (pp. 1311–1321). https://doi.org/10.1145/3485447.3512178

[16] Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., & Yu, P. S. (2020). Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 315–324). https://doi.org/10.1145/3340531.3411903

[17] Liu, Z., Dou, Y., Yu, P. S., Deng, Y., & Peng, H. (2020). Alleviating the inconsistency problem of applying graph neural network to fraud detection. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 1569–1572). https://doi.org/10.1145/3397271.3401253

[18] Shi, F., Cao, Y., Shang, Y., Zhou, Y., Zhou, C., & Wu, J. (2022). H2-FDetector: A GNN-based fraud detector with homophilic and heterophilic connections. In *Proceedings of the ACM Web Conference 2022* (pp. 1486–1494). https://doi.org/10.1145/3485447.3512195

[19] Li, X., Liu, S., Li, Z., Han, X., Shi, C., Hooi, B., Huang, H., & Cheng, X. (2020). FlowScope: Spotting money laundering based on graphs. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(04), 4731–4738. https://doi.org/10.1609/aaai.v34i04.5906

[20] Cheng, D., Ye, Y., Xiang, S., Ma, Z., Zhang, Y., & Jiang, C. (2023). Anti-money laundering by group-aware deep graph learning. *IEEE Transactions on Knowledge and Data Engineering, 35*(12), 12444–12457. https://doi.org/10.1109/TKDE.2023.3272396

[21] Song, K., Dhraief, M. A., Xu, M., Cai, L., Chen, X., Mithal, A., & Chen, J. (2024). Identifying money laundering subgraphs on the blockchain. In *Proceedings of the 5th ACM International Conference on AI in Finance* (pp. 195–203). https://doi.org/10.1145/3677052.3698635

[22] Blanuša, J., Cravero Baraja, M., Anghel, A., von Niederhäusern, L., Altman, E., Pozidis, H., & Atasu, K. (2024). Graph feature preprocessor: Real-time subgraph-based feature extraction for financial crime detection. In *Proceedings of the 5th ACM International Conference on AI in Finance* (pp. 222–230). https://doi.org/10.1145/3677052.3698674

[23] Lin, J., Guo, X., Zhu, Y., Mitchell, S., Altman, E., & Shun, J. (2024). FraudGT: A simple, effective, and efficient graph transformer for financial fraud detection. In *Proceedings of the 5th ACM International Conference on AI in Finance* (pp. 292–300). https://doi.org/10.1145/3677052.3698648

[24] Li, K., Yang, T., Zhou, M., Meng, J., Wang, S., Wu, Y., Tan, B., Song, H., Pan, L., Yu, F., Sheng, Z., & Tong, Y. (2024). SEFraud: Graph-based self-explainable fraud detection via interpretative mask learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 5329–5338). https://doi.org/10.1145/3637528.3671534

[25] Altman, E., Blanuša, J., von Niederhäusern, L., Egressy, B., Anghel, A., & Atasu, K. (2023). Realistic synthetic financial transactions for anti-money laundering models. *Advances in Neural Information Processing Systems, 36*, 29851–29874. https://doi.org/10.52202/075280-1300

[26] Li, M., & Xu, S. (2026). Trustworthy artificial intelligence in financial decision-making: A systematic review of explainability, fairness, and accountability. Journal of Sustainability, Policy, and Practice, 2(3), 104-114.

[27] Wu, C., & Pan, Z. (2024). An Integrated Graph Neural Network and Reinforcement Learning Framework for Intelligent Drug Discovery. Journal of Advanced Computing Systems, 4(6), 19-29.

[28] Li, Y., Zhang, X., & Hao, C. (2026). Graph Neural Network-Based Cross-Market Risk Contagion Analysis between US Equity Sectors and Treasury Yields during the 2023 Regional Banking Stress. Journal of Sustainability, Policy, and Practice, 2(4), 154-168.

[29] Chen, Z., & Wang, M. (2025). Evaluating the Quality of Large Language Model-Generated Explanations in Recommendation Tasks: A Multi-Dimensional Comparative Analysis. Journal of Global Engineering Review, 3(1), 54-63.

[30] Rao, G., Trinh, T. K., Chen, Y., Shu, M., & Zheng, S. (2024). Jump prediction in systemically important financial institutions' CDS prices. Spectrum of Research, 4(2).

[31] Kang, A. & Ma, X. (2025). AI-Based Pattern Recognition and Characteristic Analysis of Cross-Border Money Laundering Behaviors in Digital Currency Transactions. Pinnacle Academic Press Proceedings Series 5, 1-19

[32] Wang, Z., & Kang, A. (2025). FTAFO: A Federated Transparent Adaptive Financial Optimizer for Reducing Third-Party Dependencies in Workflow Management. Journal of Science, Innovation & Social Impact, 1(1), 329-339.

[33] Kang, A., Zhang, K., & Chen, Y. (2025, July). AI-Assisted Analysis of Policy Communication during Economic Crises: Correlations with Market Confidence and Recovery Outcomes. In 2025 International Forum on Computing and Intelligent Systems (pp. 159-173). Pinnacle Academic Press.

[34] Li, Y. & Mo, Z. (2026). Does a Predicted 0.9 Mean 90 Percent? A Comparative Evaluation of Post-hoc Probability Calibration Methods for Default-Probability Scoring under Class Imbalance. Academia Nexus Journal 5 (2)

[35] Wu, X., Li, Y., Liu, Y., Pang, S., Wang, L., Wu, C., & Huo, H. (2019, June). Jointly detecting and retrieving vehicles from road image sequences based on CNN. In 2019 IEEE Intelligent Vehicles Symposium (IV) (pp. 2530-2535). IEEE.

[36] Wu, C., Guan, H., & Weng, H. (2024). Forecasting Hospital Resource Demand Using Gradient Boosting: An Operational Analytics Approach for Bed Allocation and Patient Flow Management. Journal of Computing Innovations and Applications, 2(1), 74-85.

[37] Chen, Y. (2026). Comparing prompting strategies and backbone choices for LLM agents on tabular credit-card fraud detection: An empirical study. Artificial Intelligence and Machine Learning Review, 7(3), 1–11

[38] Peng, Y. and Gao, S., 2026. An Empirical Comparison of XGBoost and LightGBM for Capital Expenditure Deviation Prediction in US FERC-Regulated Rate-Base Transmission Projects. Journal of Sustainability, Policy, and Practice, 2(4), pp.90-102.

[39] Li, L., Li, Y., Wu, C., Dong, H., Jiang, P., & Wang, F. (2021, January). Detail fusion GAN: High-quality translation for unpaired images with GAN-based data augmentation. In 2020 25th International Conference on Pattern Recognition (ICPR) (pp. 1731-1736). IEEE.

[40] Li, Y., Zhang, X., & Hao, C. (2026). Graph Neural Network-Based Cross-Market Risk Contagion Analysis between US Equity Sectors and Treasury Yields during the 2023 Regional Banking Stress. 2026 2nd International Conference on Digital Intelligence and Computing

[41] Chen, Y. (2026). Comparing Prompting Strategies and Backbone Choices for LLM Agents on Tabular Credit-Card Fraud Detection: An Empirical Study. Journal of Innovative Technologies 2 (1)

[42] Long, L., & Hu, J. (2026). Multi-Objective Particle Swarm Optimization for Site Selection and Policy Subsidy Maximization of Foreign Renewable Energy Enterprises in the United States. Artificial Intelligence and Machine Learning Review, 7(2), 54-69.

[43] Luo, C., & Wang, X. (2026). Latency-Throughput Tradeoffs of ONNX Runtime, TensorRT-LLM, vLLM, and Triton: An Empirical Comparison on 1B-3B Parameter LLM Inference. Spectrum of Research, 6(1).

[44] Graph Neural Network-Based Cross-Market Risk Contagion Analysis between U.S. Equity Sectors and Treasury Yields during the 2023 Regional Banking Stress

[45] Fu, H., Ming, Y., Jiang, Y., & Fan, C. (2018, August). Effective SVD-based deep network compression for automatic speech recognition. In International Workshop on Face and Facial Expression Recognition from Real World Videos (pp. 35-47). Cham: Springer International Publishing.

[46] Z. Feng, D. Zhang, Y. Wang (2024). Intraday Liquidity Patterns and Their Implications for Market Risk Assessment: Evidence from Global Equity Markets. Artificial Intelligence and Machine Learning Review

[47] Kang, A., Li, Z., & Meng, S. (2023). AI-Enhanced Risk Identification and Intelligence Sharing Framework for Anti-money laundering in Cross-Border Income Swap Transactions. Journal of Advanced Computing Systems 3 (5), 34-47

[48] Wu, Z., Feng, Z., & Dong, B. (2024). Optimal feature selection for market risk assessment: A dimensional reduction approach in quantitative finance. Journal of Computing Innovations and Applications 2 (1), 20-31

[49] Li, Y., Zhao, F., & Hu, J. (2026). Identifying Cross-Market Risk Contagion Amplifiers via Graph Attention Networks: Empirical Evidence from US Financial Stress Periods. Journal of Computing Innovations and Applications 4 (1), 164-175

[50] Shang, W., Zhao, F., & Liu, M. (2026). A Comparative Study of Spatiotemporal Clustering and Classification Approaches for Security Incident Risk Assessment in UN Peacekeeping Operations. Journal of Sustainability, Policy, and Practice, 2(3), 1-11.

[51] Yiyi, C. (2025). Federated learning-based framework for privacy-protected cross-border financial risk evaluation: Analyzing US-asia investment flows. Journal of Sustainability, Policy, and Practice, 1(4), 50-65

[52] Wei, C., Ge, L. & Brooks, N. (2024). Graph-based representation learning for financial fraud and anomaly transaction detection. Journal of Computing Innovations and Applications 2 (1), 153-164

[53] Li, Y. & Fu, X. (2026). Comparative Evaluation of Graph Neural Networks for Cross-Market Risk Contagion Path Identification in Multi-Layer Financial Networks. 2026 2nd International Conference on Artificial Intelligence and Advanced

[54] Li, Y., Buchberger, B., Liu, Y., Capone, C., Adak, T., Ojha, S., ... & Baran, P. S. (2026). General Access to C-Glycosides via Redox-Neutral Radical Cross-Coupling of Glycohydrazides.

[55] Wei, C. & Wu, C. (2024). Credit risk transmission mechanism and prevention strategies in supply chain finance: A core enterprise perspective. Artificial Intelligence and Machine Learning Review 5 (2), 101-115

[56] Ming, Y., Fu, H., Jiang, Y., & Yu, H. (2020). Variational Bayesian Group-Level Sparsification for Knowledge Distillation. IEEE Access, 8, 126628-126636.

[57] Rao, GL., Wang, Z., & Liang, J. (2025). Reinforcement Learning for Pattern Recognition in Cross-Border Financial Transaction Anomalies: A Behavioral Economics Approach to AML. Applied and Computational Engineering 142, 116-127

[58] Zhang, D. & Wang, Y. (2025). AI-driven quality assessment and investment risk identification for carbon credit projects in developing countries. 2025 International Forum on Computing and Intelligent Systems, 76-92

[59] Zhang, D. & Ma, X. (2025). Machine Learning-Based Credit Risk Assessment for Green Bonds: Climate Factor Integration and Default Prediction Analysis. Journal of Sustainability, Policy, and Practice 1 (2), 121-135

[60] Li, Y. & Zhang, S. (2025). Machine Learning-Based Credit Risk Early Warning System for Small and Medium-Sized Financial Institutions: An Ensemble Learning Approach with Interpretable Risk Indicators. Journal of Science, Innovation & Social Impact 1 (1), 372-383

[61] Li, Y., Wu, C., Li, L., Liu, Y., & Zhu, J. (2021). Caption generation from road images for traffic scene modeling. IEEE Transactions on intelligent transportation systems, 23(7), 7805-7816.

[62] Chen, Y. & Tang, T. (2026). Evaluating Prompt Engineering Strategies for Few-Shot Cyber Threat Intelligence Entity and Relation Extraction from Multi-Source Reports. Journal of Science, Innovation & Social Impact 2 (2), 153-164

[63] Trinh, TK. & Zhang, D. (2024). Algorithmic fairness in financial decision-making: Detection and mitigation of bias in credit scoring applications. Journal of Advanced Computing Systems 4 (2), 36-49

[64] Cao, H., Hu, J., & Luo, C. (2025). Behavioural Feature Analysis for Anomalous Click Detection in Mobile Advertising Environments: Toward In-App Browser-Specific Detection. Journal of Computing Innovations and Applications, 3(2), 96-105.

[65] Z. Feng, C. Ni, S. Zhou (2025). Option-Implied Information for Forward-Looking Market Risk Assessment: Evidence from Commodity Derivatives Markets. Spectrum of Research

[66] Rao, GL., Lu, T., Yan, L., & Liu, Y. (2024). A Hybrid LSTM-KNN Framework for Detecting Market Microstructure Anomalies: Evidence from High-Frequency Jump Behaviors in Credit Default Swap Markets. Journal of Knowledge Learning and Science Technology

[67] Rao, GL., Trinh, TK., Chen, Y., Shu, M., & Zheng, S. (2024). Jump Prediction in Systemically Important Financial Institutions' CDS Prices. Spectrum of Research 4 (2)

[68] Li, Y. & Ling, Z. (2026). Real-Time Multi-Risk Early Warning for Community Banks: An Application of Ensemble Anomaly Detection and Explainable Artificial Intelligence. Journal of Advanced Computing Systems 6 (2), 15-27

[69] Ge, L. & Rao, G. (2025). MultiStream-FinBERT: A Hybrid Deep Learning Framework for Corporate Financial Distress Prediction Integrating Accounting Metrics, Market Signals, and Textual Disclosures. Pinnacle Academic Press Proceedings Series 3, 107-122

[70] Yuan, D. & Zhang, D. (2025). APAC-sensitive anomaly detection: Culturally-aware AI models for enhanced AML in US securities trading. 2025 International Conference on Computer, AI, Systems and Automation, 108-121

[71] Li, Y. & Mo, Z. (2026). Backtesting Value-at-Risk and Expected Shortfall Across Calm and Turbulent Markets: An Empirical Comparison of Historical Simulation, GARCH-Family, Filtered Historical Simulation. Annals of Applied Sciences 7 (1)

[72] Cao, H., & Liu, M. (2026). Empirical Evaluation of Constraint Discovery Techniques for Business Logic Consistency Verification in Payroll Data Migration. Journal of Sustainability, Policy, and Practice, 2(3), 92-103.

[73] Li, Y. (2026). Enhancing Financial Compliance Transparency through Automated Data Governance and Intelligent Risk Reporting. Journal of Science, Innovation & Social Impact 2 (1), 299-313

[74] Y. Li, S. Fan, H. Wang (2025). Machine Learning-Based Identification of Anomalous Trading Behavior Patterns Among Asia-Pacific Investors in U.S. Securities Markets. Spectrum of Research

[75] Tang, T., & Yu, M. (2024). A Comparative Evaluation of LLM-Generated Semantic Tags versus Classical Text Features (TF-IDF, LDA, BERT Embeddings) for User-Interest Enrichment in Short-Video Recommendation. Artificial Intelligence and Machine Learning Review, 5(1), 129-140.

[76] Jiang, Z., & Wang, M. (2025). Evaluation and Analysis of Chart Reasoning Accuracy in Multimodal Large Language Models: An Empirical Study on Influencing Factors. Pinnacle Academic Press Proceedings Series, 3, 43-58.

[77] Liu, W., Rao, G., & Lian, H. (2023). Anomaly Pattern Recognition and Risk Control in High-Frequency Trading Using Reinforcement Learning. Journal of Computing Innovations and Applications 1 (2), 47-58

[78] Wu, C., Li, Y., Li, L., Wang, L., & Liu, Y. (2020, October). Caption Generation from Road Images for Traffic Scene Construction. In 2020 IEEE Intelligent Vehicles Symposium (IV) (pp. 1271-1276). IEEE.

[79] Wu, Z., Zhang, Z., Zhao, Q., & Yan, L. (2025). Privacy-preserving financial transaction pattern recognition: A differential privacy approach. Preprints

[80] Wang, M., & Zhu, L. (2024). Linguistic analysis of verb tense usage patterns in computer science paper abstracts. Academia Nexus Journal, 3(3).

[81] Li, Y. & Fu, H. (2026). Interpretable Gradient Boosting for Transmission Capital Expenditure Deviation: A Multi-Decade Benchmark on FERC-Regulated Utilities. Academia Nexus Journal 5 (2)

[82] Sun, M., & Wang, H. (2025). AI-powered customer engagement sequence analysis and conversion funnel optimization in multi-channel e-commerce marketing. Journal of Industrial Engineering and Applied Science, 3(5), 7-20

[83] Ming, Y., & Fu, H. (2020, July). Variational bayesian sparsification for distillation compression. In 2020 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). IEEE.

[84] Cao, H. (2024). Privacy-Preserving Click Pattern Anomaly Detection for Mobile In-App Browser Advertising Fraud. COMPUTING INNOVATIONS AND APPLICATIONS 2 (2), 151-161

[85] Does an LLM-Based Feedback Agent Move the Needle? An Empirical Study of Student Correctness and Hint Dependency on the ASSISTments 2017 Interaction Logs

[86] Kang, A., Zhang, K., & Chen, Y. (2025). AI-Assisted Analysis of Policy Communication during Economic Crises: Correlations with Market Confidence and Recovery Outcomes. Pinnacle Academic Press Proceedings Series 3, 159-173

[87] Yu, M., & Li, Z. (2025). An Empirical Comparison of Discrete Video Tokenization Schemes for Video Question Answering and Video Captioning. Artificial Intelligence and Machine Learning Review, 6(2), 27-50.

[88] Dong, B., Zhang, D., & Xin, J. (2024). Deep reinforcement learning for optimizing order book imbalance-based high-frequency trading strategies. Journal of Computing Innovations and Applications 2 (2), 33-43

[89] Kang, A., Xin, J., & Ma, X. (2024). Anomalous cross-border capital flow patterns and their implications for national economic security: An empirical analysis. Journal of Advanced Computing Systems 4 (5), 42-54

[90] Kang, A. & Yu, K. (2025). The Impact of Financial Data Visualization Techniques on Enhancing Budget Transparency in Local Government Decision-Making. Spectrum of Research 5 (2)

[91] Kang, A., Li, C., & Meng, S. (2025). The Impact of Government Budget Data Visualization on Public Financial Literacy and Civic Engagement. Journal of Economic Theory and Business Management 2 (4), 1-16

[92] Wu, X., Li, Y., Hao, Z., Wu, C., Wang, X., & Liu, Y. (2019, November). Image style transformation based on structure GAN. In 2019 Chinese Automation Congress (CAC) (pp. 2002-2007). IEEE.

[93] P. Li, W. Liu, Q. Zheng (2024). An Empirical Study on the Quality Assessment of Code Comments Generated by Large Language Models for Different Programming Paradigms. Spectrum of Research

[94] Hu, J., Wang, X., & Lai, J. (2026). Benchmarking Learned Cardinality Estimation Techniques for Analytical Query Processing in Data Warehouses. Journal of Computer Technology and Applied Mathematics, 3(3), 1-8.

[95] Guo, Y., Li, Y., Chen, S., Wu, Y., Poll, O., Ren, Z., ... & Baran, P. S. (2025). Scalable total synthesis of saxitoxin and related natural products. Nature, 646(8084), 351-357.

[96] Chen, Y., & Hu, J. (2026). Graph Neural Network-Based Cascading Disruption Path Identification in Multi-Tier Rare Earth Processing Networks. Journal of Global Engineering Review, 4(1), 99-112.

[97] Kang, A., Min, S., & Yuan, D. (2024). Comparative Analysis of Foreign Exchange Market Shock Transmission and Recovery Resilience Among Major Economies Under Geopolitical Conflicts: Evidence from the Russia-Ukraine Crisis. Journal of Computing Innovations and Applications, 2(1), 46-61.

[98] Min, S. & Wei, C. (2023). Comparative analysis of filter-based feature selection methods for high-dimensional data in classification tasks. Journal of Advanced Computing Systems 3 (8), 25-38

[99] Zhao, F., & Tang, T. (2026). AI-Based Sentiment Analysis for Stock Market Prediction: A Systematic Literature Review. Journal of Sustainability, Policy, and Practice, 2(3), 115-124.