International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.4 (2025) pp. 8279-8287
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Real-Time Graph-Based Anomaly Detection for Capital Markets Using Stream Processing

Saravanan Thirumazhisai Prabhagaran*

Anna University, Chennai, Tamil Nadu, India
* **Corresponding Author Email:** saravanan.prabhagaran@gmail.com- **ORCID:** 0000-0002-5247-9950

Article Info:

DOI: 10.22399/ijcesen.4211
Received : 01 March 2025
Accepted : 28 March 2025

Keywords

Graph-Based Anomaly Detection,
Capital Market Surveillance,
Stream Processing,
Graph Neural Networks (GNNs),
Real-Time Financial Analytics

Abstract:

The growing complexity and speed of trading activities in capital markets have rendered rule-based anomaly detection systems incapable of following real-time monitoring. The paper examines the model of integrating graph-based modeling and stream processing architecture on financial transactions as an effective framework to detect anomalous behaviors of financial transactions. Via the representation of the market entities and their relations as dynamic graphs and the application of the machine learning model, leveraging a graph neural network to them, it is possible to detect market anomalies, like fraud, market manipulation, and insider trading, with a more in-depth understanding of the context and a faster pace. Stream processing engines (e.g., Apache Kafka, Flink) facilitate large-scale throughput, low-latency data ingestion, whereas graph forms describe non-linear as well as dynamic relationships between brokers, traders, and instruments. In the paper, architectural, machine-learning, and compliance-based deployment criteria needed to operationalize such systems are discussed. It also covers more complex subjects, such as cross-market graph correlation, federated learning, and explainability in high-stakes settings. Results indicate that graph-based real-time anomaly detection systems bring a dramatic improvement in scalability, accuracy, and compliance, as well as represent the first essential step towards proactive financial market surveillance.

1. Introduction: The Need for Real-Time Graph Intelligence in Capital Markets

Capital markets in the modern, hyper-connected, algorithmically driven financial landscape are fast, complicated, and extremely large in terms of their transaction volumes. Billions of financial trades, including equities, derivatives, forex, and fixed income instruments, take place every day on decentralized networks consisting of brokers, exchanges, institutional traders, and automated market makers. These trades are completed with millisecond latency, producing an environment in which market manipulation, fraud, and systemic risk can propagate much more quickly than is possible with prior systems of risk management or post-trade surveillance [1][2].In the past, financial surveillance networks were dependent on rule-based analysis and threshold anomaly detection, where a pattern like front-running or wash trading was detected by checking possible patterns against pre-defined behavioral models. Nevertheless, this practice is becoming inadequate. Market abuse methods as they are used now are flexible, interconnected, and disguised by a variety of accounts or routes, as well as through complex algorithm methods. They are likely to avoid fixed detection policies through pretending to be normal, as well as using fragmented liquidity across exchanges [3][4]. This requires a paradigm change in anomaly detection, one that can mingle real-time responsiveness with context-awareness. The graph-based models appear to be the apt solution to the same. This construction of the capital market, like a graph of nodes (entities) and edges (transactions, relationships), allows analysts and surveillance systems to draw conclusions about sophisticated patterns, including the strange centrality of participants, occult loops in the transactions, and the separation of cross-institutional collusion. As opposed to linear or tabular analytics, graph theory gives the means to gather relational and topological knowledge, which is essential in discovering illicit behaviours of networked data structures [5][6].On one hand, with this development in data modeling

comes the emergence of stream processing technologies. With the need to execute sub-second decisions in the financial markets, conventional batch-based analytics cannot satisfy the performance requirements. The stream processing systems like Apache Kafka, Apache Flink, and Apache Storm support event computing, data ingestion in real-time, and time-based analytics with a latency very close to zero. With this, trading platforms and surveillance teams have the ability to work with data on the run, instead of having to be analyzed after the fact [7][8]. With stream processing, one can combine graph analytics in order to continuously build and update graph representations of market behavior as new events are being fed into it. This allows finding inconsistencies not only in raw transactional data, but in the development of behavioral relationships over time. We give some possible examples: the unusual value of the centrality score of a node, a sudden network of highly connected subgraphs might indicate potential manipulation schemes in near real-time [9]. The regulatory aspect also promotes the implementation of such real-time anomaly detection systems that are highly developed. Regulatory initiatives around the world, including MiFID II in Europe, Reg SCI in the United States, and the cyber risk framework by SEBI in India, have an operational requirement to implement effective surveillance systems to support the integrity of the markets, real-time reporting, and operational resilience by financial institutions. Detection of anomalies or failure to take prompt action to flag suspicious activity in institutions also exposes the institutions to not only monetary penalties but also to deterioration of investor confidence and tarnishing of their image [10][11]. That is why the unification of graph models and stream analytics is not a technical improvement anymore but rather a requirement of operations. The organization needs to design systems that are scalable, explainable, and that dynamically learn in a fast-changing financial environment. New developments point towards using parallelism of graph-based machine learning, real-time processing frameworks, and domain-specific knowledge graphs in conjunction being one of the most appealing ways to proactively monitor the market [12][13].In this context, the following parts of this article discuss the basic elements, the obstacles, and the advancements in real-time graph-based anomaly detection. In the second part, it starts with the stream processing platform that guides the operation of these intelligent surveillance systems.

2. Stream Processing Foundations for Capital Market Surveillance

Since the capital markets produce massive amounts of data in real time, tick data, transaction records, and order book updates, any system that will process this data must have both built-in throughput and sub-microsecond latency. The real-time requirements of market surveillance do not suit the traditional batch-oriented data systems. Consequently, stream processing architectures have become the core of real-time anomaly detection systems, which promise the possibility of continuously ingesting, transforming, and analyzing the event as it arises [5][6].There is an emerging use of modern streaming platforms, including Apache Kafka, Apache Flink, Apache Pulsar, and Amazon Kinesis, to support event-driven pipelines. Such systems also work with a rapid-fire stream of data ingestion via exchanges and brokerage APIs as well as trading algorithms. This data is consumed by the stream processors with stateful operators that can perform sliding window analysis, aggregations, joins, as well as transformations. Applied in capital markets settings, these are capabilities applied to identify inconsistencies in trading volumes, price changes, order routing, and participant behavior in the financial market on a short-term basis [7].Event-time handling is one of the stream processing developments of capital market surveillance. Unlike processing-time computation, which is pegged on the system clock, event-time processing employs the actual timestamp attached to every transaction, making sure that delayed and out-of-order data are appropriately understood. This is important in such a way that in markets where network congestion or latencies might have led to data skew, and therefore might give error alarms unless recovered by mechanisms of watermarking and line timestamps alignment [8].Scalability is one more requirement of financial stream systems. As exchanges like NSE or NASDAQ can execute millions of events in a single second, the architecture would have to allow horizontal scale-out, fault tolerance, and exactly-once processing semantics. To reach this, the contemporary pipelines use distributed computing clusters, message brokers, commonly replicated and checkpointing mechanisms that maintain the state during malfunctions. Stream graphs can even be partitioned into many jobs or tasks to make parallel processing possible, which contributes immensely to the performance as well as resilience [9]. Practically, the graph construction in real-time is established under the stream pipelines. When data comes in, it is ordered and enhanced by transformations, e.g., replacing account IDs with user entities, or combining trades with metadata, or grouping trade records into sessions. When this data is enriched, it is then piped into graph engines

either directly in-memory or by use of intermediate storage engines such as Redis, Cassandra, or time-series databases [10].Moving on to structural analytics, the following section explores the capability of graph-based models of financial markets to capture and characterize complex patterns of behavior and thus detect anomalies in a more involved way than through the application of statistical measures.

3. Graph-Based Representation of Financial Market Interactions

After the data of trading is ingested and pre-processed in real-time through the stream processing engines, the second important process in the process of anomaly detection is the transformation of this data into a structured representation that may involve complex relationships. Such is the reason why graph-based representations only come in handy. However, available traditional tabular data work on entities in isolation, unlike graphs, in which interdependent behaviors and network dynamics can be modeled; both of which tend to be the ultimate indicators of anomalous or fraudulent market operations [8][11].In order to better explain how graph representations are amenable to complex interactions in the marketplace, the following table represents typical nodes and edges that occur in real-time graph-based financial transaction graphs, as well as the typical attributes that can be added to each entity to provide more information about it.In finance, a graph is often used where the nodes can be traders, brokers, bank accounts, trading venues, or instruments. Edges refer to relationships or interactions, including trades made, money moved, or order book activity shared. Attributes can be added to these graphs, including trade volume, instrument, time, and price, to create a multidimensional picture of each relationship. These graphs change over time as more and more events are processed, and by doing so, a dynamic or temporal graph is created that enables the surveillance systems to monitor changes in behaviors and relational drift [12].An empirical illustration would include detecting collusive trading involving circular trade and coining of accounts that are involved in successive self-dealing. Such patterns can seem statistically unnoticeable in tabular data. Nevertheless, the patterns, when presented in the form of repetitive patterns in a graphical map of transactions, resemble red flags. Similarly, unusual centrality can also be identified by graphs, i.e., when a node (such as a specific broker) is suddenly made a conduit in a network that has never had the connectivity, lending itself to the possibility of a liquidity funnel or intermediary manipulation [13]. Community detection algorithms are also supported by graphs, and the algorithms can be used to subdivide the trading ecosystem into closely-knit clusters. Anomalies occur when there is a violation of expected community bounds, such as a retail account doing business in an institutional cluster or dealing with offshore counterparties in a manner incompatible with the activity in its past. There are real-time algorithms such as Louvain, Label Propagation, and Spectra Clustering that can be used to track structural changes and give a probability score to anomalous behavior [14].To additionally increase interpretability, numerous platforms use graph schemas or ontologies that encode financial domain knowledge. To give an example, a node with a label of a retail trader will possess familiar restrictions, or transaction size, counterparties, and timing, that may be tested as it is traversed through the graph. Should a retail trader begin operating according to the schema with the algorithmic timing precision, then this operation will become a possible anomaly that is to be investigated [15]. Graph representations must hence be a strong abstraction to represent the domain knowledge and monitor behavioral dynamics and subtle relational anomalies that cannot be easily surfaced. The application of advanced machine learning techniques here is founded on this structural modeling, and it is discussed in the section below.

4. Machine Learning Models for Graph-Based Anomaly Detection

When trading data has been converted to a graph structure, the rest is to determine which behaviors or nodes are abnormal. That is where graph data-specific machine learning models are deployed. Anomalies in a capital market tend to be volatile and circumstantial, and as such, there may be no strict parameters or guidelines to be followed. Machine learning offers an avenue to learn ordinary behaviors and identify anomalies based on semi-supervised, unsupervised, or self-supervised learning models [13][16]. Graph neural networks (GNNs) are among the most popular implementation approaches. These models generalize those of traditional neural networks (to graphs), allowing node embeddings to be learned to preserve both node properties and any topological context. Anomaly detectors have been trained on models like Graph Convolutional Networks (GCNs), GraphSAGE, and Graph Attention Networks (GATs) to assign a score to each node or subgraph on how abnormal it acts to previously

learned norms of behavior [17]. Most real-world market surveillance systems, however, experience the scarcity of labeled data, particularly of rare cases of fraud. To solve this, unsupervised (Autoencoders, One-Class SVMs, and Isolation Forests) are frequently applied. Such models are trained with data that are assumed to be normal and are subsequently applied to score the outliers on reconstruction error or statistical separation [18].Temporal graph modeling, an alternative direction, is to add recurrent modules to GNNs (e.g., LSTM, GRU) or make them spatio-temporal networks. They allow tracing the development of the relationships over time, which is crucial in capital markets, where numerous schemes shift slowly or respond depending on monitoring feedback. Temporal GNNs have the capability to identify patterns, including the recurrence of a trader on the edge of various suspicious subgraphs, even though individually each interaction may seem normal [19]. In recent years, self-supervised learning has become widely used because of its tendency to take the employment of substantial amounts of unlabelled data. Training of models is done to make predictions regarding masked nodes, missing edges, or future graph conditions. Any difference in the performance of the prediction may be considered as a proxy for anomaly detection. An example of this can be given when a model trained on regular periods in the market is used in the off-hours trading and gives high error; the anomaly can indicate an effort to avoid the detection systems [20]. Nevertheless, explainability in models has been an urgent issue despite such developments. It is not enough that the compliance teams and regulators have to know anomalies detected by financial institutions, but they can also justify alerting them. Consequently, explainable techniques of post-hoc, like SHAP values, counterfactual analysis, and subgraph highlighting, are being integrated into real-time dashboards to aid decision-making [21]. Such machine learning solutions significantly increase the complexity of surveillance systems and also bring along deployment loss and operational dangers, discussed in the following section.

5. Deployment Challenges and Regulatory Implications

Although graph analytics and machine learning applied to stream processing pipelines are powerful tools to perform anomaly detection tasks, there are compelling technical, operational, and regulatory issues when seeking to put such systems into practice in a capital market environment. Production environments that these systems have to work in are non-tolerant to latency, cannot tolerate any downtime, and cannot tolerate regulatory compliance failure.Since compliance, security, and explainability are parts of the real-time detection system deployments having such criticality, the accompanying table presents the principal regulatory and operational factors that institutions should take into consideration in deploying graph-based surveillance in capital markets.Complex system architecture is one of the key challenges. Real-time surveillance needs to run on various types of data, ranging from exchange feeds, broker APIs, intrinsic order management systems (OMS), and after-trade settlement systems. The data in these sources is frequently of heterogeneous format and may have different timestamp representations or identifiers. This can be addressed by incorporating both schema harmonization, metadata enrichment, and entity resolution tiers in deployment architectures so that the nodes in the graph can be used to faithfully represent unique actors and transactions [17][22]. The other key deployment aspect is latency and fault tolerance. They have relatively short windows: A capital market works within milliseconds, and high-frequency trading has very micro timescales. The construction of graphs and the inference level providing information required through machine learning should be done without causing any bottlenecks, which can slow down the occurrence of alerts or the development of blind spots. This needs the deployment of spread-out stream handling frameworks, such as Apache Flink or Kafka Streams, or in-memory databases or graph shivers, such as Neo4J or TigerGraph. Fault-tolerance is achieved by checkpointing, replicated stores of state, and failover clusters; however, the DevOps cost of configuring and maintaining such systems is high [23].Adding to the technical sphere, there lies a major regulatory and compliance issue. Financial regulations are moving towards all automated decision-making systems to be explainable, auditable, and non-discriminatory. This becomes especially difficult for the black-box models like the deep neural networks or self-supervised GNNs. Surveillance systems will thus have to be explainable by design, using interpretable features, rule-based overrides, and ex-post explanations that are standards-compliant with regulations, such as MiFID II, Dodd-Frank, and GDPR [24]. There is also data governance as a constraint. Legal provisions like GDPR, CCPA, or local data residency laws formulate strict policies that require funds-centric data to be collected, stored, and shared in a specific manner. In order to support this, more surveillance systems are taking the form of federated systems in which model

learning is done locally with data particular to the institution, and only aggregate insights or anomaly scores are reported back to central authorities. This ensures privacy, which facilitates coordination among the entities and across jurisdictions [25].Safety comes first, too. Since such systems feed and learn on sensitive trading data in real-time, they are appealing points of attack by an adversary (in the form of data poisoning, model inversion, or deceptive patterns of input to beat detection). To alleviate this, platforms employ zero-trust security solutions, TLS encrypts data being moved, and frequently check model integrity [26-30]. Finally, there is a human factor to consider. The key to success in deployment is the capacity of compliance officers and risk managers, along with IT teams, to interpret alerts, tune thresholds, and confirm models. In many organizations, there are what is known as a human-in-the-loop review cycle, where it is the analysts who first review flagged anomalies before it is escalated to enforcement. This type of collaborative model minimizes false matches and gives confidence in automated solutions.These high-dimensional complexities require a carefully considered deployment roadmap that is innovative, yet reliable, interpretable, and compliant. It is to be expected that, as the systems mature, more attention will be paid to real-time coordination across platforms, venues, and regulators, and that is what is to be done in the future-oriented consideration in the next section.

6. Real-Time Explainability and Alert Prioritization in Graph-Based Anomaly Detection

With the increasing maturity of anomaly detection systems, the challenge exists increasingly to explain and take action in real time on the output of those systems. In the high-value stakes environment of capital markets, regulatory requirements and operational mandates require that system-generated alerts be not only interpretable but also triage-ready. Though machine learning models, and especially graph neural networks (GNNs), have proven up to the task of learning complex, non-linear patterns of behavior, the outcomes such networks generate are usually quite difficult to interpret, especially by compliance officers and risk analysts, unless further contextualization is applied to the result [20][21]. The real-time explainability methods have been developed that help to narrow this gap and comprise: subgraph highlighting, attribution scoring, and rule-based enrichment. The point of these methods is to add evidence to the anomaly scores, e.g., to know which edge (transaction) or node (entity) was most helpful in triggering the alert. In addition, the historical behavioral baselines can be used in integrating alerts to provide context (separating benign but unexpected bursts (e.g., earnings day trading) and truly suspicious patterns) [10][12]. The other vital requirement is the prioritization of alerts. In a scenario where thousands of accounts and millions of transactions are going on, the false positives may saturate monitoring personnel. The current graph-based risk scorable also incorporates anomaly scores with node centrality, previous alert data, and cross-market interactivity to produce a queue on priority of investigation [22][26]. Notifications with respect to the accounts already marked as flagged or crossing regulatory borders are frequently prioritized, so a small number of human sources can monitor locations with a high density of risk. The explanatory and prioritizing layer (on the real-time level) guarantees that the machine-driven surveillance is not too far apart to be accepted by human judgment and regulatory responsibility. There is also a rising need to use such tools to scale oversight efforts as anomaly detection continues to be applied to more asset types and locations.

7. Cross-Market Graph Correlation and Federated Anomaly Detection

One of the key weaknesses of the existing market surveillance systems is the venue-specific nature that surrounds them. The majority of graph-based anomaly detectors are built and utilized in the data range of an individual exchange, broker, or institution. Nevertheless, present-day capital market anomalies (spreading risks across many infrastructures or jurisdictions) like spoofing, layering, or the pump-and-dump scheme usually take place within various venues or jurisdictions. It requires a cross-market paradigm of graph correlation and federated detection [4][13][21]. As they can relate activity across exchanges, e.g. large spike in trades in one exchange and strange orders in another, the market surveillance solutions allow them to identify more clever threat vectors. This cannot be just synchronization of real-time data but requires graph schema alignment, entity resolution, and anomaly scoring model alignment. In this respect, technologies like federated graph learning are attracting attention. Each institution is given the opportunity to locally train models with its proprietary graph and may only give high-level detectives or anomaly vectors to a central coordinator, thus ensuring the privacy of the data and allowing global patterns to be detected [21][23]. These architectures are quite applicable in the context of cross-border trading environments,

crypto-asset monitoring, as well as consortium-based market surveillance. In such instances, graph metadata exchange protocols, secure multi-party computation, and cross-market anomaly dictionaries will be paramount to make them interoperable as well as trusted. Cross-market graph analytics is an emerging area that will offer future potential to increase global integrity of capital markets and decrease systemic risk [5] [27-30].New federalized surveillance and pre-validation on multi-market correlation logically lead to a subsequent stage in which what are presently anomaly detection systems are not only devices of the institution, but systems of risk warning to engage in shared risk sentinel functions across the ecosystem. This prepares the way for the enlarged thoughts of the subsequent conclusion.

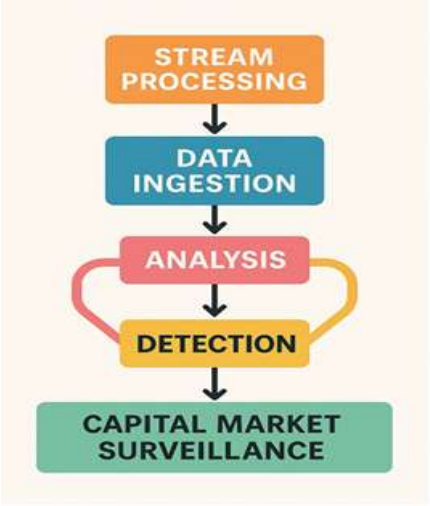

Figure 1: *Diagram illustrating the foundational stages of stream processing for capital market surveillance, from data ingestion to detection.*

Table 1: *Key Node and Edge Types in Real-Time Financial Graphs*

Graph Element	Entity Type	Example Attributes	Analytical Purpose
Node	Trader Account	ID, account type, location, risk score	User profiling and behavioral tracking
Node	Financial Instrument	Ticker, asset class, market, price volatility	Instrument clustering and anomaly propagation
Node	Broker or Exchange	Venue ID, jurisdiction, compliance tier	Cross-market connectivity and routing analysis
Edge	Trade Transaction	Volume, price, timestamp, direction	Flow tracking and cycle detection
Edge	Fund Transfer	Amount, bank ID, method, source-destination link	Movement tracing for fraud or AML analysis
Edge	Order Book Interaction	Quote type, order size, and timing granularity	Latency arbitrage and spoofing pattern detection

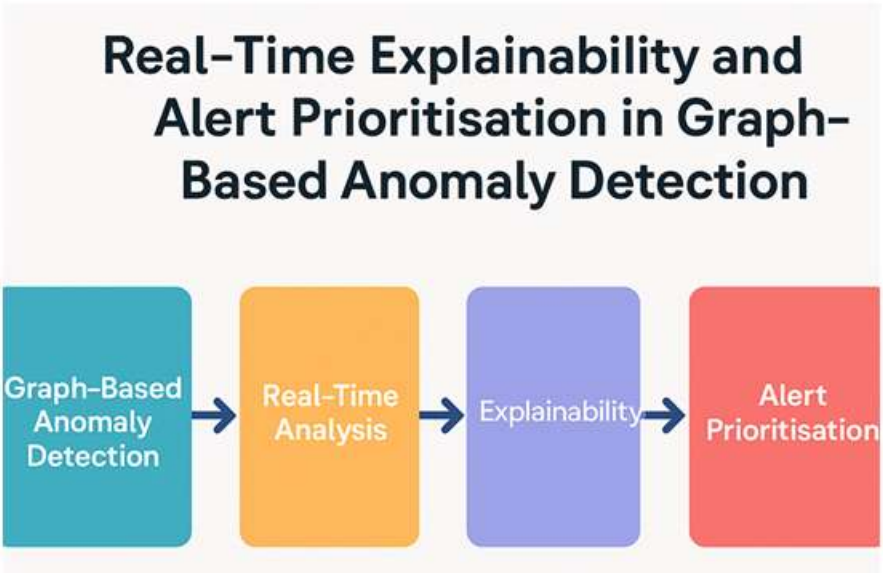

Figure 2: *Flowchart illustrating real-time explainability and alert prioritization in graph-based anomaly detection workflows.*

Table 2: *Compliance and Operational Challenges in Real-Time Financial Anomaly Detection*

Challenge Area	Requirement Description	Strategic Response
Regulatory Compliance	Explainability, auditability under MiFID II, Dodd-Frank	Use interpretable models and anomaly traceability logs
Data Privacy	GDPR, CCPA, data sovereignty regulations	Federated learning, anonymized sharing, encryption
Latency Constraints	Real-time detection within milliseconds	Use of in-memory graph engines and distributed pipelines
Model Validation	Internal model governance and external audit	Human-in-the-loop feedback and periodic retraining
Cybersecurity Risk	Threats to stream integrity and model inversion attacks	Zero-trust architecture and adversarial testing

4. Conclusions and Future Outlook

As the nature of the capital markets and their organization has changed, novel instruments of control, supervision, and credibility are required. Static-rule-based historical data-focused detection systems are insufficient in the world of algorithmic trades, liquidity fragmentation across exchanges, and almost instant settlement. Here, anomaly detection in financial systems is robust on the convergence of graph analytics, stream processing, and machine learning. By expressing entities and interactions as changing graphs, financial institutions acquire a structural prism through which to treat changing trading patterns, network abnormalities, and other forms of collusions that cannot be captured before. Together with a real-time stream processing framework, these models can be continuously updated and tested within a few seconds of data creation, and they are able to provide quick, intelligent reactions to signs of suspicious moves. Intelligence and flexibility are gained by the use of graph-based machine learning models, particularly Graph Neural Networks and temporal graph models. Such models prove especially helpful in finding weak and distant connections and changes in behavior that elude traditional statistical methods. Nevertheless, scalability, explainability, and compliance are still issues to address, which means that the stakeholders will have to work interdisciplinarily between the fields of AI, finance, and legal practice. In the future, research will move to multi-modal anomaly detection, where it will not only capture trade and transaction data but also text, visual signals, and behavioral signals, i.e., chat logs, analyst notes, social media sentiment, and biometric authentication logs. The latency will also be reduced further with edge processing, and it will also help in the detection of regionalized risk, especially in local markets. It is possible that federated learning allows collaboration between institutions while still maintaining data privacy. A more promising path is the development of self-adaptive surveillance solutions, which are able to learn online and retrain instantly in reaction to shifts in concepts. When accompanied by active explainability mechanisms, such systems will become trusted guides to human agents, as opposed to black boxes. The potential to create trusted autonomous surveillance eco-systems in capital markets is quickly emerging, and with further ethics, cybersecurity, and graph science advancements, ever more likely. Overall, it is not merely possible to detect anomalies in real-time graphs based on stream processing, but also necessary for the security of the modern financial markets. It represents a transition of reactive post-trade analysis to proactive, dynamic, and intelligent surveillance and will enable capital market infrastructure to satisfy transparency, latency, and scale requirements of the algorithmic era.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

1. Akoglu, L., Tong, H., & Koutra, D. (2015). Graph-based anomaly detection and description: a survey. *Data mining and knowledge discovery*, *29*, 626-688.
2. Rasul, I., Shaboj, S. I., Rafi, M. A., Miah, M. K., Islam, M. R., & Ahmed, A. (2024). Detecting Financial Fraud in Real-Time Transactions Using Graph Neural Networks and Anomaly Detection. *Journal of Economics, Finance and Accounting Studies*, *6*(1), 131-142.
3. Ye, C., Li, Y., He, B., Li, Z., & Sun, J. (2021, June). Gpu-accelerated graph label propagation for real-time fraud detection. In *Proceedings of the 2021 International Conference on Management of Data* (pp. 2348-2356).
4. Deng, C., Duan, Y., Jin, X., Chang, H., Tian, Y., Liu, H., ... & Zhuang, J. (2024). Deconstructing The Ethics of Large Language Models from Long-standing Issues to New-emerging Dilemmas: A Survey. *arXiv preprint arXiv:2406.05392*.
5. Gudimetla, A. R. REAL-TIME FRAUD DETECTION: INTEGRATING EVENT-DRIVEN ARCHITECTURES WITH GRAPH NEURAL NETWORKS.
6. Pazho, A. D., Noghre, G. A., Purkayastha, A. A., Vempati, J., Martin, O., & Tabkhi, H. (2023). A survey of graph-based deep learning for anomaly detection in distributed systems. *IEEE Transactions on Knowledge and Data Engineering*, *36*(1), 1-20.
7. Morishima, S. (2021). Scalable anomaly detection in blockchain using graphics processing unit. *Computers & Electrical Engineering*, *92*, 107087.
8. Ekle, O. A., & Eberle, W. (2024). Anomaly detection in dynamic graphs: A comprehensive survey. *ACM Transactions on Knowledge Discovery from Data*, *18*(8), 1-44.
9. Trinh, T. K., & Wang, Z. (2024). Dynamic graph neural networks for multi-level financial fraud detection: A temporal-structural approach. Annals of Applied Sciences, 5(1).
10. Odofin, O. T., Abayomi, A. A., Uzoka, A. C., Adekunle, B. I., Agboola, O. A., & Owoade, S. (2024). Designing Event-Driven Architecture for Financial Systems Using Kafka, Camunda BPM, and Process Engines.
11. Liu, Y., Ding, K., Lu, Q., Li, F., Zhang, L. Y., & Pan, S. (2023). Towards self-interpretable graph-level anomaly detection. *Advances in Neural Information Processing Systems*, *36*, 8975-8987.
12. George, J. G. (2023). Advancing Enterprise Architecture for Post-Merger Financial Systems Integration in Capital Markets laying the Foundation for Machine Learning Application. *Aus. J. ML Res. & App*, *3*(2), 429.
13. Goyal, A., Liu, J., Bates, A., & Wang, G. (2024). ORCHID: Streaming Threat Detection over Versioned Provenance Graphs. *arXiv preprint arXiv:2408.13347*.
14. Hu, W., Yang, F., Mao, X., Chen, R., Fan, K., & Xie, J. (2024, January). Ranking the spreading influence of nodes in weighted networks by combining node2vec and weighted K-Shell decomposition. In *2024 4th International Conference on Neural Networks, Information and Communication (NNICE)* (pp. 588-597). IEEE.
15. Liu, J., Zhang, Y., Meng, K., Xu, Y., & Dong, Z. Y. (2022, November). Risk-averse graph learning for real-time power system emergency load shedding. In *2022 IEEE PES Innovative Smart Grid Technologies-Asia (ISGT Asia)* (pp. 520-524). IEEE.
16. Liu, F., Jung, J., Feinstein, W., D'Ambrogia, J., & Jung, G. (2024, October). Aggregated Knowledge Model: Enhancing Domain-Specific QA with Fine-Tuned and Retrieval-Augmented Generation Models. In *Proceedings of the 4th International Conference on AI-ML Systems* (pp. 1-7).
17. da Silva Veith, A., de Assuncao, M. D., & Lefevre, L. (2021). Latency-aware strategies for deploying data stream processing applications on large cloud-edge infrastructure. *IEEE transactions on cloud computing*.
18. Troupiotis-Kapeliaris, A., Kastrisios, C., & Zissis, D. (2025). Vessel Trajectory Data Mining: a review. *IEEE Access*.
19. Hassan, H. B., Barakat, S. A., & Sarhan, Q. I. (2021). Survey on serverless computing. *Journal of Cloud Computing*, *10*(1), 39.
20. Bruno, D. R., Berri, R. A., Barbosa, F. M., & Osório, F. S. (2023). CARINA Project: Visual perception systems applied for autonomous vehicles and advanced driver assistance systems (ADAS). *IEEE Access*, *11*, 69720-69749.
21. Fu, D., Bao, W., Maciejewski, R., Tong, H., & He, J. (2023). Privacy-preserving graph machine learning from data to computation: A survey. *ACM SIGKDD Explorations Newsletter*, *25*(1), 54-72.
22. Khanzadeh, S., Neto, E. C. P., Iqbal, S., Alalfi, M., & Buffett, S. (2025). An exploratory study on domain knowledge infusion in deep learning for automated threat defense. *International Journal of Information Security*, *24*(1), 1-19.
23. Wang, H., Lu, Y., Shutters, S. T., Steptoe, M., Wang, F., Landis, S., & Maciejewski, R. (2018). A visual analytics framework for spatiotemporal trade network analysis. *IEEE transactions on visualization and computer graphics*, *25*(1), 331-341.
24. Zakrzewicz, M., Wojciechowski, M., & Gławiński, P. (2019). Solution pattern for anomaly detection in financial data streams. In *New Trends in Databases and Information Systems: ADBIS 2019 Short Papers, Workshops BBIGAP, QAUCA, SemBDM, SIMPDA, M2P, MADEISD, and Doctoral Consortium, Bled, Slovenia, September 8–11, 2019, Proceedings 23* (pp. 77-84). Springer International Publishing.
25. Saadati, P., Abdelnour-Nocera, J., & Clemmensen, T. (2020). Proposed system for a socio-technical design framework for improved user collaborations with automation technologies. In *Human Computer*

Interaction and Emerging Technologies. Cardiff University Press.

26. Reynisson, K., Schreyer, M., & Borth, D. (2024). GraphGuard: Contrastive Self-Supervised Learning for Credit-Card Fraud Detection in Multi-Relational Dynamic Graphs. *arXiv preprint arXiv:2407.12440*.
27. Krichen, M. (2023). Convolutional neural networks: A survey. *Computers*, *12*(8), 151.
28. Zhong, H., Yang, D., Shi, S., Wei, L., & Wang, Y. (2024). From data to insights: the application and challenges of knowledge graphs in intelligent audit. *Journal of Cloud Computing*, *13*(1), 114.
29. Venugopal, K., & Jagadeesh, B. (2024). Theoretical Insights Into User Security and Privacy in Social. *Human Impact on Security and Privacy: Network and Human Security, Social Media, and Devices: Network and Human Security, Social Media, and Devices*, 289.
30. Dou, F., Ye, J., Yuan, G., Lu, Q., Niu, W., Sun, H., ... & Song, W. (2023). Towards artificial general intelligence (AGI) in the internet of things (iot): Opportunities and challenges. *arXiv preprint arXiv:2309.07438*.