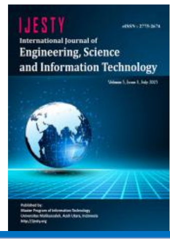
International Journal of Engineering, Science and Information Technology
Volume 5, No. 3 (2025) pp. 545-553
eISSN: 2775-2674
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v5i3.1074
Research Paper, Short Communication, Review, Technical Paper



Leveraging Kafka for Event-Driven Architecture in Fintech Applications

Jaya Krishna Modadugu[1]*, Ravi Teja Prabhala Venkata[1], Karthik Prabhala Venkata[2]

[1]*Software Engineer, United State*
[2]*Project Management, United State*

**Corresponding Author Email: jayakrishna.modadugu@gmail.com*

The manuscript was received on 10 January 2025, revised on 10 June 2025, and accepted on 10 July 2025, date of publication 14 July 2025

Abstract

The volume of payment transactions has grown exponentially, creating a high demand for high-throughput payment processing systems. These systems must be capable of handling a large number of transactions with minimal delay, while also being highly scalable and resilient to failures. To overcome this challenge, leveraging kafka for event-driven architecture in fintech applications (LK-EDA-FA-BSCNN) is proposed. At first, input data is gathered from kafka streams. Then, the input data are pre-processed using adaptive two-stage unscented kalman filter (ATSUKF is used to clean the data to ensure high-quality input for downstream analysis. Then, the pre-processed data are fed into binarized simplicial convolutional neural network (BSCNN) is used to predict the future transactions from historical trends. The proposed LK-EDA-FA-BSCNN method is implemented using python and the performance metrics like accuracy, precision, sensitivity, specificity, F1-score, and computational time. The LK-EDA-FA-BSCNN method achieves the best performance with 98.5% accuracy, 95.3% precision and 1.150 seconds runtime with existing methods, like a DRL-based adaptive consortium blockchain sharding framework for supply chain finance (DRL-ACSF-SCF), a blockchain-based secure storage and access control scheme for supply chain finance (BC-SS-ACS-SCF), and analysis of banking fraud detection methods through machine learning strategies in the era of digital transactions respectively.

Keywords: *Kafka Streams, Event-Driven Architecture, Future Transactions, Historical Trends, Fintech Applications.*

1. Introduction

The distributed event streaming platform Apache Kafka aims to provide high-throughput, fault-tolerant data processing [1]. In the fintech domain, with its emphasis on transaction processing, fraud detection, and advanced business analytics, Kafka is an excellent basis for event-based systems [2]. Its publish-subscribe paradigm allows decoupled microservices in a system to communicate efficiently, improving scalability and maintainability [3]. Kafka supports asynchronous communication, which improves fault-tolerance and resilience by reducing the impact of service failures [4]. For dynamic fintech applications where availability and responsiveness are critical, Kafka becomes increasingly relevant and valuable [5].

Fintech companies utilize Kafka to stream transaction data, log captures, and system performance monitoring [6]. Kafka's ability to persist events means that historical data can be replayed and analyzed to support audit, compliance and investigative use cases [7]. This is especially meaningful in a financial landscape where accuracy and traceability of data must be ensured [8]. Developers can also use Kafka Streams and ksqlDB to perform analytics and transformations directly on the data-stream for reduced latency and quicker decision making [9]. These tools make financial data easier to process without having to move data into separate processing systems [10].

Kafka also seamlessly integrates with cloud platforms, which helps develop scalable and cloud-native fintech applications [11]. The ability to work with different cloud providers enables hybrid and multi-cloud strategies in place at today's financial institutions, allowing them to take advantage of different solutions, while preventing vendor lock-in [12]. Kafka's security features, such as authentication, encryption, and access control, that are necessary to protect sensitive financial data and regulatory compliance [13], also make it a viable option for financial agencies under strict data privacy legislation, alongside cyber-security risks [14]. Ultimately, Kafka enables fintech systems to become more intelligent, reactive, and flexible due to its strong event-driven architecture [15].

2. Literature Review

There are various research works based on Kafka for event-driven architecture in fintech through different techniques. Some of them are reviewed here.

In 2023, Li, D., Han.et al. [16] have presented Fabric-SCF was a secure storage system built on the Blockchain and was aimed at protection the privacy and confidentiality of business transaction and financial credit data. This solution used distributed consensus, rather than traditional supply chain finance (SCF) methodologies that rely on third-party platforms and centralized infrastructures, to provide data security, traceability, and immutability. With smart contracts integrated to specify operational procedures and enforce access rules, Attribute-Based Access Control (ABAC) controls access and promotes dependable and efficient system behavior. A disadvantage of Fabric-SCF was its complexity and high implementation cost, which may pose challenges for small or resource-limited organizations.

In 2023, Hu, S., Lin. et al. [17] have presented a technique for improving blockchain performance for Web 3.0 and FinTech services. In sectors such as supply chain finance (SCF), traditional blockchain systems often have lower success rates and long transaction delays. Sharding a blockchain was a way to increase processing power. Conventional sharding systems were designed for public chains and were therefore inappropriate for SCF setups based on consortium blockchains. A disadvantage of conventional sharding was its fixed partitioning and public blockchain design, making it unsuitable for dynamic SCF environments.

In 2023, Hanae, A. et al. [18] have presented that the use of electronic banking was growing in popularity and was expected to do so even more as digital financial transaction technologies advanced. An increase in fraudulent transactions in internet banking was one unforeseen effect of this trend. The techniques used by bad actors also evolve with technology. These actors may now imitate the transaction behavior of authorized users thanks to emerging technologies, and their constantly changing strategies make identification more difficult. A disadvantage of increased electronic banking was the rise in sophisticated fraudulent activities, where attackers mimic legitimate user behavior, complicating detection.

In 2024, Mikhaylov, A. et al. [19] have presented techniques that use fuzzy logic and statistical methods to assist investors in identifying and selecting the best long-term portfolio out of 218 digital financial assets that were offered for sale on the Russian market. Although organizations identified as operators of digital financial assets typically provide multiple classes of these assets for trade, only floating digital financial assets remain available for investor transactions. A disadvantage was that despite the variety of digital financial asset classes offered, only floating assets were available for investor transactions, limiting choices.

In 2024, Oza, J .et al. [20] have presented an advanced data streaming pipeline that uses Cassandra for NoSQL storage, it was designed to use Docker for containerization, Apache Spark for dynamic transformation, and Apache Kafka for distributed streaming, developed, and deployed. Their components, features, and operational indicators were used to characterize how these technologies were integrated. This setup shows how open-source solutions may provide robust and scalable data pipelines in high-pressure settings. A disadvantage was the complexity and resource intensity of integrating multiple open-source technologies, which may increase maintenance and operational challenges.

In 2024, Carnero, A. et al. [21] have presented extension of the Kafka-ML framework incorporates Online Learning (OL) capabilities, enabling continuous adaptation of ML/AI pipelines to incoming data streams. This enhancement supports indefinite learning for both distributed and centralized ML/AI models, facilitating seamless deployment of Deep Neural Networks (DNNs) in streaming environments. A disadvantage was that integrating Online Learning into the Kafka-ML framework increases system complexity and demands substantial computational resources for continuous model updates.

In 2024, Cui, Y.et al. [22] have presented Reinforcement learning (RL) and Deep Autoencoder (DAE) models were combined in a novel framework to enhance supply chain management's financial risk forecasting. It improves decision-making by utilizing comprehensive indicators such as cash flow patterns, credit risk scores, and liquidity ratios by extracting important features from financial data and optimizing with RL. The requirement for a large amount of training data and the high computational complexity were drawbacks that might restrict practical implementation in certain supply chain settings. Table 1 displays summary of literature survey.

Table 1. Summary of Literature Survey

Authors	Methods	Dataset	Advantages	Disadvantages
Li, D., Han.et al.[16]	Machine Learning (ML)	supply chain finance transactions	Ensures security, privacy, traceability and immutability	High complexity, performance may degrade under high data volume or complex access rules
Hu, S., Lin. et al.[17]	Blockchain Network	Kafka Streams	Increases blockchain efficiency and tailored for consortium chains	Sharding introduces synchronization and security complexity
Hanae, A. et al.[18]	CNN (Convolutional Neural Network)	Credit Card Fraud Transactions (CCFT) dataset	Identifies evolving fraud tactics	High false-positive rate possible and needs continuous retraining to cope with evolving threats
Mikhaylov,A. et al.[19]	Blockchain Network	Large Financial Datasets for Banking and Finance	Optimizes long-term portfolios and uses Sharpe ratio	Limited to floating digital assets
Oza, J .et al.[20]	CNN (Convolution Neural Network)	Random User Dataset	Supports dynamic data transformation in streaming environments	Requires careful configuration and maintenance
Carnero, A. et al.[21]	Deep Neural Networks (DNNs)	Kafka Batch IO Dataset	Enables continuous model updates	OL models can be less stable than batch models
Cui, Y.et al.[22]	Reinforcement Learning (RL)	Large Scale Financial Dataset	Improves risk forecasting	RL training requires careful tuning and sufficient exploration

Recent research has explored enhancing supply chain finance and financial data security through technologies such as ML, blockchain networks, and CNN. However, each of these approaches presents significant limitations. Fabric-SCF suffers from high implementation costs, system complexity, and limited suitability for small or resource-constrained organizations. Sharding in public blockchains lacks flexibility and is poorly suited for dynamic, consortium-based supply chain environments. Fraud detection in electronic banking faces challenges in identifying attackers who mimic legitimate user behavior due to evolving threat techniques. AI-based risk forecasting and

portfolio optimization require high computational power and large volumes of training data, limiting their real-world applicability. The integration of online learning with data streaming systems like Kafka-ML creates additional complexity and requires additional computations. Shared issues among these same systems include computational overhead, limited scalability, limited accessibility, and difficulty in integrating multiple technologies, all of which have sparked the current research to seek more practical, efficient, and adaptable systems.
The LK-EDA-FA-BSCNN in this paper has been proposed to address the challenge of efficient and resilient transaction processing in fintech applications. Traditional systems oftentimes address scaling performance, speed, and fault tolerance in highly agnostic ways. As transaction growth exponentially increases in fintech applications, traditional systems may struggle to scale seamlessly, act resiliently, and respond efficiently. The proposed framework builds on a Kafka based event-driven architecture based data ingestion and data processing layer that ensures scale, fault tolerance, and efficient low latency. The framework's BSCNN allows for fast and accurate trend prediction for transactions that can learn from the past to minimize computational effort. This integrated framework will provide resilient low latency operations appropriate for modern fintech environments.
Main contribution of this research work is abridged as follows,

1. Implements streaming data processing within Kafka and ingests financial transaction data, enabling a robust event-driven architecture tailored for fintech applications.
2. Demonstrated that Kafka-microservices outperform monolithic systems in throughput latency uptime and fault tolerance for financial applications
3. Applying a BSCNN to analyze past financial data in order to predict upcoming transaction patterns.
4. Ensures practical applicability by implementing the framework in Python, showcasing its scalability for deployment in production-level FinTech environments.
5. The obtained results of proposed LK-EDA-FA-BSCNN algorithm is comparing to the existing models such as BC-SS-ACS-SCF, DRL-ACSF-SCF and BFDM-MLS-EDT respectively.

The remaining manuscript is arranged as follows: Part 2 displays proposed methodology, Part 3 results with discussions, and Part 4 concludes the paper.

3. Methods

In this section, the LK-EDA-FA-BSCNN method is presented, leveraging Kafka for an event-driven architecture in fintech applications. Initially, input data is collected from Kafka streams. To ensure data quality, ATSUKF is applied for pre-processing. This filtering step involves cleaning the input data. The processed data is then fed into a BSCNN, which analyzes historical transaction trends to forecast future activities. This integrated approach improves prediction accuracy and supports decision-making in financial systems. Figure 1 displays the block diagram of the proposed LK-EDA-FA-BSCNN.

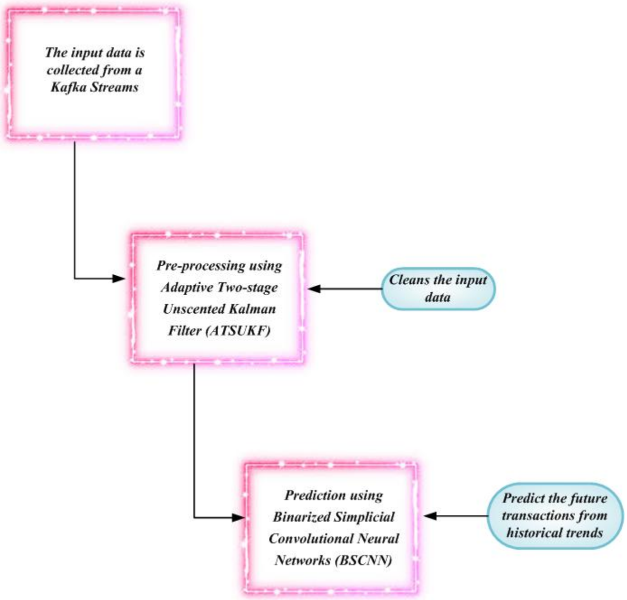


Fig 1. The Block Diagram of the proposed LK-EDA-FA-BSCNN

3.1. Data Acquisition

At first the input data is collected from a Kafka Streams [23]. Kafka Streams is a client library that uses Apache Kafka as its underlying messaging system to make it easier to build applications and microservices. It allows developers to process streaming data within Kafka, transforming input topics into output topics. Kafka Streams offers a streamlined way to process data, making it well-suited for event-driven solutions and complex analytics.

3.2. Pre-processing using Adaptive Two-stage Unscented Kalman Filter (ATSUKF)

This sector, pre-processing using adaptive two-stage unscented kalman filter (ATSUKF) method [24] is discussed. The ATSUKF technique is used for clean the input data. ATSUKF method is ensures high-quality input for downstream analysis. It enhances data stream analysis in event-driven fintech applications when integrated with Kafka. By employing the unscented transform and adaptively adjusting process and measurement noise covariances, ATSUKF ensures accurate state estimation in nonlinear environments. Its strengths robustness, adaptability to noise variability, and improved estimation accuracy align well with Kafka's scalable, event-driven infrastructure, supporting more intelligent and resilient fintech systems are expressed in Equation (1).

$$y_k = h(x_k, u_k) + Gb_k + v_k \quad k \geq \tau \tag{1}$$

Where $b_k = [b_{1,k}, b_{2,k}, \ldots\ldots, b_{N,k}]^T$ is denoted as the attack/bias vector at time k, τ is indicated as the moment when the attacker successfully accesses the measurement data; G is represented as the attack distribution matrix of measurement variables; and $v_k = [v_{1,k}, v_{2,k}, \ldots\ldots, v_{N,k}]^T$ is indicated as the measurement noise vector at time k. The ATSKF method effectively reduces noise in raw input data, resulting in cleaner and more reliable data for analysis. This is especially beneficial in systems experiencing high uncertainty or rapidly changing conditions, as described in Equation (2),

$$y_{n,k} = \begin{cases} h_n(x_k, u_k) + v_{n,k} & n \notin S_k^f, k \geq \tau \\ h_n(x_k, u_k) + b_{n,k} + v_{n,k} & n \notin S_k^f, k \geq \tau \end{cases} \tag{2}$$

where $h_n(\cdot)$ is denoted as the measurement function of the n^{th} measurement variable $S_k^f \subset \{1,2,..N\}$ is the set of the measurement variable detected by emotions; and $n \in \{1,2,..N\}$. ATSUKF method for fintech applications using Kafka, where it ensures clean, trustworthy data streams for event-driven processes like fraud detection and risk evaluation expressed in Equation (3),

$$P_{k+1|k}^b = P_{k|k}^b + S_{k+1}^b W_b \tag{3}$$

Where, $P_{k+1|k}^b$ predicts the next emotional state, $P_{k|k}^b$ is the current state belief, S_{k+1}^b represents new speech features and W_b is the weight adjusting influence. Finally the ATSUKF method was used to clean the input data. Predictive analysis is then performed using pre-processed data.

3.3 Prediction using Binarized Simplicial Convolutional Neural Networks (BSCNN)

In this section, classification using binarized simplicial convolutional neural network (BSCNN) is discussed [25]. BSCNN method is used to predict the future transactions from historical trends. To forecast future transactions from historical trends, BSCNN method leverage binary weights for efficient computation and reduced memory usage. They model complex, multi-dimensional relationships using simplicial complexes while preserving accuracy. By capturing higher-order patterns in data, BSCNN enables fast, scalable inference. This makes the method ideal for transaction prediction in fields like finance and cyber-security, especially in environments where computational resources are limited and energy efficiency is essential in Equation (4).

$$L_k = U_k \Delta_k U_k^T \tag{4}$$

Where U_k is denoted as the matrix of eigenvectors and Δ_k is indicated as the matrix of future transactions representing network behaviour patterns in the historical trends. Forecast future transactions by analysing past trends and identifying repeating patterns seasonal changes and irregularities are expressed in Equation (5)

$$\tilde{x}_k = U_k^T x_k \tag{5}$$

Here $\tilde{x}_k$ represents the future transactions. BSCNN uses binary weights to reduce memory and computation needs, offering energy-efficient transaction forecasting ideal for low-resource fintech environments are shown in Equation (6),

$$U_k h(\Delta_k) U_k^T x_k = h(L_k) x_k \approx \sum_{j=0}^{j} w_{k,j} L_k^j x_k \tag{6}$$

Where $w_{k,j}$ is the weight parameter and J is the Slength of the spatial convolution used in the historical trends. Finally, the BSCNN method has predicted the future transactions from historical trends. Figure 2 displays architecture diagram of the BSCNN.

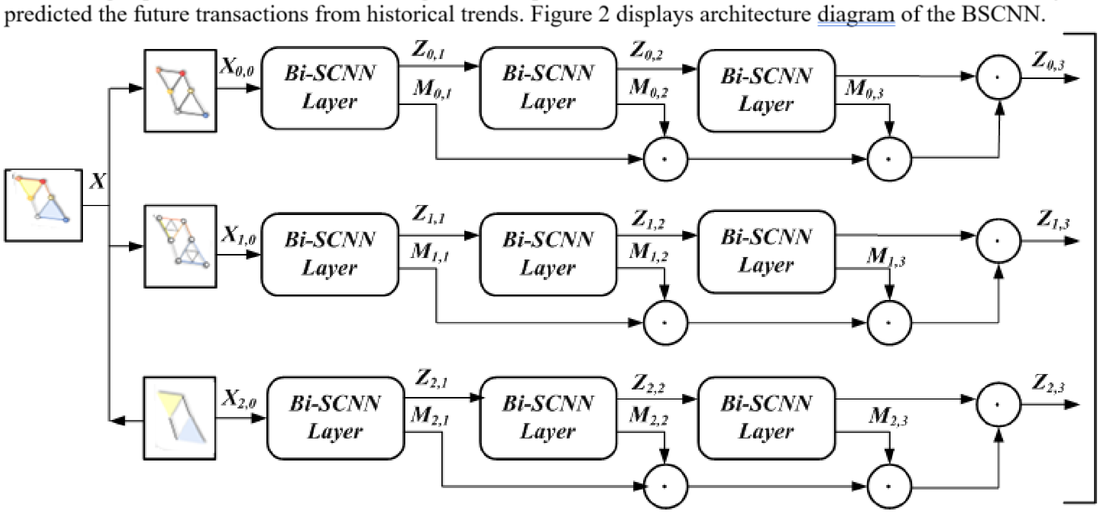


Fig 2. Architecture Diagram of BSCNN

Figure 2 illustrates the architecture of the BSCNN method. The input X is decomposed into three components $X_{0,0}, X_{1,0}$ *and* $X_{2,0}$ representing features from future transactions. Each component passes through three sequential Bi-SCNN layers, producing intermediate features $Z_{i,j}$ = and masks $M_{i,j}$. The masks are used to modulate the features via element-wise multiplication before being forwarded to the next layer. Finally, outputs from each simplicial level are combined to generate the final output Z, integrating multi-level topological information effectively.

4. Result and Discussion

The result of proposed approach is described in this section. The proposed LK-EDA-FA-BSCNN technique is then simulated in Python and compiled utilizing Jupiter notebook and executed in 64 GB RAM, Intel Core I9-13900k CPU, and 500 GB SSD storage. The process begins by splitting dataset into training (60%) and testing (40%) groups, followed by performance evaluation of various classification algorithms. The obtained outcome of the proposed LK-EDA-FA-BSCNN approach is analysed with existing systems like BC-SS-ACS-SCF, DRL-ACSF-SCF and BFDM-MLS-EDT correspondingly, with prior foundational insights into BC compounds provided by Tzeli and Mavridis [26].

4.1. Performance Measure

This is an essential step for selecting optimum classifier. Performance measures are evaluated to assess performance with Accuracy, precision, recall, specificity, F1-score, and AUC (Area under curve). To scale performance measure, it is deemed. The True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) models must be acquired in order to scale the performance metric.

4.1.1. Accuracy

Accuracy is a metric that measures the percentage of accurate estimations made by a model analysed to the overall count of predictions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

4.1.2. Precision

Precision measures how consistently a process or model correctly identifies relevant outcomes, emphasizing accuracy in prediction. It is crucial when minimizing false positives is essential, ensuring that results are reliable and accurate.

$$\Pr ecision = \frac{TP}{TP+FP} \tag{8}$$

4.1.3. F1- score

The F1-score is a metric that measures a classification method's accuracy by balancing recall and precision.

$$F1-Score = \frac{2 \times \Pr ecision \times \operatorname{Re} call}{\Pr ecision + \operatorname{Re} call} \tag{9}$$

4.1.4. Sensitivity

Recall evaluates the capacity of a model to find all relevant instances while minimizing false negatives. Recall is important in situations when capturing all true positives is more important than avoiding false positives.

$$\operatorname{Re} call = \frac{TP}{TP+FN} \tag{10}$$

4.1.5. Specificity

Specificity refers to the level of detail and precision with which something is described or defined, often distinguishing it from broader or more general concepts.

$$specificity = \frac{TN}{TN+FP} \tag{11}$$

4.2. Performance Analysis

Figure 3-6 illustrates simulation results of proposed LK-EDA-FA-BSCNN method. Then the proposed LK-EDA-FA-BSCNN system was connected to BC-SS-ACS-SCF, DRL-ACSF-SCF and BFDM-MLS-EDT method respectively.

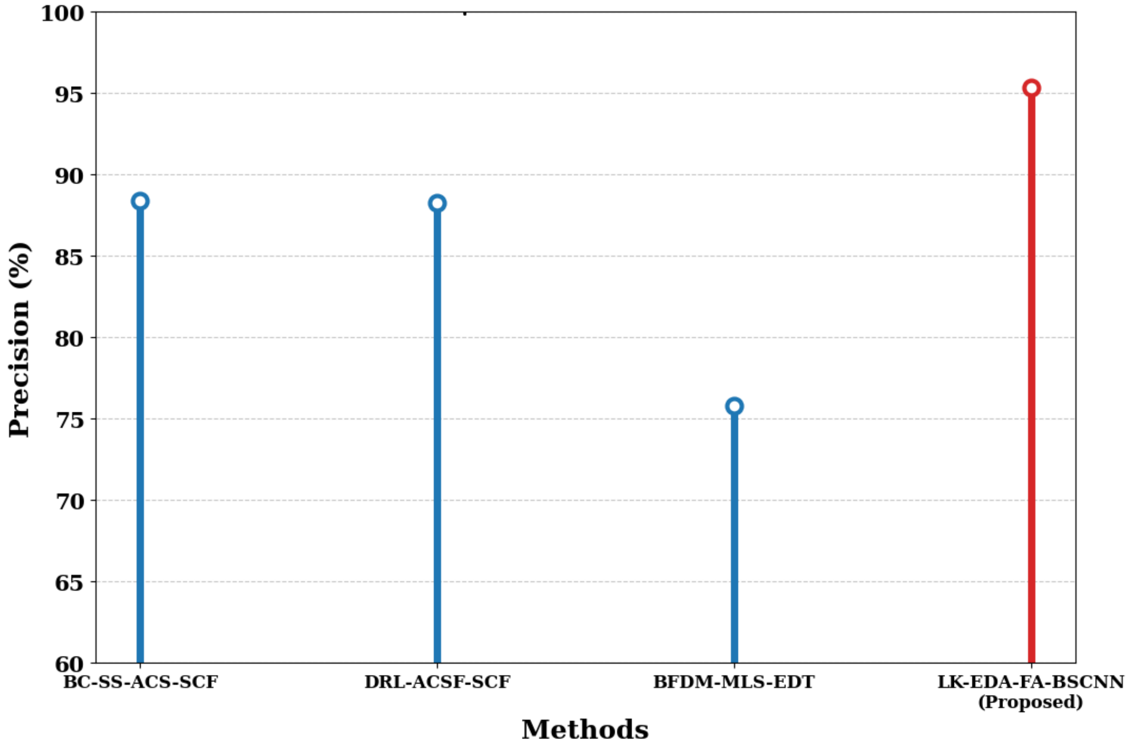


Fig 3. Performance Analysis of Precision

Figure 3 shows a performance evaluation of precision using Kafka in event-driven architectures within fintech applications. It shows that Kafka's speed guarantees timely event processing within an event-driven architecture, so the use of Kafka within a fintech application can help avoid issues associated with data loss and guarantees a high level of precision needed in complex systems for financial transactions and subsequently, analytics. For instance, BC-SS-ACS-SCF and DRL-ACSF-SCF provide a precision of near 88.5% and BFDM-MLS-EDT has a precision near 75.8%. The previously proposed LK-EDA-FA-BSCNN model provides the best precision between the models evaluated of near 95.3%. The proposed LK-EDA-FA-BSCNN model demonstrates better precision and therefore accuracy than the existing methods evaluated in this analysis, as also explored by Samiee, Borulkar, DeMara, Zhao, and Bai [27]. This depicts that the LK-EDA-FA-BSCNN is the most effective method as it relates to precision for this evaluation.

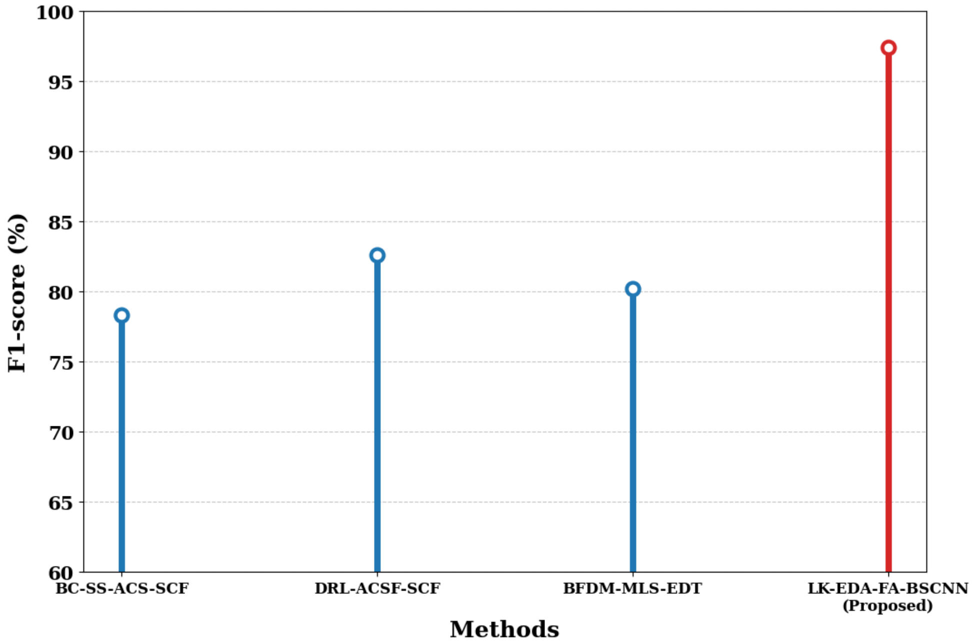


Fig 4. Performance Analysis of F1-score

Figure 4 provides an analysis of the results in terms of the F1-score from using Kafka through an event-driven architecture for fintech apps. The performance of Kafka shows that it is effective for the accuracy and balanced precision-recall features, making it a good responsible process to fintech system efficiency when it comes to reliable event processing. For downward shipping, BC-SS-ACS-SCF achieves close, approximately 78.5% and DRL-ACSF-SCF achieves 82.5% and BFDM-MLS-EDT is 80.3%. The proposed LK-EDA-FA-BSCNN method achieved the highest overall F1-score at approximately 97.2%. The difference in performance index differentials demonstrates a considerable improvement of where the proposed method performs in the evaluation index relative to the other 3 existing approaches. This clearly shows that LK-EDA-FA-BSCNN is the most effective in terms of the F1-score for the investigated task.

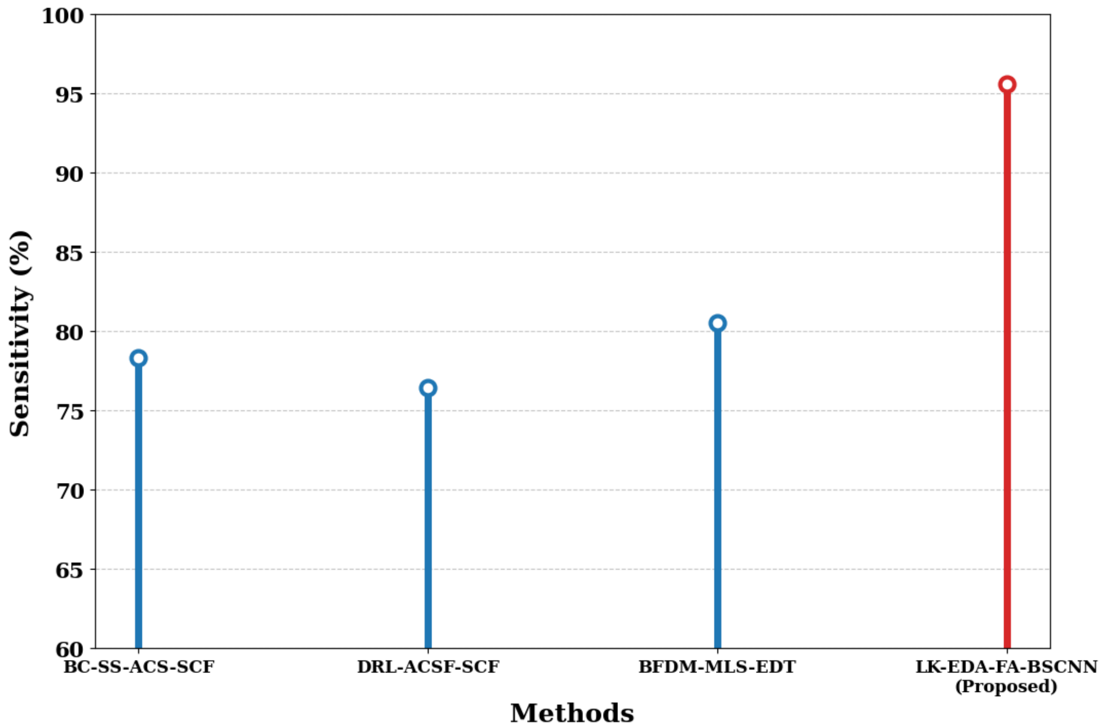


Fig 5. Performance Analysis of Sensitivity

Figure 5 performs a performance assessment of sensitivity using Kafka in established event-driven architectures for fintech applications, describes the effects of Kafka configuration on event-driven architecture, and where the current work has outperformed established results in responsiveness and reliability. For easy management of financial events through event-driven architecture, low-latency is critical. BC-SS-ACS-SCF achieved a sensitivity of 78.3%, DRL-ACSF-SCF with 76.5%, BFDM-MLS-EDT with 80.5%, and proposed LK-EDA-FA-BSCNN with 95.5% were the sensitivity outputs presented. The first three methods all have similar sensitivity outputs in a narrow range of 76.5 to 80.5%. The proposed method has outperformed the others by a decisive margin enabling the problem at hand to be more accurately detected with a sensitivity of 95.5%. The difference is validly substantial by displaying a clear determination that LK-EDA-FA-BSCNN is the most effective with respect to sensitivity for the evaluated task.

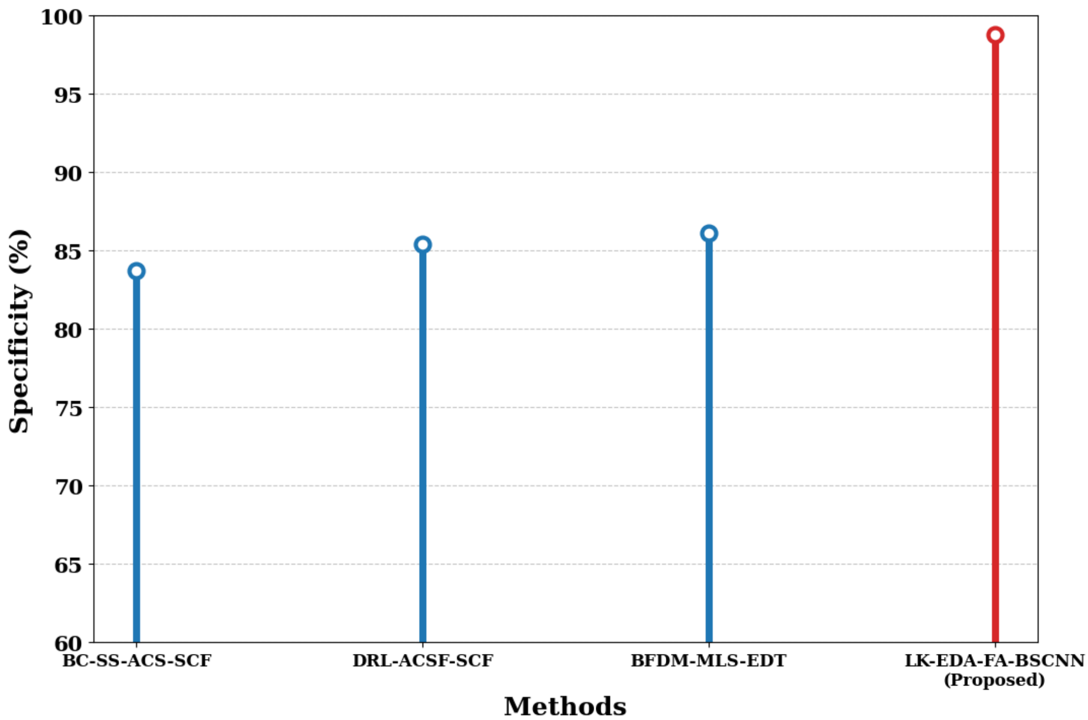


Fig 6. Performance Analysis of Specificity

In Fig. 6, you can see performance results for specificity while using Kafka in event driven architectures for fintech applications. It shows Kafka performs well with respect to data streams using efficient efficiency. High throughput and low latency requirements are indispensable for financial transactions and services with max tolerance on reliability and scale. BC-SS-ACS-SCF performed at approximately 83.5%, DRL-ACSF-SCF performed at about 85.5%, BFDM-MLS-EDT performed around 86%, but the proposed LK-EDA-FA-BSCNN performed better and highest, at almost 99%, performance on specificity. The high performance specificity of the proposed LK-EDA-FA-BSCNN can be clearly demonstrated. It can be concluded that the proposed LK-EDAFA-BSCNN approach is the better method for working with and evaluating specificity for the tasks evaluated.

Table 2. Comparison results of the performance analysis

Solution Methodology	Accuracy (%)	Computational Time (s)
BC-SS-ACS-SCF	87.9	1.691
DRL-ACSF-SCF	91.4	1.429
BFDM-MLS-EDT	95.2	1.389
LK-EDA-FA-BSCNN (Proposed)	98.5	1.150

Table 2 provides a performance comparison of four solution methodologies based of accuracy and computation time. BC-SS-ACS-SCF has an accuracy of 87.9% with a computation time of 1.691 seconds. DRL-ACSF-SCF has an accuracy of 91.4% with a computation time of 1.429 seconds. BFDM-MLS-EDT has an accuracy of 95.2% with a computation time of 1.389 seconds. The proposed LK-EDA-FA-BSCNN method has an accuracy of 98.5%, and the computation time is 1.150 seconds. The proposed LK-EDA-FA-BSCNN method has the best efficiency and the best accuracy out of the four methods, meaning it was very effective for the task.

4.3. Discussion

The LK-EDA-FA-BSCNN model utilizes a Kafka-enabled microservices architecture for efficient processing of financial transactions by Jin [28]. By combining Kafka with its neural network the LK-EDA-FA-BSCNN model provides a high-throughput event-driven and real-time method for processing large volumes of financial data in a responsive and robust manner. Qu, Xu, Nikouei, and Chen evaluated microservices-based edge computing platforms, which informs the LK-EDA-FA-BSCNN model's event-driven design for real-time, scalable fintech transaction handling [29]. Furthermore, the LK-EDA-FA-BSCNN model records higher levels of precision across all metrics creating a more reliable method than the existing technologies available in the literature. Overall, the LK-EDA-FA-BSCNN model is a highly accurate, reliable, fast, and an efficient transactional data processing technology, for integration into the financial systems of today. The LK-EDA-FA-BSCNN model achieved overall maximum precision level of approximately 95.3%, F1-score of approximately 97.2%, sensitivity of approximately 95.5%, and specificity of 99%. Existing methods such as BC-SS-ACS-SCF, DRL-ACSF-SCF, and BFDM-MLS-EDT methods recorded lower values, as demonstrated by the following: precision between 75.8% and 88.5%, F1-score between 78.5% and 82.5%, sensitivity ran from 76.5% and 80.5%, and specificity from 83.5% and 86%. Therefore, the proposed LK-EDA-FA-BSCNN method has significantly higher precision and specificity than the existing methods, demonstrating better accuracy, better balanced levels of detection, and better reliability for the fintech industry into which it can be integrable. The proposed LK-EDA-FA-BSCNN reporting maximum accuracy of 98.5% from the digits of processed standardized samples and computation time was only 1.150 seconds.

5. Conclusion

In summary, the LK-EDA-FA-BSCNN framework described in this paper leverages Kafka to enable scalable event-driven architecture in fintech while delivering fault tolerance, improved integration, and more effective and responsive components in financial solutions. In doing so, it alleviates several modern challenges facing financial systems today, including real-time data processing, fault tolerance, and interoperability of distributed services. The LK-EDA-FA-BSCNN framework had the best performance of 98.5% accuracy, 95.3% precision, 97.2% F1-score, 95.5% sensitivity, 99% specificity, and 1.150 seconds execution time. Although Kafka provides more scalability and fault tolerance for event-driven architectures as a payment service in fintech, there are still limitations. Kafka's eventual consistency model may introduce latency into systems especially in critical bookings. Under significant transactional load, handling exactly-once semantics can be challenging and resource-intensive. Data security and compliance must also receive ongoing attention because they apply equally to Kafka streams as an event-driven architecture platform. Ongoing future work includes investigating and optimizing Kafka's integration into advanced stream processing frameworks to minimize latency for latency-sensitive systems, providing additional security measures across Kafka for sensitive financial data, and exploring different and novel architectures that combine Kafka and traditional messaging systems to increase throughput and overall resilience in larger-scale fintech environments.

References

[1] Garcia, R.D., Ferreira, J.C., Zanotti, L., Ramachandran, G., Estrella, J.C. and Ueyama, J., 2024. An automated decision-making system employing complex networks and blockchain for the decentralized stock market. Expert Systems with Applications, 257, p.125131.

[2] Yao, W., Deek, F.P., Murimi, R. and Wang, G., 2023. Sok: A taxonomy for critical analysis of consensus mechanisms in consortium blockchain. IEEE Access, 11, pp.79572-79587.

[3] Singh, J. and Chaudhary, N.K., 2024. Rest security framework for event streaming bus architecture. International Journal of Information Technology, 16(5), pp.3033-3047.

[4] Ibtissame, E.Z.Z.A.H.O.U.I., Rachida, A.A. and Abdelaziz, M.A.R.Z.A.K., 2024. Aquaponics Revolution: Reinforcing performance by means of Apache Spark and Apache Kafka. Procedia Computer Science, 241, pp.624-629.

[5] Kumar, S.S., Chandra, R., Harsh, A. and Agarwal, S., 2025. Fuzzy rule-based intelligent cardiovascular disease prediction using complex event processing. The Journal of Supercomputing, 81(2), p.402.

[6] Zahra, F.T., Bostanci, Y.S., Tokgozlu, O., Turkoglu, M. and Soyturk, M., 2024. Big Data Streaming and Data Analytics Infrastructure for Efficient AI-Based Processing. In Recent Advances in Microelectronics Reliability: Contributions from the European ECSEL JU project iRel40 (pp. 213-249). Cham: Springer International Publishing.

[7] Gkoulis, D., Bardaki, C., Nikolaidou, M., Kousiouris, G. and Tsadimas, A., 2024. A Hybrid Simulation Platform for quality-aware evaluation of complex events in an IoT environment. Simulation Modelling Practice and Theory, 133, p.102919..

[8] Steindl, G., Schwarzinger, T., Schreiberhuber, K. and Ekaputra, F.J., 2024. Towards Semantic Event-handling for building Explainable Cyber-physical Systems. IEEE Open Journal of the Industrial Electronics Society.

[9] Aishwarya, C.K., Lahari, C.S. and Saheb, S.H., 2024. Data Analytics Tools and Applications for Business and Finance Systems. In Data-Driven Modelling and Predictive Analytics in Business and Finance (pp. 18-34). Auerbach Publications.

[10] Rosa-Bilbao, J., Boubeta-Puig, J., Lagares-Galán, J. and Vella, M., 2025. Leveraging complex event processing for monitoring and automatically detecting anomalies in Ethereum-based blockchain networks. Computer Standards & Interfaces, 91, p.103882.

[11] Al-dahasi, E.M., Alsheikh, R.K., Khan, F.A. and Jeon, G., 2025. Optimizing fraud detection in financial transactions with machine learning and imbalance mitigation. Expert Systems, 42(2), p.e13682.

[12] Rama, M., Kiranteja, V., Sireeshai, M., Vamsi, M. and Sohel, S., 2025. Analyzing the effectiveness of machine learning algorithms in detecting fraud transactions. In Hybrid and Advanced Technologies (pp. 116-124). CRC Press.

[13] Banirostam, H., Banirostam, T., Pedram, M.M. and Rahmani, A.M., 2025. Analysis and Evaluation of Various Fraud Detection Methods for Electronic Payment Cards Transactions in Big Data. Journal of Signal Processing Systems, pp.1-22.

[14] Sau, A., Banerjee, A. and Vedhavathy, T.R., Improving fraud detection efficiency: Leveraging machine learning strategies. In Hybrid and Advanced Technologies (pp. 33-40). CRC Press.

[15] Odufisan, O.I., Abhulimen, O.V. and Ogunti, E.O., 2025. Harnessing Artificial Intelligence and Machine Learning for Fraud Detection and Prevention in Nigeria. Journal of Economic Criminology, p.100127.

[16] Li, D., Han, D., Crespi, N., Minerva, R. and Li, K.C., 2023. A blockchain-based secure storage and access control scheme for supply chain finance. The Journal of Supercomputing, 79(1), pp.109-138.

[17] Hu, S., Lin, J., Du, X., Huang, W., Lu, Z., Duan, Q. and Wu, J., 2023. ACSarF: a DRL-based adaptive consortium blockchain sharding framework for supply chain finance. Digital Communications and Networks.

[18] Hanae, A., Youssef, G. and Saida, E., 2023, December. Analysis of Banking Fraud Detection Methods through Machine Learning Strategies in the Era of Digital Transactions. In 2023 7th IEEE Congress on Information Science and Technology (CiSt) (pp. 105-110). IEEE.

[19] Mikhaylov, A. and Bhatti, M.I.M., 2024. The link between DFA portfolio performance, AI financial management, GDP, government bonds growth and DFA trade volumes. Quality & Quantity, pp.1-18.

[20] Oza, J., Patil, A., Maniyath, C., More, R., Kambli, G. and Maity, A., 2024, May. Harnessing Insights from Streams: Unlocking Real-Time Data Flow with Docker and Cassandra in the Apache Ecosystem. In 2024 IEEE Recent Advances in Intelligent Computational Systems (RAICS) (pp. 1-6). IEEE.

[21] Carnero, A., Martín, C., Jeon, G. and Díaz, M., 2024. Online learning and continuous model upgrading with data streams through the kafka-ml framework. Future Generation Computer Systems, 160, pp.251-263.

[22] Cui, Y. and Yao, F., 2024. Integrating deep learning and reinforcement learning for enhanced financial risk forecasting in supply chain management. Journal of the Knowledge Economy, pp.1-20.

[23] Joshi, P.K., Building High-Throughput Payment Transaction Systems with Kafka and Micro services.

[24] Hou, D., Sun, Y., Dinavahi, V. and Wang, Y., 2024. Adaptive two-stage unscented Kalman filter for dynamic state estimation of synchronous generator under cyber attacks against measurements. Journal of Modern Power Systems and Clean Energy, 12(5), pp.1408-1418.

[25] Yan, Y. and Kuruoglu, E.E., 2025. Binarizedsimplicial convolutional neural networks. Neural Networks, 183, p.106928.

[26] D. Tzeli and A. Mavridis, "First-Principles Investigation of the Boron and Aluminum Carbides BC and AlC and Their Anions BC^- and AlC^-. 1," *The Journal of Physical Chemistry A*, vol. 105, no. 7, pp. 1175–1184, Jan. 2001, doi: https://doi.org/10.1021/jp003258k.

[27] Ashkan Samiee, Payal Borulkar, R. F. DeMara, P. Zhao, and Y. Bai, "Low-Energy Acceleration of Binarized Convolutional Neural Networks Using a Spin Hall Effect Based Logic-in-Memory Architecture," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, pp. 928–940, May 2019, doi: https://doi.org/10.1109/tetc.2019.2915589.

[28] JIN, "An Introduction to Apache Kafka System Architecture," *Medium*, Mar. 27, 2022. https://aws.plainenglish.io/apache-kafka-system-architecture-cc74e7d47904

[29] Q. Qu, R. Xu, Seyed Yahya Nikouei, and Y. Chen, "An Experimental Study on Microservices based Edge Computing Platforms," *arXiv (Cornell University)*, Jul. 2020, doi: https://doi.org/10.1109/infocomwkshps50562.2020.9163068.