

An optimized architecture for real-time fraud detection in big data systems, ecosystems, and environments

Gaber E. Abutaleb¹, Abdallah A. AlHabshy¹, Beriham R. Elemery^{2,3}, Ebeid A. Ebeid¹,
Kamal A. ElDahshan¹

¹Department of Mathematics, Faculty of Science, Al-Azhar University, Cairo, Egypt

²Department of Applied, Mathematical, and Actuarial Statistics, Faculty of Commerce, Damietta University, Damietta, Egypt

³Universidade NOVA de Lisboa, NOVA at TKH, Cairo, Egypt

Article Info

Article history:

Received Dec 3, 2024

Revised Apr 8, 2025

Accepted Jul 2, 2025

Keywords:

Apache Spark

Big data technologies in fraud detection

Big data technologies machine learning algorithms real-time processing

Fraud detection

ABSTRACT

The exponential growth of data in recent years has created significant challenges in fraud detection. Fraudulent activities are increasingly widespread across sectors, such as banking, web networks, health insurance, and telecommunications. This trend highlights a growing need for big data technologies such as Hadoop, Spark, Storm, and HBase to enable real-time detection and analysis of data fraud. This study aims to enhance understanding of the fraud classifications and their spread in various sectors. Fraud detection involves analyzing data and developing machine learning (ML) models or traditional rule-based systems to identify abnormal activities as they occur. The analysis in this paper examines both the advantages and limitations of these solutions, particularly regarding scalability and performance. This paper evaluates the methods and big data tools used in fraud detection and prevention through a comprehensive literature review, emphasizing the implementation challenges. This review discusses existing solutions, operational environments, and the ML algorithms and traditional rules employed. The main objective of this study is to address these challenges by proposing an innovative architecture that equips organizations with the latest knowledge and methodologies in big data technologies for real-time fraud detection and prevention.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Gaber E. Abutaleb

Department of Mathematics, Faculty of Science, Al-Azhar University

Cairo, 11511, Egypt

Email: gaber_abutaleb@azhar.edu.eg

1. INTRODUCTION

The widespread reliance on the Internet has made numerous entities more susceptible to fraudulent activities, leading to significant financial losses [1]. Detecting and differentiating fraudsters from legitimate users in the vast sea of digital data has become a significant challenge. The explosion of digital transactions has fueled a rise in fraudulent activity, threatening financial institutions and businesses [2]. Traditional fraud detection methods struggle to keep pace with the evolving tactics of fraudsters.

Big data, encompassing structured and unstructured data from various sources, holds the key to addressing this challenge [3]. As fraudsters continuously adapt their techniques, organizations must employ machine learning (ML) and artificial intelligence (AI) to detect sophisticated deviations from normal patterns among millions of legitimate transactions. Implementing robust measures to detect and prevent fraudsters' access to services has become a top priority for entities, safeguarding the interests of legitimate users.

In recent years, there has been an increasing amount of literature that has developed frameworks for fraud detection in big data environments. Baldominos *et al.* [4] created a scalable Hadoop-based architecture for ML on big data streams, while Di Mauro and Di Sarno [5] proposed a real-time processing system using Apache Storm and Yahoo SAMOA, achieving 90% accuracy in Skype traffic detection. Zhao *et al.* [6] built a network anomaly detection framework using Hadoop, Kafka, and Storm for academic networks. Dai *et al.* [7] developed a four-layer credit card fraud detection system incorporating distributed storage and streaming analysis. Stojanovic *et al.* [8] introduced a self-adaptive approach for industrial quality control anomaly detection. Cui and He [9] proposed a cloud-based framework combining Hadoop with Weka's algorithms, while Balasupramanian *et al.* [10] created a system using principal component analysis (PCA) and self-organizing maps (SOM). Melo-Acosta *et al.* [11] addressed class imbalance in credit card fraud detection using balanced random forest (BRF) on Apache Spark. Othman *et al.* [12] introduced the Spark-Chi-support vector machines (SVM) model for intrusion detection, and Carcillo *et al.* [13] developed scalable real-time fraud finder (SCARFF), a scalable system combining Kafka, Spark, and Cassandra with advanced ML techniques. Nair *et al.* [14] created a health prediction system using Spark and Twitter data, while Cao *et al.* [15] developed TitAnt, a millisecond-speed fraud detection system for Alipay combining offline training with online prediction. Zhou *et al.* [16] introduced a financial fraud detection approach using Node2Vec with Spark GraphX and Hadoop. Habeeb *et al.* [17] created a real-time anomaly detection framework with their SSWLOFCC algorithm achieving 96.51% accuracy. Saheed *et al.* [18] developed credit card fraud models using PCA with various ML techniques, and Tawde *et al.* [19] proposed an online payment fraud detection system using PySpark for large-scale transaction analysis.

Although there are many excellent review papers on this topic, no single architectural design effectively addresses and manages these challenges. Previous research on big data fraud detection frameworks has faced several limitations. Many solutions operated exclusively with batch ML algorithms over Hadoop, lacking support for real-time stream processing. Most frameworks were applied to single datasets, limiting their generalizability and robustness. Several approaches did not incorporate unsupervised ML algorithms, data enrichment features (such as IP geolocation), or visualization tools. Some frameworks provided only near real-time rather than true real-time processing, with accuracy levels requiring improvement. Integration with multiple detection algorithms was often insufficient. Additionally, some systems lacked rule-building engines, data enrichment operations, and manual verification components for managing suspicious alerts.

The current study contributes to the expansion of knowledge in this field by addressing important issues. First, a comprehensive overview of fraud, its classifications, and the areas most susceptible to fraudulent activities. Second, study the techniques used in detecting fraudulent activities, highlighting both the advantages and limitations of these solutions. Third, Additionally, it examines the latest big data technologies and their applications in detecting fraudulent operations. Fourth, presents a thorough review of much of the literature addressing fraud prevention in big data, with a comparative analysis of real-time solutions. Tables 1 and 2 demonstrate how my study relates to and differs from previous work, highlighting the methodologies used by other researchers and positioning my contributions within the existing body of knowledge. Fifth, propose an innovative architecture that equips organizations with the latest knowledge and methodologies in big data technologies for real-time fraud detection and prevention.

The proposed architecture addresses these limitations through several key contributions. It utilizes multiple big data technologies including Apache Spark and Storm for improved scalability and efficient stream processing. The architecture supports both supervised and unsupervised ML algorithms. It also incorporated data enrichment operations including IP geolocation, email, and bank card analysis. Additionally, the architecture includes a rule-building engine alongside ML algorithms and features a manual verification component for suspicious alert management. These enhancements collectively represent a more robust, versatile, and accurate approach to fraud detection in big data environments.

The remainder of the paper is organized as follows: section 2 provides the method of this study. In section 3 provides an overview of fraud, its characteristics, and the use of big data technologies for fraud detection. In section 4 explains methodologies and techniques for detecting real-time fraud in big data systems, as well as their limitations. Section 5 describes the literature review methodology and provides an analysis of it. In section 6 presents a comparative analysis of the literature review, highlighting its impact on fraud-related areas and the proposed architecture in this study. The conclusion highlights the paper's contributions and suggests future research directions.

2. METHOD

This study follows a systematic approach to analyzing fraud detection techniques in big data environments. The approach consists of multiple stages, including understanding fraud and its classifications,

analyzing existing fraud detection techniques, evaluating big data technologies, literature review, comparative analysis, and architectural design to address the challenges associated with real-time fraud detection. The following steps outline the research methodology:

- Understanding fraud and its classifications: this phase explores fraud types, their classifications, and vulnerable domains. It also examines the limitations of traditional detection methods and how ML and big data technologies address these challenges.
- Literature review: a literature review was conducted on fraud detection methods, ML models, and big data tools like Hadoop, Spark, Storm, and HBase. The review covers studies from 2014 to 2024, focusing on peer-reviewed research addressing fraud detection in big data environments.
- Comparative analysis of big data fraud detection techniques: the study conducts a comparative analysis of fraud detection techniques by examining various research studies across different domains, including financial transactions, network security, and online fraud detection. It evaluates learning types (supervised, unsupervised, rule-based), algorithms (decision trees (DT), random forest (RF), SVM, neural networks, DBSCAN), and big data tools. The trade-offs in accuracy, scalability, and real-time processing are analyzed, along with the role of streaming technologies like Kafka and Flume.
- Architectural design and optimization: based on the findings from the literature review and comparative analysis, an optimized architecture for real-time fraud detection is proposed. The proposed system integrates multiple big data technologies, including Apache Spark and Storm, to enhance scalability and stream processing efficiency. It also supports a hybrid fraud detection approach, combining ML algorithms (both supervised and unsupervised) with rule-based detection mechanisms. Additional data enrichment operations, such as IP geolocation, email, and bank card analysis, are incorporated to improve fraud detection accuracy.
- Implementation considerations: the proposed architecture is modular and scalable, supporting integration with existing infrastructures. It includes a rule-building engine for custom fraud detection rules and ML models trained on historical data. A manual verification component ensures human oversight in critical fraud cases.

3. FRAUD DETECTION OVERVIEW

An overview of fraud detection is presented in this section, covering the fraud definition, its classification, the areas most affected, and the challenges in detecting fraud. Each aspect is discussed in detail as follows:

3.1. Fraud

Fraud involves falsifying information, making false representations, or abusing positions of power within an organization with the intent of personal enrichment at the expense of others. The Association of Certified Fraud Examiners (ACFE) defines fraud as “the use of one’s occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization’s resources or assets” [1]. Fraud results in significant financial losses and has detrimental effects on the economy, organizations, and individuals [2].

3.2. Fraud classification

Fraud can be classified in multiple ways: by perpetrator, victim, and scheme type. Among various types, frauds against organizations are common, and financial statement frauds are the most costly. This paper highlights two major classifications:

- i) Internal vs. external fraud: internal fraud is committed by individuals within an organization who misuse their positions for personal gain, including acts like embezzlement, falsifying reports, and kickbacks. Managers represent the greatest risk, while employees present a lower risk. External fraud, in contrast, is perpetrated by individuals or entities outside the organization exploiting its vulnerabilities through schemes like identity theft and phishing [3].
- ii) Victim-based fraud: this classification identifies the affected party:
 - Institutional fraud targets the organization itself and includes employee embezzlement, vendor fraud, and customer-related fraud.
 - Management fraud harms shareholders or debtholders of the organization, often through financial statement manipulation by senior management.
 - Investment and consumer fraud affects individuals, with investment fraud involving schemes like Ponzi schemes, while consumer fraud includes identity theft, credit card fraud, and internet fraud.
 - Miscellaneous fraud refers to cases in which someone exploits another person’s confidence to deceive them. This category includes fraudulent activities such as bankruptcy fraud and tax evasion [2], [20].

3.3. Fraud areas

Fraud affects growing risks across sectors, especially in finance and data-sensitive industries. As digitalization increases, fraudsters are employing more sophisticated methods to exploit online financial activities. Key sectors vulnerable to fraud include financial services [21], insurance [22], telecommunications [23], healthcare [24], [25], and computer intrusion [26], [27]. Figure 1 illustrates the areas most commonly at risk for fraud.

Financial services fraud encompasses a wide range of illegal practices, including credit card fraud involving unauthorized transactions and card information theft [28]. Identity theft remains one of the most prevalent forms, leading to financial losses and legal ramifications [29]. Ponzi schemes deceive investors with promises of high returns [30], while money laundering attempts to conceal the origins of illegally obtained funds through various methods such as foreign transfers and cryptocurrency transactions [21].

Web network fraud operates through the Internet using malicious services and software to exploit victims. This includes phishing attacks, lottery scams, romance scams, and ransomware attacks that target individuals through various digital channels [31]. Internal fraud, occurring within organizations, poses a significant threat through activities like embezzlement, asset misappropriation, and document falsification, leading to both financial losses and reputational damage [32]. Customs fraud involves evading or reducing duty payments through deceptive practices such as underreporting values, misclassifying goods, and forging documents. This undermines fair market competition and results in significant economic losses [33].

Computer intrusion, involves unauthorized access to systems and networks, manifesting through malware attacks, denial of service operations, and SQL injections, often resulting in data theft and system damage [26]. Insurance fraud manifests across multiple sectors, including automobile, home, and crop insurance, where individuals or entities make false claims or exaggerate damages to obtain undue benefits [34]. Telecommunications fraud is becoming a major threat to telecom operators across their service offerings. Subscription fraud involves criminals obtaining services with premeditated plans to avoid payment. In superimposed fraud, scammers gain unauthorized control of legitimate accounts by cloning devices and cards [23]. Healthcare fraud involves various deceptive practices in the medical sector. This includes submitting inflated invoices, generating fake bills, and engaging in prescription drug fraud where medications are obtained for non-medical purposes or dispensed to individuals without legitimate need [35].

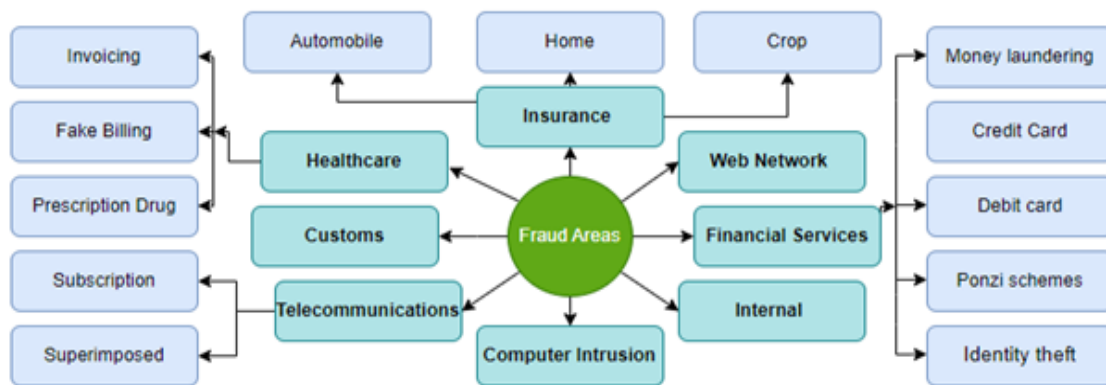


Figure 1. Fraud areas

3.4. Fraud detection challenges

Detecting fraud presents substantial challenges due to inherent complexities that often reduce system accuracy and increase false alarms. Particularly in banking and e-commerce, these alarms can lead to financial losses. Effective fraud detection systems require overcoming these challenges to minimize errors and improve robustness. This paper highlights the toughest challenges that these systems face:

- Concept drift: the shift in data over time due to changes in user and fraudster behavior, decreases model accuracy as historical data becomes outdated. To counteract this, regular model updates and continuous monitoring are crucial, ensuring that models remain aligned with the changing nature of inputs over time [36].
- Imbalanced data: fraud detection faces an imbalanced dataset where legitimate activities vastly outnumber fraudulent ones. This imbalance can bias models, increasing false alarms. Addressing it involves techniques like under sampling, oversampling, or the synthetic minority oversampling

technique (SMOTE), alongside specialized algorithms and evaluation metrics such as precision, recall, and F1-score [37].

- Data volume and variety: data volume and variety refers to the massive amount and different types of data that organizations must process for fraud detection. The high data volume and diversity common in sectors like banking and e-commerce, complicate fraud detection as fraudsters use multiple channels and data types. Processing such vast, multidimensional datasets requires substantial computational resources and advanced algorithms that can handle large, diverse data efficiently [38].
- Real-time detection: real-time detection is the process of identifying anomalies, patterns, or events as they happen, allowing for immediate response to potential threats. In banking, real-time fraud detection is essential for stopping fraudulent activities before they cause damage. This capability protects financial assets and preserves the reputation of both companies and customers, making it a crucial component of modern financial security systems[39].
- False positives: occur when legitimate transactions are wrongly identified as fraudulent, causing customer dissatisfaction. Reducing false positives involves refining detection methods, analyzing customer behavior, and combining rule-based and anomaly detection techniques, while human review of flagged transactions aids in minimizing mistakes [40].

4. BIG DATA-BASED FRAUD DETECTION

Big data refers to massive datasets that surpass traditional processing capabilities and are defined by the “Five Vs” [41]. Volume indicates vast amounts of data, variety covers diverse data types, velocity highlights real-time processing needs, veracity ensures data quality and reliability, and value focuses on extracting meaningful insights. Big data plays a crucial role in fraud detection by enabling the analysis of vast amounts of data from various sources to uncover patterns, anomalies, and suspicious activities. This section will explore the approaches and big data tools utilized for detecting fraud within large datasets.

Rule-based systems: these systems utilize predefined rules derived from historical data to enable immediate transaction surveillance and quick decision-making in detecting fraudulent activities. This approach forms the foundation of many fraud detection strategies.

AI-driven fraud detection: this is an advanced technological approach that benefits AI and ML algorithms to automatically identify, predict, and prevent fraudulent activities across various domains. By analyzing vast amounts of data in real-time, these intelligent systems detect complex patterns, anomalies, and potential security threats that traditional rule-based methods might overlook. The methods used in AI-driven fraud detection can include, but are not limited to:

- Machine learning: comprises four main categories: 1) supervised learning uses labeled data with techniques like DT, logistic regression (LR), RF, and SVM; 2) unsupervised learning works without labeled data, employing K-means, local outlier factor (LOF), and DBScan to identify patterns [42]; 3) semi-supervised learning combines both approaches using generative adversarial networks (GANs) and S3VM; 4) reinforcement learning trains algorithms to make sequential decisions, where agents maximize rewards through beneficial actions in complex environments [43].
- AI-powered fraud detection strategies: utilize advanced anomaly detection algorithms and AI-driven pattern recognition to analyze complex data across multiple dimensions [39]. These approaches integrate natural language processing for detecting suspicious activities, enabling comprehensive risk assessment through intelligent technology patterns [44]. Adaptive learning systems continuously update fraud detection models, creating dynamic and evolving defense mechanisms that can anticipate and mitigate emerging fraudulent activities with unprecedented accuracy.

Advanced analytics: in fraud detection combine real-time transaction scoring, behavioral biometrics, network graph analysis [45], and predictive modeling. These techniques enable instant risk assessment by analyzing user interactions, mapping fraud networks, and anticipating suspicious activities through sophisticated algorithms.

Advanced technical methods: for security and verification integrate a comprehensive suite of cutting-edge technologies. These approaches include biometric authentication, multi-factor authentication techniques, blockchain-based verification systems, and sophisticated encryption and tokenization methods that work in concert to create robust security frameworks [22], [46].

Big data visualization: instead of just reading raw numbers and text, data visualization turns them into pictures like charts and graphs. This makes it easier to see what the data means and spot trends [47]. Data visualization tools like Tableau or Power BI further empower analysts by transforming complex data sets into clear, interactive formats. Big data tools for fraud detection as follows:

- Processing frameworks: the implementation of these approaches relies on several major big data tools, primarily divided into processing frameworks and data movement solutions. Processing frameworks

- include Apache Spark [48], which provides distributed computing for batch and real-time processing; Apache Hadoop [49], offering distributed storage and processing capabilities; Apache Storm [50], delivering real-time processing with low latency; and Apache Flink [51], for high-throughput stream processing.
- Data movement and streaming: for data movement and streaming, organizations utilize tools such as Apache Flume for efficient log data collection and movement [52], Apache Kafka for distributed event streaming [53], and Amazon Kinesis for AWS-based real-time data processing [54]. These tools and approaches collaborate to create comprehensive fraud detection systems capable of processing massive amounts of data in real-time, identifying patterns, and flagging suspicious activities across various industries. This integrated approach ensures robust protection against fraudulent activities while maintaining efficient processing capabilities.

5. LITERATURE REVIEW

Numerous studies have examined fraud detection systems, especially with big data's rise. This section explores key literature on fraud detection in big data, covering methodologies like rule-based strategies and both supervised and unsupervised learning algorithms. Baldominos *et al.* [4] proposed a scalable architecture for ML on big data streams using the Hadoop ecosystem. It consists of a batch processing module for non-urgent tasks and a stream processing module for real-time applications, both relying on HDFS and HBase for storage [43]. The system includes a dashboard for batch results and a RESTful API for ML services, applied in web advertising and gaming behavior prediction. The evaluation highlights its ability to handle concurrent requests and provide accurate real-time predictions, making it suitable for real-time analytics as a service.

Di Mauro and Di Sarno [5] proposed an architecture for real-time big data processing and analysis, integrating Apache Storm as the stream processing engine and Yahoo SAMOA for distributed streaming ML. They utilized the vertical hoeffding tree algorithm to recognize hidden Skype traffic in network streams. Experimental results demonstrated 90% accuracy in identifying Skype traffic.

Zhao *et al.* [6] introduced a framework for real-time network traffic anomaly detection using ML and big data tools like Apache Hadoop, Kafka, and Storm. This system analyzes real-time network flow data from the University of Missouri–Kansas city's campus network by integrating batch processing with real-time analysis. Kafka handles data storage while Storm facilitates streaming analysis. The framework encompasses data ingestion, preprocessing, anomaly detection, and ML components. Preliminary results indicate its effectiveness in identifying network anomalies, highlighting its potential for efficient network management in academic settings.

Dai *et al.* [7] proposed a big data-based online credit card fraud detection framework with a four-layer design: a distributed storage layer for handling large transaction data, a batch training layer for model training using Hadoop and Spark [48], a key-value sharing layer for fast model data access via NoSQL database [55]–[57], and a streaming detection layer for real-time analysis using storm. This hybrid framework highlights the scalability, fault tolerance, and high performance of big data technologies in fraud detection.

Stojanovic *et al.* [8] introduced a data-driven approach for real-time anomaly detection in industrial quality control, suitable for complex manufacturing processes with non-linear parameter correlations. This self-adaptive method utilizes both historical and real-time data to enhance accuracy, particularly in manufacturing critical components like microwave oven fans. The system efficiently detects deviations using real-time processing technologies (e.g., Storm) alongside historical data processing (e.g., Hadoop). Its architecture comprises six components: data storage, processing, analytics, user interaction, integration, and security layers.

Cui and He [9] proposed a cloud-based anomaly detection framework that uses Hadoop's distributed processing and Weka's ML algorithms. Traffic data is stored in HDFS and processed with MapReduce, while the best-performing algorithm—selected from Naïve Bayes, DT, and SVM—is identified through Weka. The framework includes data processing, mining, and detection modules, achieving over 90% prediction accuracy. Evaluation with self-refit and 10-fold cross-validation shows the DT excels in classification accuracy, offering greater efficiency than traditional single-point methods.

Balasupramanian *et al.* [10] proposed a framework to prevent online fraud by combining big data analytics and ML. Their approach involves collecting and preprocessing transaction data, extracting features, and reducing dimensionality with PCA. A SOM model is then trained to evaluate transactions, with suspicious activity triggering alerts or card blocks. The framework recommends using HDFS or Spark for data storage and keeping trained data in memory for speed, aiming to detect fraud proactively by analyzing user behavior from historical data.

Melo-Acosta *et al.* [11] presented a credit card fraud detection system addressing class imbalance, mixed labeled/unlabeled data, and high transaction volume. Their approach uses BRF and combines supervised with semi-supervised learning through co-training, implemented on Apache Spark for scalability. This method significantly improves performance, achieving a 24% higher geometric mean than traditional RFs, with the BRF and co-trained BRF meta-classifier showing the best results.

Othman *et al.* [12] proposed the Spark-Chi-SVM model for intrusion detection, utilizing apache spark for efficient big data processing. The model includes dataset preprocessing, feature selection with ChiSqSelector, and classification using SVMWithSGD, which is tested on the KDD dataset. Results show high performance and speed.

Carcillo *et al.* [13] introduced SCARFF, a scalable fraud detection system that combines big data tools like Kafka, Spark, and Cassandra with advanced ML to address issues such as data imbalance and feedback latency. SCARFF processes streaming data for near real-time alerts, utilizing Kafka for fault-tolerant data collection, Spark for feature engineering and classification, and Cassandra for storage. Its ML engine employs a weighted ensemble of classifiers, while its open-source framework, deployable via Docker, allows for reproducibility and testing with artificial datasets, making it highly effective for large-scale fraud detection.

Nair *et al.* [14] proposed a real-time remote health status prediction system leveraging Apache Spark and Twitter data. The system focuses on applying ML models to streaming big data for health prediction. Users tweet their health attributes, which are then processed by the application in real time. The system extracts attributes and applies a ML model, such as a DT, to predict the user's health status. The prediction is instantly messaged to the user for appropriate action. The application is implemented in Scala, integrating a DT model with Twitter streaming data handling, and can be deployed on-premise or in the cloud, such as Amazon EC2.

The TitAnt system was developed by Cao *et al* [15]. Ant financial is a high-speed fraud detection framework that predicts online transaction fraud in milliseconds. It combines offline training and online real-time prediction: offline, transaction data is processed in MaxCompute for feature extraction, and models are trained with KunPeng. Online, the model server receives periodic model updates and retrieves data from Ali-HBase to make instant predictions on Alipay transactions. Despite delayed labels, TitAnt meets strict latency demands by integrating distributed algorithms and user node embeddings, showcasing effective, real-time fraud detection in financial transactions.

Zhou *et al.* [16] introduced an intelligent, distributed approach for detecting financial fraud using Node2Vec, a graph embedding algorithm that learns network topologies and represents them as low-dimensional vectors. This approach enhances classification and prediction using deep neural networks and utilizes Apache Spark GraphX and Hadoop clusters for parallel data processing. The workflow includes four main modules: data preprocessing, feature extraction, graph embedding, and prediction. Node2Vec on Spark GraphX efficiently captures vertex features, improving deep neural network classification and boosting fraud detection accuracy.

Habeeb *et al.* [17] developed a real-time anomaly detection framework using a composite streaming clustering approach with big data tools like Spark MLlib, Kafka, and HBase. They introduced SSWLOFCC, a novel algorithm in Spark MLlib, achieving 96.51% accuracy, 13.333s execution time, and 194.33 MB memory usage, outperforming K-means and HDBSCAN. Tested on DARPA, MACCDC, and DEFCON21 datasets, the framework improves real-time anomaly detection by optimizing computational cost, accuracy, and data visualization.

Saheed *et al.* [18] introduced ML models for predicting credit card fraud, focusing on a new detection model that utilizes PCA for feature selection and various supervised ML techniques (K-nearest neighbor (KNN), ridge classifier, gradient boosting, quadratic discriminant analysis, AdaBoost, and RF) for classification. The model is tested on German and Taiwan credit card datasets to distinguish fraudulent from legitimate transactions.

Tawde *et al.* [19] proposed a novel approach to detecting online payment fraud using big data techniques, particularly leveraging PySpark. After data preprocessing, ML algorithms like RF, DT, Naive Bayes, and LR from Spark ML are applied, enabling scalable, distributed classification. The system is designed to help large organizations analyze extensive transaction data to identify potential fraud or anomalies effectively.

6. RESULTS AND DISCUSSION

This analysis examines fraud detection methods from selected studies (Table 1), exploring trends in authors, publication years, learning types, algorithms, tools, and targeted domains. It also reviews detection techniques and algorithms, recommending a proposed architecture for organizations' 6.2 tools and technologies. Big data platforms like Spark, Hadoop, HBase, and Storm are widely used for efficient large-

scale data processing. Kafka and flume complement these by managing real-time data streams, crucial for fraud detection in dynamic environments. Spark stands out for its versatility in both batch and stream processing.

Table 1. Comparative analysis of the examined big data fraud detection systems

Author	year	Learning types	Algorithm	Tools	Domain
Baldominos <i>et al.</i> [4]	2014	Supervised, unsupervised	DT, neural-network, K-means, RF, Markov-chains	Hadoop, HBase Mahout	Advertisements to web visitors, social games
Di Mauro and Di Sarno [5]	2014	Supervised	Vertical hoeffding tree	Storm, SAMOA	Hidden Skype traffic
Zhao <i>et al.</i> [6]	2015	Supervised	Naïve-Bayesian, SVM, DT	Storm, Kafka Hadoop	Network traffic, anomaly detection
Dai <i>et al.</i> [7]	2016	Supervised, unsupervised	DBSCAN, HMM, SOM, neural network, LR, DT, Naive Bayes	Storm, HBase, Spark, Hadoop	Credit card fraud detection
Stojanovic <i>et al.</i> [8]	2016	Rules	Rules	Storm, Hadoop, Spark, HBase	Quality control anomaly detection
Cui and He [9]	2016	Supervised	Naïve-Bayes, DT, SVM	Hadoop	Anomaly detection
Balasupramanian, <i>et al.</i> [10]	2017	Unsupervised	SOM	Spark, Hadoop	Online transaction fraud detection
Melo-Acosta <i>et al.</i> [11]	2017	Supervised	RF and co-trained BRF	Spark, Hive	Credit card fraud detection
Othman <i>et al.</i> [12]	2018	Semi-supervised	Chi-SVM	Spark	Intrusion detection
Carcillo <i>et al.</i> [13]	2018	Supervised	RF	Spark, Kafka, Cassandra	Credit card fraud detection
Nair <i>et al.</i> [14]	2018	Supervised	DT	Spark	Health status prediction
Cao <i>et al.</i> [15]	2019	Supervised, rules, Unsupervised	DW, S2V, NRL, LR, GBDT, rule-based, isolation-forest	Ali-HBase, KunPeng MapReduce	Financial transaction fraud
Zhou <i>et al.</i> [16]	2021	Supervised	Node2Vec, deepwalk, SVM	Spark Hadoop	Internet financial fraud detection
Habeeb <i>et al.</i> [17]	2022	Unsupervised	SSWLOFCC, IF, LOF, K-means, HDBSCAN, agglomerative clustering	Flume, Kafka, Spark, HBase	Intrusion detection, hacking detection, network fraud detection
Saheed <i>et al.</i> [18]	2022	Supervised	KNN, PCA, RF	N/M	Credit card fraud detection
Tawde <i>et al.</i> [19]	2024	Supervised	RF, DT, NB, LR	Spark	Online payment

6.1. Learning types and algorithm usage

As shown in Table 1, multiple learning approaches are applied in fraud detection, including supervised, unsupervised, and rule-based learning. Supervised learning dominates the field, with algorithms such as DTs, SVM, RFs, CNN, and long short-term memory (LSTM) widely used. Yussiff *et al.* [58] proposed an intelligent approach for detecting online credit card fraud using the extreme gradient boosting (XGBoost) algorithm. These techniques are favored due to their predictive capabilities, which aid in identifying fraudulent patterns in large datasets.

Unsupervised learning is also prominent, with clustering algorithms like K-means, HDBSCAN, and SOM employed for detecting outliers and unknown fraud patterns. Moreover, rule-based methods are utilized in specific cases where predefined fraud criteria are essential for anomaly detection. The analysis of fraud detection algorithms shows that clustering methods, particularly K-means, are highly favored for unsupervised learning scenarios, while DTs and SVM are dominant in supervised learning contexts. This preference suggests that clustering is instrumental in discovering hidden patterns in unlabeled data, while classification algorithms enhance prediction accuracy and help identify fraudulent instances within labeled datasets.

6.2. Suitable fraud detection methods across various domains

Credit card fraud detection: several studies, including those by Dai *et al.* [7], Carcillo *et al.* [13], utilize supervised techniques, often using algorithms like RFs, DTs, Naive Bayes, and artificial neural network (ANN), combined with tools such as Spark, HBase, and Kafka for real-time processing. Financial transaction fraud: Zhou *et al.* [16] applied techniques such as Node2Vec and SVM for detecting internet financial fraud, utilizing Spark and Hadoop for data handling. According to Yussiff *et al.* [58], RF was found to be the most effective machine-learning algorithm for fraud detection on financial e-platforms. It ranked first in both frequency of usage and performance analysis, achieving an average accuracy of 96.67%.

Intrusion and network fraud detection: algorithms like isolation forest, LOF, and HDBSCAN are utilized for detecting anomalies in network data. Habeeb *et al.* [17] employed Spark, HBase, and Kafka in intrusion detection to monitor for hacking activities and fraud within networks. Health and quality control: in health prediction and quality control anomaly detection, studies like those of Nair *et al.* [14] and Stojanovic *et al.* [8] use DTs and rule-based systems to detect anomalies in health statuses and ensure data integrity in quality control processes.

6.3. The proposed architecture

This research provides a vision for organizations to build a strong architecture to combat fraud in real time. Figure 2 illustrates the proposed architecture that integrates rule enforcement, AI models, and big data technologies across multiple layers. The storage layer is used to store data. It is recommended to use HBase, Hadoop HDFS, or Cassandra [59], providing a scalable, distributed storage solution that supports both real-time data access and batch processing requirements for fraud detection.

The training layer processes historical data and improves the AI model. Spark, flume [52], and Hadoop's processing capabilities can be combined with Mahout's ML algorithms [60], while Hive provides a powerful data warehousing function for comprehensive analysis of past fraud patterns [61]. The integration layer serves as the primary data ingestion point, utilizing powerful messaging systems like Kafka, Redis [62], RabbitMQ [63], and ActiveMQ [64] to handle data from external sources.

The real-time fraud detection layer serves as the core analytics component, leveraging technologies like Kafka streams, Spark streaming, or Storm to process live data streams. It integrates complex fraud detection rules alongside ML models to identify advanced fraud patterns. Additionally, data enrichment enhances detection accuracy by incorporating contextual insights from internal and external sources. Key techniques include IP-based geolocation, VPN detection, device fingerprinting, transaction analysis, behavioral biometrics, BIN numbers, and email verification, providing a comprehensive approach to fraud prevention.

The user interface layer delivers an intuitive experience through data visualization tools, accompanied by a rules management interface and an alert management system that promptly notifies users of potential fraud activities, enabling quick response to suspicious patterns. The proposed architecture delivers a balanced approach to fraud detection, combining real-time processing with historical analysis to create a robust defense against fraudulent activities.

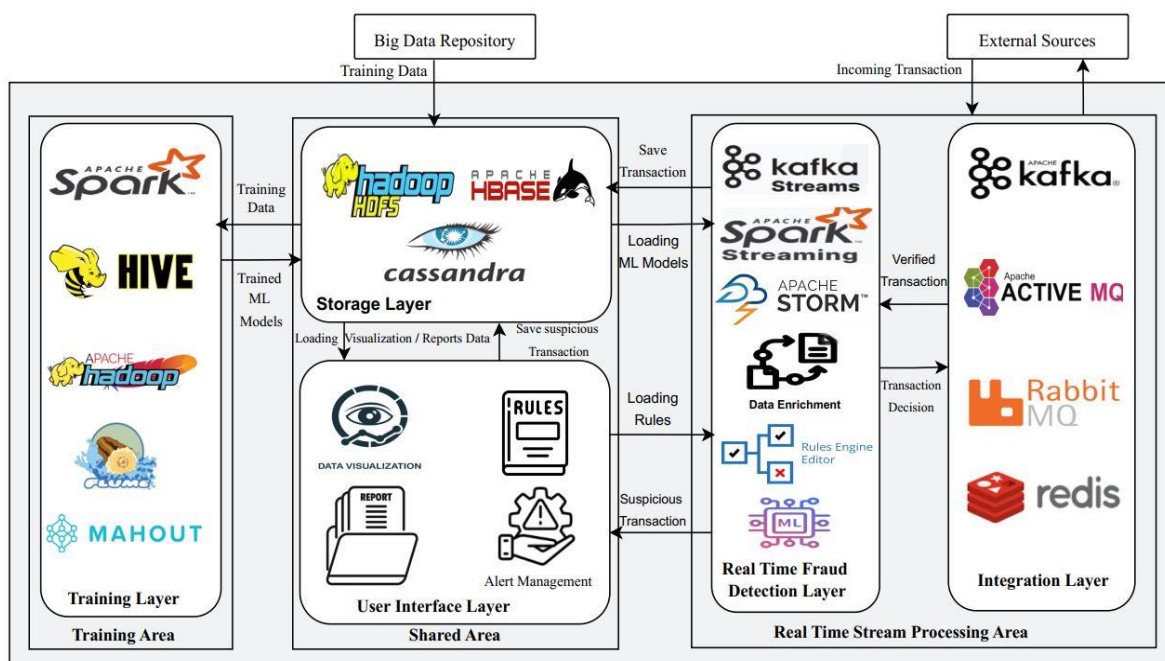


Figure 2. The proposed real-time fraud detection architecture

6.4. Comparative analysis with related works

This study presents a novel optimized fraud detection architecture that addresses several limitations identified in existing works. Unlike previous studies that rely solely on Hadoop for batch processing and lack real-time stream processing capabilities (e.g., Baldominos *et al.* [4], Di Mauro and Di Sarno [5]), the architecture integrates Spark, Hadoop, and Storm, enabling efficient real-time detection alongside batch processing. Additionally, as summarized in Table 2, some frameworks (e.g., Zhao *et al.* [6], Cui and He [9]) lack supervised ML support, data enrichment, and visualization tools. The study approach overcomes these gaps by incorporating supervised learning, IP geolocation enrichment, and visualization features, enhancing interpretability and decision-making.

Table 2. Comparison of limitations in related works and advantages of the proposed architecture

Reference	Limitations in existing works	Improvements in the proposed architecture
Baldominos <i>et al.</i> [4]	In this research, the authors implement and evaluate batch ML algorithms that run only on Hadoop and lack real-time stream processing capability.	The proposed architecture utilizes Spark and Storm technologies for enhanced scalability and stream processing efficiency.
Di Mauro and Di Sarno [5]	In this research, the authors implement and evaluate batch unsupervised ML algorithms that operate solely on Hadoop and do not possess real-time stream processing capabilities.	The proposed architecture utilizes Spark and Hadoop to enable efficient batch training using unsupervised ML algorithms.
Zhao <i>et al.</i> [6]	In this research work, the authors rely solely on Hadoop for batch training. Unsupervised ML, data enrichment (such as IP geolocation), and visualization tools are unavailable.	The proposed architecture utilizes Spark and Hadoop for batch training with unsupervised ML algorithms and integrates IP geolocation enrichment along with visualization tools.
Cui <i>et al.</i> [9]	Their solution works on one dataset via Hadoop batch training only. Needs accuracy improvement and lacks real-time processing.	The proposed architecture utilizes Spark and Hadoop for batch training while ensuring accuracy meets requirements and providing full real-time processing support.
Dai <i>et al.</i> [7]	The framework needs integration with multiple detection algorithms.	The proposed architecture is already well-integrated with multiple detection algorithms.
Stojanovic <i>et al.</i> [8]	Learning types and ML algorithms were unspecified.	The proposed architecture supports the different learning types and rules.
Balasupramanian <i>et al.</i> [10]	No supervised ML was used. The solution offers near real-time detection only and lacks real-time processing tools.	The proposed architecture supports supervised ML algorithms and enables real-time detection and processing.
Melo-Acosta <i>et al.</i> [11]	Framework lacks real-time processing; Spark only handles batch training.	The proposed architecture enables real-time processing with Spark and Hadoop for batch training.
Carcillo <i>et al.</i> [13]	The framework handles transactions near real-time only and lacks rule-based support.	The proposed architecture processes transactions in real-time; rule-based support.
Othman <i>et al.</i> [12]	The model lacks scalability and uses a single-prediction model.	The proposed architecture uses Storm for scalable real-time processing and multiple prediction models.
Nair <i>et al.</i> [14]	The system uses solely DT ML with no preset rules.	The proposed architecture combines ML algorithms with preset rules.
Cao <i>et al.</i> [15]	System missing rule engine, data enrichment (IP/email/card), and manual alert verification capabilities.	The proposed architecture supports a rule builder engine, data enrichment (IP/email/card), and manual alert verification capabilities.
Zhou <i>et al.</i> [16]	The methodology lacks real-time processing, alternative solution integration, and manual verification for suspicious alerts.	The proposed architecture enables real-time processing, integrates with any solution, and includes manual verification.
Habeeb <i>et al.</i> [17]	Framework lacks supervised ML, real-time processing, rule building, data enrichment, and manual verification, offering only near real-time detection.	The proposed architecture supports supervised ML, real-time processing, rule building, data enrichment, manual verification, and real-time detection.
Saheed <i>et al.</i> [18]	No big data tech was specified. Lacks real-time processing, integration options, and manual verification for suspicious transactions.	The proposed architecture utilizes big data technologies to provide real-time processing, seamless integration capabilities, and manual verification for suspicious transactions.
Tawde <i>et al.</i> [19]	The method lacks real-time processing, integration options, and manual verification for suspicious transactions.	The proposed architecture offers real-time processing, integrates with alternatives, and includes manual verification for suspicious transactions.

Furthermore, as shown in Table 2, prior works (e.g., Stojanovic *et al.* [8], Dai *et al.* [7], and You and Shi [23]) do not explicitly define learning types or require integration with multiple detection algorithms. The proposed architecture improves upon these limitations by supporting various ML algorithms and enabling seamless integration with multiple systems. Moreover, frameworks such as Melo-Acosta *et al.* [11] and Carcillo *et al.* [13] lack real-time capabilities and rule-based support, whereas the architecture ensures real-time transaction processing with rule-based detection mechanisms for greater adaptability.

Implications and future work: the proposed architecture significantly improves fraud detection efficiency by enabling real-time fraud detection, scalability, and multi-model prediction capabilities. This advancement, as highlighted in Table 1, is crucial for handling dynamic fraud patterns that cannot be effectively addressed by traditional batch-processing models.

By bridging the gaps in previous studies and offering a robust, scalable, and real-time fraud detection architecture, the presented work provides a substantial intellectual contribution to the field, advancing both practical applicability and theoretical understanding of fraud analytics. In the future, our approach can be enhanced with advanced XAI techniques for better transparency and adaptive learning for evolving fraud trends. By addressing gaps in previous studies, this work offers a scalable, real-time fraud detection framework that advances both practical and theoretical fraud analytics.

7. CONCLUSION

This study has demonstrated the transformative impact of big data technologies and ML approaches on modern fraud detection systems. The proposed optimized architecture reveals several significant findings with important implications for both the research community and industry practitioners. The dominance of supervised learning techniques (DT, RF, LR, and SVM) across various fraud detection domains represents a critical advancement in the field's ability to identify known fraud patterns with increasing accuracy. However, the research suggests that the integration of these techniques with unsupervised learning approaches creates a more robust detection architecture capable of identifying both established and emerging fraud patterns addressing a fundamental challenge in this rapidly evolving domain. The successful implementation of big data platforms (Spark, Hadoop, HBase) combined with real-time processing tools (Kafka, Storm) demonstrates not merely technological adoption but a necessary evolution in fraud detection capabilities. This technological advancement enables organizations to process the unprecedented volume, velocity, and variety of modern transaction data—a capability that was previously unattainable with traditional data processing methods.

This study's findings have significant implications for organizational strategy in fraud detection. The clear superiority of multi-faceted approaches combining diverse learning techniques with scalable big data technologies suggests that organizations should move away from siloed, single-technology solutions toward integrated architectures that utilize complementary strengths of different approaches. The proposed architecture addresses existing limitations in the field through multiple innovations: enhanced scalability, support for hybrid ML algorithms, comprehensive data enrichment operations, integrated rule-building engines, and human-in-the-loop verification components. These advancements collectively represent a new paradigm in fraud detection that balances automated intelligence with human expertise.

Looking forward, these findings open several promising research directions. Future work should focus on developing self-adapting algorithms capable of continuously evolving alongside fraud patterns, optimization techniques that maintain real-time processing capabilities while handling increasingly complex models, and frameworks for effective knowledge transfer among different fraud domains. The potential application of these approaches extends beyond traditional financial fraud to emerging areas such as IoT security, digital identity verification, and decentralized finance systems. In conclusion, this research enhances our understanding of modern fraud detection technologies while laying the groundwork for next-generation systems capable of adapting to increasingly sophisticated fraud tactics.

ACKNOWLEDGEMENTS

We extend our sincere gratitude to the Faculty of Science, Al-Azhar University, Cairo, Egypt, for its invaluable scientific support and resources, which were instrumental in the success of this research.

FUNDING INFORMATION

The authors state no funding is involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Gaber Elsayed		✓	✓		✓		✓		✓	✓	✓			✓
Abutaleb														
Abdallah A. Alhabshy	✓		✓	✓		✓			✓	✓		✓	✓	
Berihan R. Elemery	✓			✓		✓	✓	✓		✓	✓	✓		
Ebeid Ali		✓		✓				✓		✓	✓	✓		
Kamal Abdelraouf	✓			✓	✓	✓			✓	✓		✓	✓	
Eldahshan														

C : **C**onceptualization

M : **M**ethodology

So : **S**oftware

Va : **V**alidation

Fo : **F**ormal analysis

I : **I**nterpretation

R : **R**esources

D : **D**ata Curation

O : **O**riginal Draft

E : **E**diting

Vi : **V**isualization

Su : **S**upervision

P : **P**roject administration

Fu : **F**unding acquisition

CONFLICT OF INTEREST STATEMENT

The authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES




- [1] J. R. Dorronsoro, F. Ginel, C. Sánchez, and C. Santa Cruz, "Neural fraud detection in credit card operations," *IEEE Transactions on Neural Networks*, vol. 8, no. 4, pp. 827–834, 1997, doi: 10.1109/72.595879.
- [2] W. S. Albrecht, C. O. Albrecht, C. C. Albrecht, and M. F. Zimbelman, *Fraud examination*. 2006.
- [3] B. J. T. Wells, V. Kanhere, and P. D., *Principles of Fraud Examination*. 2014.
- [4] A. Baldominos, E. Albacete, Y. Saez, and P. Isasi, "A scalable machine learning online service for big data real-time analysis," in *2014 IEEE Symposium on Computational Intelligence in Big Data (CIBD)*, Dec. 2014, pp. 1–8, doi: 10.1109/CIBD.2014.7011537.
- [5] M. Di Mauro and C. Di Sarno, "A framework for Internet data real-time processing: A machine-learning approach," in *2014 International Carnahan Conference on Security Technology (ICCST)*, Oct. 2014, vol. 2014-October, no. October, pp. 1–6, doi: 10.1109/CCST.2014.6987044.
- [6] S. Zhao, M. Chandrashekar, Y. Lee, and D. Medhi, "Real-time network anomaly detection system using machine learning," in *2015 11th International Conference on the Design of Reliable Communication Networks (DRCN)*, Mar. 2015, pp. 267–270, doi: 10.1109/DRCN.2015.7149025.
- [7] Y. Dai, J. Yan, X. Tang, H. Zhao, and M. Guo, "Online credit card fraud detection: a hybrid framework with big data technologies," in *2016 IEEE Trustcom/BigDataSE/ISPA*, Aug. 2016, pp. 1644–1651, doi: 10.1109/TrustCom.2016.0253.
- [8] L. Stojanovic, M. Dinic, N. Stojanovic, and A. Stojadinovic, "Big-data-driven anomaly detection in industry (4.0): an approach and a case study," in *2016 IEEE International Conference on Big Data (Big Data)*, Dec. 2016, pp. 1647–1652, doi: 10.1109/BigData.2016.7840777.
- [9] B. Cui and S. He, "Anomaly detection model based on Hadoop Platform and Weka Interface," in *2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, Jul. 2016, pp. 84–89, doi: 10.1109/IMIS.2016.50.
- [10] N. Balasupramanian, B. G. Ephrem, and I. S. Al-Barwani, "User pattern based online fraud detection and prevention using big data analytics and self organizing maps," in *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT)*, Jul. 2017, vol. 2017-Janua, pp. 691–694, doi: 10.1109/ICICT.2017.8342647.
- [11] G. E. Melo-Acosta, F. Duitama-Munoz, and J. D. Arias-Londono, "Fraud detection in big data using supervised and semi-supervised learning techniques," in *2017 IEEE Colombian Conference on Communications and Computing (COLCOM)*, Aug. 2017, pp. 1–6, doi: 10.1109/ColComCon.2017.8088206.
- [12] S. M. Othman, F. M. Ba-Alwi, N. T. Alsohybe, and A. Y. Al-Hashida, "Intrusion detection model using machine learning algorithm on Big Data environment," *Journal of Big Data*, vol. 5, no. 1, p. 34, Dec. 2018, doi: 10.1186/s40537-018-0145-4.
- [13] F. Carcillo, A. Dal Pozzolo, Y.-A. Le Borgne, O. Caelen, Y. Mazzer, and G. Bontempi, "SCARFF: a scalable framework for streaming credit card fraud detection with spark," *Information Fusion*, vol. 41, pp. 182–194, May 2018, doi: 10.1016/j.inffus.2017.09.005.
- [14] L. R. Nair, S. D. Shetty, and S. D. Shetty, "Applying spark based machine learning model on streaming big data for health status prediction," *Computers & Electrical Engineering*, vol. 65, pp. 393–399, Jan. 2018, doi: 10.1016/j.compeleceng.2017.03.009.

- [15] S. Cao, X. Yang, C. Chen, J. Zhou, X. Li, and Y. Qi, "Titant: Online real-time transaction fraud detection in ant financial," *Proceedings of the VLDB Endowment*, vol. 12, no. 12, pp. 2082–2093, Aug. 2019, doi: 10.14778/3352063.3352126.
- [16] H. Zhou, G. Sun, S. Fu, L. Wang, J. Hu, and Y. Gao, "Internet financial fraud detection based on a distributed big data approach with Node2vec," *IEEE Access*, vol. 9, pp. 43378–43386, 2021, doi: 10.1109/ACCESS.2021.3062467.
- [17] R. A. Ariyaluran Habeeb *et al.*, "Clustering-based real-time anomaly detection—a breakthrough in big data technologies," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 8, Aug. 2022, doi: 10.1002/ett.3647.
- [18] Y. K. Saheed, U. A. Baba, and M. A. Raji, "Big data analytics for credit card fraud detection using supervised machine learning models," in *Big Data Analytics in the Insurance Market*, Emerald Publishing Limited, 2022, pp. 31–56.
- [19] S. D. Tawde, S. Arora, and Y. S. Thakur, "Online payment fraud detection for big data," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14501, 2024, pp. 324–337.
- [20] O. E. Akinbowale, P. Mashigo, and M. F. Zerihun, "The integration of forensic accounting and big data technology frameworks for internal fraud mitigation in the banking industry," *Cogent Business & Management*, vol. 10, no. 1, Dec. 2023, doi: 10.1080/23311975.2022.2163560.
- [21] A. N. Bakry, A. S. Alsharkawy, M. S. Farag, and K. R. Raslan, "Automatic suppression of false positive alerts in anti-money laundering systems using machine learning," *The Journal of Supercomputing*, vol. 80, no. 5, pp. 6264–6284, Mar. 2024, doi: 10.1007/s11227-023-05708-z.
- [22] A. Y. A. B. Ahmad, "Fraud prevention in insurance: biometric identity verification and AI-based risk assessment," in *2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS)*, Apr. 2024, pp. 1–6, doi: 10.1109/ICKECS61492.2024.10616613.
- [23] H. You and T. Shi, "Identifying and intercepting telecommunications fraud numbers on the internet through big data technology," *International Journal of Network Security*, vol. 26, no. 5, pp. 786–793, 2024.
- [24] R. Fabrikant, P. E. Kalb, P. H. Bucy, and M. D. Hopson, *Health care fraud: Enforcement and compliance*. Law Journal Press, 2023.
- [25] S. S. Kaddi and M. M. Patil, "Ensemble learning based health care claim fraud detection in an imbalance data environment," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 3, p. 1686, Dec. 2023, doi: 10.11591/ijeecs.v32.i3.pp1686-1694.
- [26] K. A. ElDahshan, A. A. AlHabshy, and B. I. Hameed, "Meta-heuristic optimization algorithm-based hierarchical intrusion detection system," *Computers*, vol. 11, no. 12, p. 170, Nov. 2022, doi: 10.3390/computers11120170.
- [27] I. Idrissi, M. Azizi, and O. Moussaoui, "An unsupervised generative adversarial network based-host intrusion detection system for internet of things devices," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 25, no. 2, p. 1140, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp1140-1150.
- [28] R. Bin Sulaiman, V. Schetinin, and P. Sant, "Review of machine learning approach on credit card fraud detection," *Human-Centric Intelligent Systems*, vol. 2, no. 1–2, pp. 55–68, 2022, doi: 10.1007/s44230-022-00004-0.
- [29] C. M. Gupta and D. Kumar, "Identity theft: a small step towards big financial crimes," *Journal of Financial Crime*, vol. 27, no. 3, pp. 897–910, Oct. 2020, doi: 10.1108/JFC-01-2020-0014.
- [30] R. Anggriawan, M. E. Susila, M. H. Sung, and D. Irynta, "The rising tide of financial crime: a ponzi scheme case analysis," *Lex Scientia Law Review*, vol. 7, no. 1, pp. 307–346, May 2023, doi: 10.15294/lesrev.v7i1.60004.
- [31] A. Metwally, D. Agrawal, and A. El Abbadi, "Using association rules for fraud detection in web advertising networks," *VLDB 2005 - Proceedings of 31st International Conference on Very Large Data Bases*, vol. 1, pp. 169–180, 2005.
- [32] M. Jans, N. Lybaert, and K. Vanhoof, "Internal fraud risk reduction: Results of a data mining case study," *International Journal of Accounting Information Systems*, vol. 11, no. 1, pp. 17–41, Mar. 2010, doi: 10.1016/j.accinf.2009.12.004.
- [33] T.-D. Mai, K. Hoang, A. Baigutanova, G. Alina, and S. Kim, "Customs fraud detection in the presence of concept drift," in *2021 International Conference on Data Mining Workshops (ICDMW)*, Dec. 2021, vol. 2021-Decem, pp. 370–379, doi: 10.1109/ICDMW53433.2021.00052.
- [34] F. Aslam, A. I. Hunjra, Z. Fiti, W. Louhichi, and T. Shams, "Insurance fraud detection: evidence from artificial intelligence and machine learning," *Research in International Business and Finance*, vol. 62, p. 101744, Dec. 2022, doi: 10.1016/j.ribaf.2022.101744.
- [35] A. Y. B. R. Thaifur, M. A. Maidin, A. I. Sidin, and A. Razak, "How to detect healthcare fraud? 'A systematic review,'" *Gaceta Sanitaria*, vol. 35, pp. S441–S449, 2021, doi: 10.1016/j.gaceta.2021.07.022.
- [36] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys*, vol. 46, no. 4, pp. 1–37, Apr. 2014, doi: 10.1145/2523813.
- [37] X. Liu, J. Wu, and Z. Zhou, "Exploratory under-sampling for class-imbalance learning," in *Sixth International Conference on Data Mining (ICDM'06)*, Dec. 2006, pp. 965–969, doi: 10.1109/ICDM.2006.68.
- [38] B. K. Jha, G. G. Sivasankari, and K. R. Venugopal, "Fraud detection and prevention by using big data analytics," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, Mar. 2020, pp. 267–274, doi: 10.1109/ICCMC48092.2020.ICCMC-00050.
- [39] R. A. Ariyaluran Habeeb, F. Nasaruddin, A. Gani, I. A. Targio Hashem, E. Ahmed, and M. Imran, "Real-time big data processing for anomaly detection: A Survey," *International Journal of Information Management*, vol. 45, pp. 289–307, Apr. 2019, doi: 10.1016/j.ijinfomgt.2018.08.006.
- [40] G. Baader and H. Krcmar, "Reducing false positives in fraud detection: Combining the red flag approach with process mining," *International Journal of Accounting Information Systems*, vol. 31, pp. 1–16, Dec. 2018, doi: 10.1016/j.accinf.2018.03.004.
- [41] Y. Demchenko, J. J. Cuadrado-Gallego, O. Chertov, and M. Aleksandrova, "Big data algorithms, MapReduce and Hadoop ecosystem," in *Big Data Infrastructure Technologies for Data Analytics*, Cham: Springer Nature Switzerland, 2024, pp. 145–198.
- [42] K. A. ElDahshan, G. E. Abutaleb, B. R. Elemery, E. A. Ebeid, and A. A. AlHabshy, "An optimized intelligent open-source MLaaS framework for user-friendly clustering and anomaly detection," *The Journal of Supercomputing*, vol. 80, no. 18, pp. 26658–26684, Dec. 2024, doi: 10.1007/s11227-024-06420-2.
- [43] S. Vimal, K. Kayathwal, H. Wadhwa, and G. Dhama, "Application of deep reinforcement learning to payment fraud," *arXiv*, 2021, [Online]. Available: <http://arxiv.org/abs/2112.04236>.
- [44] J. F. Rodríguez, M. Papale, M. Carminati, and S. Zanero, "A natural language processing approach for financial fraud detection," *CEUR Workshop Proceedings*, vol. 3260, pp. 135–149, 2022.
- [45] S.-J. Yu and J.-S. Rha, "Research trends in accounting fraud using network analysis," *Sustainability*, vol. 13, no. 10, p. 5579, May 2021, doi: 10.3390/su13105579.




- [46] A. Abozeid, A. A. AlHabshy, and K. ElDahshan, "A software security optimization architecture (SoSOA) and its adaptation for mobile applications," *International Journal of Interactive Mobile Technologies (IJIM)*, vol. 15, no. 11, p. 148, Jun. 2021, doi: 10.3991/ijim.v15i11.20133.
- [47] L. T. Mohammed, A. A. AlHabshy, and K. A. ElDahshan, "Big data visualization: a survey," in *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, Jun. 2022, pp. 1–12, doi: 10.1109/HORA55278.2022.9799819.
- [48] M. Zaharia, "Apache Spark." <https://spark.apache.org/> (accessed 30-06, 2024).
- [49] M. C. D. Cutting, "Apache Hadoop." <https://hadoop.apache.org/> (accessed 30-06, 2024).
- [50] N. Marz, "Apache Storm." <https://storm.apache.org/> (accessed 06-30, 2024).
- [51] A. S. Foundation, "Apache Flink." <https://flink.apache.org/> (accessed 06-30, 2024).
- [52] A. S. Foundation, "Apache Flume." <https://flume.apache.org/> (accessed 06-30, 2024).
- [53] N. N. A. S. Foundation, "APACHE KAFKA." <https://kafka.apache.org/> (accessed 06-30, 2024).
- [54] Amazon, "Amazon Kinesis." <https://aws.amazon.com/kinesis/> (accessed 06-30, 2024).
- [55] A. S. Foundation, "Apache HBase." <https://hbase.apache.org/> (accessed 06-30, 2024).
- [56] K. A. ElDahshan, A. A. AlHabshy, and G. E. Abutaleb, "Data in the time of COVID-19: a general methodology to select and secure a NoSQL DBMS for medical data," *PeerJ Computer Science*, vol. 6, p. e297, Sep. 2020, doi: 10.7717/peerj-cs.297.
- [57] K. A. ElDahshan, A. A. AlHabshy, and G. E. Abutaleb, "A comparative study among the main categories of NoSQL databases," *Al-Azhar Bulletin of Science*, vol. 31, no. 2, pp. 51–60, Dec. 2020, doi: 10.21608/absb.2020.210374.
- [58] A.-S. Yussiff *et al.*, "The best machine learning model for fraud detection on e-platforms: a systematic literature review," *Computer Science and Information Technologies*, vol. 5, no. 2, pp. 195–204, Jul. 2024, doi: 10.11591/csit.v5i2.p195-204.
- [59] A. S. Foundation, "Apache Cassandra." https://cassandra.apache.org/_/index.html (accessed 6-11, 2024).
- [60] A. S. Foundation, "Apache Mahout." <https://mahout.apache.org/> (accessed 6-11, 2024).
- [61] I. Facebook, "Apache Hive." <https://hive.apache.org/> (accessed 1-11, 2024).
- [62] S. S. Redis, "Redis." <https://redis.io/> (accessed 1-11, 2024).
- [63] VMware, "RabbitMQ." <https://www.rabbitmq.com/> (accessed 1-11, 2024).
- [64] A. S. Foundation, "Apache ActiveMQ." <https://activemq.apache.org/> (accessed 3-11, 2024).

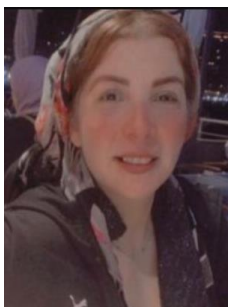
BIOGRAPHIES OF AUTHORS






Gaber E. Abutaleb    earned his B.Sc. in mathematics and computer science from Al-Azhar University in 2013, followed by an M.Sc. in computer science from Al-Azhar University in 2020. He served as a teaching assistant in 2019. His current position is as an assistant lecturer of computer science, at the Faculty of Science, Al-Azhar University, Cairo, Egypt. His research interests include database security, software security, and machine learning. He can be contacted at email: gaber_abutaleb@azhar.edu.eg.






Assoc. Prof. Abdallah A. AlHabshy    is an associate professor of computer Science in the Mathematics Department at Al-Azhar University, specializing in cybersecurity, autonomous systems, and big data security. With over 19+ years of academic and research experience, he is a leading expert in intrusion detection, software security, machine learning applications, and network anomaly detection. His research has led to numerous high-impact publications, including advancements in AI-enhanced behavior detection, optimized machine learning frameworks for anomaly detection, and secure data warehousing. Beyond research, he is dedicated to teaching, mentoring, and contributing to international conferences and collaborative initiatives. His work bridges theory and practice, driving robust cybersecurity and big data solutions while advancing the security and resilience of autonomous systems and distributed data environments. He can be contacted at email: abdallah@azhar.edu.eg.



Assoc. Prof. Beriham R. Elemery    is an associate professor of statistics with a Ph.D. (2011) and 15+ years of teaching experience in governmental and private universities. Currently Vice Dean for Higher Studies and Postgraduates and Academic Director for International Programs at NUB University. Specializes in applied statistics, biostatistics, and statistical industrial analysis. Published internationally in experimental design, industrial engineering, and mathematical statistics. Reviewer for Scopus-indexed journals, including IEEE Access, and has supervised theses and participated in international conferences. She can be contacted at email: berihanelemery@gmail.com.



Dr. Ebeid A. Ebeid    obtained his B.Sc. degree in computer science from the Department of Mathematics, Faculty of Science, Al Azhar University, Cairo, Egypt in 2002 and his M.Sc. degree from the same faculty in 2015. He was involved as a teaching assistant for the Department of Mathematics, Faculty of Science, Al Azhar University in 2012 then a lecturer assistant in 2016. He received his Ph.D. degree in 2020 and he is currently a lecturer of computer science at the same faculty. His research interests include biometrics, pattern recognition, computer vision, machine learning, and AI. He can be contacted at email: ebeidali78@yahoo.com.



Prof. Dr. Kamal A. ElDahshan    earned his graduate degree from Cairo University and a Ph.D. from Université de Technologie de Compiègne, France. He has taught at Université de Technologie de Compiègne and is now a professor at Al-Azhar University, Cairo. With international experience across four continents, he has held positions at Institut National de Télécommunications in Paris and Virginia Tech. He has served as a consultant for the Egyptian Cabinet, a senior advisor to the Ministry of Education, and deputy director of the National Technology Development Centre. He is a Fellow of Open Educational Resources (U.S. Department of State), an ALECSO expert, and a British Computer Society fellow. He can be contacted at email: dahshan@gmail.com.