applied sciences

MDPI

Review

An Introduction to Machine Learning Methods for Fraud Detection

Antonio Alessio Compagnino [1,2,*], Ylenia Maruccia [2,3], Stefano Cavuoti [2,3], Giuseppe Riccio [2,3], Antonio Tutone [1,2], Riccardo Crupi [4] and Antonio Pagliaro [1,2,5,*]

1 INAF IASF Palermo, Via Ugo La Malfa 153, I-90146 Palermo, Italy; antonio.tutone@inaf.it
2 ICSC—Centro Nazionale di Ricerca in HPC, Big Data e Quantum Computing, I-40121 Bologna, Italy; ylenia.maruccia@inaf.it (Y.M); stefano.cavuoti@gmail.com (S.C.); giuseppe.riccio@inaf.it (G.R.)
3 INAF Osservatorio Astronomico di Capodimonte, I-80131 Napoli, Italy
4 Intesa Sanpaolo S.p.A, I-10138 Torino, Italy; riccardo.crupi@intesasanpaolo.com
5 Istituto Nazionale di Fisica Nucleare Sezione di Catania, Via Santa Sofia, 64, I-95123 Catania, Italy
* Correspondence: antonio.compagnino@inaf.it (A.A.C.); antonio.pagliaro@inaf.it (A.P.)

Abstract

Financial fraud represents a critical global challenge with substantial economic and social consequences. This comprehensive review synthesizes the current knowledge on machine learning approaches for financial fraud detection, examining their effectiveness across diverse fraud scenarios. We analyze various fraud types, including credit card fraud, financial statement fraud, insurance fraud, and money laundering, along with their specific detection challenges. The review outlines supervised, unsupervised, and hybrid learning approaches, discussing their applications and performance in different fraud detection contexts. We examine commonly used datasets in fraud detection research and evaluate performance metrics for assessing these systems. The review is further grounded by two case studies applying supervised models to real-world banking data, illustrating the practical challenges of implementing fraud detection systems in operational environments. Through our analysis of the recent literature, we identify persistent challenges, including data imbalance, concept drift, and privacy concerns, while highlighting the emerging trends in deep learning and ensemble methods. This review provides valuable insights for researchers, financial institutions, and practitioners working to develop more effective, adaptive, and interpretable fraud detection systems capable of operating within real-world financial environments.

Keywords: machine learning; deep learning; fraud detection; data-driven finance

Academic Editor: Jose María Alvarez Rodríguez

Received: 25 September 2025
Revised: 20 October 2025
Accepted: 24 October 2025
Published: 5 November 2025

Citation: Compagnino, A.A.; Maruccia, Y.; Cavuoti, S.; Riccio, G.; Tutone, A.; Crupi, R.; Pagliaro, A. An Introduction to Machine Learning Methods for Fraud Detection. *Appl. Sci.* **2025**, *15*, 11787. https://doi.org/10.3390/app152111787

1. Introduction

Financial fraud represents one of the most pressing challenges facing modern business sectors, with severe impacts extending beyond individual organizations to affect the entire global economy. Recent comprehensive surveys reveal that 56% of companies worldwide have experienced some form of fraud, with financial fraud being particularly prevalent and economically damaging [1]. The sophistication and scale of fraudulent activities continue to evolve, rendering traditional detection approaches increasingly inadequate for addressing contemporary threats [2].

The emergence of machine learning (ML) as a powerful analytical tool has revolutionized fraud detection capabilities, enabling organizations to identify complex patterns and

anomalies in vast financial datasets that might indicate fraudulent activities. The strategic application of appropriate ML techniques is crucial for identifying emerging threats while simultaneously minimizing false fraud alarms that can disrupt legitimate business operations [3].

Despite these technological advances, significant challenges persist in developing robust fraud detection systems. These challenges are not merely technical but are deeply rooted in the nature of financial data and the adversarial dynamics of fraud. The key obstacles include extreme class imbalance, where fraudulent transactions are needles in a haystack (often less than 1% of the total), making it difficult for models to learn their characteristics [4–6]. Furthermore, models face constant concept drift as fraudsters continuously evolve their tactics, rendering models trained on historical data obsolete [7]. Finally, the increasing complexity of models, especially deep learning approaches, raises critical issues of interpretability, which is often a strict requirement in highly regulated financial environments [8,9]. These multifaceted challenges significantly complicate the accurate and timely detection of evolving financial fraud schemes.

This comprehensive review aims to provide a systematic analysis of ML techniques for financial fraud detection, addressing the current state of research and practical applications. However, unlike many surveys that remain at a theoretical level, this work aims to bridge the gap between academic research and operational reality. Unlike other reviews, this work bridges the gap between theory and practice through two in-depth case studies on real and proprietary banking data, critically analyzing the operational trade-offs and performance in realistic scenarios. We examine different categories of financial fraud, analyze various ML approaches, explore commonly used datasets, and evaluate performance metrics that are critical for assessing system effectiveness. By synthesizing the recent advances and identifying research gaps, this review provides valuable insights for researchers and practitioners seeking to develop more effective and practical fraud detection systems.

2. Types of Financial Fraud

Financial fraud encompasses a diverse array of deceptive activities designed to achieve illegal financial gain through various mechanisms. A thorough understanding of different fraud types and their characteristics is fundamental for developing targeted and effective detection systems. Based on our comprehensive analysis of the literature, we present a systematic classification of financial fraud into two primary categories: external and internal fraud.

2.1. External Fraud

External fraud represents deceptive activities perpetrated by stakeholders operating outside an organization's direct control. Our literature analysis reveals that 54% of the examined articles focus on investigating various forms of external fraud.

2.1.1. Credit Card and Payment Fraud

Credit card fraud emerges as the most extensively researched type of external fraud in the literature [4,10,11]. Globally, losses from this type of fraud are projected to exceed USD 40 billion annually by 2027. This category involves unauthorized transactions conducted using stolen card information or through sophisticated deceptive mechanisms. Research in this domain typically analyzes transaction patterns, purchase behaviors, temporal aspects of card usage, and geographic anomalies to identify indicators of potential fraud [12].

The scope of payment fraud has expanded significantly with digital transformation, now encompassing online payment fraud, digital wallet transactions, contactless payments, and point-of-sale fraud. The increasing adoption of diverse digital payment systems has

attracted substantial research attention, reflecting the growing complexity of payment fraud schemes [13–15].

2.1.2. Loan Fraud

Loan fraud involves sophisticated deceptive practices throughout the loan lifecycle, including fraudulent loan applications, identity theft schemes, income misrepresentation, and property value inflation [16–18]. ML models designed for loan fraud detection typically analyze applicant data patterns, credit history inconsistencies, application timing anomalies, and cross-referencing patterns to identify potential fraud indicators.

2.1.3. Insurance Fraud

Insurance fraud represents a significant category encompassing various schemes across different insurance sectors. The FBI estimates that the total cost of insurance fraud (non-health insurance) is more than USD 40 billion per year in the United States alone. This includes fraudulent claims in health insurance programs involving document forgery, fraudulent billing practices, and false medical prescriptions [19,20]. Additionally, automobile insurance fraud involving sophisticated collusion networks between policyholders and repair shops presents unique detection challenges [21–23]. ML techniques in this domain analyze claim patterns, policyholder behavioral histories, service provider network relationships, and temporal claim sequences to detect potential fraudulent activities.

2.2. Internal Fraud

Internal fraud consists of deceptive activities perpetrated by individuals with authorized access within an organization. Approximately 46% of the analyzed studies focus on various forms of internal fraud.

2.2.1. Financial Statement Fraud

Financial statement fraud involves sophisticated manipulation of financial reports to misrepresent an organization's true financial performance and position [24–26]. This category is typically studied using comprehensive data from financial regulatory bodies such as the SEC, major stock exchanges, and auditing firms [27]. ML techniques in this domain analyze financial ratios, reporting pattern anomalies, textual elements of financial statements, and cross-period consistency to identify potential fraud indicators.

2.2.2. Money Laundering

Money laundering fraud involves complex schemes designed to disguise the origin of illegally obtained funds by integrating them into legitimate financial transaction flows [28,29]. According to the United Nations, the estimated amount of money laundered globally in one year is 2–5% of the global GDP, or USD 800 billion–USD 2 trillion in current US dollars. ML models for money laundering detection analyze suspicious patterns in transaction networks, customer behavioral anomalies, account activity sequences, and cross-institutional transaction flows that may indicate laundering activities [30].

2.2.3. Tax Fraud

Tax fraud encompasses deliberate misrepresentation of information to tax authorities with the intent to reduce tax liability through various schemes [31,32]. ML techniques in this domain analyze tax return data inconsistencies, business activity patterns, taxpayer network relationships, and cross-referencing with external data sources to identify potential evasion patterns [33].

2.2.4. Asset Misappropriation

This category includes various forms of unauthorized asset manipulation, including unauthorized payment schemes [34,35], dormant account fraud exploitation, smurf fraud techniques, and bulk fraud operations [36]. ML approaches for detecting asset misappropriation analyze transaction patterns, user access logs, timing anomalies, and authorization sequences to detect unusual activities that may indicate misappropriation.

Understanding the specific characteristics and indicators of each fraud type is essential for developing targeted ML approaches that address the unique patterns and challenges associated with different fraud categories. The diversity of fraud types necessitates varied and sophisticated approaches to detection as different fraud schemes exhibit distinct behavioral patterns, temporal characteristics, and network relationships that require specialized analytical techniques.

2.3. Emerging and Complex Fraud Typologies

In addition to these well-established categories, recent developments have introduced new and complex fraud typologies that warrant dedicated attention [37,38]. These include the following:

- **Money Muling**: Individuals are recruited—knowingly or unknowingly—to transfer illicit funds across accounts, masking their criminal origin. These schemes often involve SEPA transfers, instant payments, and prepaid card top-ups, and are associated with device or SIM changes and recurring beneficiary patterns.
- **Account Takeover (ATO)**: Unauthorized access to user accounts via stolen credentials or social engineering, enabling fraudulent transactions under the guise of legitimate users.
- **Authorized Push Payment (APP) Fraud**: Victims are deceived into initiating payments to fraudsters, typically through impersonation or urgency-based manipulation.
- **Synthetic Identity Fraud**: Fraudsters construct fictitious identities by combining real and fabricated personal data, allowing them to open accounts and obtain credit undetected.
- **Social Engineering and Impersonation**: Psychological manipulation techniques are used to extract sensitive information or induce financial actions, often targeting customer support or vulnerable individuals.
- **Business Email Compromise (BEC)**: Fraudsters hijack or spoof corporate email accounts to redirect payments or obtain confidential data through deceptive communications.

These emerging fraud types often blur the boundaries between internal and external threats and require adaptive detection strategies that incorporate behavioral analytics, cross-channel monitoring, and real-time anomaly detection.

3. Review Methodology

This review is based on a systematic literature search designed to identify relevant and impactful research at the intersection of machine learning and financial fraud detection. The process was guided by a defined protocol to ensure comprehensive coverage and minimize bias. Our analysis synthesizes the findings from over 120 peer-reviewed articles, conference proceedings, and technical reports published in the last decade, specifically from January 2014 to December 2023, to capture the most current trends and challenges in the field.

The literature search was conducted across several major academic databases, including Scopus, IEEE Xplore, ACM Digital Library, and Web of Science. Search queries were formulated by combining keywords from three core areas:

- **Fraud Typologies:** “financial fraud", “credit card fraud”, “insurance fraud”, “money laundering”, “payment fraud”, and “financial statement fraud”.
- **Learning Paradigms:** “machine learning”, “deep learning”, “supervised learning”, “unsupervised learning”, “anomaly detection”, and “ensemble methods”.
- **Specific Algorithms:** “Random Forest”, “Support Vector Machine”, “neural network”, “LSTM”, “autoencoder”, and “XGBoost”.

Our inclusion criteria required articles to (i) apply one or more ML techniques to a financial fraud detection task, (ii) present empirical results with performance evaluation, and (iii) be published in English. We excluded studies that were purely theoretical, lacked sufficient methodological detail for replication, or focused on non-financial domains (e.g., click fraud and academic plagiarism). The selection involved an initial screening of titles and abstracts, followed by a full-text review to determine final eligibility for inclusion in this synthesis.

4. Machine Learning Approaches for Fraud Detection

Machine learning approaches for fraud detection can be systematically categorized into three primary methodologies: supervised, unsupervised, and hybrid methods. Each approach offers distinct advantages and limitations depending on the specific fraud detection scenario and available data characteristics.

4.1. Supervised Learning

Supervised learning represents the predominant approach in fraud detection research, accounting for approximately 57% of the techniques employed in the reviewed literature. In this methodology, models are trained on carefully labeled datasets where transactions are explicitly classified as either fraudulent or legitimate, enabling the algorithm to learn discriminative patterns.

4.1.1. Classification Techniques

Random Forest (RF)

Random Forest emerges as the most widely adopted supervised technique for fraud detection, appearing in 34 studies within our review. This ensemble method combines multiple decision trees to improve classification accuracy while effectively reducing overfitting tendencies [39,40]. RF models have demonstrated consistently high accuracy rates in credit card fraud detection and financial statement fraud identification, frequently achieving accuracy rates exceeding 95% [41].

RF overcomes the fundamental limitations of individual decision trees, where splits occur considering the entire dataset, making DT algorithms prone to overfitting and high variance. Instead, RF considers random subsets of data to build multiple decision trees, and, for each tree, a random subset of features is selected from all available features. Furthermore, it employs optimization criteria for finding the best split for each node using Gini index or entropy for classification tasks, or RSS for regression problems.

Extra Trees

Extra Trees (Extremely Randomized Trees) represent a computationally efficient variant of Random Forest [42,43]. This algorithm is particularly well-suited for applications with noisy data or a large number of features.

While similar to Random Forest in using an ensemble of decision trees, Extra Trees introduces a greater degree of randomness in how splits are chosen. Specifically, for each node, a random subset of features is considered, and the split point for each feature is selected randomly rather than being optimized based on criteria like Gini impurity or

entropy. This randomized approach reduces variance and computational cost, often leading to robust performance, although it may sometimes result in a slight increase in bias.

Logistic Regression (LR)

Logistic Regression appears in 32 studies and is particularly valued for its simplicity and interpretability characteristics. It models the probability of a transaction being fraudulent based on its feature characteristics [44,45]. LR demonstrates particular effectiveness for binary classification tasks common in fraud detection, although it may struggle with highly complex and non-linear fraud patterns [46].

Support Vector Machine (SVM)

Support Vector Machine is utilized in 29 studies, separating data points using a hyperplane that maximizes the margin between different classes [47,48]. SVM proves effective for both linear and non-linear classification through various kernel functions, making it suitable for diverse fraud detection scenarios with different data characteristics [49].

In practice, SVM is a linear model used in two primary scenarios: when data are linearly separable (classes are well separated) and when data are not linearly separable (no sharp separation exists in the original space). In the first case, we can identify a line (one-dimensional problem) or hyperplane (multi-dimensional problem) that effectively separates the classes. This separating hyperplane is called the decision boundary.

While multiple separators may perfectly discriminate between classes, the choice of separator significantly impacts how new data points are classified. We compute the perpendicular distance from each training observation to a given separating hyperplane; the smallest distance from observations to the hyperplane is known as the margin. The decision boundary should be positioned as far as possible from data points of both classes. Training points that touch the margin are support vectors, which "support" the maximal margin hyperplane.

For non-linearly separable data, SVM employs the kernel trick, transforming data into higher-dimensional space where linear separation becomes possible. The kernel function implicitly maps data to high-dimensional space, allowing for non-linear decision boundaries in the original space.

Decision Trees (DTs)

Decision trees are employed in 29 studies, creating models that predict target variable classes by learning decision rules from data features [50,51]. DTs are particularly valued for their interpretability, which is crucial in fraud detection, where explaining why a transaction was flagged as fraudulent may be legally or operationally necessary [52].

Decision trees utilize tree-like structures to deliver consequences based on input decisions. They are particularly important for anomaly and fraud detection in industries like finance and banking, where companies deploy decision trees to filter out anomalous or fraudulent loan applications and identify fraudulent customers. The recursive binary tree structure provides excellent interpretability, with the feature space fully described by a single tree.

The goal is to find trees that minimize the Residual Sum of Squares (RSS), a general characteristic of DTs except for Extra Trees. Unfortunately, finding the best binary partition in terms of minimum sum of squares is not feasible in practice because it is computationally infeasible to consider every possible partition of the feature space. For this reason, we employ a top-down greedy approach known as recursive binary splitting.

Naive Bayes (NB)

Naive Bayes appears in 19 studies, applying Bayes' theorem with an independence assumption between features [53,54]. While this independence assumption rarely holds in real-world financial data, NB often performs surprisingly well in fraud detection tasks, particularly when dealing with limited training data [44].

This probabilistic classifier represents one of the fastest, most accurate, and reliable supervised learning algorithms. It assumes features are normally distributed and independent of each other. The algorithm uses prior probability P(H) of hypothesis H being true and posterior probability P(H | D) of data D given that hypothesis H is true.

K-Nearest Neighbor (KNN)

K-Nearest Neighbor is used in 14 studies, representing an instance-based learning algorithm that classifies transactions based on their proximity to known samples in the feature space [55,56]. KNN proves effective for detecting fraud patterns that cluster in feature space but may struggle with high-dimensional data [57].

KNN is a non-parametric supervised approach used for both classification and regression problems. The algorithm works by finding the k most similar objects to a given object x based on distance or similarity measures between x and all objects in the dataset. It then assigns a label to x based on the most frequent label among its k neighbors using either majority voting or weighted voting approaches.

Artificial Neural Networks (ANNs)

Artificial neural networks are employed in 17 studies, capable of modeling complex non-linear relationships between features and fraud likelihood [58,59]. While effective, traditional ANNs may require significant computational resources and careful hyperparameter tuning [60].

Neural networks are information processing models inspired by biological neuron systems, composed of highly interconnected processing elements known as "neurons." These networks are adaptive systems that can change their internal structure by adjusting input weights. Each input variable is multiplied by respective weights, then summed together to form net output, with bias added.

The ANN creates three layers in the neural network: input layer, hidden layer, and output layer. The first layer receives raw input, processes it, and passes processed information to hidden layers, which pass information to the output layer that produces the final output. The advantage of neural networks is their adaptive nature, learning from provided information to optimize weights for better prediction in unknown outcome situations.

XGBoost

XGBoost appears in 13 studies, representing a gradient boosting framework known for exceptional performance and computational speed [61,62]. It sequentially builds decision trees, with each tree correcting errors made by previous trees, making it particularly effective for imbalanced datasets common in fraud detection [63].

Boosting (XGBoost) is an ensemble technique creating a collection of predictors where learners are learned sequentially. Early learners fit simple models to data, then analyze data for errors. The goal is to solve errors from prior trees at every step. Models can have different importance or weights, and datasets are weighted so that observations incorrectly classified by previous classifiers receive greater importance in subsequent model training.

4.2. Unsupervised Learning

Unsupervised learning techniques are employed in approximately 18% of the reviewed studies. These methods identify patterns and anomalies without requiring labeled training data, making them particularly valuable for detecting novel fraud patterns that may not be represented in historical labeled datasets.

4.2.1. Isolation Forest

Isolation Forest appears in 19 studies, isolating observations by randomly selecting features and randomly selecting split values between maximum and minimum values of selected features [64,65]. It proves particularly effective for identifying outliers, which frequently represent fraudulent transactions [66].

4.2.2. Autoencoders

Autoencoders are used in 10 studies, representing neural networks that learn to compress and reconstruct data, with reconstruction error serving as an anomaly score [14,67]. They are particularly effective for dimensionality reduction and capturing complex patterns in transaction data [68].

Autoencoders are latent variable models that discover latent variables (variables not directly observed but inferred from direct observations). They consist of an encoder that learns mapping from data X to low-dimensional latent space Z and a decoder that performs the reverse operation, increasing dimensions from latent space Z back to original dimensions.

The process represents a form of compression, keeping the core information in data. The encoder learns mapping from data X to low-dimensional latent space Z, while the decoder learns to use latent features Z to reconstruct the original data. Using the distance between reconstructed data and real data, we can determine if a signal represents an anomaly or normal behavior.

4.2.3. K-Means Clustering

K-means appears in 7 studies, partitioning data into k clusters with each observation belonging to the cluster with the nearest mean [69,70]. It can identify groups of similar transactions, with those falling outside established clusters potentially representing fraudulent activities [23].

The K-means algorithm divides N samples into K disjoint clusters of equal variances, minimizing within-cluster sum-of-squares (WSS) while maximizing between-cluster sum of squares (BSS). Each cluster is represented by the mean of contained observations.

4.2.4. Hidden Markov Models (HMMs)

HMMs are employed in 7 studies, representing statistical models that assume the system being modeled follows a Markov process with unobserved states [71,72]. HMMs are particularly useful for modeling sequential data like transaction sequences, where they can learn normal spending behaviors and flag significant deviations [73].

4.2.5. Local Outlier Factor (LOF)

LOF appears in 13 studies, comparing the local density of a point with local densities of its neighbors, identifying samples with substantially lower density than neighbors as potential outliers [65,74].

4.3. Deep Learning Approaches

Deep learning approaches constitute a rapidly growing segment of fraud detection research, with approximately 34 studies employing these sophisticated techniques either independently or in combination with traditional ML methods.

4.3.1. Long Short-Term Memory (LSTM)

LSTM networks are used in 8 studies, representing specialized recurrent neural networks designed to model temporal sequences and long-range dependencies [75,76]. They prove particularly effective for analyzing sequential transaction data, capturing temporal patterns that may indicate fraudulent behavior [77].

LSTMs are special kinds of recurrent neural networks capable of learning long-term dependencies in data through internal mechanisms called gates that regulate information flow. These gates can learn which data in a sequence is important to keep or forget, allowing LSTM models to store information for extended periods.

4.3.2. Convolutional Neural Networks (CNNs)

CNNs appear in 7 studies. Originally designed for image processing, CNNs can be effectively applied to structured data for fraud detection [58,78]. The core idea is to treat structured financial data as a grid, similar to an image, and apply convolutional filters (kernels) to it. They automatically learn hierarchical features from data, where initial layers capture simple patterns and deeper layers combine them to identify more complex relationships, which is valuable for identifying sophisticated fraud patterns [79].

Using conv2d implementation, the most popular function for building convolutional layers requires setting input channels, output channels, kernel size, stride, padding, and dilation parameters. The kernel is the filter that slides over the data, the stride defines the step size of this movement, and padding adds a border to control the output dimensions. The general formula to calculate the output size of a convolution is

$$\text{Output Size} = \left\lfloor \frac{\text{Input Size} + 2 \times \text{Padding} - \text{Kernel Size}}{\text{Stride}} \right\rfloor + 1 \quad (1)$$

where

- **Input Size**: The dimensions (height and width) of the input data.
- **Kernel Size**: The dimensions of the filter used to scan the data.
- **Padding**: A border of zeros added to the input, primarily to control the output's spatial dimensions.
- **Stride**: The step size, or how many pixels the filter moves at a time across the input.

In this formula, the **input size** is adjusted by adding twice the **padding** (to account for both sides) and subtracting the **kernel size**. This result is then divided by the **Stride** (the step size of the filter). The floor function $\lfloor \cdot \rfloor$ ensures the output dimension is an integer, and one is added to finalize the count of possible kernel positions. This determines the spatial dimensions of the feature map produced by the layer. This process allows the network to build a rich multi-level representation of the input data to effectively classify transactions.

4.3.3. Recurrent Neural Networks (RNNs)

RNNs are employed in 7 studies, representing networks that maintain memory of previous inputs, making them suitable for sequential data analysis [80,81]. They can model temporal dependencies in transaction sequences, helping to identify unusual patterns [82].

RNNs are dynamic networks that account for temporal aspects, exhibiting cyclic behavior that allows forward and backward processing to reconstruct temporal sequences. They are particularly well-suited for sentiment analysis and other sequential data processing tasks.

4.3.4. Generative Adversarial Networks (GANs)

GANs appear in 7 studies, consisting of generator and discriminator networks that compete against each other [83,84]. In fraud detection, GANs can generate synthetic

fraud samples to improve classifier training, particularly valuable for highly imbalanced datasets [5].

While promising for data augmentation, GANs face challenges such as training instability and "mode collapse," where the generator produces a limited variety of samples. Furthermore, evaluating the quality of synthetic data remains an open research question, making their practical deployment complex.

GANs utilize two competing neural networks: a generator (G) that creates samples from random noise and a discriminator (D) that distinguishes between real and fake samples. The generator is trained indirectly through its ability to "fool" the discriminator.

Deep Convolutional Generative Adversarial Networks (DCGANs) extend GANs using convolutional architectures for improved image generation and feature extraction capabilities.

4.4. Hybrid and Ensemble Methods

Many studies employ hybrid approaches that strategically combine multiple ML techniques to improve overall detection performance and robustness.

4.4.1. Supervised-Unsupervised Hybrids

Approximately 15% of studies combine supervised and unsupervised techniques, such as using unsupervised learning for feature extraction followed by supervised classification [85,86].

4.4.2. Ensemble Methods

Techniques including bagging (5 studies), boosting (59 studies), and stacking (4 studies) combine multiple models to improve prediction accuracy [87–89]. These approaches can effectively mitigate individual model weaknesses and improve overall detection performance [52].

4.4.3. Deep Learning Hybrids

Some studies combine traditional ML with deep learning approaches, such as using deep learning for feature extraction and traditional ML for final classification [4,90].

The prevalence of supervised learning in fraud detection research reflects the availability of labeled historical data in many financial institutions. However, the growing interest in unsupervised and deep learning approaches indicates recognition of their potential to identify novel and evolving fraud patterns that may not be well-represented in historical training data.

To provide a clear comparative overview of the most common techniques discussed in the literature, Table 1 summarizes their key characteristics, advantages, and disadvantages in the context of financial fraud detection. This synthesis highlights the critical trade-offs between model performance, complexity, and interpretability that practitioners and researchers must navigate.

Table 1. Comparative analysis of machine learning approaches in fraud detection.

Algorithm	Advantages	Disadvantages	Complexity	Interpretability	Handling Imbalance	Application Examples
Random Forest (RF)	- Most widely adopted supervised technique. - Reduces overfitting and improves accuracy. - Consistently high performance (>95% accuracy).	- Less interpretable than a single decision tree. - Case Study 2 shows difficulty improving recall even with `class_weight`.	Moderate-High: Requires building and aggregating multiple decision trees.	Low-Moderate: Not a complete black box; feature importance can be extracted.	Can handle imbalanced data via parameters like `class_weight`, although with varying effectiveness.	Credit card fraud and financial statement fraud identification.

Table 1. *Cont.*

Algorithm	Advantages	Disadvantages	Complexity	Interpretability	Handling Imbalance	Application Examples
Logistic Regression (LR)	- Valued for simplicity and high interpretability. - Effective for binary classification.	- May struggle with complex non-linear fraud patterns.	Low: Simple linear model; fast to train.	Very High: Coefficients directly indicate the influence of each feature.	Often requires preprocessing (e.g., resampling) as it can be biased towards the majority class.	Binary classification tasks common in fraud detection.
XGBoost	- Known for exceptional performance and computational speed. - Particularly effective with imbalanced datasets.	- More complex to tune than simpler models. - Risk of overfitting if not carefully configured.	High: A gradient boosting framework with numerous hyperparameters to tune.	Low: As an ensemble of trees, it is difficult to interpret, although it provides feature importance scores.	Highly effective as it sequentially builds trees to correct errors of prior ones.	Fraud detection in contexts with highly imbalanced data.
Isolation Forest	- Unsupervised, requires no labeled data. - Effective at isolating outliers, which often represent fraud.	- Less effective if fraud patterns are complex and mimic normal behavior.	Moderate: Based on an ensemble of trees, often computationally efficient.	Low: Provides an anomaly score but not a clear reason why a point was isolated.	Natively designed for this purpose; its goal is to isolate rare data points, making it ideal for imbalanced data.	Anomaly and outlier detection.
Autoencoders	- Unsupervised; learn complex patterns and perform dimensionality reduction. - Anomaly score is based on reconstruction error.	- Designing the neural network architecture can be complex.	High: Requires designing and training a neural network.	Very Low: A quintessential "black-box" model; reconstruction error indicates an anomaly but not the cause.	Natively suited: Learns the representation of "normal" (majority class) data and fails to reconstruct anomalies well.	Anomaly detection and dimensionality reduction in transactional data.
LSTM	- Specialized for modeling temporal sequences and long-range dependencies. - Ideal for analyzing sequences of transactions.	- Requires data in a sequential format. - Computationally expensive to train.	Very High: A recurrent neural network with a complex internal "gate" architecture.	Very Low: The recurrent nature and temporal dependencies make it extremely difficult to interpret.	Can identify deviations from "normal" sequences, but often requires standard imbalance techniques in the final classification layer.	Analysis of sequential transactional data to capture temporal fraud patterns.

5. Datasets for Fraud Detection Research

The performance and practical applicability of fraud detection models depend significantly on the quality, characteristics, and representativeness of the datasets used for training and evaluation. Our comprehensive analysis identified several commonly used datasets in fraud detection research, each with distinct characteristics relevant to different fraud types and detection scenarios.

5.1. Credit Card Fraud Detection Datasets

5.1.1. Credit Card Fraud Detection Dataset

This dataset from the Machine Learning Group at Université Libre de Bruxelles represents the most widely used resource in credit card fraud detection research, appearing in 15 studies [10,91,92]. It contains anonymized credit card transactions from European cardholders in September 2013, with only 492 frauds out of 284,807 transactions (0.172% fraud rate). Most features underwent PCA transformation for confidentiality protection, with only 'Time' and 'Amount' retained as original features [93].

5.1.2. German Credit Data

Created by Professor Hofmann for the UCI ML repository, this dataset focuses on credit risk classification [94]. It contains 1000 instances with 20 attributes describing individual characteristics and credit information. It has been utilized in studies [90,95,96].

5.1.3. Australian Credit Approval

This UCI ML repository dataset contains 690 instances and 14 attributes related to credit card applications [97]. It has been employed in studies [95,96,98].

5.1.4. Default of Credit Card Clients

This dataset from the UCI ML repository focuses on defaulted payments of credit card customers in Taiwan [99]. It includes 30,000 instances with 24 attributes covering credit data and payment history from April to September 2005. It has been used by [90,96,98].

5.2. Financial Statement Fraud Datasets

5.2.1. China Stock Market and Accounting Research (CSMAR)

This comprehensive dataset provides information on China's stock markets and financial statements of listed companies between 1998 and 2016 [100]. It includes 35,574 samples with 337 annual fraud cases, used in studies [17,101,102].

5.2.2. Compustat

This database contains financial and economic information on US and Canadian companies [103]. It includes data on 228 companies, with half showing fraud in their financial information. It has been used in studies [25,104].

5.3. Synthetic Datasets

5.3.1. PaySim Mobile Money Simulator

This synthetic dataset was generated using aggregated data from a mobile money service in an African country [105]. It contains 6,362,620 samples with 8213 fraudulent transactions. It has been used in studies [106–108].

5.3.2. BankSim Payment Simulator

Based on a sample of transactional data from a Spanish bank, this synthetic dataset includes 594,643 transactions, with approximately 1.2% (7200) labeled as fraud [105]. It has been used by [90,96,98].

5.4. Other Specialized Datasets

5.4.1. Insurance Company Benchmark (COIL 2000)

This dataset contains information about customers of an insurance company, including product usage and sociodemographic data [109]. It has 9822 instances with 86 attributes and has been used in insurance fraud detection studies [19,110].

5.4.2. Bitcoin Network Transactional Metadata

This dataset contains Bitcoin transaction metadata from 2011 to 2013, with 30,000 instances and 11 attributes related to Bitcoin transactions and flows [111]. It has been used for analyzing anomalies in cryptocurrency transactions [112].

5.5. Characteristics of Fraud Detection Datasets

Our analysis reveals several important characteristics that significantly impact fraud detection research.

5.5.1. Class Imbalance

Most fraud detection datasets exhibit extreme class imbalance, with fraudulent transactions typically constituting less than 1% of all transactions. This imbalance accurately reflects real-world fraud prevalence but presents significant challenges for model training and evaluation [40,113].

5.5.2. Feature Transformation

Many datasets, particularly those containing sensitive financial information, undergo feature transformation (e.g., PCA) to protect privacy. While necessary for data protection, this transformation can obscure the interpretability of resulting models and limit domain knowledge integration [10].

5.5.3. Temporal Aspects

Some datasets preserve crucial temporal information, allowing for analysis of how fraud patterns evolve over time. This temporal dimension is essential for developing models that can adapt to emerging fraud strategies and concept drift.

5.5.4. Real vs. Synthetic Data

While most studies (approximately 93%) utilize real-world data, there is growing interest in synthetic datasets that can simulate diverse fraud scenarios without privacy concerns. These synthetic datasets prove particularly valuable for testing model robustness against various fraud strategies and for scenarios where real fraud data is scarce [105].

5.6. Guidelines for Future Dataset Development

Based on the identified challenges, we propose the following guidelines to advance research in this area:

- **Establishment of Standardized Benchmarks**: The community would benefit greatly from the creation and maintenance of shared, large-scale, and contemporary benchmark datasets. This would allow for a more direct and fair comparison of different models and techniques.
- **Privacy-Preserving Data Sharing**: Techniques like federated learning should be further explored to enable collaborative model training across different institutions without centralizing sensitive data. This could lead to more robust models trained on a wider variety of fraud patterns.
- **Advanced Synthetic Data Generation**: While synthetic datasets exist, future work should focus on generating data that more accurately captures complex multi-modal fraud scenarios and temporal dynamics using advanced generative models like Wasserstein GANs (WGANs) or variational autoencoders (VAEs).

The choice of dataset significantly impacts both model performance and practical applicability of fraud detection systems. Datasets that accurately reflect the complexity and evolving nature of real-world fraud are essential for developing effective detection systems. However, the limited availability of recent, comprehensive, and publicly accessible fraud datasets remains a significant challenge for researchers in this field.

6. Performance Metrics for Fraud Detection

Evaluating the performance of fraud detection models requires appropriate metrics that address the specific challenges of the domain, particularly class imbalance and asymmetric misclassification costs. The selection of evaluation metrics must carefully consider the operational context and business requirements of fraud detection systems.

6.1. Supervised Learning Metrics

6.1.1. Accuracy

Accuracy measures the proportion of correct predictions (both fraud and non-fraud) to total predictions:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2}$$

While commonly reported, accuracy can be misleading in highly imbalanced datasets, where simply classifying all transactions as non-fraudulent would yield deceptively high accuracy [114].

6.1.2. Precision and Recall

These two metrics are crucial for imbalanced classification. Precision measures the proportion of correctly identified fraudulent transactions among all transactions classified as fraudulent:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

High precision indicates a low false positive rate (also known as Type I Error), which is critically important for minimizing unnecessary interventions in legitimate transactions [115]. Conversely, recall (or sensitivity) measures the proportion of actual fraudulent transactions that were correctly identified:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

High recall is essential for minimizing financial losses and ensuring regulatory compliance as it directly relates to a low false negative rate (Type II Error, which is equal to $1 - \text{Recall}$) [115,116].

6.1.3. F1-Score

F1-score is the harmonic mean of precision and recall, providing a balanced measure of model performance:

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

This metric is particularly valuable for fraud detection as it balances the need to minimize both false positives and false negatives, making it especially useful for imbalanced datasets [117].

6.1.4. Area Under the ROC Curve (AUC–ROC)

AUC–ROC represents the model's ability to discriminate between fraudulent and non-fraudulent transactions across different threshold settings:

$$\text{AUC–ROC} = \int_0^1 \text{TPR}(\text{FPR}^{-1}(t))dt \tag{6}$$

Higher AUC–ROC values indicate better model performance, with values closer to 1 representing near-perfect discrimination [17].

6.2. Unsupervised Learning Metrics

Evaluating unsupervised learning models for fraud detection presents unique challenges as these models do not rely on labeled training data. Several specialized metrics have been developed for this purpose.

6.2.1. Silhouette Coefficient

This metric measures how well data points are clustered, with values ranging from -1 to 1:

$$s(j) = \frac{y - x}{\max(x, y)} \tag{7}$$

where x is the average distance between point j and other points in its cluster, and y is the minimum average distance to points in another cluster [118].

The average silhouette coefficient measures clustering quality by assessing how well each data point fits within its assigned cluster. Higher average silhouette width indicates better clustering performance.

6.2.2. Davies–Bouldin Index

This metric evaluates clustering quality based on the ratio of within-cluster scatter to between-cluster separation:

$$\mathrm{DB} = \frac{1}{K}\sum_{i=1}^{k}\max_{j\neq i}\left(\frac{\alpha_i + \alpha_j}{d(c_i, c_j)}\right) \tag{8}$$

where K is the number of clusters, c_i and c_j are cluster centroids, and α_i and α_j are the average distances of all elements in clusters i and j to their respective centroids [118].

6.2.3. Rand Index

This measures the similarity between two data clusterings:

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \times 100 \tag{9}$$

It is particularly useful when external labels are available for evaluation purposes [118]. Having clustering algorithms with very high silhouette coefficient but low external measures indicates poor performance when observations are assigned to wrong clusters.

6.3. Practical Considerations in Metric Selection

The selection of appropriate performance metrics should consider several domain-specific factors:

6.3.1. Class Imbalance

In highly imbalanced datasets, metrics like precision, recall, F1-score, and AUC–ROC provide more meaningful assessments than accuracy alone [117].

6.3.2. Operational Context

The operational context of fraud detection systems should inform metric selection. For example, in credit card fraud detection, minimizing false positives (high precision) may be prioritized to avoid customer inconvenience, while, in money laundering detection, minimizing false negatives (high recall) may be more critical due to regulatory requirements [119].

6.3.3. Cost Sensitivity

The asymmetric costs of false positives and false negatives should be carefully considered. Cost-sensitive metrics that incorporate the financial impact of misclassifications can provide more practically relevant performance assessments [115].

6.3.4. Time Sensitivity

For real-time fraud detection systems, metrics that incorporate detection latency are valuable as early detection can significantly reduce financial losses [75].

The comprehensive evaluation of fraud detection systems requires multiple complementary metrics that collectively address the complex requirements of the domain. While no single metric can capture all relevant aspects of performance, carefully selected metric combinations can provide meaningful insights into a system's practical utility and operational effectiveness.

6.4. SoTA Micro-Benchmark on Standard Dataset (ULB 2013)

In Table 2, we report **AUPRC** for five commonly used methods on the ULB 2013 credit-card dataset. Tree/boosting methods lead AUPRC; kNN lags; values provide an external baseline for our proprietary studies [93,120–122].

Table 2. SoTA micro-benchmark on ULB 2013 (AUPRC).

Dataset	Method	AUPRC
ULB 2013	Logistic Regression (L1)	0.724
ULB 2013	Random Forest	0.871
ULB 2013	XGBoost	0.867
ULB 2013	AdaBoost	0.808
ULB 2013	K-Nearest Neighbor	0.585

7. Case Study 1 : Application of Supervised Models on a Real-World Banking Dataset

To provide practical context for the reviewed machine learning techniques and performance metrics, this section presents a comprehensive case study based on the application of selected supervised learning models to a real-world financial dataset.

While this case study focuses on tree-based supervised models, it is worth noting that other studies have demonstrated the potential of different approaches. For instance, deep learning models like LSTMs have shown promise in capturing sequential patterns in transaction data, while hybrid models combining unsupervised anomaly detection with supervised classifiers have been effective in identifying novel fraud typologies.

7.1. Dataset and Preprocessing

The dataset for this study was derived from actual transactional data provided by a financial institution, comprising two main sources: bank transfers (“Bonifici”) and cardless payments (“Cardless”). The raw data, initially consisting of over 427,000 records across multiple interconnected tables, underwent a multi-stage preprocessing pipeline to prepare it for modeling.

First, a data cleaning stage was performed to ensure data quality. Records with critical missing values (e.g., transaction amount and timestamp) were excluded, and duplicate entries were removed to prevent model bias. Following this, a feature engineering process was undertaken to extract relevant variables.

From the cleaned dataset, a specific subset of 48,559 instances was carefully selected. This selection was not random but targeted a distinct operational period for which a meticulously verified ground truth was available. Each transaction in this subset was authoritatively labeled as either legitimate or fraudulent by domain experts. This process, while reducing the dataset size, was critical to guarantee a high-quality dataset for a controlled and reliable evaluation of model performance.

The resulting features included a combination of numerical and categorical attributes. To ensure compatibility with the machine learning models, categorical features were transformed using one-hot encoding. In line with best practices for tree-based ensembles, numerical features were left unscaled as these models are not sensitive to feature magnitude.

Table 3 summarizes the key characteristics of the final dataset used for the experiments. A significant and challenging characteristic of this dataset is the severe class imbalance, with a fraud rate of approximately 1.43%.

Table 3. Summary of the banking dataset characteristics.

Characteristic	Value
Total Instances	48,559
Fraudulent Instances	696
Fraud Rate	~1.43%
Initial "Bonifici" Records	242,792 records, 52 features
Initial "Cardless" Records	184,729 records, 50 features
Example Numerical Features	`num_impor`, `num_hour`, `num_day_of_month`, `num_month`, `num_day_of_week`, `num_longi`, `num_latit`
Example Categorical Features	`cat_CountryCodeBIC_f415b`, `cat_bank_code_4f875`

7.2. Experimental Setup and Models

Three supervised learning algorithms, discussed comprehensively in 4, were selected for this comparative analysis:

- Random Forest (RF).
- Extreme Gradient Boosting (XGBoost).
- Extra Trees (ETs).

The models were trained on the processed dataset to classify transactions as either legitimate (class 0) or fraudulent (class 1). The primary evaluation was conducted using confusion matrices and comprehensive classification reports, focusing on metrics particularly relevant to imbalanced datasets, such as precision, recall, and F1-score for the minority (fraud) class. Feature importance was also extracted from each model to identify key indicators contributing to fraud classification.

7.2.1. Data Splits and Validation

We adopt a temporal split to prevent leakage: training on transactions from period $T_0 \rightarrow T_{n-1}$ and testing on T_n. Within training, we use a stratified k-fold cross-validation ($k = 5$) to select hyperparameters. We fix a global random seed (e.g., 42) and repeat each experiment $r = 3$ times; we report mean $\pm$ standard deviation.

7.2.2. Preprocessing and Leakage Prevention

Categorical features are one-hot encoded; numerical features are left unscaled for tree ensembles. To avoid customer-level leakage, all transactions belonging to the same customer (or account identifier available) are assigned to the *same* split. Feature derivations use only past information relative to transaction time (no look-ahead).

7.2.3. Hyperparameter Tuning

We perform randomized search with $N = 50$ configurations per model, selecting by validation AUPRC.

- **RF/ET**: `n_estimators` $\in \{200, 400, 800\}$, `max_depth` $\in \{None, 8, 12, 16\}$, `max_features` $\in \{$`sqrt`, `log2`, $0.5\}$, `min_samples_leaf` $\in \{1, 2, 5, 10\}$.
- **XGB**: `n_estimators` $\in \{300, 600, 1000\}$, `learning_rate` $\in [0.02, 0.2]$, `max_depth` $\in \{4, 6, 8\}$, `subsample`, `colsample_bytree` $\in [0.6, 1.0]$, `reg_alpha`, `reg_lambda` $\in [0, 10]$.

7.2.4. Probability Calibration and Thresholding

We calibrate scores with isotonic regression on the validation folds and choose the operating threshold to minimize expected cost:

$$\text{Cost} = C_{\text{FN}} \cdot \text{FN} + C_{\text{FP}} \cdot \text{FP},$$

with C_{FN}/C_{FP} set by business constraints (e.g., 50:1). We also report threshold-free metrics (AUPRC and AUROC).

7.2.5. Evaluation Metrics

In addition to precision/recall/F1 and FPR already reported, we add (i) area under the precision–recall curve (AUPRC), (ii) Precision@K and Recall@K (K = top 0.1% and 0.5% scored transactions), (iii) calibration (Brier score; reliability plot), and (iv) cost curves for several C_{FN}/C_{FP} ratios.

7.3. Results and Analysis

Test split and prevalence. The held-out test set contains $N = 5000$ transactions with a fraud prevalence of 2.37% (119 positives). We report **AUPRC** and **Recall@K** (K = 0.1% and 0.5% of top-ranked transactions), in addition to standard metrics. AUPRC is computed from out-of-sample fraud probabilities; Recall@K is the recall achieved when screening the top K% transactions ranked by the model score (K = 0.1% $\Rightarrow$ 5 alerts; K = 0.5% $\Rightarrow$ 25 alerts). (see Figure 1 and Table 4)

Table 4. Case Study 1—test set results ($N = 5000$; prevalence 2.37%). We report accuracy, F1 (weighted), AUPRC, and Recall@K.

Model	Accuracy	F1 (w)	AUPRC	Recall@0.1%	Recall@0.5%
Random Forest	0.9838	0.9801	0.6188	0.0336	0.2017
Gradient Boosting	0.9810	0.9756	0.5153	0.0336	0.1765
KNN	0.9806	0.9755	0.4123	0.0336	0.1933
Logistic Regression	0.9768	0.9733	0.3263	0.0336	0.1261
AdaBoost	0.9756	0.9675	0.1914	0.0336	0.1008
SVC (probability)	0.9772	0.9672	0.4957	0.0336	0.1849

Best model and confusion matrix. The best overall model is *Random Forest* (F1(w) $=$ 0.9801; AUPRC $=$ 0.6188). At the default decision threshold, its confusion matrix is

$$\begin{bmatrix} \text{TN} = 4878 & \text{FP} = 3 \\ \text{FN} = 78 & \text{TP} = 41 \end{bmatrix}$$

($\text{precision}_{\text{fraud}} = 0.93$; $\text{recall}_{\text{fraud}} = 0.34$).

Operational screening (Recall@K). Ranking by the fraud score, Random Forest retrieves the following: (i) Recall@0.1% $= 0.0336 \Rightarrow$ among the top 5 alerts, it catches $\sim$4 frauds (of 119); (ii) Recall@0.5% $= 0.2017 \Rightarrow$ among the top 25 alerts, it catches $\sim$24 frauds. These figures provide an actionable triage view for limited human capacity and complement threshold-based metrics.

Takeaways. (1) *Ranking quality:* Tree/boosting methods (RF and GBDT) dominate AUPRC, confirming better ordering of rare positives under heavy imbalance. (2) *Capacity-aware recall:* With only 25 top-ranked alerts, RF recovers roughly one-fifth of all frauds—useful when manual review bandwidth is tight. (3) *Threshold vs. ranking:* High accuracy with low fraud recall at a fixed threshold suggests cost-aware thresholding or top-K review as preferred operating modes in production.

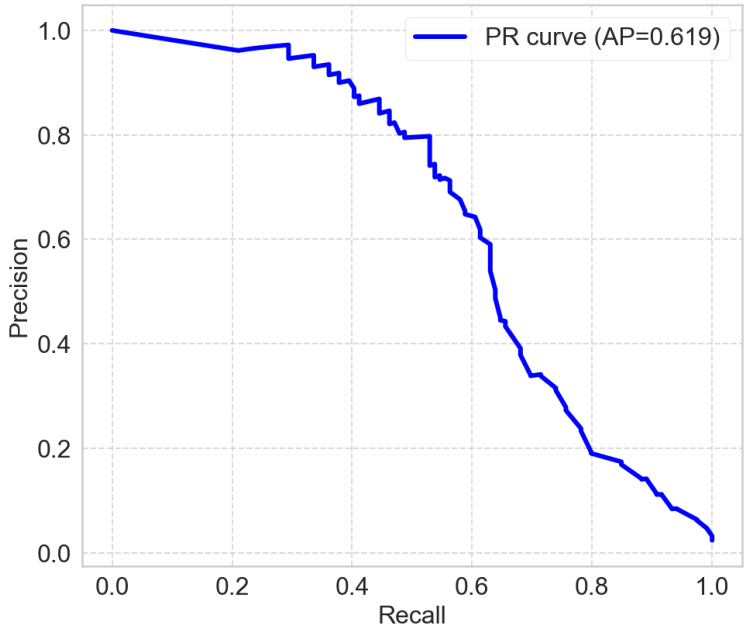

Figure 1. Case Study 1—precision–recall curve for the best model (Random Forest). The average precision (AUPRC) is 0.619 on the test set.

7.4. Performance Analysis and Operational Implications

The detailed results presented highlight the persistent challenges in operational fraud detection. While the overall accuracy is high, it remains a misleading metric in this imbalanced context. A more insightful analysis focuses on the trade-off between ranking quality (AUPRC) and the model's performance at a fixed decision threshold.

The best-performing model, Random Forest, achieves an AUPRC of 0.619, indicating a good ability to rank fraudulent transactions higher than legitimate ones. However, this ranking strength does not translate to sufficient performance at a standard decision threshold. The model's recall for the fraud class is only 0.34. Operationally, this is the most critical finding: the system, even at its best, fails to detect roughly two-thirds (66%) of all fraudulent transactions, representing an unacceptable level of risk for a financial institution.

While the model's high precision (0.93) ensures that the few alerts it generates are highly reliable, this comes at the severe cost of a large number of missed frauds (false negatives). This starkly illustrates that even a well-ranked model can be operationally ineffective if deployed with a simple threshold-based strategy.

The persistently low recall suggests that the challenge may be inherent to the data itself, where fraudulent patterns are too subtle or similar to legitimate behavior to be distinguished by standard models alone. While this study focused on tree-based ensembles, the literature indicates that more advanced strategies could yield better results. These include **data-level approaches** like synthetic oversampling (e.g., SMOTE) to create a more balanced training set [113,116], **algorithm-level approaches** such as cost-sensitive learning that assigns a higher penalty to misclassifying frauds [12,115], and **hybrid models** that combine anomaly detection with supervised classification [85,86].

Ultimately, this analysis confirms that the supervised methods tested, when used in isolation, are insufficient for robust standalone deployment. The fundamental challenge of balancing adequate fraud detection (high recall) with an acceptable level of customer friction (high precision) requires moving beyond standard models toward either more sophisticated architectures or different operational paradigms, such as the top-K screening explored in our Recall@K analysis.

7.5. Implications for Fraud Detection Research

This case study underscores several persistent challenges outlined throughout this review:

1. **Class Imbalance Impact**: The low prevalence of fraud (1.43% in this experiment) makes it extremely difficult for models to learn distinctive fraud patterns without specialized techniques and careful algorithm selection.
2. **Misclassification Cost Asymmetry**: The relatively low recall for fraud detection means a significant proportion of frauds would remain undetected. Conversely, even a low FPR can translate into numerous false alarms, emphasizing the critical need for cost-sensitive learning approaches.
3. **Model Selection and Optimization**: The performance varies significantly between models, with ET showing slight advantages in F1-score and recall for fraud in this specific instance. This highlights the necessity for careful model selection, hyperparameter optimization, and potentially ensemble approaches.
4. **Need for Advanced Methodologies**: The results suggest that basic supervised models, even with feature engineering, may not suffice for robust fraud detection. This points toward the importance of exploring hybrid methods, advanced ensemble techniques, deep learning approaches, or sophisticated unsupervised anomaly detection systems.

This practical application demonstrates that, while machine learning offers powerful analytical tools, achieving effective and operationally viable fraud detection systems requires addressing inherent challenges through sophisticated methodologies, careful evaluation protocols, and approaches specifically tailored to the financial context and business requirements.

8. Case Study 2: Application of Random Forest on a Real-World Bank Transfer Dataset

8.1. Dataset

The initial dataset was extracted from the Intesa Sanpaolo (ISP) database, organized in a table containing 90,314 records of anonymous users' bank transfers. The dataset exhibits strong class imbalance, with only 3285 fraudulent transactions among 90,314 total transactions ($\sim$3.6%). All the data were encrypted and anonymized to ensure privacy and compliance with regulatory requirements. In particular, all the categorical and textual variables were hashed using the SHA-256 algorithm. For computational purposes and to enable the running of machine learning algorithms, the hashed information was further transformed into numerical representations through appropriate encoding strategies.

Each transaction is described by a set of variables that underwent a feature engineering process to derive temporal, spatial, financial, technical, and contextual features. Temporal information includes the hour, day, day of the week, and a weekend indicator, while spatial data capture the latitude and longitude of the transaction origin. Financial attributes comprise the transaction amount, currency code, divisibility flags (by 2, 3, 5, or 10), and decimal patterns (e.g., 0.00, 0.50, and 0.99). The dataset also includes metadata such as the Bank Identifier Code (BIC) country code, bank codes, client type, and mobile carrier information.

Security and authentication indicators are available to characterize the transaction environment, including flags for secure app usage, fingerprint authentication, instant payment, and additional elements such as fingerprint certificate, secure session ID, and digital signature. Furthermore, the IP address associated with each transaction has been decomposed into its four octets and encrypted, ensuring full compliance with privacy standards.

Finally, the dataset contains semantic information derived from the transaction description field (causal field), represented as a fixed-length `Word2Vec` embedding (10 dimensions), and a set of flags describing the access environment (e.g., access mode, device model, operating system, application name and version, user agent, and connection type).

8.2. Experiments

In classification problems, it is common to encounter imbalanced data, where certain classes are significantly underrepresented compared to others. Typical examples include fraud detection, where fraudulent transactions may represent less than 5% of the data, or medical diagnostics, where a disease might only appear in a small fraction of the population. When trained on such data without any corrective measures, a classifier tends to be biased toward the majority class, often resulting in high overall accuracy but poor performance on the minority class. This imbalance severely limits the model's usefulness in real-world scenarios, where correct identification of the minority class is often the primary objective. To address this issue, the Random Forest algorithm offers the `class_weight` parameter, which allows assigning different weights to classes with the aim of penalizing misclassifications of minority classes more heavily. By introducing `class_weight`, the decision criteria used to construct each individual tree (such as Gini impurity or entropy) are modified to account for the relative importance of each class. This adjustment leads to splits that are more sensitive to minority samples, thereby improving the model's ability to identify them. For these reasons, we performed three classification experiments adopting the Random Forest algorithm with different setups of the `class_weight`parameter. In particular, in the first experiment, it was set to "`None`" (the default). In this way, all the classes are treated equally, which is generally inappropriate in the presence of class imbalance. In the second experiment, it was set to "`balanced`". The algorithm automatically adjusts weights inversely proportional to class frequencies in the entire training set, effectively compensating for imbalance and encouraging the model to pay more attention to minority classes. In the third experiment, setting it to "`balanced_subsample`" results in weights being computed in the same way as "`balanced`", but these weights are calculated independently for each bootstrap sample drawn to train an individual tree. This can lead to slightly different weighting across trees, potentially improving robustness when class distributions vary across different subsets of the data.

8.3. Results and Critical Analysis

To evaluate the impact of different `class_weight` settings on model performance, we applied the Random Forest models to a held-out test set. The test split contains 23,310 transactions, with a fraud prevalence of 3.39%. In addition to standard metrics, we report **AUPRC** and **Recall@K**, which are particularly informative in imbalanced classification scenarios. AUPRC is computed from out-of-sample fraud probabilities, while Recall@K measures the fraction of fraud cases captured when screening the top K% of transactions ranked by the model score (K = 0.1% corresponds to $\sim$23 alerts; K = 0.5% corresponds to $\sim$117 alerts). These metrics allow us to assess how well each `class_weight` configuration improves the identification of the minority (fraud) class in a realistic setting.

The results of the three experiments, summarized in Table 5, are strikingly similar and provide clear insight into the limitations of simple algorithmic adjustments for class imbalance. This uniformity is the first and most important finding: simple algorithmic adjustments like using `class_weight` (both `balanced` and `balanced_subsample`) did not lead to any substantial performance improvements. Moreover, despite the high overall accuracy and a relatively high weighted F1-score (both around 0.97), these global metrics

are misleading in the presence of extreme class imbalance and mask the model's critical failure to identify the minority class.

Table 5. Case Study 2—test set results (N = 23,310; prevalence = 3.39%). We report accuracy, F1 (weighted), AUPRC, and Recall@K.

`class_weight`	Accuracy	F1 (w)	AUPRC	Recall@0.1%	Recall@0.5%
`None` (Default)	0.9769	0.9721	0.6969	0.0291	0.1403
`balanced`	0.9753	0.9694	0.6787	0.0291	0.1365
`balanced_subsample`	0.9753	0.9694	0.6787	0.0291	0.1365

Among the three configurations, the model trained with `class_weight = None` achieved the best overall balance between precision and recall. The corresponding confusion matrix on the held-out test set is reported below:

$$\begin{bmatrix} \text{TN} = 22{,}487 & \text{FP} = 32 \\ \text{FN} = 506 & \text{TP} = 285 \end{bmatrix}$$

with $\text{precision}_{\text{fraud}} = 0.90$ and $\text{recall}_{\text{fraud}} = 0.36$. In Figure 2, the PR curve for `class_weight = None` is shown.

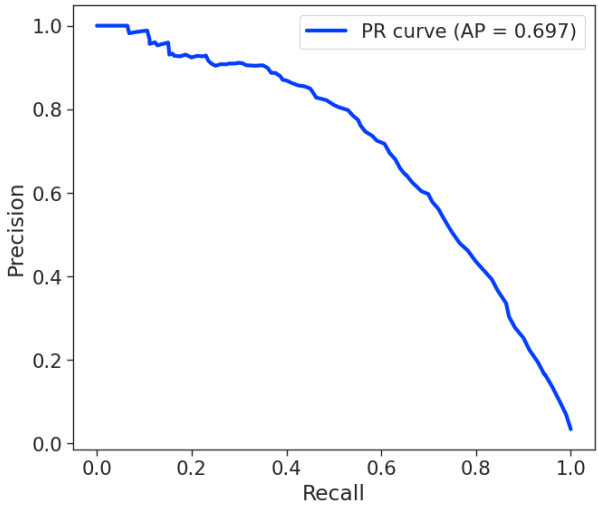

Figure 2. Case Study 2—precision–recall curve for `class_weight = None`. The average precision (AUPRC) is 0.697 on the test set.

Even though the AUPRC is moderately high, suggesting that the model is able to distinguish between classes, the recall at low detection rates is critically low. For example, at a threshold of 0.1%, the model only identifies approximately 2.9% of fraudulent transactions.

From an operational perspective, this is a significant finding: even when the model is explicitly instructed to penalize errors on the minority class more heavily, it still fails to identify the vast majority of fraudulent transactions. This outcome highlights a key limitation of the `class_weight` parameter. While it adjusts the cost function, it does not generate new information or alter the feature space. If the fraudulent transactions are located in dense overlapping regions with legitimate transactions, simply increasing their weight is often insufficient to allow the algorithm to find a clean generalizable separation boundary. The model remains unable to learn a distinct pattern from the few available positive samples.

Therefore, these results strongly suggest that more sophisticated strategies are required. A more promising path would be to explore data-level techniques that directly modify the

dataset, such as oversampling (e.g., SMOTE) to increase the number of minority samples or undersampling to reduce the majority class. Another powerful alternative, supported by the literature, lies in using advanced ensemble methods specifically designed for imbalanced data or hybrid approaches that can isolate anomalous patterns before a final classification occurs [12,85].

In conclusion, this case study serves as a practical demonstration that tackling severe class imbalance requires more than a single parameter tweak. It underscores the necessity of employing multifaceted strategies that either reshape the data or use algorithms inherently designed to handle such challenging conditions.

9. Challenges and Future Directions

Despite significant advances in ML-based fraud detection, several persistent challenges continue to limit system effectiveness, while new opportunities are emerging in this rapidly evolving field.

9.1. Current Challenges

9.1.1. Data Imbalance and Feature Engineering

The extreme class imbalance in fraud datasets remains a fundamental and persistent challenge. Fraudulent transactions typically constitute less than 1% of all transactions, making it exceptionally difficult for models to learn discriminative features for the minority class without specialized techniques [4,5]. Compounding this issue is the complexity of feature engineering, which is crucial for success but requires substantial domain expertise and is often labor-intensive [123].

Furthermore, drawing inspiration from advances in computer vision, such as methods for learning robust feature representations in complex scenes for accurate human parsing [124], could offer valuable pathways to improve feature extraction and model generalization in the intricate domain of financial transactions.

9.1.2. Concept Drift and Real-Time Requirements

Fraud patterns evolve continuously, leading to concept drift, where a model's performance degrades as the relationship between features and fraud likelihood changes. Developing models that can adapt to these evolving patterns without requiring frequent retraining remains a significant challenge [7]. This is further complicated by modern financial systems that require real-time detection, imposing strict constraints on model complexity and computational latency.

Moreover, the adversarial nature of fraud detection shares similarities with challenges in information security. Techniques developed for robust watermarking against geometric attacks [125], which focus on resisting complex interference, may provide useful insights for building fraud detection systems resilient to dynamic environments and sophisticated adversarial attempts.

9.1.3. The "Black-Box" Problem and the Need for Explainable AI (XAI)

As fraud detection models become increasingly complex, their interpretability decreases, creating a "black box" that is problematic in regulatory environments. Financial regulations, such as the GDPR's "right to explanation," demand that institutions be able to justify automated decisions. This makes explainability not just a desirable feature but a critical compliance requirement [8,9].

Explainable AI (XAI) provides a pathway to address this challenge. Techniques like SHAPs (SHapley Additive exPlanations) and LIMEs (Local Interpretable Model-Agnostic Explanations) are gaining significant traction for providing transparency [7]. For the models

in our case studies, these methods could be applied to provide crucial insights. For instance, for a transaction flagged by our Random Forest model, SHAP values could reveal precisely which features—such as an unusually high transaction amount (num_impor), an atypical time of day (num_hour), or a high-risk country code—were the primary drivers behind the fraud classification. This transforms an opaque alert into an actionable and auditable insight for fraud analysts.

9.1.4. Data Privacy, Security, and the Promise of Federated Learning

Fraud detection systems require access to vast amounts of sensitive financial data, raising significant privacy and regulatory concerns [20]. Centralizing customer data from multiple sources creates a high-value target for cyberattacks and complicates compliance with data protection laws. Federated learning emerges as a powerful privacy-preserving paradigm to overcome this obstacle [126].

Instead of pooling raw data, federated learning allows multiple institutions to collaboratively train a shared model without ever exposing their confidential data. Each institution trains the model locally on its own dataset, and only the resulting model updates (anonymous parameters or gradients) are sent to a central server for aggregation. This approach allows a global model to learn from a much wider and more diverse set of fraud patterns—improving its accuracy and robustness—while ensuring that sensitive customer information never leaves the secure perimeter of each participating institution.

9.1.5. Algorithmic Bias and Ethical Implications

A significant, and often overlooked, challenge is the risk of embedding societal biases into automated fraud detection systems. If historical data used for training contains biases, an ML model will learn and amplify them. For example, a model might unfairly associate a higher risk of fraud with transactions originating from low-income neighborhoods or with specific demographic groups simply because of historical data imbalances or prejudiced policing practices reflected in the data. This can lead to discriminatory outcomes, where certain groups of legitimate customers are subjected to higher rates of declined transactions or account blockages.

The ethical implications of such biases are profound, posing significant reputational and legal risks to financial institutions [9]. Addressing this requires a dedicated focus on fairness throughout the model lifecycle. This includes conducting bias audits on datasets, employing fairness-aware machine learning algorithms, and regularly monitoring model predictions to ensure equitable treatment across different user groups. The pursuit of accuracy must be balanced with a commitment to fairness and ethical responsibility.

9.2. Emerging Trends and Opportunities

9.2.1. Deep Learning Advances

Deep learning approaches, particularly those designed for sequential data analysis (e.g., LSTM and GRU), show considerable promise for capturing complex temporal patterns in transaction sequences and behavioral data [76,90].

9.2.2. Graph-Based Methods

Graph-based approaches that model complex relationships between entities (customers, merchants, transactions, and accounts) are emerging as effective tools for detecting sophisticated fraud schemes involving multiple participants and complex network structures [127].

These methods excel at uncovering coordinated fraudulent activities, such as money laundering rings or collusion networks, that are difficult to detect with transaction-level

models. However, their computational complexity and the need for graph-specific data structures can be barriers to implementation.

9.2.3. Federated Learning

Federated learning approaches allow models to be trained across multiple institutions without sharing raw sensitive data, addressing privacy concerns while leveraging broader data sources and collective intelligence [126].

9.2.4. Explainable AI Integration

Techniques for explaining model decisions are becoming increasingly important in fraud detection applications, with methods like SHAPs (SHapley Additive exPlanations) and LIMEs (Local Interpretable Model-Agnostic Explanations) gaining significant traction for regulatory compliance and operational transparency [7].

Grad-CAM represents a valuable approach for model debugging, using gradients as importance measures in feature space. It does not require specific CNN architectures and can be applied to any gradient-based neural network model. SHAP values, derived from cooperative game theory, assess feature contributions by analyzing prediction changes when features are removed from feature sets.

9.2.5. Synthetic Data Generation

Advanced techniques for generating realistic synthetic fraud data, such as GANs and variational autoencoders, are being explored to address data scarcity and privacy concerns while providing diverse training scenarios [84,107].

9.2.6. Hybrid and Ensemble Approaches

Sophisticated combinations of multiple ML techniques, leveraging the complementary strengths of different approaches, show significant promise for improving detection performance across diverse fraud scenarios and operational contexts [81,128].

9.2.7. Transfer Learning Applications

Transfer learning approaches that can leverage knowledge gained from one fraud detection task to improve performance on related tasks with limited data are emerging as valuable techniques for addressing data scarcity in specific domains [6].

9.3. Future Research Directions

Based on our comprehensive analysis, several promising directions for future research emerge.

9.3.1. Advanced Unsupervised Techniques

Further exploration of sophisticated unsupervised and semi-supervised techniques is needed, particularly for detecting novel fraud patterns not represented in historical labeled data and for addressing the challenge of limited labeled fraud examples [35,108].

9.3.2. Real-Time Adaptive Systems

Development of fraud detection systems that can adapt to evolving fraud patterns in real time, without requiring periodic retraining or manual intervention, would effectively address the persistent challenge of concept drift [7].

9.3.3. Comprehensive Evaluation Frameworks

Standardized frameworks for evaluating fraud detection systems that consider multiple performance dimensions (accuracy, interpretability, computational efficiency, adapt-

ability, and regulatory compliance) would facilitate meaningful comparisons between approaches and accelerate research progress [117].

9.3.4. Domain-Specific Approaches

Tailored approaches for specific fraud types (e.g., money laundering, insurance fraud, and tax evasion) that incorporate domain-specific knowledge, regulatory constraints, and operational requirements could significantly improve detection effectiveness [28,32].

9.3.5. Cross-Organizational Collaboration

Frameworks for collaborative fraud detection across organizations that preserve data privacy while leveraging broader pattern recognition capabilities could enhance detection of sophisticated fraud schemes that span multiple institutions [112].

9.3.6. Behavioral Analysis Integration

Integration of advanced behavioral analytics that model normal user behavior patterns and detect subtle deviations could improve fraud detection accuracy while reducing false positives and enhancing user experience [36].

9.3.7. Ethical Considerations

Research on the ethical implications of automated fraud detection, including fairness, bias mitigation, transparency, and accountability, is increasingly important as these systems become more widespread and influential [9].

10. Conclusions

This comprehensive review has presented a systematic analysis of machine learning techniques for financial fraud detection, examining the current state of the research, practical applications, and emerging trends. Our analysis confirms the dominance of supervised learning methods in the literature, but it also highlights the persistent fundamental challenges that limit their effectiveness in real-world operational environments.

The unique contribution of this paper lies in bridging theoretical knowledge with a critical assessment of practical application. We achieve this through two case studies on proprietary real-world banking data, which move the discussion from abstract performance to concrete operational trade-offs. Crucially, our experiments reveal a persistent and critical limitation: even with hyperparameter tuning and class weight adjustments, the models struggle to achieve adequate recall for the minority (fraud) class. This finding is not merely a technical result; it is a strategic insight demonstrating that standard supervised models, when used in isolation, are often insufficient to overcome the extreme class imbalance inherent in financial fraud data.

This identified gap directly informs a more strategic vision for future research. The low recall obtained highlights the need to explore hybrid architectures with resampling or cost-sensitive models. Future work should focus on hybrid models that combine unsupervised anomaly detection for identifying novel threats with supervised classifiers trained on enriched datasets using advanced resampling (e.g., SMOTE variations) or generative (e.g., GANs) techniques. Furthermore, to meet the regulatory and operational demands for transparency, the integration of Explainable AI (XAI) is no longer optional. Techniques like SHAPs and LIMEs must be embedded into the development lifecycle to transition from "black-box" predictors to decision-support systems that provide clear actionable rationales for fraud alerts.

As financial systems become increasingly interconnected, the path forward requires a shift towards more adaptive, interpretable, and collaborative systems. Research should prioritize real-time adaptive models that are capable of countering concept drift, and privacy-

preserving frameworks like federated learning to enable cross-institutional collaboration without centralizing sensitive data. By focusing on these strategic directions—inspired directly by the practical limitations observed—the research community can develop more robust and effective defenses against the persistent and evolving threat of financial fraud.

Success in this endeavor will require close collaboration between researchers, practitioners, financial institutions, and regulatory bodies to ensure that technological advances translate into practical, transparent, and ethically sound solutions that protect the integrity of the global financial ecosystem.

Author Contributions: Conceptualization, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; methodology, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; software, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; validation, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; formal analysis, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; investigation, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; data curation, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; writing—original draft preparation, A.P., A.A.C., Y.M., S.C., G.R. and R.C.; writing—review and editing, A.P., A.A.C., and A.T.; visualization, A.P. and A.A.C.; supervision, A.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: All analyses were conducted on anonymized records under institutional agreements with strict access controls and audit trails. Only features necessary for fraud detection were processed (data minimization), and results were reported at aggregate level without any attempt at re-identification.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in Case Study 1 and Case Study 2 consist of proprietary bank transactions. The data presented in this study are available on request from the corresponding author. The data are not publicly available due to legal and contractual restrictions.

Acknowledgments: The authors acknowledge supercomputing resources and support from ICSC—Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing—and hosting entity, funded by European Union—NextGenerationEU. The views and opinions expressed are those of the authors and do not necessarily reflect the views of Intesa Sanpaolo, its affiliates, or its employees.

Conflicts of Interest: Author Riccardo Crupi was employed by the company Intesa Sanpaolo S.p.A. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. PricewaterhouseCoopers. *Encuesta Global de Crimen y Fraude Económico de PwC Colombia 2022–2023*; PricewaterhouseCoopers: London, UK, 2022.
2. Reurink, A. Financial fraud: A literature review. *J. Econ. Surv.* **2018**, 32, 1292–1325. [CrossRef]
3. Ahmed, M.; Mahmood, A.N.; Islam, M.R. A survey of anomaly detection techniques in financial domain. *Future Gener. Comput. Syst.* **2016**, *55*, 278–288. [CrossRef]
4. Femila, Roseline, J.F.; Naidu, G.; Samuthira Pandi, V.; Alamelu alias Rajasree, S.; Mageswari, D.N. Autonomous credit card fraud detection using machine learning approach. *Comput. Electr. Eng.* **2022**, *102*, 108132. [CrossRef]
5. Tingfei, H.; Guangquan, C.; Kuihua, H. Using variational auto encoding in credit card fraud detection. *IEEE Access* **2020**, *8*, 149841–149853. [CrossRef]
6. Dantas, R.M.; Firdaus, R.; Jaleel, F.; Neves Mata, P.; Mata, M.N.; Li, G. Systemic acquired critique of credit card deception exposure through machine learning. *J. Open Innov. Technol. Mark. Complex.* **2022**, *8*, 192. [CrossRef]
7. Dal Pozzolo, A.; Caelen, O.; Le Borgne, Y.A.; Waterschoot, S.; Bontempi, G. Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Syst. Appl.* **2022**, *41*, 4915–4928. [CrossRef]
8. Makki, S.; Assaghir, Z.; Taher, Y.; Haque, R.; Hacid, M.S.; Zeineddine, H. An experimental study with imbalanced classification approaches for credit card fraud detection. *IEEE Access* **2019**, *7*, 93010–93022. [CrossRef]

9. Rakowski, R.; Polak, P.; Kowalikova, P. Ethical aspects of the impact of AI: The status of humans in the era of artificial intelligence. *Society* **2021**, *58*, 196–203. [CrossRef]
10. Alarfaj, F.K.; Malik, I.; Khan, H.U.; Almusallam, N.; Ramzan, M.; Ahmed, M. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access* **2022**, *10*, 39700–39715. [CrossRef]
11. Madhurya, M.J.; Gururaj, H.L.; Soundarya, B.C.; Vidyashree, K.P.; Rajendra, A.B. Exploratory analysis of credit card fraud detection using machine learning techniques. *Glob. Transitions Proc.* **2022**, *3*, 31–37. [CrossRef]
12. Sahin, Y.; Bulkan, S.; Duman, E. A cost-sensitive decision tree approach for fraud detection. *Expert Syst. Appl.* **2013**, *40*, 5916–5923. [CrossRef]
13. Vanini, P.; Rossi, S.; Zvizdic, E.; Domenig, T. Online payment fraud: From anomaly detection to risk management. *Financ. Innov.* **2023**, *9*, 66.
14. Srokosz, M.; Bobyk, A.; Ksiezopolski, B.; Wydra, M. Machine-learning-based scoring system for antifraud CISIRTs in banking environment. *Electronics* **2023**, *12*, 251. [CrossRef]
15. Zhou, H.; Chai, H.; Qiu, M. Fraud detection within bankcard enrollment on mobile device based payment using machine learning. *Front. Inf. Technol. Electron. Eng.* **2018**, *19*, 1537–1545. [CrossRef]
16. Xiong, T.; Ma, Z.; Li, Z.; Dai, J. The analysis of influence mechanism for internet financial fraud identification and user behavior based on machine learning approaches. *Int. J. Syst. Assur. Eng. Manag.* **2022**, *13*, 996–1007. [CrossRef]
17. Chen, Y.; Wu, Z. Financial fraud detection of listed companies in China: A machine learning approach. *Sustainability* **2022**, *15*, 105. [CrossRef]
18. Kumar, S.; Ahmed, R.; Bharany, S.; Shuaib, M.; Ahmad, T.; Tag Eldin, E.; Rehman, A.U.; Shafiq, M. Exploitation of machine learning algorithms for detecting financial crimes based on customers' behavior. *Sustainability* **2022**, *14*, 13875.
19. Sathya, M.; Balakumar, B. Insurance fraud detection using novel machine learning technique. *Int. J. Intell. Syst. Appl. Eng.* **2022**, *10*, 374–381.
20. Van Capelleveen, G.; Poel, M.; Mueller, R.M.; Thornton, D.; Van Hillegersberg, J. Outlier detection in healthcare fraud: A case study in the Medicaid dental domain. *Int. J. Account. Inf. Syst.* **2016**, *21*, 18–31. [CrossRef]
21. Aslam, F.; Hunjra, A.; Ftiti, Z.; Louhichi, W.; Shams, T. Insurance fraud detection: Evidence from artificial intelligence and machine learning. *Res. Int. Bus. Financ.* **2022**, *62*, 101744.
22. Nian, K.; Zhang, H.; Tayal, A.; Coleman, T.; Li, Y. Auto insurance fraud detection using unsupervised spectral ranking for anomaly. *J. Financ. Data Sci.* **2016**, *2*, 58–75. [CrossRef]
23. Subudhi, S.; Panigrahi, S. Use of optimized fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection. *J. King Saud Univ.-Comput. Inf. Sci.* **2020**, *32*, 568–575. [CrossRef]
24. Chen, S. Detection of fraudulent financial statements using the hybrid data mining approach. *SpringerPlus* **2016**, *5*, 89. [CrossRef] [PubMed]
25. Dutta, I.; Dutta, S.; Raahemi, B. Detecting financial restatements using data mining techniques. *Expert Syst. Appl.* **2017**, *90*, 374–393. [CrossRef]
26. Hajek, P.; Henriques, R. Mining corporate annual reports for intelligent detection of financial statement fraud—A comprehensive survey of machine learning methods. *Knowl.-Based Syst.* **2017**, *128*, 139–152. [CrossRef]
27. Lokanan, M.; Tran, V.; Vuong, N.H. Detecting anomalies in financial statements using machine learning algorithm. *Asian J. Account. Res.* **2019**, *4*, 181–201. [CrossRef]
28. Alsuwailem, A.A.S.; Salem, E.; Saudagar, A.K.J. Performance of different machine learning algorithms in detecting financial fraud. *Comput. Econ.* **2022**, *62*, 1631–1667. [CrossRef]
29. Lokanan, M.E. Predicting money laundering using machine learning and artificial neural networks algorithms in banks. *J. Appl. Secur. Res.* **2022**, *19*, 20–24. [CrossRef]
30. Rocha-Salazar, J.J.; Segovia-Vargas, M.J.; Camacho-Miñano, M.M. Money laundering and terrorism financing detection using neural networks and an abnormality indicator. *Expert Syst. Appl.* **2021**, *169*, 114470. [CrossRef]
31. Baghdasaryan, V.; Davtyan, H.; Sarikyan, A.; Navasardyan, Z. Improving tax audit efficiency using machine learning: The role of taxpayer's network data in fraud detection. *Appl. Artif. Intell.* **2022**, *36*, 2012002. [CrossRef]
32. Savić, M.; Atanasijević, J.; Jakovetić, D.; Krejić, N. Tax evasion risk management using a hybrid unsupervised outlier detection method. *Expert Syst. Appl.* **2022**, *193*, 116409. [CrossRef]
33. Vanhoeyveld, J.; Martens, D.; Peeters, B. Value-added tax fraud detection with scalable anomaly detection techniques. *Appl. Soft Comput.* **2020**, *86*, 105895. [CrossRef]
34. Arévalo, F.; Barucca, P.; Téllez-León, I.E.; Rodríguez, W.; Gage, G.; Morales, R. Identifying clusters of anomalous payments in the salvadorian payment system. *Lat. Am. J. Cent. Bank.* **2022**, *3*, 100050. [CrossRef]
35. Rubio, J.; Barucca, P.; Gage, G.; Arroyo, J.; Morales-Resendiz, R. Classifying payment patterns with artificial neural networks: An autoencoder approach. *Lat. Am. J. Cent. Bank.* **2020**, *1*, 100013. [CrossRef]

36. Hamza, C.; Lylia, A.; Nadine, C.; Nicolas, C. Semi-supervised method to detect fraudulent transactions and identify fraud types while minimizing mounting costs. *Int. J. Adv. Comput. Sci. Appl.* **2023**, *14*, 861–870. [CrossRef]
37. UK Finance. Fraud-The Facts 2021, The Definitive Overview of Payment Industry Fraud. 2021. Available online: https://www.ukfinance.org.uk (accessed on 20 October 2025).
38. Langan, T. *Internet Crime Report 2023*; Tech. rep.; Federal Bureau of Investigation: Washington, DC, USA, 2023.
39. Amusan, E.; Alade, O.; Fenwa, O.D.; Emuoyibofarhe, J.O. Credit card fraud detection on skewed data using machine learning techniques. *LAUTECH J. Comput. Inform.* **2021**, *2*, 49–56.
40. Ahirwar, A.; Sharma, N.; Bano, A. Enhanced SMOTE & fast random forest techniques for credit card fraud detection. *Solid State Technol.* **2020**, *63*, 4721–4733.
41. Jonnalagadda, V.; Gupta, P.; Sen, E. Credit card fraud detection using random forest algorithm. *Int. J. Adv. Res. Ideas Innov. Technol.* **2019**, *5*, 1–5.
42. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
43. Pagliaro, A. Forecasting Significant Stock Market Price Changes Using Machine Learning: Extra Trees Classifier Leads. *Electronics* **2023**, *12*, 4551. [CrossRef]
44. Itoo, F.; Singh, S. Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. *Int. J. Inf. Technol.* **2021**, *13*, 1503–1511. [CrossRef]
45. Adityasundar, N.; SaiAbhigna, T.; Lakshman, B.; Phaneendra, D.; MohanKumar, N. Credit card fraud detection using machine learning classification algorithms over highly imbalanced data. *J. Sci. Technol.* **2020**, *5*, 138–146. [CrossRef]
46. Soh, W.W.; Yusuf, R.M. Predicting credit card fraud on a imbalanced data. *Int. J. Data Sci. Adv. Anal.* **2019**, *1*, 12–17. [CrossRef]
47. Li, C.; Ding, N.; Zhai, Y.; Dong, H. Comparative study on credit card fraud detection based on different support vector machines. *Intell. Data Anal.* **2021**, *25*, 105–119. [CrossRef]
48. Zhang, D.; Bhandari, B.; Black, D. Credit card fraud detection using weighted support vector machine. *Appl. Math.* **2020**, *11*, 1275–1291. [CrossRef]
49. Pavithra, T.; Thangadurai, K. The improving perdition of credit card fraud detection on PSO optimized SVM. *Int. J. Anal. Exp. Modal Anal.* **2019**, *XI*, 478–485.
50. Bandyopadhyay, S.; Thakkar, V.; Mukherjee, U.; Dutta, S. Emerging approach for detection of financial frauds using machine learning. *Preprints* **2021**. https//doi.org/10.20944/preprints202108.0028.v1.
51. Hammed, M.; Soyemi, J. An implementation of decision tree algorithm augmented with regression analysis for fraud detection in credit card. *Int. J. Comput. Sci. Inf. Secur.* **2020**, *18*, 79–88.
52. Barahim, A.; Alhajri, A.; Alasaibia, N.; Altamimi, N.; Aslam, N.; Khan, I.U. Enhancing the credit card fraud detection through ensemble techniques. *J. Comput. Theor. Nanosci.* **2019**, *16*, 4461–4468. [CrossRef]
53. Borse, D.D.; Patil, S.H.; Dhotre, S. Credit card fraud detection using naive Bayes and robust scaling techniques. *Int. J.* **2021**, *10*, 1–5.
54. Gupta, A.; Lohani, M.C.; Manchanda, M. Financial fraud detection using naive Bayes algorithm in highly imbalance data set. *J. Discret. Math. Sci. Cryptogr.* **2021**, *24*, 1559–1572. [CrossRef]
55. Chowdari, G.B. Supervised machine learning algorithms for detecting credit card fraud. *Epra Int. J. Res. Dev.* **2021**, *6*, 131–134. . [CrossRef]
56. Kumar, R.; Student, P.G.; Budihul, R. An efficient approach for credit card fraud detection. Int. *J. Innov. Sci. Res. Technol.* **2020**, *5*, 1057–1073.
57. Manlangit, S.; Azam, S.; Shanmugam, B. Novel machine learning approach for analyzing anonymous credit card fraud patterns. *Int. J. Electron. Commer. Stud.* **2019**, *10*, 175–202. [CrossRef]
58. Agarwal, V. Identity theft detection using machine learning. *Int. J. Res. Appl. Sci. Eng. Technol.* **2021**, *9*, 1943–1946. [CrossRef]
59. Asha, R.B.; KR, S.K. Credit card fraud detection using artificial neural network. *Glob. Transitions Proc.* **2021**, *2*, 35–41.
60. Oumar, A.W.; Augustin, P. Credit card fraud detection using ANN. *Int. J. Comput. Sci. Mob. Comput.* **2019**, *8*, 257–260.
61. Niu, X.; Wang, L.; Yang, X. A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv* **2019**, arXiv:1904.10604. [CrossRef]
62. Faraj, A.A.; Mahmud, D.A.; Rashid, B.N. Comparison of different ensemble methods in credit card default prediction. *UHD J. Sci. Technol.* **2021**, *5*, 20–25. [CrossRef]
63. Dalal, S.; Seth, B.; Radulescu, M.; Secara, C.; Tolea, C. Predicting fraud in financial payment services through optimized hyper-parameter-tuned XGBoost model. *Mathematics* **2022**, *10*, 4679. [CrossRef]
64. Meenu; Gupta, S.; Patel, S.; Kumar, S.; Chauhan, G. Anomaly detection in credit card transactions using machine learning. *Int. J. Innov. Res. Comput. Sci. Technol.* **2020**, *8*, 2347–5552.
65. Vijayakumar, V.; Divya, N.S.; Sarojini, P.; Sonika, K. Isolation forest and local outlier factor for credit card fraud detection system. *Int. J. Eng. Adv. Technol.* **2020**, *9*, 261–265. [CrossRef]

66. Palekar, V.; Kharade, S.; Zade, H.; Ali, S.; Kamble, K.; Ambatkar, S. Credit card fraud detection using isolation forest. *Int. Res. J. Eng. Technol.* **2020**, *7*, 1–6.
67. Misra, S.; Thakur, S.; Ghosh, M.; Saha, S.K. An autoencoder based model for detecting fraudulent credit card transaction. *Procedia Comput. Sci.* **2020**, *167*, 254–262. [CrossRef]
68. Wu, E.; Cui, H.; Welsch, R.E. Dual autoencoders generative adversarial network for imbalanced classification problem. *IEEE Access* **2020**, *8*, 91265–91275. [CrossRef]
69. Abdulsalami, B.A.; Kolawole, A.A.; Ogunrinde, M.A.; Lawal, M.; Azeez, R.A.; Afolabi, A.Z. Comparative analysis of back-propagation neural network and K-means clustering algorithm in fraud detection in online credit card transactions. *Fountain J. Nat. Appl. Sci.* **2019**, *8*, 315. [CrossRef]
70. Deb, K.; Ghosal, S.; Bose, D. A comparative study on credit card fraud detection. *EngrXiv* **2021**. [CrossRef]
71. Singh, P. and Singh Umrao, L. Credit card fraud detection system using data mining. *Open Nano Res. J.* **2024**, *20*, 4. [CrossRef]
72. Singh, M.; Kumar, S.; Kumar, S.; Garg, T. Credit card fraud detection using hidden Markov model. *Int. J. Eng. Comput. Sci.* **2019**, *8*, 24878–24882. [CrossRef]
73. Danaa, A.A.A.; Daabo, M.I.; Abdul-Barik, A. Detecting electronic banking fraud on highly imbalanced data using hidden Markov models. *Earthline J. Math. Sci.* **2021**, *7*, 315–332. [CrossRef]
74. Sugidamayatno, S.; Lelono, D. Outlier detection credit card transactions using local outlier factor algorithm (LOF). *Indones. J. Comput. Cybern. Syst.* **2019**, *13*, 409–420. [CrossRef]
75. Alghofaili, Y.; Albattah, A.; Rassam, M.A. A financial fraud detection model based on LSTM deep learning technique. *J. Appl. Secur. Res.* **2020**, *15*, 498–516. [CrossRef]
76. Benchaji, I.; Douzi, S.; El Ouahidi, B. Credit card fraud detection model based on LSTM recurrent neural networks. *J. Adv. Inf. Technol.* **2021**, *12*, 113–118. [CrossRef]
77. Cheon, M.J.; Lee, D.H.; Joo, H.S.; Lee, O. Deep learning based hybrid approach of detecting fraudulent transactions. *J. Theor. Appl. Inf. Technol.* **2021**, *99*, 4044–4054.
78. Aswathy, M.; Samuel, L. Credit card fraud detection using hybrid models. *Int. Res. J. Eng. Technol.* **2019**, *6*, 2019.
79. Chen, J.I.Z.; Lai, K.L. Deep convolution neural network model for credit-card fraud detection and alert. *J. Artif. Intell.* **2021**, *3*, 101–112.
80. Bandyopadhyay, S.; Dutta, S. Detection of fraud transactions using recurrent neural network during COVID-19: Fraud transaction during COVID-19. *J. Adv. Res. Med. Sci. Technol.* **2020**, *7*, 16–21. [CrossRef]
81. Forough, J.; Momtazi, S. Ensemble of deep sequential models for credit card fraud detection. *Appl. Soft Comput.* **2021**, *99*, 106883. [CrossRef]
82. Sadgali, I.; Sael, N.; Benabbou, F. Bidirectional gated recurrent unit for improving classification in credit card fraud detection. *Indones. J. Electr. Eng. Comput. Sci.* **2021**, *21*, 1704–1712. [CrossRef]
83. Ba, H. Improving detection of credit card fraudulent transactions using generative adversarial networks. *arXiv* **2019**, arXiv:1907.03355. [CrossRef]
84. Fiore, U.; De Santis, A.; Perla, F.; Zanetti, P.; Palmieri, F. Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Inf. Sci.* **2019**, *479*, 448–455. [CrossRef]
85. Dzakiyullah, N.R.; Pramuntadi, A.; Fauziyyah, A.K. Semi-supervised classification on credit card fraud detection using autoencoders. *J. Appl. Data Sci.* **2021**, *2*, 1–7. [CrossRef]
86. Pratap, B.G.; Vijayaraghavulu, P. A hybrid method for credit card fraud detection using machine learning algorithm. *Int. J. Comput. Electr. Adv. Commun. Eng.* **2021**, *10*, 46–50.
87. Husejinovic, A. Credit card fraud detection using naive Bayesian and c4.5 decision tree classifiers. *Period. Eng. Nat. Sci.* **2020**, *4*, 1–5. [CrossRef]
88. Lin, T.H.; Jiang, J.R. Credit card fraud detection with autoencoder and probabilistic random forest. *Mathematics* **2021**, *9*, 2683. [CrossRef]
89. Karthik, R.; Navinkumar, R.; Rammkumar, U.; Mothilal, K.C. Supervised machine learning algorithms for credit card fraudulent transaction detection. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2019**, *2019*, 2456–3307.
90. Esenogho, E.; Mienye, I.D.; Swart, T.G.; Aruleba, K.; Obaido, G. A neural network ensemble with feature engineering for improved credit card fraud detection. *IEEE Access* **2022**, *10*, 16400–16407. [CrossRef]
91. Baker, M.R.; Mahmood, Z.N.; Shaker, E.H. Ensemble learning with supervised machine learning models to predict credit card fraud transactions. *Rev. Intell. Artif.* **2022**, *36*, 509–518. [CrossRef]
92. Fanai, H.; Abbasimehr, H. A novel combined approach based on deep autoencoder and deep classifiers for credit card fraud detection. *Expert Syst. Appl.* **2023**, *217*, 119562. [CrossRef]
93. Machine Learning Group—ULB. *Credit Card Fraud Detection (Dataset)*; Kaggle: San Francisco, CA, USA, 2013.
94. Hofmann, H. *Statlog (German Credit Data)*; UCI Machine Learning Repository: Berkeley, CA, USA, 1994.

95. Lee, H.; Choi, E.; Kim, I.; Choi, D.; Go, W.; Lee, K.; Yim, H.; Lee, T. Feature selection practice for unsupervised learning of credit card fraud detection. *J. Theor. Appl. Inf. Technol.* **2018**, *96*, 408–417.
96. Pumsirirat, A.; Yan, L. Credit card fraud detection using deep learning based on auto-encoder and restricted Boltzmann machine. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 18–25. [CrossRef]
97. Quinlan, R. *Statlog (Australian Credit Approval)*; UCI Machine Learning Repository: Berkeley, CA, USA, 1997.
98. Seera, M.; Lim, C.P.; Kumar, A.; Dhamotharan, L.; Tan, K.H. An intelligent payment card fraud detection system. *Ann. Oper. Res.* **2021**, *334*, 445–467. [CrossRef]
99. Yeh, I.C. *Default of Credit Card Clients*; UCI Machine Learning Repository: Berkeley, CA, USA, 2016.
100. *China Stock Market & Accounting Research (CSMAR)*; Wharton University of Pennsylvania: Philadelphia, PA, USA, 2022.
101. Achakzai, M.A.K.; Juan, P. Using machine learning meta-classifiers to detect financial frauds. *Financ. Res. Lett.* **2022**, *48*, 102915. [CrossRef]
102. Shou, M.; Bao, X.; Yu, J. An optimal weighted machine learning model for detecting financial fraud. *Appl. Econ. Lett.* **2023**, *30*, 410–415. [CrossRef]
103. Compustat. *S&P Global Market Intelligence*; Compustat: Centennial, CO, USA, 2022.
104. Whiting, D.G.; Hansen, J.V.; McDonald, J.B.; Albrecht, C.; Albrecht, W.S. Machine learning methods for detecting patterns of management fraud. *Comput. Intell.* **2012**, *28*, 505–527. [CrossRef]
105. López-Rojas, E. *Synthetic Financial Datasets for Fraud Detection*; Kaggle: San Francisco, CA, USA, 2017.
106. Alwadain, A.; Ali, R.F.; Muneer, A. Estimating financial fraud through transaction-level features and machine learning. *Mathematics* **2023**, *11*, 1184. [CrossRef]
107. Hwang, J.; Kim, K. An efficient domain-adaptation method using GAN for fraud detection. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 94–103. [CrossRef]
108. Moreira, M.Â.L.; de Souza Rocha Junior, C.; de Lima Silva, D.F.; de Castro Junior, M.A.P.; de Araújo Costa, I.P.; Gomes, C.F.S.; dos Santos, M. Exploratory analysis and implementation of machine learning techniques for predictive assessment of fraud in banking systems. *Procedia Comput. Sci.* **2022**, *214*, 117–124. [CrossRef]
109. Putten, P. *Insurance Company Benchmark (COIL 2000)*; UCI Machine Learning Repository: Berkeley, CA, USA, 2000.
110. Huang, D.; Mu, D.; Yang, L.; Cai, X. CoDetect: Financial fraud detection with anomaly feature detection. *IEEE Access* **2018**, *6*, 19161–19174. [CrossRef]
111. Omershafiq. *Bitcoin Network Transactional Metadata*; Kaggle: San Francisco, CA, USA, 2019.
112. Ashfaq, T.; Khalid, R.; Yahaya, A.; Aslam, S.; Alsafari, S.; Hameed, I. A machine learning and blockchain bases efficient fraud detection mechanism. *Sensors* **2022**, *22*, 7162. [CrossRef]
113. Almhaithawi, D.; Jafar, A.; Aljnidi, M. Example-dependent cost-sensitive credit cards fraud detection using SMOTE and Bayes minimum risk. *SN Appl. Sci.* **2020**, *2*, 1–12.
114. Ramírez-Alpízar, A.; Jenkins, M.; Martínez, A.; Quesada-López, C. Use of data mining and machine learning techniques for fraud detection in financial statements: A systematic mapping study. *Rev. Iber. Sist. Tecnol. Inform.* **2020**, *2020*, 97–109.
115. Bakumenko, A.; Elragal, A. Detecting anomalies in financial data using machine learning algorithms. *Systems* **2022**, *10*, 130. [CrossRef]
116. Zhao, Z.; Bai, T. Financial fraud detection and prediction in listed companies using SMOTE and machine learning algorithms. *Entropy* **2022**, *24*, 1157. [CrossRef] [PubMed]
117. Shahana, T.; Lavanya, V.; Bhat, A.R. State of the art in financial statement fraud detection: A systematic review. *Technol. Forecast. Soc. Change* **2023**, *192*, 122527. [CrossRef]
118. Viera, J.; Aguilar, J.; Rodríguez-Moreno, M.; Quintero-Gull, C. Analysis of the behavior pattern of energy consumption through online clustering techniques. *Energies* **2023**, *16*, 1649. [CrossRef]
119. Nicholls, J.; Kuppa, A.; Le-Khac, N.A. Financial cybercrime: A comprehensive survey of deep learning approaches to tackle the evolving financial crime landscape. *IEEE Access* **2021**, *9*, 163965–163986. [CrossRef]
120. Andrade-Arenas, L.; Yactayo-Arias, C. Comparative analysis of machine learning models for credit card fraud detection using SMOTE for class imbalance. *Int. J. Saf. Secur. Eng.* **2025**, *15*, 893–901. [CrossRef]
121. Saito, T.; Rehmsmeier, M. The precision–recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE* **2015**, *10*, e0118432. [CrossRef]
122. Dal Pozzolo, A.; Caelen, O.; Johnson, R.A.; Bontempi, G. Calibrating probability with undersampling for unbalanced classification. In Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence (SSCI), Cape Town, South Africa, 7–10 December 2015.
123. Lucas, Y.; Portier, P.E.; Laporte, L.; He-Guelton, L.; Caelen, O.; Granitzer, M.; Calabretto, S. Towards automated feature engineering for credit card fraud detection using multi-perspective HMMs. *Future Gener. Comput. Syst.* **2020**, *102*, 393–402. [CrossRef]

124. Li, W.; Zhao, R.; Wang, X. From simple to complex scenes: Learning robust feature representations for accurate human parsing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, Boston, MA, USA, 7–12 June 2015; pp. 26–34.
125. Wang, C.; Zhang, J.; Gao, Z.; Wang, Z.; Liu, N. Light-field image multiple reversible robust watermarking against geometric attacks. *IEEE Access* **2018**, *6*, 41193–41207. [CrossRef]
126. Salam, M.A.; Fouad, K.M.; Elbably, D.L.; Elsayed, S.M. Federated learning model for credit card fraud detection with data balancing techniques. *Neural Comput. Appl.* **2024**, *36*, 6231–6256. [CrossRef]
127. Cheng, D.; Wang, X.; Zhang, Y.; Zhang, L. Graph neural network for fraud detection via spatial-temporal attention. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3800–3813. [CrossRef]
128. Chullamonthon, P.; Tangamchit, P. Ensemble of supervised and unsupervised deep neural networks for stock price manipulation detection. *Expert Syst. Appl.* **2023**, *220*, 119698. [CrossRef]