AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 7(1) (2026) 85-116
https://doi.org/10.22060/AJMC.2025.24642.1446

Review Article

A systematic survey and empirical comparison of hybrid methods for imbalanced fraud detection: Combining resampling and machine learning

Behnam Yousefimehr, Mehdi Ghatee*

Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Iran

ABSTRACT: The accurate identification of fraudulent activities has been a significant focus of computational research, leading to the development of diverse methodologies ranging from traditional statistical tests to advanced machine learning and deep learning models. A persistent and critical challenge undermining these approaches is the inherent class imbalance present in most real-world fraud datasets, where genuine transactions vastly outnumber fraudulent ones, often causing models to exhibit bias toward the majority class. To mitigate this issue, a promising paradigm has emerged: hybrid frameworks that synergistically integrate data resampling techniques with robust machine learning algorithms. These frameworks are particularly valuable for their potential to facilitate accurate, real-time automated detection systems. This survey provides a comprehensive examination of the efficacy and impact of such hybrid techniques on the field of fraud detection. To quantitatively evaluate their performance, we conduct a rigorous numerical study using auto insurance fraud as a case study. Employing the car fraud datasets, we perform a detailed comparative analysis of various detection algorithms, each coupled with different resampling methods. Our empirical results demonstrate that the performance of each fraud detection algorithm is profoundly contingent upon the specific resampling strategy employed, highlighting the necessity for careful methodological selection tailored to the dataset's characteristics. Code for analysis is available at `https://github.com/behnamy2010/Car-Claims-Compression`.

Review History:

Received:28 August 2025
Revised:20 September 2025
Accepted:03 December 2025
Available Online: 01 January 2026

Keywords:

Imbalanced learning
Oversampling
Undersampling
Ensemble learning
Auto insurance fraud detection

MSC (2020):

68-XX; 68Txx; 68T05; 68T07; 62R07

1. Introduction

Traditional fraud detection methods primarily rely on manual investigations that are often time-consuming and costly. Figure 1 illustrates the number of publications on fraud detection across various sources. Since 2004, computer science has been at the forefront of developing fraud detection techniques, with an increase in publications related to networks and systems observed in 2022-2024.

Figure 2 categorizes the subject areas of these publications, identifying computer science, engineering, mathematics, decision science, and business as the most relevant fields. Machine learning methods have demonstrated significant potential for enhancing fraud detection capabilities. Additionally, some research has explored the integration of machine learning and blockchain technologies for fraud detection [12]. While numerous surveys have addressed

*Corresponding author.
E-mail addresses: behnam.y2010@aut.ac.ir (B. Yousefimehr), ghatee@aut.ac.ir (M. Ghatee)

the applications of machine learning in fraud detection [2, 11, 20, 21, 33, 42, 100, 102, 105, 115, 116, 123, 158, 165], only a handful have focused on the impact of resampling methods on imbalanced fraud datasets. For instance, [61] examined imbalanced learning techniques specifically within the insurance sector. Research has demonstrated that the efficacy of machine learning methods in detecting fraud is closely tied to the balance of the training data. A hybrid approach is developed when data balancing techniques and machine learning methods are simultaneously employed. Figure 3 illustrates a common framework structure for addressing the fraud detection problem, which is the focus of this survey. This survey paper aims to review the literature on hybrid methods for fraud detection and will follow these steps to investigate such methods:

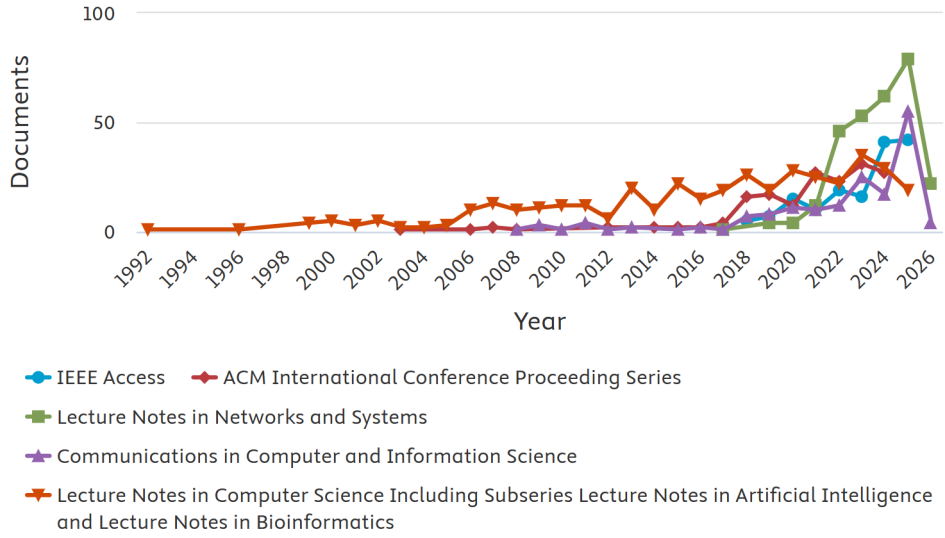

Figure 1: Number of publications in Scopus containing "Fraud Detection" in the title, abstract, and keywords (9,982 documents).

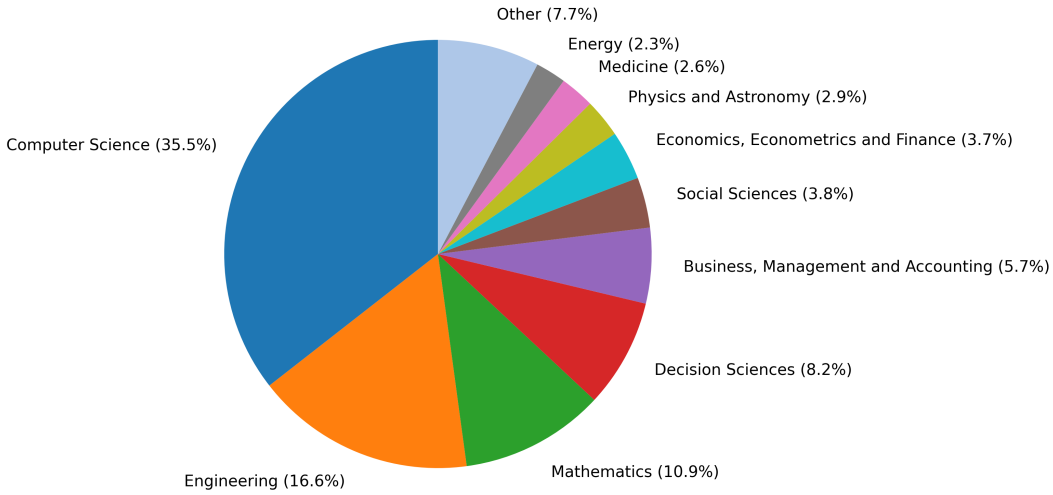

Figure 2: Subject area of publications in Scopus containing "Fraud Detection".

1. *Conducting a comprehensive review of the latest hybrid techniques in fraud detection:* A case study will be included, focusing on auto insurance fraud and addressing imbalanced data challenges through resampling techniques and cost-sensitive learning methods. This survey systematically evaluates published manuscripts up to November 20, 2025, utilizing databases such as Web of Science (WoS), Scopus, Google Scholar, and Microsoft Academic. This methodology is adopted from [92]. Relevant keywords for this study encompass 'fraud detection', 'anomaly detection', 'outlier detection', 'imbalance methods', 'machine learning', 'hybrid methods', 'data mining', 'concept drift', 'resampling', 'oversampling', 'undersampling', 'smart insurance', 'car insurance', 'auto insurance', and 'Usage-Based insurance'.
2. *Broaden the scope of the search for fraud detection:* The search is done in general to gather practical strategies for identifying fraud detection, and then, the auto insurance fraud case-study will be followed directly.

3. *Examining the growth trend of related topics by analyzing the volume of articles associated with specified keywords:* This approach aims to elucidate the developmental trajectory of prominent scientific issues and to serve as a guide for researchers targeting relevant articles in specific areas of interest.
4. *Presenting an overview of the current cutting-edge machine learning methods utilized in auto insurance fraud detection:* The analysis covers computational methods on auto insurance fraud detection, focusing on the details more meticulously than in prior studies.
5. *Establishing qualitative and quantitative comparison categories for this survey:* The qualitative comparison will include a taxonomy of auto insurance fraud detection literature, whereas the quantitative comparison will involve assessing hybrid methods using datasets and showcasing the reported outcomes. The hybrid methods will be evaluated in the quantitative section based on the Carclaimtxt and AICD datasets. Due to the complexity of implementing various hybrid methods, this comparison is confined to five tables. Nonetheless, we believe this analysis provides valuable insights into effective hybrid methods, paving the way for future research.

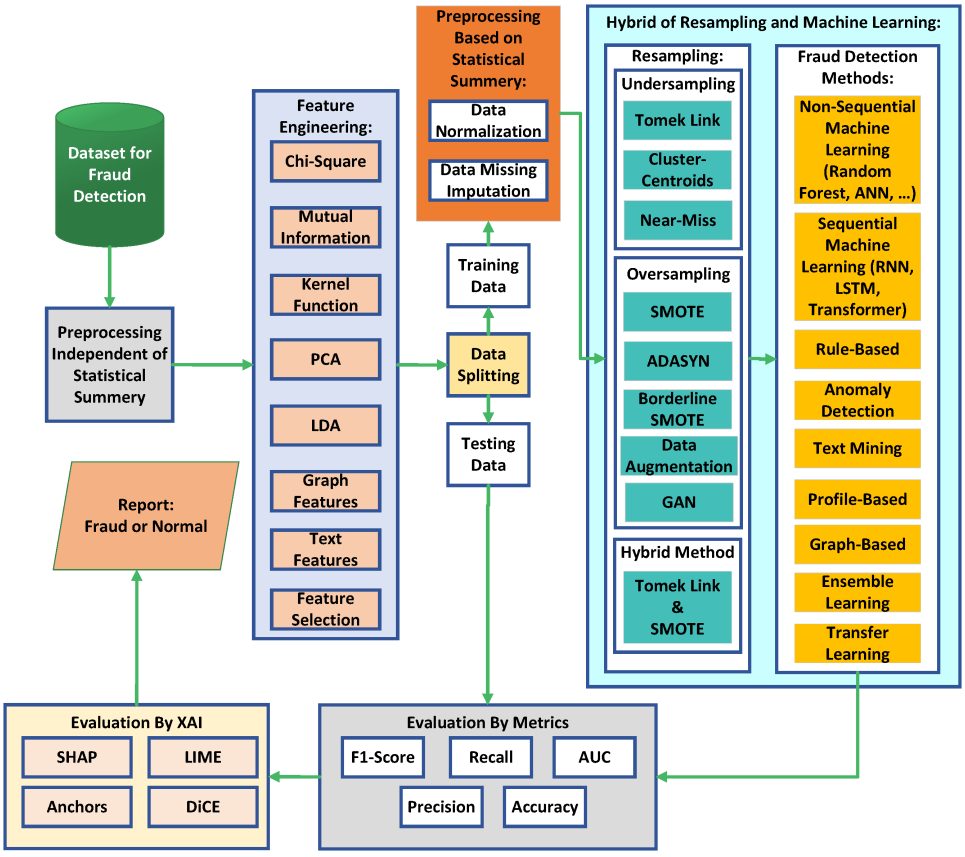

Figure 3: The general structure of a fraud detection system considered in this survey.

This paper significantly contributes to the existing body of research by thoroughly analyzing the utilization of a hybrid of machine learning techniques and data resampling methods for detecting car insurance fraud. It surpasses recent studies by examining individual machine-learning algorithms and conducting a comparative analysis of different algorithms in the context of car insurance fraud detection. By evaluating and comparing various algorithms, this survey offers valuable insights into the strengths and limitations of different approaches, assisting researchers and practitioners in making well-informed decisions concerning choosing a hybrid method tailored to their specific needs.

Additionally, this survey tackles a crucial challenge encountered by researchers in the field, namely, imbalanced fraud data. The imbalanced nature of fraud data poses a significant obstacle to effective detection, and this paper particularly emphasizes the potential impact of balancing techniques in enhancing the accuracy and robustness of car insurance fraud detection systems.

Finally, this survey outlines the future research directions in car insurance fraud detection, pinpointing areas that require further exploration and examination. By summarizing potential topics, this paper provides a roadmap for researchers to continue advancing effective techniques for detecting car insurance fraud.

The remainder of the paper is structured as follows: Section 2 presents an auto insurance case study. Section 3 outlines challenges in fraud detection, and this is followed by a comparison of the hybrid methods used in the auto insurance case study in Section 4, as well as a numerical experiment on hybrid methods in the case study in Section 5. The paper concludes with a summary and future directions in the final section.

2. A case study: Auto insurance fraud

Auto insurance companies define very different agreements and need fraud detection for these agreements [8, 15, 129]. [162] reviewed the intelligent risk control in China's insurance industry in 2019 and estimated fraud and leakage of nearly 20% of total annual losses (more than 20 billion yuan). In the UK, insurers identified 72,600 fraudulent insurance claims, including 42,500 fraudulent car insurance claims, in 2022, valued at £1.1 billion. The Association of British Insurers (ABI) estimates that the same amount of fraud goes undetected, so UK insurers invest at least £200 million annually in fraud detection projects[1]. Fraud statistics in the insurance industry in the US also show a very high value [75]. Figure 4 shows insurance fraud statistics, including $35.1 billion in auto insurance fraud. As the volume of claims increases, fraud detection becomes increasingly important. Identifying fraudulent claims is critical to preventing financial losses and maintaining customer trust. Different models have been studied to

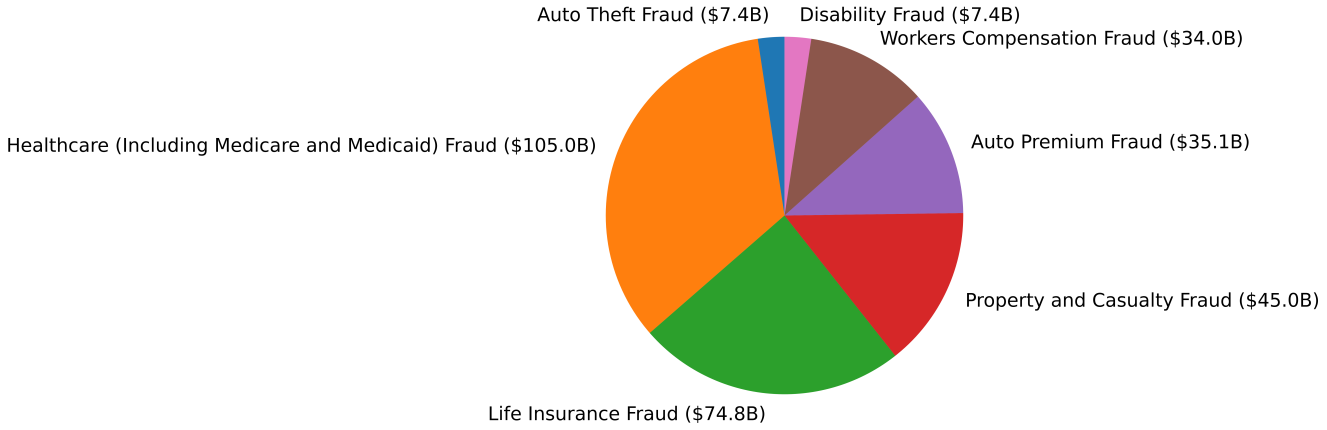

Figure 4: Fraud statistics in the insurance industry in the United States [75].

detect fraud. To train these models, the researchers need pervasive datasets containing examples of both fraudulent and legitimate claims. Most of the datasets on car insurance are private because of business limitations. There are several publicly available datasets for car insurance fraud problems. They typically contain driver and vehicle characteristics and some details about the accident or incident. Table 1 lists the most essential datasets in this category. Statista platform[2] has also published reports regarding datasets and research on fraudulent insurance claims in the United Kingdom, Norway, India, Russia, Poland, etc.

Table 1: The most popular car insurance fraud detection datasets

Dataset	# Samples	# Features	Fraud ratio	Ref.	Link
Carclaimtxt	15420	32	6%	[113]	https://www.kaggle.com/datasets/shivamb/vehicle-claim-fraud-detection
Car Insurance Claim Data (CICD)	10302	26	27%	-	https://www.kaggle.com/datasets/xiaomengsun/car-insurance-claim-data
Car Insurance Claim Prediction (CICP)	58592	43	6%	-	https://www.kaggle.com/datasets/ifteshanajnin/carinsuranceclaimprediction-classification
Auto Insurance Claims Data (AICD)	1000	39	25%	[98]	https://www.kaggle.com/datasets/buntyshah/auto-insurance-claims-data

[1] https://www.abi.org.uk/products-and-issues/topics-and-issues/fraud/
[2] https://www.statista.com/

3. Fraud detection challenges

A fraud detection system, as illustrated in Figure 3, encounters various challenges that will be addressed in the subsequent sections.

3.1. Data Quality

The quality of data used for training machine learning models is of paramount importance. Insurance companies often provide limited, anonymized data for research purposes, making it difficult to detect fraudulent activities using these restricted datasets. Since the datasets of different insurance companies are kept separate, fraud perpetrated by an individual may go unnoticed across various datasets. Furthermore, due to concerns about maintaining commercial confidentiality, collaboration among insurance companies remains limited. However, sharing information among these companies could facilitate the identification of patterns by examining the network of relationships among individuals.

3.2. Imbalanced Data

Imbalanced data refers to situations where there are significantly fewer instances of fraudulent cases compared to non-fraudulent cases. This imbalance poses challenges in effectively learning the characteristics of fraudulent samples [59]. To tackle this issue, techniques such as resampling and cost-sensitive learning can be employed [170]. The extent of the imbalance is determined by the ratio of majority to minority classes. Various balancing methods have been extensively researched within the realm of fraud detection. Figure 5 illustrates recent trends in related studies concerning fraud detection and resampling techniques.

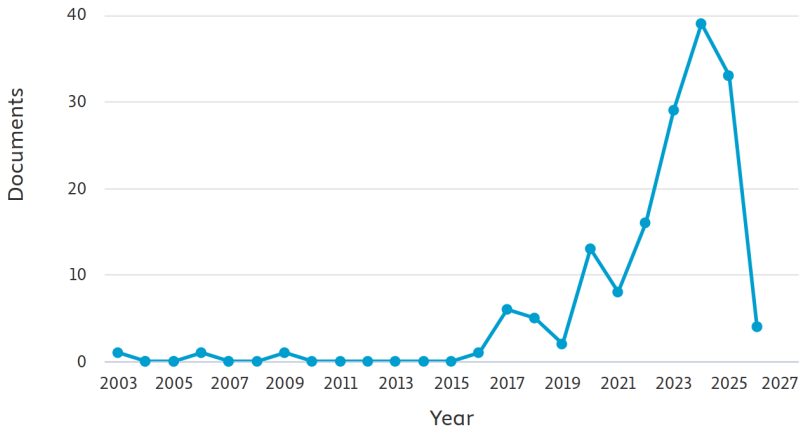

Figure 5: Number of publications in Scopus containing "Fraud Detection" and "Resampling" in the title, abstract, and keywords (159 documents).

It's important to note that balancing methods should only be applied to the training data. Balancing the data before splitting it into training and testing sets can result in data leakage, which is the use of information during the training process that is not ordinarily available at the time of prediction [71]. Unfortunately, some researchers have overlooked this issue, which is a significant mistake. There are examples of studies that address data leakage in fraud detection, such as [16, 136].

Various methods can be employed to balance a dataset. Balancing the data is essential for improving the accuracy and reliability of fraud detection systems by addressing the uneven distribution of fraudulent and non-fraudulent instances in the dataset. Some researchers have explored modern clustering algorithms to determine the similarity between data during the balancing phase, as seen in the work of [6].

3.2.1. Resampling:

Resampling methods fall into Oversampling, Undersampling, and Hybrid techniques. Their approaches have been compared in Figure 6.

Undersampling involves reducing or removing some data from the majority class. This can be done randomly, known as random undersampling, or through informed undersampling using statistical calculations and data cleaning methods to further filter the majority class data [34]. Some popular undersampling methods include Tomek link [147], Cluster-Centroids, and Near-Miss [91].

Oversampling, on the other hand, adds new samples to the minority class. It can be done using original data,

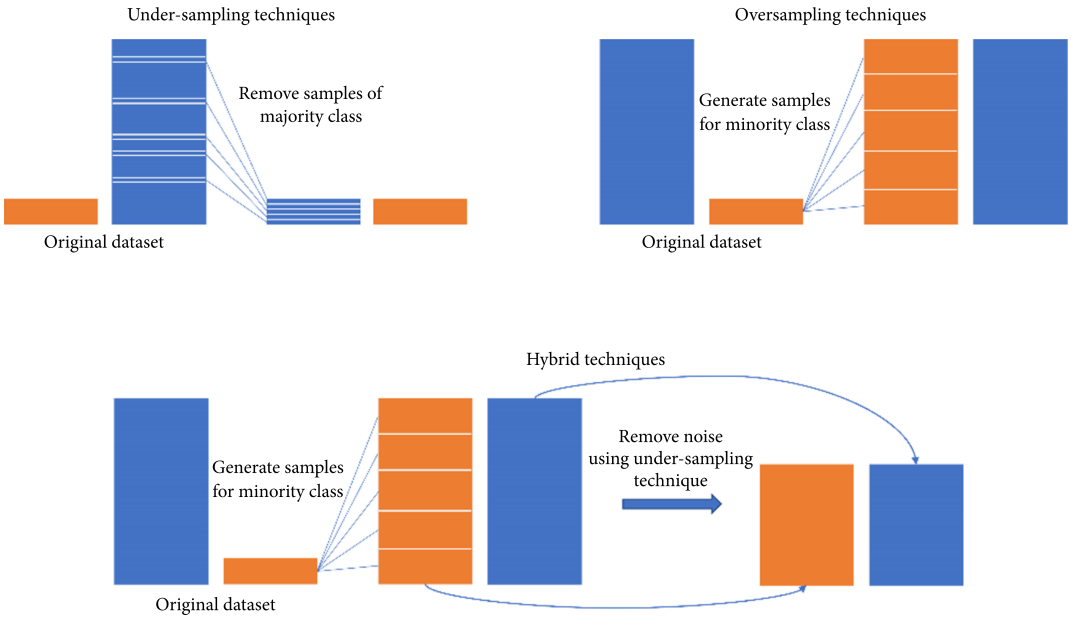

Figure 6: A visualized comparison between the approaches of oversampling, undersampling, and hybrid techniques (Adopted from [79]).

known as random oversampling, or by generating synthetic data, such as the Synthetic Minority Oversampling Technique (SMOTE) [31]. Overfitting is a common problem in oversampling due to the frequent addition of data, which can cause the decision boundary to become unclear [34]. Popular oversampling methods include SMOTE [31], ADASYN [58], BorderlineSMOTE [54], data augmentation [142, 163], and Generative Adversarial Networks (GAN) [46, 51].

Hybrid sampling combines oversampling and undersampling methods to balance the data. It involves reducing the majority class data using undersampling and increasing the minority class data using oversampling [34]. An example of a hybrid algorithm entails increasing the minority class using SMOTE and reducing the majority class using Tomek link undersampling. For further details, see [18].

Figure 7 compares the number of publications in Scopus that used these techniques for fraud detection. It demonstrates that oversampling, undersampling, and GAN are the most effective methods for fraud detection in the literature. For an overview of GAN applications for fraud detection, refer to [143]. Additionally, for references on data augmentation for car insurance fraud detection, see [1, 88, 99].

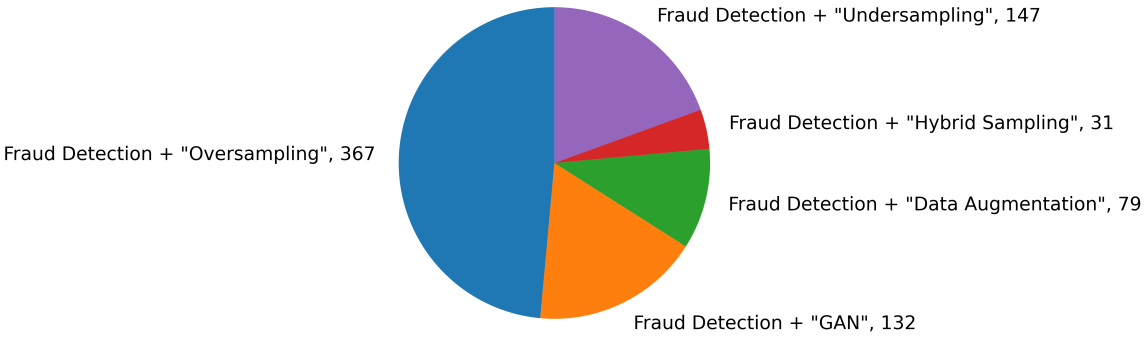

Figure 7: Pie chart of publications containing "Fraud Detection" & different resampling methods in the title, abstract, and keywords of Scopus publications).

3.2.2. Cost-sensitive algorithms for imbalanced class

Cost-sensitive learning is a subfield of machine learning that addresses classification problems where misclassification costs are not equal [41]. In fraud detection, a common approach involves extending a machine learning algorithm to account for different errors for classes [84]. Cost sensitivity focuses on minimizing the costs associated with failing to detect fraudulent activities, where missing a fraud sample is generally considered more costly than a false alarm in a legitimate sample [168].

3.3. Concept drift

Concept drift, within machine learning, refers to the change in relationships between input and output data over time. Patterns and relationships in such data often change over the prediction time [152, 174]. Concept drift occurs in supervised learning problems with real-time data collection or time-series forecasting. Some methods to address this challenge include continuous data collection [174], periodic model updates [148], using the sliding window for recurrent neural networks [66], employing ensemble learning techniques [95], and influencing the users' profiles in the learning model [78]. Sequence-based models are effective in tackling this challenge. Monitoring the model performance continuously and retraining the model as necessary can also help address this challenge. [5] reviewed the credit card fraud detection research using concept drift methods. The same study has been done by [131]. Based on Scopus results, Figure 8 shows the number of publications that include "fraud detection", and "Concept drift" in recent years. Table 2 describes various approaches to fraud detection and managing concept drift in online transactions.

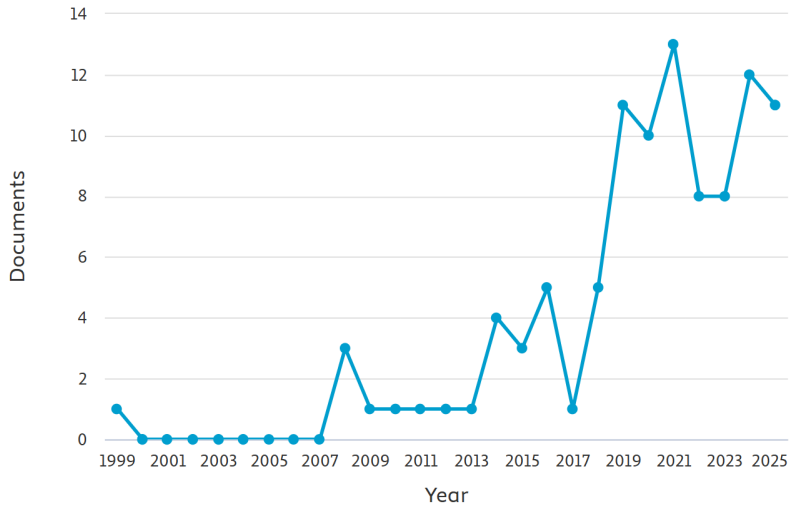

Figure 8: Number of publications in Scopus containing "Fraud Detection" & "Concept Drift" in the title, abstract, and keywords (100 documents).

3.4. Fraud detection techniques

In the literature, many techniques are used for fraud detection. Figure 9 compares the frequency of different methods used for fraud detection based on Scopus sources. This figure shows that random forest, decision tree, SVM, Graph Neural Network (GNN), and Convolutional Neural Network (CNN) are the most popular methods for fraud detection. However, the choice of a technique depends on the features of the data and the type of fraud. In the following sections, we explain some branches of methods used for fraud detection in the literature. Additionally, the quality of features is a critical issue for the success of machine learning algorithms. The effect of feature extraction for fraud detection can be seen in [125].

3.4.1. Rule-based techniques

Rule-based systems use a set of pre-defined rules to identify fraudulent activities and compare cases against a set of pre-defined rules. Then, they mark the samples as fraudulent if they match any of the regulations [77]. Rule-based techniques are simple and effective, but are limited in identifying new fraud patterns.

3.4.2. Anomaly detection techniques

This technique detects fraudulent activities by identifying outliers in the data. Usually, it uses statistical methods to detect unusual behavior in the data and flags it as fraudulent if the activity is outside the normal range. Anomaly detection techniques can identify known and unknown fraud patterns [28, 29, 53].

3.4.3. Machine learning techniques

Classical machine learning techniques such as explainable machine learning [55], decision tree [26], SVM [36], and random forest [25] are helpful to identify fraud patterns. In recent years representation learning [150], Graph Neural Network (GNN) [159, 160, 161], contrastive GNN [39], parallel models [14], ensemble learning [94], and autoencoders

Table 2: Concept drift in fraud detection researches

Ref.	Description
[90]	This paper proposed a concept drift management framework for fraud detection in online transactions. The framework employs a window-based method to manage concept drift independently from historical profiles.
[38]	It presented fraud detection systems for processing credit card transactions using different classifiers. The classifiers return alerts for the riskiest payments, and human researchers initially provide a small set of supervised samples. The delay in acquiring accurate labels and the interaction between alerts and monitored information must be carefully considered when learning in a conceptual environment.
[10]	It introduced an anomaly-based fraud detection system model that identifies fraudulent activities by deviant behavior over time. The system utilizes a combination of features from the literature and employs an effective online flow approach based on an incremental classifier to distinguish fraudulent from normal data. The framework also uses a ranked search feature selection method and a trigger-based approach for adaptive learning in the concept drift recognition stage.
[139]	This paper proposed a transaction window wrapping model for real-time fraud detection in credit card transactions. The model addresses data imbalance, noise, boundary entities, and concept drift. An incremental learning model, a cost-sensitive base learner, and a weighted vote-based combinator effectively exploit concept drift and data imbalance.
[87]	It presented a customs fraud detection system that considers the detection of new frauds in the context of new imported goods. The paper introduces an adaptive selection method to control the balance between exploitation and discovery strategies. It uses model performance trends and conceptual drift to determine the best discovery ratio at any time.
[171]	It proposed HOBA methodology for feature engineering to detect credit card fraud by deep learning models.
[133]	This paper reviewed the problem of concept drift in fraud detection, exploring different types of methods, including adaptive learning models.
[5]	This reference comprehensively compared credit card fraud detection methods using machine learning and concept drift techniques.
[7]	It proposed a hybrid machine-learning framework that combines SMOTEBoost and cost-sensitive learning to tackle data imbalances, utilizes adversarial training and FraudGAN for enhanced robustness, employs DDM and ADWIN for adaptive learning, and incorporates SHAP, LIME, and human-in-the-loop (HITL) analysis to ensure transparency and explainability.
[62]	Firstly, it pretrained a deep neural network and transferred the parameters of its hidden layers to a similar network for processing streaming data, while freezing some of the hidden layers. The outputs from both phases were then utilized to train and test an autoencoder. Concept drift is identified by examining the reconstruction error of the autoencoder in conjunction with the 3σ principle.
[37]	It introduced Temporal-Spatial-Semantic Graph Convolution (TSSGC) to capture the temporal, spatial, and semantic aspects of transaction data. A Deep Q-Network (DQN) adjusts the fraud detection threshold and feature importance in real-time, enabling the model to adapt to changing fraud patterns and reduce detection costs. Additionally, a Federated Learning approach allows for collaborative training across financial institutions while ensuring data privacy. The framework demonstrates strong resilience to concept drift and adversarial attacks, maintaining high performance over time.
[9]	It presents a comprehensive framework that integrates advanced data preprocessing, effective drift detection, and a robust detection model. The approach utilizes Mutual Information and SelectKBest for feature selection, ADASYN for class imbalance, and Convolutional Neural Networks (CNN) for recognizing complex transaction patterns. Early Drift Detection Method (EDDM) and Adaptive Windowing (ADWIN) are employed to proactively identify and respond to both gradual and sudden drift.

[43] have been also followed by researchers for fraud detection. For further investigation to identify fraud, see [108]. Figure 10 shows the same attention to using deep learning methods for fraud detection.

3.4.4. Sequential machine learning techniques

Sequential machine learning techniques such as Hidden Markov Models (HMM) [119], Recurrent Neural Networks (RNN) [60], and LSTM [167] analyze sequential data. They identify fraud patterns as a claimant changing over time. The frequencies of the papers focusing on fraud detection and RNN, LSTM, and Transformer networks have been illustrated in Figure 11. This figure shows that LSTM has recently received more attention from researchers

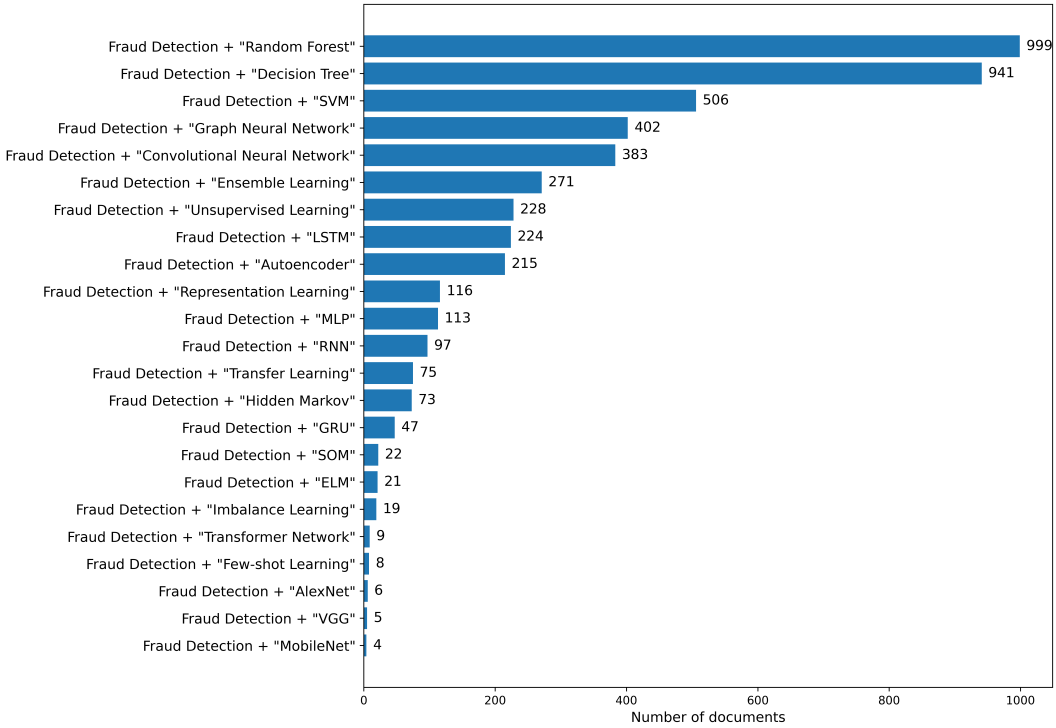

Figure 9: Frequency chart of publications containing fraud detection & different machine learning techniques in the title, abstract, and keywords of papers of Scopus.

than RNN and the Transformer network. For transformer models for fraud detection, one can refer to [83, 153, 173]. In addition, for a book containing sequential modeling to detect dynamic fraud, see [172]. These methods can also deal with the concept drift problem.

3.4.5. Text mining techniques

In auto insurance fraud detection, the Natural Language Processing (NLP) technique [35] can identify language patterns indicating fraudulent activities. For example, NLP algorithms can be trained to identify common phrases and select words used commonly in fraudulent claims. To see more details on NLP applications for fraud detection, see [45, 24, 27, 69, 30, 140].

3.4.6. Profile-based machine learning techniques

They use clustering or rule extraction methods to group similar claims and identify common features that indicate fraud. They create profiles of fraudulent behavior to flag new claims that exhibit identical characteristics [77]. These techniques are instrumental when dealing with large amounts of data, as they can identify patterns that may be difficult to detect using other methods.

3.4.7. Graph-based machine learning techniques

Graph-based machine learning techniques such as graph convolutional networks [76] and network analysis [23] analyze the relationships between the various entities involved in an auto insurance claim (the claimant, the insured, and the repairer). They identify behavior patterns indicating fraud, such as collusion between applicants, insurers, and even repair shops. Graph-based techniques can be instrumental when dealing with complex networks of entities, as they can identify patterns that may be difficult to detect using other methods [103, 155].

3.4.8. Transfer learning

Transfer learning techniques have gained prominence in the domain of fraud detection by leveraging knowledge acquired from various sources [82, 134, 137, 138, 146]. These methodologies typically adapt the characteristics of the present task by drawing upon insights from established domains [81]. This approach not only enhances the efficacy of fraud detection systems but also underscores the importance of interdisciplinary knowledge transfer in addressing complex challenges within this field.

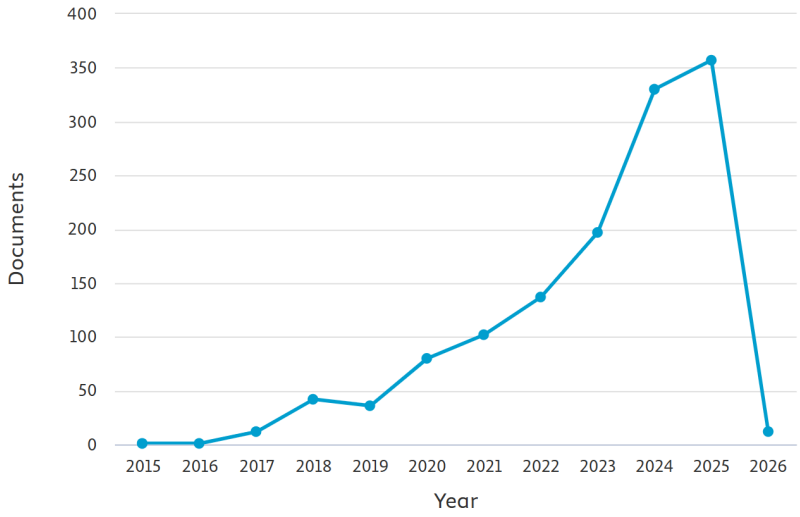

Figure 10: Frequency of publications in Scopus containing "Fraud detection" & "Deep Learning" (1,307 documents).

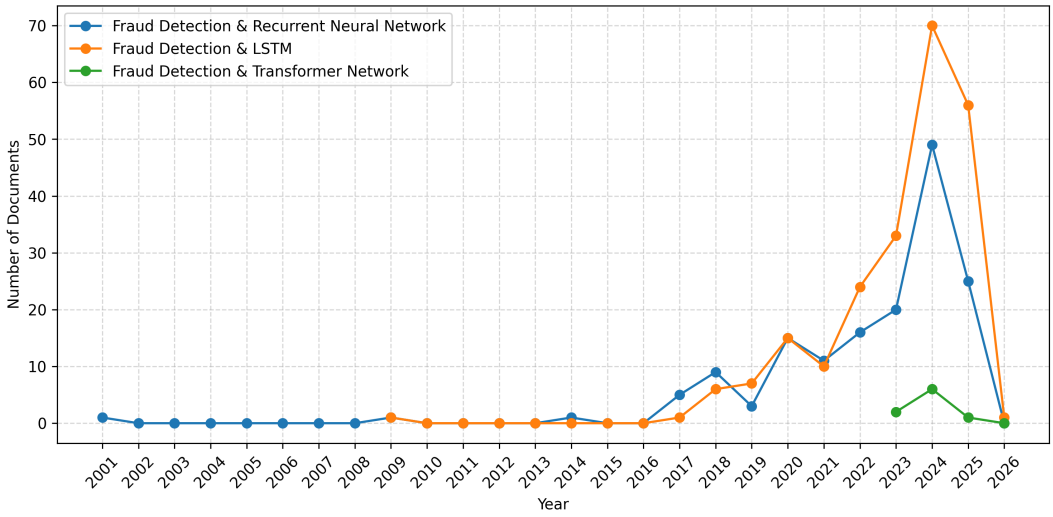

Figure 11: Frequency of publications in Scopus containing "Fraud detection" & "Recurrent Neural Network", or "LSTM" or "Transformer Network" in the recent years.

3.5. Evaluation

3.5.1. Evaluation by metrics

The accuracy paradox indicates that accuracy is not a suitable criterion for evaluating categories in imbalanced data learning. This problem arises from the imbalance between the fraud class and the normal class [4, 149]. In recent years, many alternative evaluation methods have been suggested instead of accuracy. However, this issue is still open to data fraud. Precision is the proportion of true positive predictions (i.e., actual fraud cases) among all positive predictions. Recall is the proportion of true positive predictions (i.e., actual fraud cases) that the model can identify among all actual fraud cases in the dataset. F1-score (F-measure) also takes both precision and recall by using the harmonic mean. The AUC-ROC measures the model's ability to distinguish between fraudulent and non-fraudulent cases. At various threshold settings, it plots the true positive rate (i.e., recall) against the false positive rate (i.e., the proportion of non-fraudulent cases incorrectly identified as fraudulent). A high AUC-ROC score indicates that the model can accurately distinguish between fraudulent and non-fraudulent cases.

The Geometric mean (G-mean) is a performance metric commonly used in fraud detection, especially when dealing with imbalanced datasets. The G-mean criterion considers the model's sensitivity and specificity, making it a valuable metric for evaluating a model's ability to correctly identify fraudulent and non-fraudulent cases. A high G-mean score indicates that the model can accurately identify fraudulent and non-fraudulent cases while minimizing the false positive and false negative rates. Figure 12 presents the frequency of the used evaluation metric for fraud detection in the literature. As one can note, the used metrics were usually accuracy; however, it is sensitive to

imbalanced data. Thus, more attention is expected to be paid to robust metrics against imbalanced data in the following research.

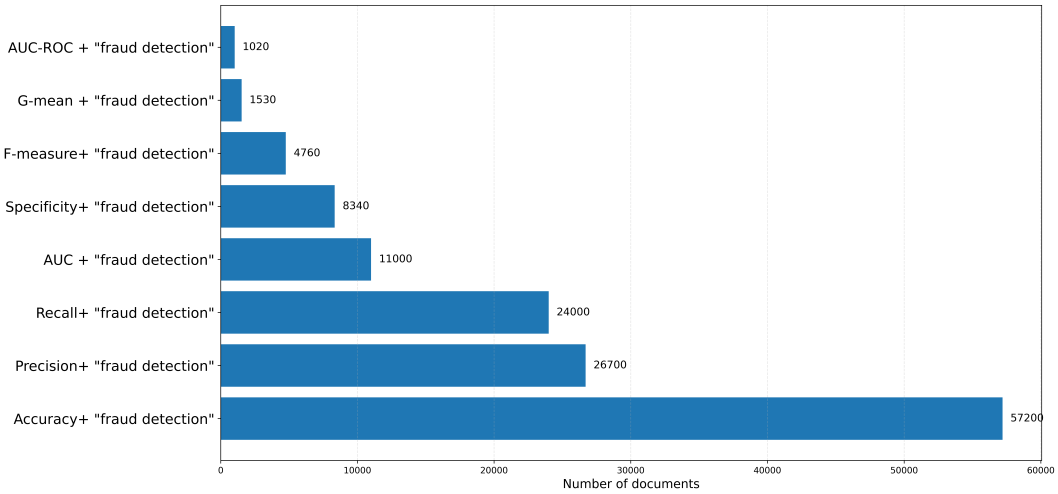

Figure 12: Frequency chart of publications containing fraud detection & different evaluation techniques (in Google Scholar).

3.5.2. Evaluation by Explainable AI (XAI)

In the context of auto insurance, AI models are used to detect fraudulent behavior patterns by analyzing large claims datasets. However, traditional AI models often operate as "black boxes," making it difficult to explain how specific decisions are reached. This lack of transparency can lead to challenges in regulatory compliance and reduce trust among users, especially when decisions impact customer premiums or result in claim rejections [13]. To build trust and implement artificial intelligence systems in critical domains, it is important to have a deep understanding of and explain the model effectively. Explainable Artificial Intelligence (XAI) focuses on advancing machine learning techniques that allow human users to understand, trust, and create transparent models [40]. When dealing with fraud detection problems, it is crucial to enhance the model's explainability by utilizing powerful Explainable AI (XAI) techniques such as SHAP (SHapley Additive exPlanations) [85], LIME (Local Interpretable Model-agnostic Explanations) [121], Anchors (High-Precision Model-Agnostic Explanations) [122], or DiCE (Diverse Counterfactual Explanations) [97] to gain meaningful insights into the model's decision-making process. These techniques enable experts to make well-informed decisions based on the interpretation results. For example, by leveraging LIME's local approximation capabilities and the grounded consistency of SHAP, it becomes possible for experts to gain a comprehensive understanding of model outputs, facilitating informed decisions based on the model's predictions and articulating these to stakeholders. XAI techniques address these issues by making the decision-making process of AI models more interpretable. For example, knowledge distillation and rule extraction simplify complex models into more understandable components, allowing insurance professionals to see which factors most heavily influence decisions [169]. This is critical not only for internal auditing purposes but also for communicating with customers and regulatory bodies [107]. For more details, refer to [49, 50, 80, 104, 110, 114, 120]. Table 3 summarizes key research that discusses the application of XAI in fraud detection, particularly in the insurance sector.

The incorporation of XAI into fraud detection systems in auto insurance not only improves the transparency and reliability of AI models but also ensures compliance with regulatory standards. As the insurance industry continues to embrace AI-driven approaches, the role of XAI is likely to expand, becoming an essential tool for both operational efficiency and ethical AI deployment. The most practical tools in this category include:

1. **SHAP:** This approach provides a unified measure of feature importance for any model using game theory. It assigns a "Shapley value" to each feature, representing its average contribution to the model's prediction across all possible feature combinations. Grounded in cooperative game theory, SHAP ensures fair distribution of each feature's importance, offering consistent and reliable explanations. In fraud detection, SHAP proves useful in identifying which variables, such as claim amount, customer history, or claim frequency, significantly influence the AI model's decision to flag a claim as potentially fraudulent. By presenting these factors in an interpretable format, SHAP helps insurance professionals understand and validate the model's behavior, ensuring transparent and justifiable decisions [85].
2. **LIME:** This popular XAI tool provides local explanations for model predictions. Unlike SHAP, which offers a global perspective on feature importance, LIME focuses on explaining individual predictions by approximating the model locally around the specific prediction of interest. It achieves this by generating a simpler,

Table 3: XAI Applications in Fraud Detection

Ref.	XAI Method(s)	Key Findings
[48]	SHAP	Enhances transparency in financial fraud detection by explaining key variables, improving interpretability, and ensuring regulatory compliance.
[44]	Attention Mechanisms	Improves fraud detection in healthcare claims by handling unstructured data, outperforming traditional models, and providing meaningful explanations.
[101]	LIME, SHAP	Offers explanations for decisions using random forest, addressing customer concerns, and mitigating credit and legal risks.
[89]	DefragTrees	Improves model transparency in insurance fraud detection by providing global explanations for tree ensemble models, enhancing predictive power and ethical considerations.
[73]	SHAP	A fully unsupervised approach containing SHAP for feature selection with an autoencoder for generating class labels for credit card fraud detection
[157]	Partial Dependence Plots (PDP), SHAP, and LIME	Used ensemble techniques, including Voting, Weighted, and Stacking methods, to combine different models and PDP, SHAP, and LIME to understand features' impact on the predictions.
[167]	SHAP, and LIME	Uses two hybrid approaches containing OCSVM, SMOTE, and random undersampling by preserving the distribution of fraud instances, then applies LightGBM and LSTM for fraud detection.

interpretable model, such as a linear model, that approximates the behavior of the complex model near the specific instance being analyzed. In the context of auto insurance fraud detection, LIME can be employed to understand why a particular claim was classified as fraudulent. For example, LIME might reveal that the model heavily weighed unusual patterns in a customer's claim history or inconsistencies in the reported accident details. This level of insight allows insurance companies to investigate specific cases more effectively and provide clear explanations to customers or regulators [121].

3. **PDP:** A Partial Dependence Plot (PDP) is a global, model-agnostic interpretation tool that visualizes the average relationship between a subset of features (typically one or two) and the predicted outcome of a machine learning model. It marginalizes (averages out) the effects of all other features to show the marginal effect of the feature(s) of interest on the prediction [52].
4. **DefragTrees:** DefragTrees (short for Defragmentation of Trees) is a post-hoc explainability method designed to enhance the interpretability of complex tree ensemble models (e.g., Random Forests, Gradient Boosted Trees) by simplifying them into a smaller, more understandable set of rules or a single decision tree. DefragTrees aims to balance interpretability and predictive performance by transforming a tree ensemble into a simplified model that retains faithfulness to the original while being human-readable [106].

While both SHAP and LIME aim to make AI models more interpretable, they do so in different ways and offer distinct advantages. SHAP is particularly valuable for providing consistent global explanations across a dataset, making it ideal for understanding overall model behavior. In contrast, LIME excels at offering detailed local explanations, making it useful for case-by-case analysis. The choice between SHAP and LIME often depends on the specific needs of the fraud detection system. If the goal is to ensure that the model's behavior is transparent across all decisions, SHAP may be more appropriate. However, if the focus is on explaining individual decisions and understanding specific instances of potential fraud, LIME may offer more actionable insights. In summary, integrating SHAP and LIME into fraud detection models enhances the interpretability and transparency of AI systems, making them more reliable and trustworthy for use in auto insurance [96].

4. Qualitative comparison on auto insurance fraud detection

This section delves into recent articles on car insurance fraud. Table 4 assesses the articles that conducted data balancing before the train-test split. In contrast, Table 5 compares articles that paid attention to the correct separation of the data.

These tables reveal the following insights:

- Carclaims is a widely used dataset for comparing fraud detection methods.
- SMOTE is among fraud detection studies' most renowned data balancing methods. Extensions of SMOTE, random undersampling, and cost-sensitive methods [67] are also prominent.

Table 4: Car insurance fraud detection containing the balancing phase before the train-test split

Ref.	Dataset	Balancing	Classifier	Feature engineering	Evaluation	Results
[141]	Carclaims	MWMOTE	DT	Chi-squared method, Recursive feature elimination, Tree-based method	F1: 93.4% AUC: 97.96% Precision: 93.92% Recall: 93.42%	1-Proper performance of tree-based methods, 2-Poor performance of undersampling and hybrid sampling methods compared to oversampling.
[99]	AICD	SMOTE	Deep learning	None	Accuracy: 100% Precision: 100% Recall: 100% F1: 100%	Better result of the balanced dataset in combination with a neural network. Overfitting should be checked before usage.
[117]	Private	SMOTE	Random forest	None	Accuracy: 98.5% Sensitivity: 100% Specificity: 98.5% AUC: 99.7%	Better performance of ensemble learners.
[109]	Carclaims	SMOTE	SVM, MLP, KNN	None	Accuracy: 79.24% Sensitivity: 92.27% Specificity: 58.35%	Using decision template to combine classifiers has improved the performance.
[57]	Carclaims	SMOTE	Random forest	None	Accuracy: 94.33% Sensitivity: 99.9% Specificity: 45.1%	1-Good explanation of the data, 2-Better performance of ensemble learning versus individual methods.
[145]	Carclaims	ADASYN	SVM, MLP, DT	None	Accuracy: 79.51% Sensitivity: 94.74% Specificity: 41.76%	1-Better performance of ADASYN than SMOTE, 2-Resampling in SVM and MLP has reduced specificity.

- Most studies concentrate on traditional machine learning methods such as Decision Trees (DT), Support Vector Machines (SVM), Multi-Layer Perceptrons (MLP), K-Nearest Neighbors (KNN), and Random Forests (RF). Recent works have also utilized deep learning and social network analysis.
- Various feature engineering methods are employed for fraud detection, including the Chi-squared method, recursive feature elimination, Principal Component Analysis (PCA), search methods, and Latent Dirichlet Allocation (LDA).
- Metrics like Recall, F-measure, Sensitivity, Specificity, and AUC offer more precise insights for comparing fraud detection techniques.
- Some studies may exhibit overfitting, mainly when reporting accuracy close to 100%. Double-checking for overfitting is advised, and consider using regularization.
- The performance of a hybrid of SMOTE and RF varied for different datasets in [117] and [57]. This indicates that there is no one-size-fits-all hybrid method for differing applications.
- Due to the lack of a complete dataset, the relation network or user profile method has been neglected for many fraud detection problems.
- In most cases, resampling is done before splitting the training and testing data, which causes data leakage. For balancing, few researchers focus on cost-sensitive methods [67].
- Usually, the ensemble learners and tree-based methods perform better compared with the single classifiers.
- In the case of big data, deep learning methods can also achieve good results.
- Special attention has not been paid to the problems of concept drift and, of course, the methods based on sequence analysis.
- In this field, feature selection is also helpful, and the presence of unrelated features causes a drop in the model's performance.

5. Comparison between hybrid methods on the case study

Our study evaluated the performance of various hybrid resampling techniques and machine learning methods for car insurance fraud detection. We selected powerful machine learning algorithms that have been tested in recent

Table 5: Car insurance fraud detection publication

Ref.	Dataset	Balancing	Classifier	Feature Engineering	Evaluation	Results
[126]	Carclaims	-	Random forest	Chi-square, Mutual information	Recall: 95%	1-The positive effect of feature selection on performance, 2-Weaker performance of logistic regression compared to random forest.
[98]	AICD	ADASYN	SVM	None	Accuracy: 98% Precision: 87% Recall: 90% F1: 90%	1-Proper performance of SVM, 2-Good explanation of the data, 3-Poor performance of SMOTE compared to ADASYN.
[65]	Carclaims	TH-SMOTE	KNN	None	Accuracy: 91.52% Specificity: 90.62% Recall: 95.1% F1: 90%	1-Proper performance of KNN in combination with SMOTE, 2-The F-Measure with optimal beta has better comparability than AUC-ROC, 3- Half-half is not always good value for resampling.
[3]	Kaggle	-	J48, NN, XGboost, Naive Bayes	Kernel function	Accuracy: 92.53% ROC: 98.1%	1-Proper performance of XGboost and Naive Bayes, 2-Low ROC in Naive Bayes.
[130]	Private	-	Spectral Ranking	-	-	1-Using clustering for labeling, 2-Higher speed, and acceptable false alarm rate, 3-Not paying attention to balancing.
[164]	Private	-	Genetic Algorithm, NN	Principal Component Analysis	Accuracy: 100%	1-Better performance than the traditional genetic algorithm in terms of convergence speed and prediction accuracy, 2-Not paying attention to balancing.
[64]	Carclaims	-	Random forest	Experimental analysis	Recall: 23.83% Precision: 19.66% F-Measure: 21.52% K-Measure: 99.46%	1-Better performance of ensemble and tree-based learners compared to individual learners, 2- Low value of recall compared to the recall of other models.
[111]	Carclaims	-	ELM	-	Sensitivity: 47.40% Specificity: 74.98% G-mean: 59.62%	ELM hyperparameter optimization has improved performance.
[17]	Carclaims	-	Inter-quartile range	CSFSubset as attribute evaluator and Genetic Search as the search method	Accuracy: 98.0% F1: 85.7%	1-The positive effect of feature selection on accuracy, 2-Weaker performance of SVM compared to inter-quartile range.
[23]	Private	-	Social network analysis	-	-	Finding relationship and fraud groups using circle in graph.
[22]	Private	-	Naive Bayes, DT	-	Accuracy: 78% Recall: 86% Precision: 70%	Ignoring balancing.
[144]	Private	Random undersam-pling	Social network analysis	-	Recall: 89.13% Precision: 65.08% F1: 75.23% Specificity: 86.67% AUC: 92.28%	1-No need for labels, 2-Random undersampling may cause some loss of important information.
[112]	Private	Cost sensitive	DT	-	Precision: 70.39%	1-CT trees behave better than C4.5 trees, 2-Decreasing recall with increasing precision.
[151]	Carclaims	Addressing class imbalance	Machine learning methods	Graph features	Recall: 83.33% Specificity: 91.05%	This paper used artificial intelligence approaches such as claim graph features, classification, and rule-based systems to uncover fraudulent activities in auto insurance.
[162]	Carclaims	-	Deep learning	-	Accuracy: %89.6 Recall: 89.6% Precision: 90.7%	1-Better performance of deep learning compared to classic algorithms, 2-Ignoring balancing.
[156]	Private	SMOTE	Deep learning	Latent Dirichlet Allocation	Accuracy: 91.4% Precision: 91.7% F1: 91.3%	Integration of textual, numeric and categorical data has improved performance.
[135]	Private	-	MRCNN	-	mAP: 0.40	Fraud detection with image.

studies, including artificial neural networks [154], random forest [25], XGBoost [32], CatBoost [118], LightGBM [72], and AdaBoost [47]. We conducted a comparative analysis using these methods and different resampling techniques on the Carclaimtxt and AICD dataset. We applied the same sampling percentage to all models to ensure a fair evaluation. All the code implementations can be found on our GitHub link [166], and their setup parameters can be accessed there. All experiments are implemented in Python using scikit-learn, imbalanced-learn, XGBoost, LightGBM, CatBoost, TensorFlow and SciKeras. We fix the random seed to 42 for NumPy, Python's random module, TensorFlow, the cross-validation splitter, the stochastic learners, and all resampling methods to ensure reproducibility. Unless stated otherwise, we evaluate each classifier–sampler combination using repeated stratified 5-fold cross-validation with 3 repeats (15 train/test splits in total); in every split, a Min–Max scaler is fitted on the training fold only and applied to both training and test data. All resampling strategies are applied strictly to the training folds, never to the test folds, so as to avoid information leakage. For oversampling and undersampling methods we use sampling_strategy = 1.0, yielding an approximately 1:1 minority-to-majority ratio in the resampled training data. To avoid optimistic bias on this relatively small dataset, we do not perform exhaustive hyperparameter tuning; instead, we adopt modest configurations close to the libraries' defaults and keep them fixed across all folds and repetitions. Additionally, we analyzed the effects of different resampling methods in hybrid implementations with different machine-learning methods. The detailed results can be accessed in Tables 6-17 for hybrids of different resampling methods and artificial neural network, random forest, XGBoost, CatBoost, LightGBM, and AdaBoost.

Figures 13–18 summarize the mean recall, F1-score, and AUC over all resampling–model hybrids on both the Carclaimtxt and AICD datasets and show that:

1. Different resampling methods lead to markedly different performances for each learning algorithm, and there is no single sampler that dominates across all models and both datasets.
2. On both datasets, hybrids with aggressive under-sampling such as ClusterCentroids and, in some cases, NearMiss often achieve very high recall (sometimes close to 100% on AICD), but this typically comes at the cost of much lower precision and F1-scores.
3. The best F1-scores are generally obtained with moderate rebalancing strategies. On Carclaimtxt, SMOTE-type oversampling and hybrid samplers (e.g., SMOTE, ADASYN, SMOTEENN) work well for neural networks and random forests, whereas on AICD, RandomOverSampler, RandomUnderSampler, and cost-sensitive learning tend to provide the most favorable recall–F1 trade-offs for tree-based ensembles and boosting methods.
4. Across both datasets, AUC values remain relatively similar and often high, even for some hybrids with very low F1-scores. This confirms that AUC alone is not sufficient to distinguish between effective and ineffective fraud detection hybrids, especially under strong class imbalance.
5. Accuracy is high for most hybrids, particularly on Carclaimtxt, but can coexist with poor F1-scores. Therefore, recall and F1-score are more informative metrics than accuracy for imbalanced fraud detection.
6. Overall, the figures highlight that the choice of resampling strategy must be tailored jointly to the base model and the dataset; inappropriate resampling can substantially reduce fraud detection performance.

In addition, Tables 6 through 17, which report mean ± standard deviation of all metrics and training time, provide the following detailed insights:

1. For artificial neural networks, on Carclaimtxt the SMOTE-family oversamplers and hybrid methods (RandomOverSampler, SMOTE, ADASYN, BorderlineSMOTE, SMOTETomek, SMOTEENN) yield very similar and clearly better recall/F1 than the no-sampling baseline, while pure under-sampling offers slightly lower F1 but shorter training times. On AICD, RandomOverSampler, SMOTE, and ADASYN provide the best balance between recall and F1; SMOTEENN and other aggressive hybrids further increase recall but with a noticeable drop in F1 and only modest gains in AUC.
2. For random forest, on Carclaimtxt all resampling methods that rebalance the data (over-, under-, or hybrid sampling) substantially improve recall and F1 relative to no sampling, with SMOTEENN and ClusterCentroids providing particularly strong recall at the cost of additional training time. On AICD, random forest already performs well without resampling; moderate strategies such as cost-sensitive learning, RandomOverSampler, and RandomUnderSampler preserve high AUC and F1, whereas ClusterCentroids yields the highest recall and F1 but with reduced accuracy.
3. For XGBoost, on Carclaimtxt the F1-score improves dramatically when moving from no sampling to cost-sensitive learning or RandomOverSampler, while further oversampling mainly affects recall and leaves AUC relatively unchanged; SMOTEENN offers a more balanced recall–F1 compromise than purely over- or under-sampled variants. On AICD, XGBoost hybrids with cost-sensitive learning, RandomUnderSampler, NearMiss, and especially ClusterCentroids substantially improve F1 compared to the baseline, with ClusterCentroids achieving the best joint recall–F1 performance. In contrast, SMOTEENN attains extremely high recall but poor F1 and AUC, illustrating that recall alone can be misleading.

4. For CatBoost, on Carclaimtxt the behavior is similar to XGBoost: oversampling and hybrid resampling slightly improve recall and F1, while AUC values remain close across samplers. On AICD, however, CatBoost combined with RandomUnderSampler (and, to a lesser extent, RandomOverSampler, SMOTE, and ADASYN) yields very strong F1 and recall, making it one of the best-performing base learners on this dataset. Again, ClusterCentroids and SMOTEENN produce very high recall but substantially lower precision and F1.
5. For LightGBM, on Carclaimtxt cost-sensitive resampling gives the best F1-score, although the differences to RandomOverSampler and related oversampling schemes are small; all rebalancing methods significantly improve recall over the no-sampling baseline. On AICD, the pattern is close to CatBoost: cost-sensitive learning and moderate over-/under-sampling (RandomOverSampler, RandomUnderSampler, SMOTE/ADASYN) achieve high F1 and recall, whereas ClusterCentroids and SMOTEENN push recall to extreme values at the expense of F1 and, for SMOTEENN, AUC.
6. For AdaBoost, on Carclaimtxt the hybrids remain relatively weak compared with tree-based ensembles and gradient boosting, with low F1-scores despite acceptable AUC. In contrast, on AICD the picture changes: AdaBoost combined with cost-sensitive learning, RandomOverSampler, or RandomUnderSampler reaches the highest F1-scores among all tested hybrids, while SMOTEENN and ClusterCentroids again trade precision and AUC for near-perfect recall. This confirms that boosting can be very competitive on some fraud datasets when paired with appropriate imbalance-handling strategies.

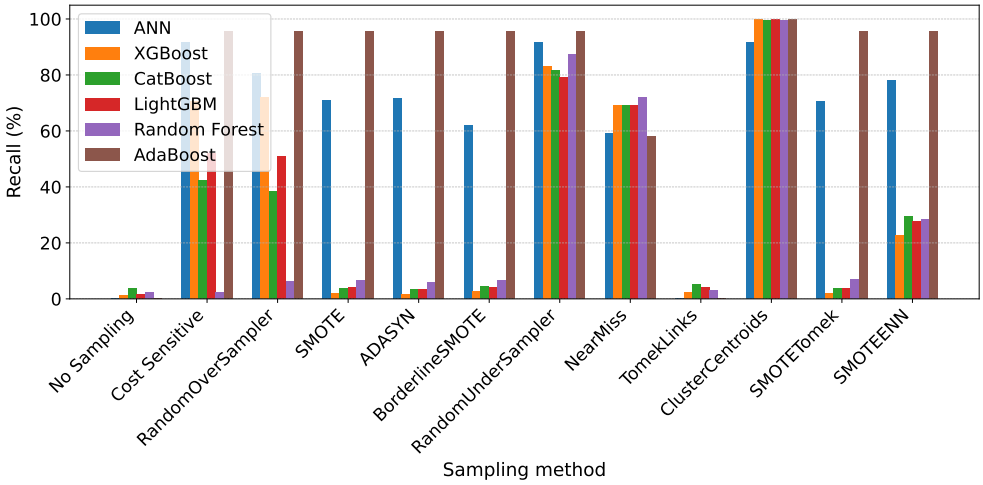

Figure 13: Mean recall (%) of fraud detection on the Carclaimtxt dataset using hybrids of different resampling methods and machine learning models.

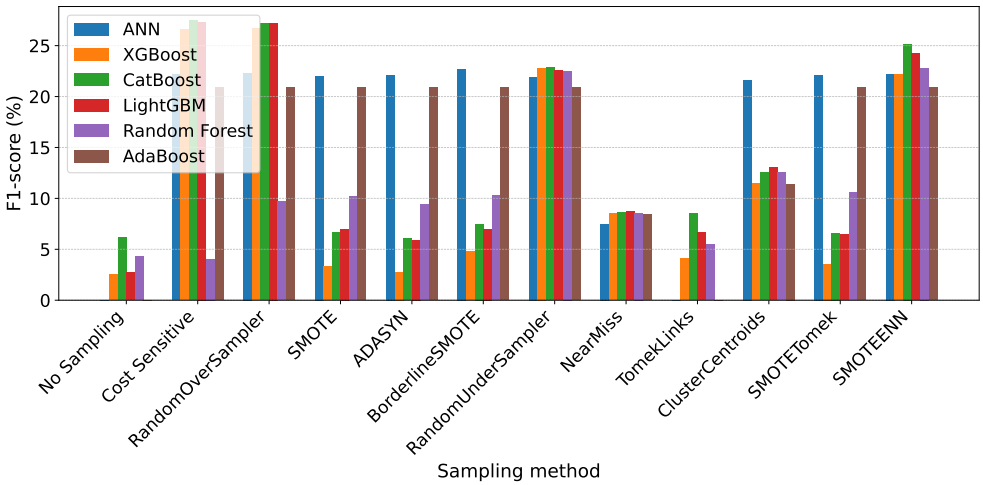

Figure 14: Mean F1-score (%) of fraud detection on the Carclaimtxt dataset using hybrids of different resampling methods and machine learning models.

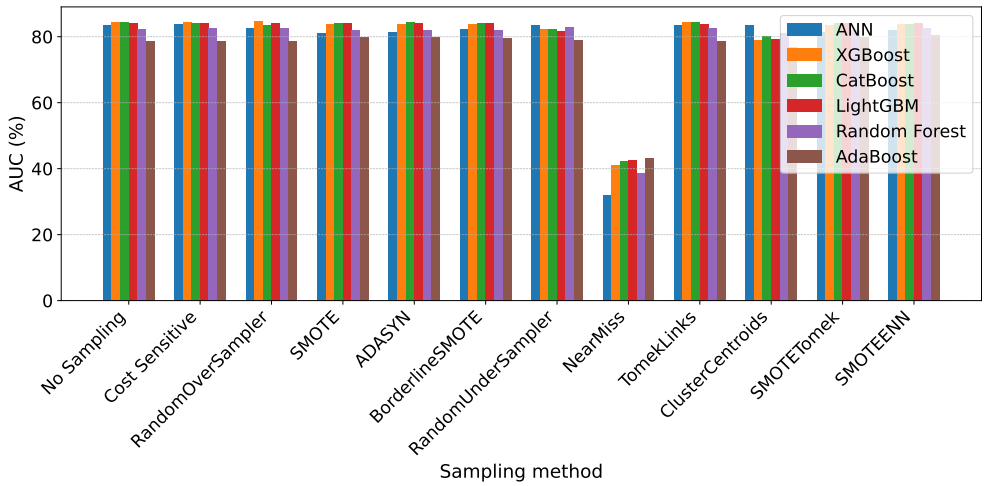

Figure 15: Mean AUC (%) of fraud detection on the Carclaimtxt dataset using hybrids of different resampling methods and machine learning models (15 repeated CV runs).

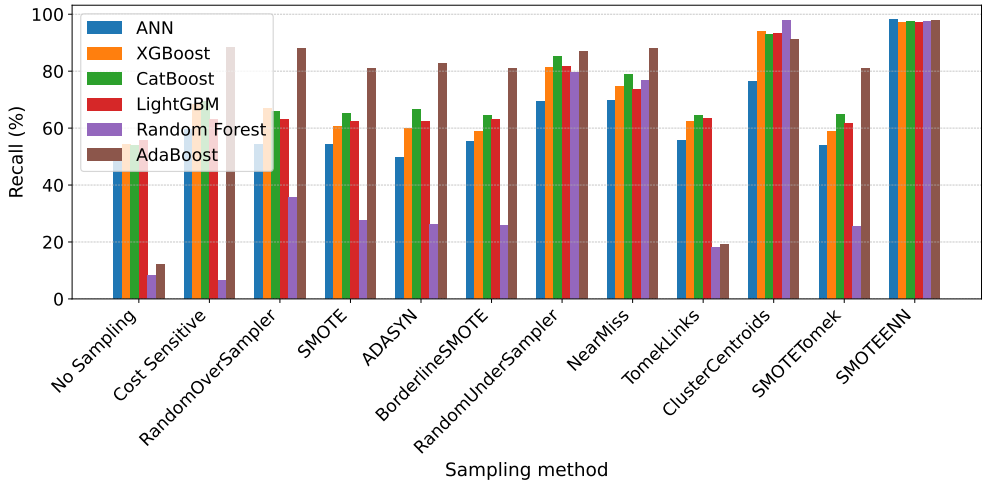

Figure 16: Mean recall (%) of fraud detection on the AICD dataset using hybrids of different resampling methods and machine learning models (15 repeated CV runs).

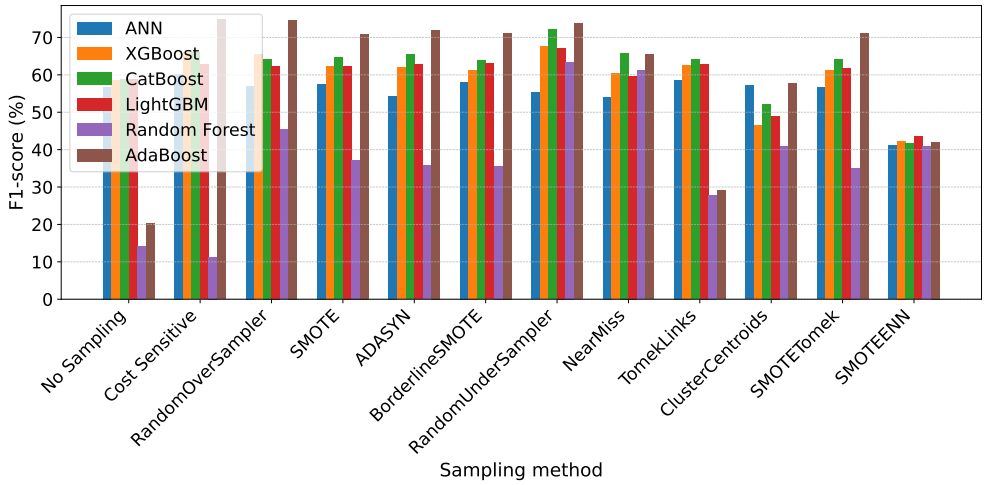

Figure 17: Mean F1-score (%) of fraud detection on the AICD dataset using hybrids of different resampling methods and machine learning models (15 repeated CV runs).

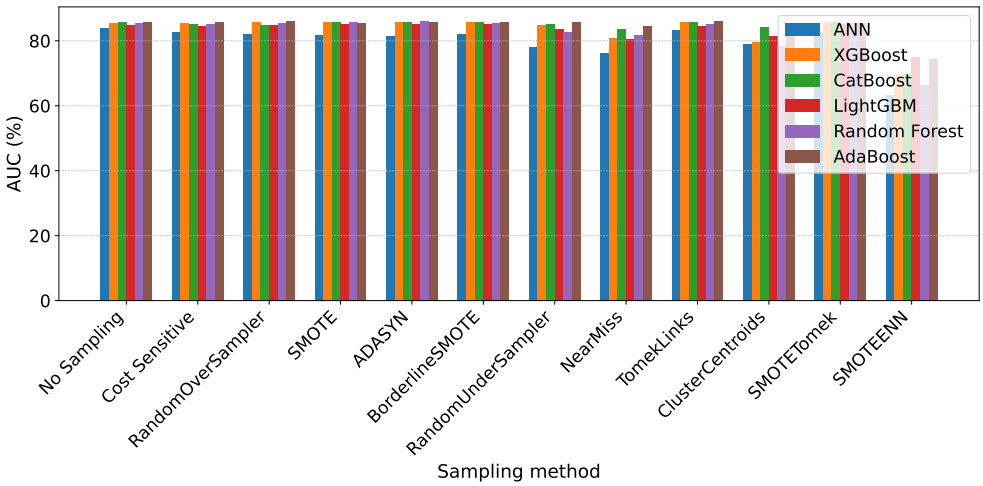

Figure 18: Mean AUC (%) of fraud detection on the AICD dataset using hybrids of different resampling methods and machine learning models (15 repeated CV runs).

Table 6: Fraud detection results on the Carclaimtxt dataset using hybrids of artificial neural networks [154] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.79±0.00	0.00±0.00	0.00±0.00	0.00±0.00	83.36±0.91	19.01±2.88	inf	16.55±1.05
Cost Sensitive [41]	66.43±1.85	12.60±0.48	91.47±3.46	22.14±0.74	83.73±0.97	19.20±2.12	6.95±0.31	17.48±1.61
RandomOverSampler [31]	70.62±1.85	12.94±0.66	80.73±4.03	22.29±1.04	82.60±1.25	17.14±1.26	6.75±0.40	28.46±0.81
SMOTE [31]	73.74±1.38	13.02±0.75	71.00±4.44	22.00±1.21	81.23±1.46	16.81±2.18	6.70±0.44	28.72±1.42
ADASYN [58]	73.68±1.60	13.09±0.75	71.67±4.10	22.13±1.18	81.28±1.20	16.81±1.44	6.66±0.44	28.34±2.40
BorderlineSMOTE [54]	77.93±1.65	13.90±1.25	62.13±7.00	22.69±2.03	82.38±1.48	17.92±3.06	6.25±0.69	31.42±8.75
RandomUnderSampler [34]	65.99±1.74	12.46±0.53	91.47±2.26	21.92±0.82	83.52±0.91	18.36±1.51	7.04±0.34	3.91±0.34
NearMiss [91]	23.86±2.94	3.98±0.24	59.00±5.39	7.46±0.45	32.02±1.23	3.53±0.06	24.19±1.56	3.96±0.42
ClusterCentroids [91]	65.38±0.94	12.26±0.37	91.60±2.90	21.62±0.63	83.40±1.08	18.67±2.05	7.16±0.25	4.06±0.44
TomekLinks [147]	94.79±0.00	0.00±0.00	0.00±0.00	0.00±0.00	83.50±1.02	19.04±2.60	inf	16.11±1.07
SMOTETomek [18]	74.07±1.54	13.13±0.93	70.60±4.58	22.13±1.48	81.29±1.50	16.83±2.19	6.65±0.56	27.78±1.43
SMOTEENN [19]	71.45±1.79	12.96±0.83	78.07±4.15	22.22±1.31	82.04±1.21	17.57±2.18	6.75±0.49	24.83±1.17

Table 7: Fraud detection results on the Carclaimtxt dataset using hybrids of random forest [25] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.62±0.12	31.95±10.27	2.33±0.82	4.31±1.46	82.26±1.66	18.40±1.66	2.52±1.44	1.79±0.32
Cost Sensitive [41]	94.61±0.10	28.25±9.20	2.20±0.94	4.06±1.68	82.45±1.59	18.58±1.74	2.94±1.39	1.69±0.29
RandomOverSampler [31]	93.90±0.19	21.31±4.12	6.33±1.80	9.71±2.43	82.59±1.58	17.81±1.63	3.85±0.89	2.53±0.35
SMOTE [31]	94.03±0.26	24.18±5.00	6.53±1.60	10.19±2.17	82.05±1.41	18.15±1.95	3.30±0.87	4.67±0.54
ADASYN [58]	94.09±0.26	23.77±6.34	5.93±2.05	9.40±2.95	82.09±1.48	18.20±1.87	3.57±1.50	4.56±0.55
BorderlineSMOTE [54]	94.01±0.35	24.26±7.57	6.60±1.88	10.28±2.83	82.07±1.49	18.57±1.82	3.78±2.47	4.62±0.49
RandomUnderSampler [34]	68.60±1.28	12.92±0.62	87.40±3.36	22.51±1.01	82.99±1.42	18.81±2.97	6.76±0.38	0.87±0.10
NearMiss [91]	20.10±1.50	4.56±0.18	71.87±3.48	8.57±0.34	38.56±2.05	4.15±0.28	20.97±0.88	0.88±0.12
ClusterCentroids [91]	26.91±9.24	6.70±0.73	99.33±1.11	12.54±1.30	81.11±1.38	13.84±1.19	14.11±1.81	0.79±0.09
TomekLinks [147]	94.52±0.15	28.04±9.29	3.07±1.28	5.49±2.22	82.49±1.34	18.58±1.67	3.29±2.84	1.66±0.21
SMOTETomek [18]	94.05±0.24	24.71±5.56	6.80±2.01	10.58±2.75	82.10±1.57	18.18±2.11	3.28±1.16	4.61±0.54
SMOTEENN [19]	89.95±0.76	19.09±2.62	28.47±3.93	22.79±2.87	82.74±1.47	18.85±2.03	4.35±0.86	4.15±0.53

Table 8: Fraud detection results on the Carclaimtxt dataset using hybrids of XGBoost [32] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.66±0.15	35.38±31.22	1.33±0.90	2.52±1.68	84.40±1.14	19.19±1.65	inf	0.16±0.02
Cost Sensitive [41]	79.65±0.84	16.38±0.65	70.67±3.02	26.59±0.98	84.50±1.08	19.43±1.91	5.11±0.24	0.17±0.02
RandomOverSampler [31]	79.35±0.71	16.38±0.63	72.13±3.07	26.70±0.97	84.79±1.04	20.22±2.26	5.11±0.24	0.35±0.26
SMOTE [31]	94.58±0.16	23.63±19.7	1.80±1.52	3.29±2.70	83.70±0.93	18.35±1.21	inf	0.27±0.01
ADASYN [58]	94.66±0.12	30.96±26.25	1.47±0.92	2.76±1.72	83.77±1.08	18.69±1.74	inf	0.34±0.27
BorderlineSMOTE [54]	94.54±0.16	26.80±13.3	2.67±1.35	4.80±2.38	83.69±1.04	18.71±1.62	inf	0.28±0.12
RandomUnderSampler [34]	70.66±1.22	13.22±0.63	83.13±4.03	22.81±1.05	82.39±1.34	16.71±2.10	6.58±0.36	0.06±0.01
NearMiss [91]	22.66±2.51	4.54±0.19	69.07±3.84	8.52±0.35	40.96±2.27	4.44±0.42	21.07±0.93	0.06±0.01
ClusterCentroids [91]	18.41±10.59	6.09±0.80	99.87±0.35	11.47±1.42	79.10±2.20	14.77±1.95	15.68±2.05	0.10±0.12
TomekLinks [147]	94.60±0.20	32.49±20.0	2.20±1.08	4.07±2.00	84.50±1.22	19.98±2.00	3.58±3.27	0.21±0.15
SMOTETomek [18]	94.63±0.15	30.37±20.50	1.93±1.44	3.56±2.57	83.59±0.89	18.36±1.46	inf	0.32±0.13
SMOTEENN [19]	91.72±0.58	21.84±3.06	22.60±3.36	22.15±2.94	83.94±0.80	19.21±1.49	3.67±0.74	0.32±0.25

Table 9: Fraud detection results on the Carclaimtxt dataset using hybrids of CatBoost [118] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.35±0.21	23.25±7.39	3.60±1.84	6.14±2.99	84.38±1.04	19.45±1.74	3.77±1.66	1.02±0.26
Cost Sensitive [41]	88.33±0.63	20.34±1.96	42.40±4.55	27.47±2.64	84.14±1.11	20.59±2.30	3.96±0.49	1.14±0.35
RandomOverSampler [31]	89.22±0.66	21.00±2.00	38.53±4.10	27.15±2.51	83.50±1.21	19.69±2.07	3.80±0.47	1.92±0.39
SMOTE [31]	94.38±0.21	25.47±9.70	3.87±1.68	6.65±2.78	84.12±1.26	19.87±2.08	3.59±2.12	2.61±0.58
ADASYN [58]	94.33±0.18	21.96±8.04	3.53±1.60	6.04±2.65	84.39±1.26	19.56±1.87	4.11±1.86	2.64±0.63
BorderlineSMOTE [54]	94.27±0.28	24.99±9.06	4.47±1.85	7.47±2.95	84.08±1.08	19.71±2.28	3.56±1.84	2.70±0.62
RandomUnderSampler [34]	71.32±1.11	13.30±0.56	81.53±4.17	22.86±0.94	82.42±1.35	16.72±2.12	6.53±0.32	0.33±0.02
NearMiss [91]	23.72±2.56	4.60±0.27	69.00±3.87	8.62±0.51	42.40±1.96	4.72±0.50	20.82±1.31	0.37±0.12
ClusterCentroids [91]	26.83±7.79	6.68±0.63	99.60±0.63	12.52±1.10	80.26±1.74	15.69±2.05	14.09±1.52	0.96±0.32
TomekLinks [147]	94.07±0.28	22.17±5.61	5.33±1.80	8.50±2.67	84.46±1.41	19.84±1.90	3.79±1.24	1.02±0.28
SMOTETomek [18]	94.34±0.20	23.73±10.81	3.87±1.92	6.59±3.18	84.06±1.05	19.60±1.89	4.11±2.48	2.70±0.62
SMOTEENN [19]	90.91±0.51	22.07±2.61	29.33±3.90	25.15±2.98	83.95±0.98	19.25±1.85	3.59±0.54	2.68±0.77

Table 10: Fraud detection results on the Carclaimtxt dataset using hybrids of LightGBM [72] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.45±0.10	13.97±9.63	1.53±1.41	2.73±2.43	83.99±1.59	18.59±1.89	inf	0.31±0.02
Cost Sensitive [41]	85.61±0.78	18.55±1.60	51.87±5.00	27.31±2.33	84.12±1.63	18.66±2.05	4.43±0.49	0.47±0.43
RandomOverSampler [31]	85.79±1.15	18.62±1.70	50.87±4.45	27.22±2.21	84.25±1.38	19.29±2.29	4.41±0.51	0.57±0.33
SMOTE [31]	94.41±0.21	27.10±12.6	4.07±1.94	7.00±3.24	84.18±1.24	18.67±1.57	3.35±1.68	0.85±0.31
ADASYN [58]	94.40±0.23	24.94±13.22	3.40±1.50	5.92±2.58	84.17±1.13	18.71±1.79	3.93±2.17	0.84±0.35
BorderlineSMOTE [54]	94.33±0.22	24.79±10.1	4.13±1.92	7.01±3.12	84.21±1.22	18.95±1.55	4.27±3.81	0.84±0.37
RandomUnderSampler [34]	71.63±1.18	13.16±0.75	79.27±3.83	22.57±1.22	81.69±1.65	16.80±2.53	6.62±0.44	0.14±0.01
NearMiss [91]	24.27±1.26	4.64±0.25	69.20±4.28	8.69±0.46	42.49±2.00	4.74±0.57	20.62±1.18	0.20±0.14
ClusterCentroids [91]	29.90±7.45	6.97±0.60	99.93±0.26	13.03±1.05	79.33±1.68	14.42±1.62	13.46±1.46	0.25±0.09
TomekLinks [147]	94.30±0.17	22.90±5.97	3.93±1.53	6.65±2.43	83.74±1.59	18.28±2.17	3.66±1.32	0.39±0.29
SMOTETomek [18]	94.42±0.23	26.89±12.40	3.73±1.71	6.48±2.84	84.25±1.18	18.93±1.57	3.70±2.91	0.94±0.46
SMOTEENN [19]	90.91±0.62	21.48±3.04	27.87±4.14	24.21±3.30	84.12±1.09	19.39±1.93	3.76±0.78	0.75±0.08

Table 11: Fraud detection results on the Carclaimtxt dataset using hybrids of AdaBoost [47] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	94.79±0.00	0.00±0.00	0.00±0.00	0.00±0.00	78.62±0.70	11.52±0.31	inf	1.73±0.18
Cost Sensitive [41]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	78.62±0.70	11.52±0.31	7.54±0.21	1.80±0.29
RandomOverSampler [31]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	78.62±0.70	11.52±0.31	7.54±0.21	2.56±0.27
SMOTE [31]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	79.79±1.45	12.47±0.88	7.54±0.21	4.43±0.32
ADASYN [58]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	79.75±1.25	12.49±0.73	7.54±0.21	4.40±0.31
BorderlineSMOTE [54]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	79.55±1.76	12.45±0.98	7.54±0.21	4.34±0.33
RandomUnderSampler [34]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	78.82±0.83	11.62±0.41	7.54±0.21	0.73±0.11
NearMiss [91]	34.32±2.53	4.54±0.30	57.93±4.45	8.42±0.56	43.14±1.86	4.50±0.18	21.12±1.53	0.75±0.13
ClusterCentroids [91]	18.88±5.28	6.05±0.38	99.73±0.46	11.40±0.68	76.72±3.66	13.18±1.98	15.60±1.02	0.91±0.10
TomekLinks [147]	94.79±0.00	0.00±0.00	0.00±0.00	0.00±0.00	78.62±0.70	11.52±0.31	inf	1.72±0.22
SMOTETomek [18]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	79.85±1.32	12.52±0.80	7.54±0.21	4.46±0.33
SMOTEENN [19]	62.19±0.98	11.71±0.28	95.60±1.92	20.87±0.47	80.43±1.21	12.73±0.78	7.54±0.21	4.10±0.27

Table 12: Fraud detection results on the AICD dataset using hybrids of artificial neural networks [154] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	81.47±2.47	66.84±6.27	49.95±10.08	56.69±7.53	83.82±1.72	60.90±4.85	0.51±0.14	3.83±0.48
Cost Sensitive [41]	80.77±2.33	61.89±5.47	59.65±9.60	60.24±5.62	82.57±1.93	59.72±4.38	0.63±0.15	3.83±0.50
RandomOverSampler [31]	79.93±2.09	60.60±4.88	54.41±8.32	57.01±5.81	82.06±2.26	58.80±3.79	0.66±0.14	4.67±0.57
SMOTE [31]	80.43±1.84	62.30±5.45	54.39±9.10	57.54±5.78	81.86±1.78	58.72±3.63	0.62±0.14	4.63±0.42
ADASYN [58]	80.03±2.27	62.83±6.99	49.67±12.16	54.39±8.44	81.46±2.28	57.96±4.01	0.61±0.17	4.63±0.53
BorderlineSMOTE [54]	80.43±2.53	62.04±6.80	55.35±7.95	58.14±5.78	81.93±2.05	58.64±4.41	0.63±0.18	4.55±0.39
RandomUnderSampler [34]	72.43±2.81	46.36±2.94	69.34±8.12	55.35±3.19	77.97±2.10	52.96±5.05	1.17±0.14	3.52±0.49
NearMiss [91]	70.80±4.10	44.69±4.00	69.62±8.53	54.11±3.47	76.33±2.95	51.82±3.06	1.26±0.22	3.44±0.62
ClusterCentroids [91]	71.70±2.97	45.89±3.07	76.51±8.54	57.14±3.11	79.13±2.56	54.37±4.40	1.19±0.15	3.49±0.61
TomekLinks [147]	80.87±2.26	63.23±5.85	55.74±10.0	58.66±6.13	83.40±1.61	60.75±4.74	0.59±0.15	3.90±0.26
SMOTETomek [18]	80.03±1.87	61.65±6.08	53.86±10.48	56.68±5.94	82.29±2.21	59.26±4.32	0.64±0.15	4.62±0.38
SMOTEENN [19]	30.57±3.37	26.05±1.01	98.25±2.38	41.17±1.26	63.39±4.04	35.71±4.00	2.84±0.14	3.71±0.36

Table 13: Fraud detection results on the AICD dataset using hybrids of random forest [25] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	75.60±0.69	52.64±9.77	8.36±3.41	14.28±5.30	85.61±2.44	60.22±3.85	0.98±0.46	1.04±0.15
Cost Sensitive [41]	75.17±1.28	45.78±21.2	6.49±3.55	11.23±5.83	85.30±2.41	58.55±4.17	inf	1.01±0.13
RandomOverSampler [31]	79.10±1.44	64.17±5.39	35.64±7.34	45.35±6.17	85.34±2.22	59.87±4.46	0.57±0.13	1.21±0.23
SMOTE [31]	77.60±2.21	60.21±9.05	27.40±8.19	37.18±8.80	85.70±2.32	60.50±5.00	0.69±0.24	1.42±0.29
ADASYN [58]	77.33±2.08	59.23±8.32	26.19±8.17	35.81±8.79	86.09±2.65	60.76±4.34	0.71±0.21	1.39±0.28
BorderlineSMOTE [54]	77.23±1.79	58.87±8.98	25.90±6.71	35.63±7.49	85.43±2.66	59.45±4.87	0.74±0.27	1.37±0.20
RandomUnderSampler [34]	77.37±3.63	53.06±4.93	79.49±7.25	63.49±4.89	82.82±3.04	57.55±5.29	0.90±0.19	0.82±0.15
NearMiss [91]	75.90±3.55	51.11±4.79	76.81±7.72	61.17±4.75	81.88±3.21	57.03±4.37	0.97±0.19	0.80±0.12
ClusterCentroids [91]	29.93±1.91	25.80±0.60	97.84±2.90	40.82±0.91	78.33±2.88	54.56±4.26	2.88±0.09	0.83±0.10
TomekLinks [147]	77.03±1.19	62.88±8.93	18.22±4.65	27.89±5.72	85.27±2.42	60.74±4.02	0.62±0.22	0.97±0.15
SMOTETomek [18]	77.27±1.80	58.98±7.21	25.37±7.41	35.09±8.03	85.56±2.55	60.72±4.17	0.72±0.20	1.40±0.29
SMOTEENN [19]	30.13±4.32	25.84±1.15	97.44±3.83	40.83±1.44	66.25±3.95	37.40±4.15	2.88±0.16	0.94±0.15

Table 14: Fraud detection results on the AICD dataset using hybrids of XGBoost [32] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	81.47±2.62	64.81±4.88	54.29±12.55	58.49±8.62	85.40±2.65	61.70±4.92	0.55±0.12	0.31±0.28
Cost Sensitive [41]	82.70±2.58	64.14±5.19	68.44±8.87	65.99±5.70	85.52±2.83	62.24±5.86	0.57±0.13	0.21±0.01
RandomOverSampler [31]	82.70±1.89	64.83±4.46	66.82±8.12	65.46±4.32	85.69±2.65	62.26±5.24	0.55±0.11	0.22±0.02
SMOTE [31]	82.20±2.53	65.14±5.46	60.59±9.94	62.41±6.62	85.83±2.75	61.25±5.77	0.55±0.14	0.65±0.33
ADASYN [58]	82.20±2.46	65.21±4.89	59.95±12.25	61.94±7.61	85.73±2.93	61.54±5.38	0.54±0.11	0.65±0.31
BorderlineSMOTE [54]	81.87±2.48	64.73±5.18	58.87±10.2	61.25±6.86	85.65±2.85	61.76±6.58	0.55±0.13	0.65±0.37
RandomUnderSampler [34]	80.70±2.51	58.10±4.22	81.37±7.86	67.52±3.82	84.83±2.80	61.56±5.47	0.73±0.13	0.14±0.01
NearMiss [91]	75.77±3.57	51.02±5.54	74.75±7.98	60.40±5.00	80.97±3.77	57.69±4.75	0.98±0.21	0.15±0.01
ClusterCentroids [91]	46.40±4.84	30.94±1.88	94.05±3.65	46.52±2.10	79.54±2.42	54.22±2.82	2.24±0.20	0.30±0.19
TomekLinks [147]	81.90±1.51	63.83±3.37	62.24±9.03	62.68±4.60	85.78±2.40	61.65±3.64	0.57±0.08	0.21±0.07
SMOTETomek [18]	81.83±2.70	64.70±5.95	58.86±11.08	61.14±7.35	85.80±2.76	61.16±5.66	0.56±0.15	0.65±0.34
SMOTEENN [19]	34.43±3.40	27.02±1.15	97.04±2.39	42.26±1.47	69.53±4.59	41.87±6.29	2.71±0.15	0.20±0.02

Table 15: Fraud detection results on the AICD dataset using hybrids of CatBoost [118] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	81.80±2.19	66.28±6.07	53.81±10.15	58.93±7.43	85.86±2.34	62.54±4.24	0.52±0.15	1.47±0.43
Cost Sensitive [41]	82.50±2.09	63.50±4.49	69.23±8.57	65.98±4.91	85.22±2.71	60.71±6.25	0.58±0.11	1.46±0.41
RandomOverSampler [31]	82.00±1.73	63.25±4.41	65.99±7.50	64.30±3.98	84.94±2.66	59.37±6.25	0.59±0.11	1.58±0.49
SMOTE [31]	82.47±2.16	64.72±5.29	65.19±8.12	64.61±4.70	85.91±2.97	61.01±6.33	0.55±0.12	4.28±0.54
ADASYN [58]	82.87±2.74	65.08±5.52	66.56±10.50	65.46±6.64	85.69±2.90	60.36±5.86	0.55±0.14	4.27±0.60
BorderlineSMOTE [54]	82.17±2.42	63.95±5.29	64.52±9.66	63.87±5.93	85.65±2.49	60.39±5.84	0.57±0.13	4.25±0.50
RandomUnderSampler [34]	83.73±2.03	62.87±3.95	85.02±6.01	72.09±2.89	85.23±3.17	61.57±6.33	0.60±0.10	1.38±0.42
NearMiss [91]	79.87±2.89	56.88±4.77	78.81±7.97	65.87±4.75	83.51±3.29	59.02±5.98	0.77±0.15	1.39±0.47
ClusterCentroids [91]	57.57±6.33	36.44±3.44	92.98±3.51	52.22±3.19	84.15±2.41	60.53±4.64	1.77±0.26	2.08±0.41
TomekLinks [147]	82.30±1.51	64.41±4.19	64.35±7.51	64.08±4.01	85.82±1.93	61.04±3.87	0.56±0.10	1.44±0.29
SMOTETomek [18]	82.30±2.21	64.29±4.57	64.80±8.15	64.23±5.02	85.77±2.67	60.19±5.55	0.56±0.12	4.35±0.58
SMOTEENN [19]	32.20±3.90	26.44±1.17	97.58±2.43	41.59±1.41	69.69±4.92	42.56±7.73	2.79±0.16	4.22±0.51

Table 16: Fraud detection results on the AICD dataset using hybrids of LightGBM [72] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	80.97±1.54	63.25±4.26	55.77±8.26	58.90±4.67	84.71±2.47	60.44±3.53	0.59±0.10	0.37±0.02
Cost Sensitive [41]	81.70±1.31	63.22±3.21	62.90±7.64	62.76±3.65	84.57±2.70	60.58±2.60	0.59±0.08	0.47±0.33
RandomOverSampler [31]	81.43±1.44	62.51±3.59	62.88±8.07	62.38±4.36	84.73±2.65	61.23±3.06	0.60±0.09	0.46±0.02
SMOTE [31]	81.70±1.79	63.31±3.60	62.22±9.96	62.36±5.35	85.28±2.37	60.34±3.72	0.58±0.09	1.18±0.52
ADASYN [58]	81.97±2.12	63.85±4.32	62.48±10.35	62.78±6.13	85.05±2.53	59.80±3.82	0.57±0.11	1.02±0.39
BorderlineSMOTE [54]	82.03±2.04	63.78±3.46	63.17±11.3	63.03±6.36	85.25±2.88	60.59±4.88	0.57±0.09	1.05±0.31
RandomUnderSampler [34]	80.43±2.44	57.46±3.74	81.52±8.15	67.22±4.29	83.69±3.25	58.63±5.63	0.75±0.11	0.26±0.29
NearMiss [91]	75.43±3.54	50.69±5.07	73.44±6.71	59.71±3.90	80.61±3.22	56.16±4.95	0.99±0.20	0.15±0.01
ClusterCentroids [91]	51.70±6.57	33.36±3.01	93.12±3.66	48.99±2.85	81.56±3.20	56.35±5.70	2.02±0.26	0.32±0.03
TomekLinks [147]	81.83±1.37	63.36±2.79	63.31±10.3	62.88±5.24	84.59±2.19	59.68±2.64	0.58±0.07	0.48±0.24
SMOTETomek [18]	81.47±1.68	62.92±3.48	61.55±9.79	61.82±5.13	85.14±2.56	59.68±3.89	0.59±0.09	1.17±0.54
SMOTEENN [19]	37.37±4.30	27.99±1.67	97.17±2.11	43.45±2.11	74.85±5.04	48.23±7.69	2.58±0.21	0.63±0.36

Table 17: Fraud detection results on the AICD dataset using hybrids of AdaBoost [47] and different sampling methods.

Resampling methods	Accuracy	Precision	Recall	F1 Score	AUC	PR-AUC	FP/TP	Time (s)
No Sampling	77.10±0.85	73.86±11.04	12.01±4.55	20.23±6.71	85.73±2.08	61.34±2.55	0.38±0.19	1.31±0.15
Cost Sensitive [41]	85.30±1.58	65.09±3.34	88.25±4.70	74.79±2.30	85.78±1.94	61.81±3.09	0.54±0.08	1.35±0.19
RandomOverSampler [31]	85.23±1.59	65.01±3.30	87.98±5.32	74.63±2.51	85.93±1.71	62.42±3.32	0.54±0.08	1.63±0.24
SMOTE [31]	83.63±1.64	63.38±3.42	80.99±6.42	70.93±2.84	85.58±1.91	60.04±3.58	0.58±0.08	2.35±0.22
ADASYN [58]	84.13±1.33	63.98±3.01	82.89±6.40	72.03±2.26	85.92±2.06	60.90±3.77	0.57±0.07	2.34±0.27
BorderlineSMOTE [54]	83.77±1.59	63.62±3.38	81.13±5.84	71.15±2.59	85.88±1.99	60.69±3.42	0.58±0.08	2.35±0.30
RandomUnderSampler [34]	84.73±2.05	64.31±3.91	87.03±6.25	73.78±3.25	85.89±2.73	62.77±4.30	0.56±0.10	1.04±0.17
NearMiss [91]	77.03±2.26	52.24±3.01	87.85±5.07	65.42±2.64	84.41±3.51	59.51±4.78	0.92±0.11	0.99±0.11
ClusterCentroids [91]	66.23±7.75	42.58±6.55	91.09±3.76	57.67±5.40	84.42±2.62	61.78±5.69	1.40±0.34	1.20±0.10
TomekLinks [147]	78.03±1.25	71.74±7.03	19.06±8.60	29.11±9.48	86.14±1.87	61.89±2.89	0.41±0.14	1.31±0.17
SMOTETomek [18]	83.67±1.63	63.42±3.39	81.12±6.72	71.00±2.90	85.56±1.92	60.16±3.83	0.58±0.08	2.37±0.23
SMOTEENN [19]	33.27±3.82	26.81±1.19	97.98±2.28	42.08±1.44	74.48±7.74	48.73±10.25	2.74±0.16	1.81±0.21

6. Conclusions and Future Directions

This paper has explored various techniques for detecting fraud, particularly within the auto insurance industry. One of the main challenges in fraud detection is dealing with imbalanced data. Proposed techniques such as resampling and cost-sensitive models have shown promising results in improving the performance of machine learning models. Additionally, ensemble learning and tree-based models have demonstrated an excellent ability to handle imbalanced datasets. This survey also emphasizes the importance of feature extraction and feature selection in improving machine learning models for fraud detection.

In the future, explainable machine learning techniques can play a significant role in auto insurance fraud detection. These methods aim to provide clarity and interpretability in the insurance industry, which is crucial in fraud detection applications. They enable insurance companies to understand the reasons for fraudulent claims better and take appropriate measures to prevent them. Furthermore, it's essential to address the problem of concept drift in this field. Collecting more multi-feature data on auto insurance fraud is vital to improving machine learning models to handle complex fraud scenarios.

It's important to note that the methods of insurance fraud are constantly evolving. Therefore, more advanced artificial intelligence techniques, especially machine learning methods, can be employed to detect fraud. Some of these methods include:

1. Using deep learning methods to analyze images sent to the insurance company, extracting essential car information, identifying any damage to the vehicle, and checking for previous claims related to the damage.
2. Employing blockchain to protect drivers' privacy and automatically calculate insurance premiums based on encrypted driving data, with audits to detect data fraud and penalize malicious drivers [63]. To see the prediction of blockchain effects on insurance fraud detection in recent years, one can refer to [68, 70, 86, 93, 124, 127, 128, 132].
3. Implementing a smartphone-based insurance system to link the car insurance system to the driver's profile [56].
4. Developing a web-based automatic claim estimator that uses user photographs to process claims and automatically determine the position and degree of car damage.
5. Preparing for the future of automated vehicles by shifting the responsibility from drivers to automated driving systems and considering the insurance model for semi-automated and autonomous vehicles in future works [74]. Thus, it is important to note the insurance model of semi-automated and autonomous vehicles in future works.

References

[1] Y. Abakarim, M. Lahby, and A. Attioui, *A bagged ensemble convolutional neural networks approach to recognize insurance claim frauds*, Applied System Innovation, 6 (2023), p. 20.

[2] A. Abdallah, M. A. Maarof, and A. Zainal, *Fraud detection system: A survey*, Journal of Network and Computer Applications, 68 (2016), pp. 90–113.

[3] S. Abdelhadi, K. Elbahnasy, and M. Abdelsalam, *A proposed model to predict auto insurance claims using machine learning techniques*, Journal of Theoretical and Applied Information Technology, 98 (2020).

[4] B. Abma, *Evaluation of requirements management tools with support for traceability-based change impact analysis*, Master's thesis, University of Twente, Enschede, (2009).

[5] O. S. Adebayo, T. A. Favour-Bethy, O. Otasowie, and O. A. Okunola, *Comparative review of credit card fraud detection using machine learning and concept drift techniques*, International Journal of Computer Science and Mobile Computing, 12 (2023), pp. 24–48.

[6] H. Ahmad, B. Kasasbeh, B. Aldabaybah, and E. Rawashdeh, *Class balancing framework for credit card fraud detection based on clustering and similarity-based selection (sbs)*, International Journal of Information Technology, 15 (2023), pp. 325–333.

[7] K. I. Al-Daoud and I. A. Abu-AlSondos, *Robust ai for financial fraud detection in the gcc: A hybrid framework for imbalance, drift, and adversarial threats*, Journal of Theoretical and Applied Electronic Commerce Research, 20 (2025), p. 121.

[8] K. G. Al-Hashedi and P. Magalingam, *Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019*, Computer Science Review, 40 (2021), p. 100402.

[9] H. M. Al Lawati, A. Zainal, B. A. S. Al-Rimy, M. Al-Azawi, M. N. Kassim, S. A. Almalki, and T. A. Alghamdi, *An integrated preprocessing and drift detection approach with adaptive windowing for fraud detection in payment systems*, IEEE Access, (2025).

[10] A. A. Z. Alabdeen, *Adaptive anomaly based fraud detection model for handling concept drift in short-term profile*, PhD's thesis, Universiti Teknologi Malaysia, (2018).

[11] A. Ali, S. Abd Razak, S. H. Othman, T. A. E. Eisa, A. Al-Dhaqm, M. Nasser, T. Elhassan, H. Elshafie, and A. Saif, *Financial fraud detection based on machine learning: a systematic literature review*, Applied Sciences, 12 (2022), p. 9637.

[12] A. A. Amponsah, A. F. Adekoya, and B. A. Weyori, *A novel fraud detection and prevention method for healthcare claim processing using machine learning and blockchain technology*, Decision Analytics Journal, 4 (2022), p. 100122.

[13] P. P. Angelov, E. A. Soares, R. Jiang, N. I. Arnold, and P. M. Atkinson, *Explainable artificial intelligence: an analytical review*, WIREs Data Mining and Knowledge Discovery, 11 (2021), p. e1424.

[14] J. M. Arockiam and A. C. S. Pushpanathan, *Mapreduce-iterative support vector machine classifier: novel fraud detection systems in healthcare insurance industry*, International Journal of Electrical and Computer Engineering (IJECE), 13 (2023), p. 756.

[15] F. Aslam, A. I. Hunjra, Z. Ftiti, W. Louhichi, and T. Shams, *Insurance fraud detection: Evidence from artificial intelligence and machine learning*, Research in International Business and Finance, 62 (2022), p. 101744.

[16] S. S. Asrori, L. Wang, and S. Ozawa, *Permissioned blockchain-based xgboost for multi banks fraud detection*, in International Conference on Neural Information Processing, Springer, 2022, pp. 683–692.

[17] T. Badriyah, L. Rahmaniah, and I. Syarif, *Nearest neighbour and statistics method based for detecting fraud in auto insurance*, in 2018 International Conference on Applied Engineering (ICAE), IEEE, 2018, pp. 1–5.

[18] G. E. Batista, A. L. Bazzan, M. C. Monard, et al., *Balancing training data for automated annotation of keywords: a case study.*, in WOB, 2003, pp. 10–18.

[19] G. E. Batista, R. C. Prati, and M. C. Monard, *A study of the behavior of several methods for balancing machine learning training data*, ACM SIGKDD explorations newsletter, 6 (2004), pp. 20–29.

[20] B. Benedek, C. Ciumas, and B. Z. Nagy, *On the cost-efficiency of automobile insurance fraud detection methods: A meta-analysis*, Global Business Review, (2023), p. 09721509231158194.

[21] B. Benedek and B. Z. Nagy, *Traditional versus ai-based fraud detection: Cost efficiency in the field of automobile insurance*, Financial and Economic Review, 22 (2023), pp. 77–98.

[22] R. Bhowmik, *Detecting auto insurance fraud by data mining techniques*, Journal of Emerging Trends in Computing and Information Sciences, 2 (2011), pp. 156–162.

[23] A. Bodaghi and B. Teimourpour, *Automobile insurance fraud detection using social network analysis*, Applications of Data Management and Analysis: Case Studies in Social Networks and Beyond, (2018), pp. 11–16.

[24] P. Boulieris, J. Pavlopoulos, A. Xenos, and V. Vassalos, *Fraud detection with natural language processing*, Machine Learning, (2023), pp. 1–22.

[25] L. Breiman, *Random forests*, Machine learning, 45 (2001), pp. 5–32.

[26] L. Breiman, J. Friedman, C. J. Stone, and R. Olshen, *Classification and regression trees*, Routledge, 1 ed., 1984.

[27] A. Calafato, C. Colombo, and G. J. Pace, *A controlled natural language for tax fraud detection*, in Controlled Natural Language: 5th International Workshop, CNL 2016, Aberdeen, UK, July 25-27, 2016, Proceedings 5, Springer, 2016, pp. 1–12.

[28] Y. Cao, Y. Ma, Y. Zhu, and K. M. Ting, *Revisiting streaming anomaly detection: benchmark and evaluation*, Artificial Intelligence Review, 58 (2024), p. 8.

[29] V. Chandola, A. Banerjee, and V. Kumar, *Anomaly detection: A survey*, ACM computing surveys (CSUR), 41 (2009), pp. 1–58.

[30] J.-W. Chang, N. Yen, and J. C. Hung, *Design of a nlp-empowered finance fraud awareness model: the anti-fraud chatbot for fraud detection and fraud classification as an instance*, Journal of Ambient Intelligence and Humanized Computing, 13 (2022), pp. 4663–4679.

[31] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, *Smote: synthetic minority over-sampling technique*, Journal of artificial intelligence research, 16 (2002), pp. 321–357.

[32] T. Chen and C. Guestrin, *Xgboost: A scalable tree boosting system*, in Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.

[33] A. Cherif, A. Badhib, H. Ammar, S. Alshehri, M. Kalkatawi, and A. Imine, *Credit card fraud detection in the era of disruptive technologies: A systematic review*, Journal of King Saud University-Computer and Information Sciences, (2022).

[34] S. Choirunnisa and J. Lianto, *Hybrid method of undersampling and oversampling for handling imbalanced data*, in 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), IEEE, 2018, pp. 276–280.

[35] K. Chowdhary and K. Chowdhary, *Natural language processing*, Fundamentals of artificial intelligence, (2020), pp. 603–649.

[36] C. Cortes and V. Vapnik, *Support-vector networks*, Machine learning, 20 (1995), pp. 273–297.

[37] Y. Cui, X. Han, J. Chen, X. Zhang, J. Yang, and X. Zhang, *Fraudgnn-rl: a graph neural network with reinforcement learning for adaptive financial fraud detection*, IEEE Open Journal of the Computer Society, (2025).

[38] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, *Credit card fraud detection and concept-drift adaptation with delayed supervised information*, in 2015 international joint conference on Neural networks (IJCNN), IEEE, 2015, pp. 1–8.

[39] Z. Deng, G. Xin, Y. Liu, W. Wang, and B. Wang, *Contrastive graph neural network-based camouflaged fraud detector*, Information Sciences, 618 (2022), pp. 39–52.

[40] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, et al., *Explainable ai (xai): Core ideas, techniques, and solutions*, ACM Computing Surveys, 55 (2023), pp. 1–33.

[41] C. Elkan, *The foundations of cost-sensitive learning*, in International joint conference on artificial intelligence, vol. 17, Lawrence Erlbaum Associates Ltd, 2001, pp. 973–978.

[42] I. Eweoya, A. Adebiyi, A. Azeta, and O. Okesola, *Fraud prediction in bank credit administration: A systematic literature review*, Journal of Theoretical and Applied Information Technology, (2019).

[43] H. Fanai and H. Abbasimehr, *A novel combined approach based on deep autoencoder and deep classifiers for credit card fraud detection*, Expert Systems with Applications, 217 (2023), p. 119562.

[44] H. Farbmacher, L. Löw, and M. Spindler, *An explainable attention network for fraud detection in claims management*, Journal of Econometrics, 228 (2022), pp. 244–258.

[45] J. Fernandez Rodriguez, *A natural language processing approach to fraud detection*, Master's thesis, Dipartimento di Elettronica Informazione e Bioingegneria, Politecnico di Milano, (2020).

[46] U. Fiore, A. De Santis, F. Perla, P. Zanetti, and F. Palmieri, *Using generative adversarial networks for improving classification effectiveness in credit card fraud detection*, Information Sciences, 479 (2019), pp. 448–455.

[47] Y. Freund and R. E. Schapire, *A desicion-theoretic generalization of on-line learning and an application to boosting*, in European conference on computational learning theory, Springer, 1995, pp. 23–37.

[48] P. Fukas, J. Rebstadt, L. Menzel, and O. Thomas, *Towards explainable artificial intelligence in financial fraud detection: Using shapley additive explanations to explore feature importance*, in Advanced Information Systems Engineering, X. Franch, G. Poels, F. Gailly, and M. Snoeck, eds., Cham, 2022, Springer International Publishing, pp. 109–126.

[49] D. Gaspar, P. Silva, and C. Silva, *Explainable ai for intrusion detection systems: Lime and shap applicability on multi-layer perceptron*, IEEE Access, (2024).

[50] V. Gonzalez, *Evaluating interpretable models for financial fraud detection*, in AMCIS 2024 Proceedings, Salt Lake City, 2024, pp. 1–5.

[51] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, *Generative adversarial networks*, Communications of the ACM, 63 (2020), pp. 139–144.

[52] B. M. Greenwell, *pdp: An R package for constructing partial dependence plots*, The R Journal, 9 (2017).

[53] W. Guan, J. Cao, Y. Gu, and S. Qian, *Gama: A multi-graph-based anomaly detection framework for business processes via graph neural networks*, Information Systems, 124 (2024), p. 102405.

[54] H. Han, W.-Y. Wang, and B.-H. Mao, *Borderline-smote: a new over-sampling method in imbalanced data sets learning*, in International conference on intelligent computing, Springer, 2005, pp. 878–887.

[55] J. T. Hancock, R. A. Bauder, H. Wang, and T. M. Khoshgoftaar, *Explainable machine learning models for medicare fraud detection*, Journal of Big Data, 10 (2023), p. 154.

[56] P. Handel, I. Skog, J. Wahlstrom, F. Bonawiede, R. Welch, J. Ohlsson, and M. Ohlsson, *Insurance telematics: Opportunities and challenges with the smartphone solution*, IEEE Intelligent Transportation Systems Magazine, 6 (2014), pp. 57–70.

[57] S. Harjai, S. K. Khatri, and G. Singh, *Detecting fraudulent insurance claims using random forests and synthetic minority oversampling technique*, in 2019 4th International Conference on Information Systems and Computer Networks (ISCON), IEEE, 2019, pp. 123–128.

[58] H. He, Y. Bai, E. A. Garcia, and S. Li, *Adasyn: Adaptive synthetic sampling approach for imbalanced learning*, in 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), IEEE, 2008, pp. 1322–1328.

[59] H. He and E. A. Garcia, *Learning from imbalanced data*, IEEE Transactions on knowledge and data engineering, 21 (2009), pp. 1263–1284.

[60] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural computation, 9 (1997), pp. 1735–1780.

[61] C. Hu, Z. Quan, and W. F. Chong, *Imbalanced learning for insurance using modified loss functions in tree-based models*, Insurance: Mathematics and Economics, 106 (2022), pp. 13–32.

[62] L. Hu, Y. Lu, and Y. Feng, *Concept drift detection based on deep neural networks and autoencoders*, Applied Sciences, 15 (2025), p. 3056.

[63] C. Huang, W. Wang, D. Liu, R. Lu, and X. Shen, *Blockchain-assisted personalized car insurance with privacy preservation and fraud resistance*, IEEE Transactions on Vehicular Technology, 72 (2022), pp. 3777–3792.

[64] B. Itri, Y. Mohamed, Q. Mohammed, and B. Omar, *Performance comparative study of machine learning algorithms for automobile insurance fraud detection*, in 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS), IEEE, 2019, pp. 1–4.

[65] B. Itri, Y. Mohamed, B. Omar, and Q. Mohamed, *Empirical oversampling threshold strategy for machine learning performance optimisation in insurance fraud detection*, International Journal of Advanced Computer Science and Applications, 11 (2020).

[66] R. K. Jagait, M. N. Fekri, K. Grolinger, and S. Mir, *Load forecasting under concept drift: Online ensemble learning with recurrent neural network and arima*, IEEE Access, 9 (2021), pp. 98992–99008.

[67] C. Jorge, R. Cao, J. M. Vilar, et al., *Cost-sensitive thresholding over a two-dimensional decision region for fraud detection*, Information Sciences, 657 (2024), p. 119956.

[68] R. Kaafarani, L. Ismail, and O. Zahwe, *An adaptive decision-making approach for better selection of blockchain platform for health insurance frauds detection with smart contracts: development and performance evaluation*, Procedia Computer Science, 220 (2023), pp. 470–477.

[69] I. Kabir, M. K. Momo, and T. Tazrian, *Fraud detection in e-commerce using natural language processing*, PhD's thesis, Brac University, (2023).

[70] K. Kapadiya, U. Patel, R. Gupta, M. D. Alshehri, S. Tanwar, G. Sharma, and P. N. Bokoro, *Blockchain and ai-empowered healthcare insurance fraud detection: an analysis, architecture, and future prospects*, IEEE Access, 10 (2022), pp. 79606–79627.

[71] S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, *Leakage in data mining: Formulation, detection, and avoidance*, ACM Transactions on Knowledge Discovery from Data (TKDD), 6 (2012), pp. 1–21.

[72] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, *Lightgbm: A highly efficient gradient boosting decision tree*, Advances in neural information processing systems, 30 (2017).

[73] R. K. Kennedy, F. Villanustre, and T. M. Khoshgoftaar, *Unsupervised feature selection and class labeling for credit card fraud*, Journal of Big Data, 12 (2025), p. 111.

[74] J. Kester, *Insuring future automobility: A qualitative discussion of british and dutch car insurer's responses to connected and automated vehicles*, Research in Transportation Business & Management, 45 (2022), p. 100903.

[75] A. Kilroy and K. A. Smith, *Insurance fraud statistics 2024, https://www.forbes.com/advisor/insurance/fraud-statistics/*, Accessed: 2024-06-25, (2024).

[76] T. N. Kipf and M. Welling, *Semi-supervised classification with graph convolutional networks*, ICLR, (2017).

[77] Y. Kou, C.-T. Lu, S. Sirwongwattana, and Y.-P. Huang, *Survey of fraud detection techniques*, in IEEE International Conference on Networking, Sensing and Control, 2004, vol. 2, IEEE, 2004, pp. 749–754.

[78] I. Koychev, *Gradual forgetting for adaptation to concept drift*, Proceedings of ECAI 2000 Workshop on Current Issues in Spatio-Temporal Reasoning, (2000).

[79] T. Le, M. T. Vo, B. Vo, M. Y. Lee, and S. W. Baik, *A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction*, Complexity, 2019 (2019).

[80] T.-T.-H. Le, A. T. Prihatno, Y. E. Oktian, H. Kang, and H. Kim, *Exploring local explanation of practical industrial ai applications: A systematic literature review*, Applied Sciences, 13 (2023), p. 5809.

[81] B. Lebichot, Y.-A. Le Borgne, L. He-Guelton, F. Oblé, and G. Bontempi, *Deep-learning domain adaptation techniques for credit cards fraud detection*, in Recent Advances in Big Data and Deep Learning: Proceedings of the INNS Big Data and Deep Learning Conference INNSBDDL2019, held at Sestri Levante, Genova, Italy 16-18 April 2019, Springer, 2020, pp. 78–88.

[82] B. Lebichot, T. Verhelst, Y.-A. Le Borgne, L. He-Guelton, F. Oble, and G. Bontempi, *Transfer learning strategies for credit card fraud detection*, IEEE access, 9 (2021), pp. 114754–114766.

[83] L. Li and J. Xu, *Graph transformer-based self-adaptive malicious relation filtering for fraudulent comments detection in social network*, Knowledge-Based Systems, 280 (2023), p. 111005.

[84] C. X. Ling and V. S. Sheng, *Class imbalance problem*, In: C. Sammut, G. I. Webb (eds) Encyclopedia of machine learning, Springer, Boston, MA (2011), p. 171.

[85] S. M. Lundberg and S.-I. Lee, *A unified approach to interpreting model predictions*, in Advances in Neural Information Processing Systems 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., Curran Associates, Inc., 2017, pp. 4765–4774.

[86] S. Mahapatra and D. Sinha, *Smart h-chain: A blockchain based healthcare framework with insurance fraud detection*, Transactions on Emerging Telecommunications Technologies, 35 (2024), p. e4911.

[87] T.-D. Mai, K. Hoang, A. Baigutanova, G. Alina, and S. Kim, *Customs fraud detection in the presence of concept drift*, in 2021 International Conference on Data Mining Workshops (ICDMW), IEEE, 2021, pp. 370–379.

[88] L. Maiano, A. Montuschi, M. Caserio, E. Ferri, F. Kieffer, C. Germanò, L. Baiocco, L. R. Celsi, I. Amerini, and A. Anagnostopoulos, *A deep-learning–based antifraud system for car-insurance claims*, Expert Systems with Applications, 231 (2023), p. 120644.

[89] A. Maillart, *Toward an explainable machine learning model for claim frequency: a use case in car insurance pricing with telematics data*, European Actuarial Journal, (2021), pp. 1–39.

[90] D. Malekian and M. R. Hashemi, *An adaptive profile based fraud detection framework for handling concept drift*, in 2013 10th International ISC conference on information security and cryptology (ISCISC), IEEE, 2013, pp. 1–6.

[91] I. Mani and I. Zhang, *knn approach to unbalanced data distributions: a case study involving information extraction*, in Proceedings of workshop on learning from imbalanced datasets, vol. 126, ICML, 2003, pp. 1–7.

[92] A. Martín-Martín, M. Thelwall, E. Orduna-Malea, and E. Delgado López-Cózar, *Google scholar, microsoft academic, scopus, dimensions, web of science, and opencitations' coci: a multidisciplinary comparison of coverage via citations*, Scientometrics, 126 (2021), pp. 871–906.

[93] J. C. Mendoza-Tello, T. Mendoza-Tello, and H. Mora, *Blockchain as a healthcare insurance fraud detection tool*, in Research and Innovation Forum 2020: Disruptive Technologies in Times of Change, Springer, 2021, pp. 545–552.

[94] I. D. Mienye and Y. Sun, *A deep learning ensemble with data resampling for credit card fraud detection*, IEEE Access, 11 (2023), pp. 30628–30638.

[95] L. L. Minku, A. P. White, and X. Yao, *The impact of diversity on online ensemble learning in the presence of concept drift*, IEEE Transactions on Knowledge and Data Engineering, 22 (2009), pp. 730–742.

[96] C. Molnar, *Interpretable machine learning*, Leanpub, 2020.

[97] R. K. Mothilal, A. Sharma, and C. Tan, *Explaining machine learning classifiers through diverse counterfactual explanations*, in Proceedings of the 2020 conference on fairness, accountability, and transparency, 2020, pp. 607–617.

[98] C. Muranda, A. Ali, and T. Shongwe, *Detecting fraudulent motor insurance claims using support vector machines with adaptive synthetic sampling method*, in 2020 61st International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS), IEEE, 2020, pp. 1–5.

[99] C. Muranda, A. Ali, and T. Shongwe, *Deep learning method for detecting fraudulent motor insurance claims using unbalanced data*, in 2021 62nd International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS), IEEE, 2021, pp. 1–5.

[100] S. Najjar-Ghabel, S. Yousefi, and P. Habibi, *Comparative analysis and practical implementation of machine learning algorithms for phishing website detection*, in 2024 9th International Conference on Computer Science and Engineering (UBMK), IEEE, 2024, pp. 1–6.

[101] M. Nallakaruppan, B. Balusamy, M. L. Shri, V. Malathi, and S. Bhattacharyya, *An explainable ai framework for credit evaluation and analysis*, Applied Soft Computing, 153 (2024), p. 111307.

[102] E. W. Ngai, Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, *The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature*, Decision support systems, 50 (2011), pp. 559–569.

[103] T. T. Nguyen, T. C. Phan, H. T. Pham, T. T. Nguyen, J. Jo, and Q. V. H. Nguyen, *Example-based explanations for streaming fraud detection on graphs*, Information Sciences, 621 (2023), pp. 319–340.

[104] S. N. Nobel, S. Sultana, S. P. Singha, S. Chaki, M. J. N. Mahi, T. Jan, A. Barros, and M. Whaiduzzaman, *Unmasking banking fraud: Unleashing the power of machine learning and explainable AI (XAI) on imbalanced data*, Information, 15 (2024), p. 298.

[105] S. J. Omar, K. Fred, and K. K. Swaib, *A state-of-the-art review of machine learning techniques for fraud detection research*, in Proceedings of the 2018 International Conference on Software Engineering in Africa, 2018, pp. 11–19.

[106] S. Onishi, M. Nishimura, R. Fujimura, and Y. Hayashi, *Why do tree ensemble approximators not outperform the recursive-rule extraction algorithm?*, Machine Learning and Knowledge Extraction, 6 (2024), pp. 658–678.

[107] E. Owens, B. Sheehan, M. Mullins, M. Cunneen, J. Ressel, and G. Castignani, *Explainable artificial intelligence (xai) in insurance*, Risks, 10 (2022).

[108] J. Pacheco, J. Chela, and G. Salomé, *Fraud detection with machine learning: model comparison*, International Journal of Business Intelligence and Data Mining, 22 (2023), pp. 434–450.

[109] S. Padhi and S. Panigrahi, *Decision templates based ensemble classifiers for automobile insurance fraud detection*, in 2019 Global Conference for Advancement in Technology (GCAT), IEEE, 2019, pp. 1–5.

[110] E. Parkar, S. Gite, S. Mishra, B. Pradhan, and A. Alamri, *Comparative study of deep learning explainability and causal ai for fraud detection*, International Journal on Smart Sensing and Intelligent Systems, 17 (2024).

[111] D. K. Patel and S. Subudhi, *Application of extreme learning machine in detecting auto insurance fraud*, in 2019 International Conference on Applied Machine Learning (ICAML), IEEE, 2019, pp. 78–81.

[112] J. M. Pérez, J. Muguerza, O. Arbelaitz, I. Gurrutxaga, and J. I. Martín, *Consolidated tree classifier learning in a car insurance fraud detection domain with class imbalance*, in Pattern Recognition and Data Mining: Third International Conference on Advances in Pattern Recognition, ICAPR 2005, Bath, UK, August 22-25, 2005, Proceedings, Part I 3, Springer, 2005, pp. 381–389.

[113] C. Phua, D. Alahakoon, and V. Lee, *Minority report in fraud detection: Classification of skewed data*, SIGKDD Explor. Newsl., 6 (2004), pp. 50–59.

[114] V. Pillai, *Enhancing transparency and understanding in ai decision-making processes*, Iconic Research and Engineering Journals, 8 (2024), pp. 168–172.

[115] S. O. Pinto and V. A. Sobreiro, *Literature review: Anomaly detection approaches on digital business financial systems*, Digital Business, (2022), p. 100038.

[116] T. Pourhabibi, K.-L. Ong, B. H. Kam, and Y. L. Boo, *Fraud detection: A systematic literature review of graph-based anomaly detection approaches*, Decision Support Systems, 133 (2020), p. 113303.

[117] I. M. N. Prasasti, A. Dhini, and E. Laoh, *Automobile insurance fraud detection using supervised classifiers*, in 2020 International Workshop on Big Data and Information Security (IWBIS), IEEE, 2020, pp. 47–52.

[118] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, *Catboost: unbiased boosting with categorical features*, Advances in neural information processing systems, 31 (2018).

[119] L. R. Rabiner, *A tutorial on hidden markov models and selected applications in speech recognition*, Proceedings of the IEEE, 77 (1989), pp. 257–286.

[120] B. Raufi, C. Finnegan, and L. Longo, *A comparative analysis of shap, lime, anchors, and dice for interpreting a dense neural network in credit card fraud detection*, in World Conference on Explainable Artificial Intelligence, Springer, 2024, pp. 365–383.

[121] M. T. Ribeiro, S. Singh, and C. Guestrin, *"why should i trust you?": Explaining the predictions of any classifier*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1135–1144.

[122] ——, *Anchors: High-precision model-agnostic explanations*, in Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, vol. 32, AAAI Press, 2018, pp. 1527–1535.

[123] M. Sabuhi, M. Zhou, C.-P. Bezemer, and P. Musilek, *Applications of generative adversarial networks in anomaly detection: a systematic literature review*, Ieee Access, 9 (2021), pp. 161003–161029.

[124] G. Saldamli, V. Reddy, K. S. Bojja, M. K. Gururaja, Y. Doddaveerappa, and L. Tawalbeh, *Health care insurance fraud detection using blockchain*, in 2020 seventh international conference on software defined systems (SDS), IEEE, 2020, pp. 145–152.

[125] Z. Salekshahrezaee, J. L. Leevy, and T. M. Khoshgoftaar, *The effect of feature extraction and data sampling on credit card fraud detection*, Journal of Big Data, 10 (2023), p. 6.

[126] M. Salmi and D. Atif, *A data mining approach for imbalanced automobile insurance fraud data with evaluation of two sampling techniques and two filters*, Journal of Information Assurance and Security, 17 (2022), pp. 122–135.

[127] B. K. Sethi, P. K. Sarangi, and A. S. Aashrith, *Medical insurance fraud detection based on block chain and machine learning approach*, in 2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), IEEE, 2022, pp. 1–4.

[128] B. K. Sethi, D. Singh, and P. K. Sarangi, *Medical insurance fraud detection based on block chain and deep learning approach*, in 2022 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON), vol. 2, IEEE, 2022, pp. 103–106.

[129] M. K. Severino and Y. Peng, *Machine learning algorithms for fraud prediction in property insurance: Empirical evidence using real-world microdata*, Machine Learning with Applications, 5 (2021), p. 100074.

[130] Z. Shaeiri and S. Kazemitabar, *Fast unsupervised automobile insurance fraud detection based on spectral ranking of anomalies*, International Journal of Engineering, 33 (2020), pp. 1240–1248.

[131] A. Shahapurkar and R. Patil, *Concept drift and machine learning model for detecting fraudulent transactions in streaming environment.*, International Journal of Electrical & Computer Engineering (2088-8708), 13 (2023).

[132] N. Shaik, N. R. Kar, B. Thankachan, A. K. Pathak, J. Singh, and S. Gupta, *Utilizing blockchain and deep learning for decentralized discovery of deceptive practices in healthcare insurance*, in 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), IEEE, 2023, pp. 445–450.

[133] S. K. Shamitha and V. Ilango, *Importance of self-learning algorithms for fraud detection under concept drift*, in International Conference on Artificial Intelligence and Sustainable Engineering: Select Proceedings of AISE 2020, Volume 2, Springer, 2022, pp. 343–354.

[134] W. Siblini, G. Coter, R. Fabry, L. He-Guelton, F. Oblé, B. Lebichot, Y.-A. L. Borgne, and G. Bontempi, *Transfer learning for credit card fraud detection: A journey from research to production*, In The Proceedings of the Data Science and Advanced Analytics (DSAA 2021) IEEE conference, (2021).

[135] R. Singh, M. P. Ayyar, T. V. S. Pavan, S. Gosain, and R. R. Shah, *Automating car insurance claims using deep learning techniques*, in 2019 IEEE fifth international conference on multimedia big data (BigMM), IEEE, 2019, pp. 199–207.

[136] A. Singla and H. Jangir, *A comparative approach to predictive analytics with machine learning for fraud detection of realtime financial data*, in 2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3), IEEE, 2020, pp. 1–4.

[137] D. Sisodia and D. S. Sisodia, *Feature space transformation of user-clicks and deep transfer learning framework for fraudulent publisher detection in online advertising*, Applied Soft Computing, 125 (2022), p. 109142.

[138] ——, *A transfer learning framework towards identifying behavioral changes of fraudulent publishers in pay-per-click model of online advertising for click fraud detection*, Expert Systems with Applications, 232 (2023), p. 120922.

[139] A. Somasundaram and S. Reddy, *Parallel and incremental credit card fraud detection model to handle concept drift and data imbalance*, Neural Computing and Applications, 31 (2019), pp. 3–14.

[140] P. Sood, C. Sharma, S. Nijjer, and S. Sakhuja, *Review the role of artificial intelligence in detecting and preventing financial fraud using natural language processing*, International Journal of System Assurance Engineering and Management, (2023), pp. 1–16.

[141] E. Soufiane, S.-E. EL Baghdadi, A. Berrahou, A. Mesbah, and H. Berbia, *Automobile insurance claims auditing: A comprehensive survey on handling awry datasets*, in WITS 2020: Proceedings of the 6th International Conference on Wireless Technologies, Embedded, and Intelligent Systems, Springer, 2022, pp. 135–144.

[142] E. Strelcenia and S. Prakoonwit, *Improving classification performance in credit card fraud detection by using new data augmentation*, AI, 4 (2023), pp. 172–198.

[143] ——, *A survey on gan techniques for data augmentation to address the imbalanced data issues in credit card fraud detection*, Machine Learning and Knowledge Extraction, 5 (2023), pp. 304–329.

[144] L. Šubelj, Š. Furlan, and M. Bajec, *An expert system for detecting automobile insurance fraud using social network analysis*, Expert Systems with Applications, 38 (2011), pp. 1039–1052.

[145] S. Subudhi and S. Panigrahi, *Effect of class imbalanceness in detecting automobile insurance fraud*, in 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), IEEE, 2018, pp. 528–531.

[146] Y. Sun, L. Lan, X. Zhao, M. Fan, Q. Guo, and C. Li, *Selective multi-source transfer learning with wasserstein domain distance for financial fraud detection*, in Intelligent Computing and Block Chain: First BenchCouncil International Federated Conferences, FICC 2020, Qingdao, China, October 30–November 3, 2020, Revised Selected Papers 1, Springer, 2021, pp. 489–505.

[147] I. Tomek, *Two modifications of cnn*, IEEE Transactions on Systems, Man, and Cybernetics, SMC-6 (1976), pp. 769–772.

[148] A. Tsymbal, *The problem of concept drift: definitions and related work*, Computer Science Department, Trinity College Dublin, 106 (2004), p. 58.

[149] F. J. Valverde-Albacete, J. Carrillo-de Albornoz, and C. Peláez-Moreno, *A proposal for new evaluation metrics and result visualization technique for sentiment analysis tasks*, in Information Access Evaluation. Multilinguality, Multimodality, and Visualization: 4th International Conference of the CLEF Initiative, CLEF 2013, Valencia, Spain, September 23-26, 2013. Proceedings 4, Springer, 2013, pp. 41–52.

[150] R. Van Belle, B. Baesens, and J. De Weerdt, *Catchm: A novel network-based credit card fraud detection method using node representation learning*, Decision Support Systems, 164 (2023), p. 113866.

[151] I. Vorobyev, *Fraud risk assessment in car insurance using claims graph features in machine learning*, Expert Systems with Applications, 251 (2024), p. 124109.

[152] H. Wang and Z. Abraham, *Concept drift detection for streaming data*, in 2015 international joint conference on neural networks (IJCNN), IEEE, 2015, pp. 1–9.

[153] H. Wang, J. Zheng, I. E. Carvajal-Roca, L. Chen, and M. Bai, *Financial fraud detection based on deep learning: Towards large-scale pre-training transformer models*, in China Conference on Knowledge Graph and Semantic Computing, Springer, 2023, pp. 163–177.

[154] S.-C. Wang and S.-C. Wang, *Artificial neural network*, Interdisciplinary computing in java programming, (2003), pp. 81–100.

[155] X. Wang, Z. Liu, J. Liu, and J. Liu, *Fraud detection on multi-relation graphs via imbalanced and interactive learning*, Information Sciences, 642 (2023), p. 119153.

[156] Y. Wang and W. Xu, *Leveraging deep learning with lda-based text analytics to detect automobile insurance fraud*, Decision Support Systems, 105 (2018), pp. 87–95.

[157] Z. Wang, X. Chen, Y. Wu, L. Jiang, S. Lin, and G. Qiu, *A robust and interpretable ensemble machine learning model for predicting healthcare insurance fraud*, Scientific Reports, 15 (2025), p. 218.

[158] J. West and M. Bhattacharya, *Intelligent financial fraud detection: a comprehensive review*, Computers & security, 57 (2016), pp. 47–66.

[159] B. Wu, K.-M. Chao, and Y. Li, *Heterogeneous graph neural networks for fraud detection and explanation in supply chain finance*, Information Systems, 121 (2024), p. 102335.

[160] B. Wu, X. Yao, B. Zhang, K.-M. Chao, and Y. Li, *Splitgnn: Spectral graph neural network for fraud detection against heterophily*, in Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, 2023, pp. 2737–2746.

[161] J. Wu, R. Hu, D. Li, L. Ren, W. Hu, and Y. Zang, *A gnn-based fraud detector with dual resistance to graph disassortativity and imbalance*, Information Sciences, 669 (2024), p. 120580.

[162] H. Xia, Y. Zhou, and Z. Zhang, *Auto insurance fraud identification based on a cnn-lstm fusion deep learning model*, International Journal of Ad Hoc and Ubiquitous Computing, 39 (2022), pp. 37–45.

[163] Z. Xu, X. Huang, Y. Zhao, Y. Dong, and J. Li, *Contrastive attributed network anomaly detection with data augmentation*, in Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2022, pp. 444–457.

[164] C. Yan, M. Li, W. Liu, and M. Qi, *Improved adaptive genetic algorithm for the vehicle insurance fraud identification model based on a bp neural network*, Theoretical Computer Science, 817 (2020), pp. 12–23.

[165] S. Yousefi, S. Najjar-Ghabel, and Z. S. Owaid, *A supervised learning-based framework for failure detection in toy cars using acoustic signal analysis*, in 2025 IEEE 7th Symposium on Computers & Informatics (ISCI), IEEE, 2025, pp. 76–81.

[166] B. Yousefimehr, *Car-claims-compression*. https://github.com/behnamy2010/Car-Claims-Compression, 2024. Accessed: 2024-08-29.

[167] B. Yousefimehr and M. Ghatee, *A distribution-preserving method for resampling combined with lightgbm-lstm for sequence-wise fraud detection in credit card transactions*, Expert Systems with Applications, 262 (2025), p. 125661.

[168] B. Yousefimehr, M. Ghatee, and A. Heydari, *Improving adhd detection with cost-sensitive lightgbm*, in 2024 14th International Conference on Computer and Knowledge Engineering (ICCKE), IEEE, 2024, pp. 109–113.

[169] B. Yousefimehr, M. Ghatee, and R. Razavi-Far, *Multi-teacher knowledge distillation framework for lightweight anomaly detection*, Neural Networks, (2025), p. 108267.

[170] B. Yousefimehr, M. Ghatee, M. A. Seifi, J. Fazli, S. Tavakoli, Z. Rafei, S. Ghaffari, A. Nikahd, M. R. Gandomani, A. Orouji, et al., *Data balancing strategies: A survey of resampling and augmentation methods*, arXiv preprint arXiv:2505.13518, (2025).

[171] X. Zhang, Y. Han, W. Xu, and Q. Wang, *Hoba: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture*, Information Sciences, 557 (2021), pp. 302–316.

[172] P. Zheng, *Dynamic Fraud Detection via Sequential Modeling*, University of Arkansas, 2020.

[173] M. Zhong, Y. Wang, J. Yan, Y. Cheng, and P. Sun, *Transformer-based comparative multi-view illegal transaction detection*, Plos one, 18 (2023), p. e0276495.

[174] I. Žliobaitė, M. Pechenizkiy, and J. Gama, *An overview of concept drift applications*, Big data analysis: new algorithms for a new society, (2016), pp. 91–114.

Please cite this article using:

Behnam Yousefimehr, Mehdi Ghatee, A systematic survey and empirical comparison of hybrid methods for imbalanced fraud detection: Combining resampling and machine learning, AUT J. Math. Comput., 7(1) (2026) 85-116
https://doi.org/10.22060/AJMC.2025.24642.1446