Expert Systems With Applications 193 (2022) 116429

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Review

Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances

Waleed Hilal [a], S. Andrew Gadsden [a,*], John Yawney [b]

[a] McMaster University, Canada
[b] Adastra Corporation, Canada

ARTICLE INFO

Keywords:
Index Terms — Anomaly
Outlier
Anomaly detection
Outlier detection
Machine learning
Deep learning
Financial fraud
Credit card fraud
Insurance fraud
Securities and commodities fraud
Insider trading
Money laundering

ABSTRACT

With the rise of technology and the continued economic growth evident in modern society, acts of fraud have become much more prevalent in the financial industry, costing institutions and consumers hundreds of billions of dollars annually. Fraudsters are continuously evolving their approaches to exploit the vulnerabilities of the current prevention measures in place, many of whom are targeting the financial sector. These crimes include credit card fraud, healthcare and automobile insurance fraud, money laundering, securities and commodities fraud and insider trading. On their own, fraud prevention systems do not provide adequate security against these criminal acts. As such, the need for fraud detection systems to detect fraudulent acts after they have already been committed and the potential cost savings of doing so is more evident than ever. Anomaly detection techniques have been intensively studied for this purpose by researchers over the last couple of decades, many of which employed statistical, artificial intelligence and machine learning models. Supervised learning algorithms have been the most popular types of models studied in research up until recently. However, supervised learning models are associated with many challenges that have been and can be addressed by semi-supervised and unsupervised learning models proposed in recently published literature. This survey aims to investigate and present a thorough review of the most popular and effective anomaly detection techniques applied to detect financial fraud, with a focus on highlighting the recent advancements in the areas of semi-supervised and unsupervised learning.

1. Introduction

Anomaly detection is a broad field that addresses the problem of identifying instances of data or events that do not conform to expected behaviour (Chandola, Banerjee, & Kumar, 2009). The term outlier detection is frequently used interchangeably with anomaly detection. Many of the techniques employed for anomaly detection are fundamentally identical but are referred to differently depending on the application domain. Other such terms for anomalies are discordant objects, exceptions, aberrations, peculiarities or contaminants (Chandola et al., 2009; Aggarwal, 2013). This paper has chosen to use the term anomaly detection; however, outlier detection is also used where appropriate, depending on the problem's approach.

The significance of identifying anomalous patterns or events is their ability to translate to significant, actionable and commonly critical information for various applications (Chandola et al., 2009). Anomalies can be induced in data for various reasons, such as malicious activity or breakdown of systems, with the common characteristic that these reasons are of interest to the analyst (Chandola et al., 2009). Anomalous behaviour in credit card data can signify identity theft or fraudulent transactions committed by an unauthorized party (Singh & Upadhyaya, 2012). Traffic patterns that are anomalous in a computer network can indicate a malicious attempt to breach or compromise the system and lead to severe disruptions or even alert to a hacked computer that is sending sensitive data to an unauthorized destination (Chandola et al., 2009; Singh & Upadhyaya, 2012). In the field of healthcare, anomaly detection techniques can identify malignant cells or regions in medical images, such as magnetic resonance imaging (MRI) scans (Spence, Parra, & Sajda, 2001; Han, Rundo, Murao, Noguchi, Shimahara, Milacski, & Satoh, 2020). Furthermore, anomalous measurements or readings from sensors within a spacecraft may indicate a faulty component, and in nature, earthquakes can be predicted by finding anomalies in precursor data (Fujimaki, Yairi, & Machida, 2005; Saradjian & Akhoondzadeh, 2011). In all the applications mentioned, there is a notion of a "normal"

* Corresponding author.
E-mail addresses: hilalw@mcmaster.ca (W. Hilal), gadsden@mcmaster.ca (S.A. Gadsden), john.yawney@adastragrp.com (J. Yawney).

https://doi.org/10.1016/j.eswa.2021.116429
Received 11 July 2021; Received in revised form 17 December 2021; Accepted 18 December 2021
Available online 31 December 2021

model of the data from which anomalies deviate (Aggarwal, 2013).

In recent years, financial fraud, which includes but is not limited to credit card fraud, insurance fraud, money laundering, healthcare fraud and securities and commodities fraud, has garnered a great deal of unwanted attention from efforts and interests seeking to prevent them. Economically, serious concerns are posed by the alarmingly increasing rates of financial fraud. The total global losses annually due to financial fraud have been shown to be in the range of billions of dollars, with some figures suggesting the yearly cost to the US being in excess of $400 billion (Bhattacharyya, Jha, Tharakunnel, & Westland, 2011; Kirkos, Spathis, & Manolopoulos, 2007). The types of crimes associated with financial fraud also have broader ramifications in industry and have been associated with funding illicit activities such as organized crime and drug trafficking (West & Bhattacharya, 2016). The losses associated with these crimes are typically worn by companies and merchants, who often end up responsible for all costs incurred from the fraud. For example, with credit card fraud, merchants end up with chargebacks, administrative costs, as well as the loss of confidence of consumers who have been victimized by these acts (Quah & Sriganesh, 2008; Sánchez, Vila, Cerda, & Serrano, 2009). Thus, the consequences of these types of frauds are dire, and the importance of developing strategies and techniques to detect them is apparent.

The purpose of this work is to review and provide a comprehensive and structured overview of the current state of research and literature published on anomaly detection techniques applied in financial fraud. By facilitating a clear discussion of the various directions of research and techniques developed, the goal is to provide a complete guide of the most recent contributions, advancements, and experimental results in the field.

We organize the paper as follows: Section II presents an overview of related surveys in the field. The motivation behind this overview is to show that past survey papers have often maintained rather narrow scopes, and consequently highlight the need for a centralized source of information for anomaly detection in the financial fraud domain. Section III of this paper defines the anomaly while detailing a high-level summary of the associated detection task and provides an overview of the nature of the problem and its associated challenges. In section IV, background information on fraud in the financial domain is contextualized, with an outline of the different types of fraudulent acts committed and insight into how they occur. In section V, a detailed review of the surveyed literature studying anomaly detection techniques applied to detect financial fraud is presented, summarizing the key findings, limitations, and suggestions from published research. We conclude with final remarks on the key findings and provide suggestions for future research avenues in section VI.

2. Related surveys

A plethora of published research literature has studied applying anomaly detection techniques in various applications, which has been the topic of focus for many survey and review papers in recent years. Of those surveys, several have focused on a broad scope of applications, strategies, and techniques that have made a significant impact on further research in various fields.

Hodge and Austin published one of the first surveys on anomaly or outlier detection methodologies in 2004, providing a comprehensive review on the subject (Hodge & Austin, 2004). The literature provides extensive background on outliers or anomalies and the challenges associated with detecting them and a thorough review of early statistical, machine learning and ensemble methods applied to the task. In 2009, Chandola *et al.* also surveyed the various anomaly detection techniques proposed in research not previously covered by Hodge and Austin, providing more insight into the various real-life applications they are employed in (Chandola et al., 2009). In 2012, a survey published by Zimek *et al.* reviewed unsupervised anomaly detection techniques specifically for high-dimensional numerical data, discussing the aspects of the 'curse of dimensionality' in great detail (Zimek, Schubert, & Kriegel, 2012). The literature involved comparisons of two categories of specialized algorithms: ones that address the presence of irrelevant features or attributes and others that are more concerned with efficiency and effectiveness issues (Zimek et al., 2012). Temporal data poses another issue for anomaly detection, a problem surveyed extensively by Gupta *et al.* in 2014 (Gupta, Gao, Aggarwal, & Han, 2014). With the advances in computational capabilities enabling the availability of various forms of temporal data, the authors extensively review the techniques that have been enabled for anomaly detection in time-series data (Gupta et al., 2014). The authors provide significant insight into various applications of temporal anomaly detection and the associated challenges in each domain.

Other survey papers exist that focus more specifically on the techniques and applications of anomaly detection that have been researched in the context of financial fraud. Bolton and Hand, authors of some of the earliest and most influential surveys of statistical fraud detection, provide an in-depth background on the various types of financial fraud and how they are committed, such as credit card fraud, insurance fraud, money laundering, and others (Bolton & Hand, 2002). In their work, published in 2002, the authors also comment on the challenges of detecting fraud in different settings while reviewing the techniques applied to detect the different types of fraud in research. Kou and Huang also published a review similar in structure, but two years more recent, highlighting the attention received by deep learning techniques applied in financial fraud detection (Kou, Lu, Sirwongwattana, & Huang, 2004). Phua *et al.* examine fraud detection in their 2010 survey from a practical data-oriented, performance-driven perspective rather than application or technique-oriented views of previous survey papers (Phua, Lee, Smith, & Gayler, 2010). Furthermore, their work extends upon the types of frauds, methods and techniques covered than previous surveys with discussion on internal fraud and the implementations of hybrid approaches.

A more methodological review of the available literature on financial fraud detection was presented by Ngai *et al.* in 2011 (Ngai, Hu, Wong, Chen, & Sun, 2011). Based on a conceptual framework for classifying the papers depending on their application and technique, it was demonstrated that there is a lack of and need for studies on money laundering, mortgage and securities and commodities fraud (Ngai et al., 2011). West and Bhattacharya extended upon the previously mentioned framework of Ngai *et al.* in 2016 by further classifying the methods proposed in the surveyed literature based on their performance (West & Bhattacharya, 2016). Research papers were compared and organized based on the accuracy, recall and specificity of the techniques outlined in their methodologies in a more quantitative approach than previous works.

Pourhabibi *et al.* (Pourhabibi, Ong, Kam, & Boo, 2020) presented a systematic literature review of different graph-based anomaly detection techniques that have been studied in published literature in the context of financial fraud. The authors extensively surveyed the methods proposed to analyze connectivity patterns in communication networks to identify suspicious behaviours (Pourhabibi et al., 2020). The framework of the review was very similar to that of the survey by Ngai *et al.* in (Ngai et al., 2011), covering the limitations associated with different techniques and providing a general overview of the four graph-based approaches: community-based, probabilistic-based, structural-based, compression-based and decomposition-based (Pourhabibi et al., 2020). The applications of these approaches included banking fraud detection, insurance fraud detection, anti-money laundering and more. Highlights of each technique and any challenges faced were discussed, which the authors analyze thoroughly. Based on the investigation and analysis in the survey paper, directions for future research efforts were suggested based on the gaps identified by the authors from the academic papers reviewed.

Most of the research efforts on detecting financial fraud in recent years have focused on credit card fraud detection techniques, which is apparent from the availability of literature on the matter, which exceeds

that of other financial fraud types by a significant margin. As such, many survey papers have been presented in the last decade evaluating the state of research in detecting credit card fraud. Delamaire *et al.* are among the first, in 2009, to review this specific topic, identifying in their work the different types of credit card fraud and the techniques that have been employed to detect them (Delamaire, Abdou, & Pointon, 2000). The authors provide a comprehensive background on the standard terms in credit card fraud and the key statistics and figures in the field. Findings of published literature were also presented, compared, and analyzed with a discussion on the various measures that can be implemented and adopted depending on the type of fraud faced. Furthermore, ethical considerations are presented in the literature regarding the ethical issues that arise from the misclassifications of genuine transactions as fraudulent, as well as the costs associated with the misclassification of fraudulent transactions as genuine (Delamaire et al., 2000).

In 2012, Zareapoor *et al.* conducted a survey focusing on the specific statistical and machine learning techniques most commonly used for credit card fraud detection (Zareapoor, 2012). In this literature, a historical background is given on each technique and a high-level overview of how they work or operate in credit card fraud detection systems. Although the authors provide commentary on the performance of the techniques mentioned in comparison to each other, we identify a significant weakness in this work: the lack of discussion on quantifiable performance measures produced by the papers reviewed by the authors in (Zareapoor, 2012). Literature published in 2017 by Adewumi and Akinyelu partially addresses the gaps in the previously mentioned works by briefly discussing the classification accuracies of techniques covered (Adewumi & Akinyelu, 2018). The authors also discuss the limitations such as high-dimensionality of data, imbalanced data sets and their impacts on the performances of reviewed research. Finally, the literature identifies a developing trend of using artificially generated data to overcome certain limitations of credit card detection systems and suggests that it may be promising to explore such methods further.

Most existing surveys provide a very general overview of the techniques applied in financial fraud detection; however, many do not highlight in enough detail the challenges and issues faced by the published research. Another significant issue is the lack of discussion on specifics relating to the performance measures employed in the methods reviewed, with many surveys only commenting on qualitative comparisons of different techniques. A lack of structure was also observed in most survey papers, many of which do not provide enough background on relevant aspects that require further understanding. Regardless of the different shortcomings of the available survey papers in the field, each is characterized by different strengths in various aspects, offering value to any audience.

In this survey, we extend the work of previous authors by covering more state-of-the-art techniques that have been applied to detect various types of financial fraud in the most recent years. We hope to address each of the individual shortcomings of previous surveys to provide an overall more comprehensive, detailed, and complete review. We will achieve this by defining the notion of an anomaly and providing a thorough background and discussion on the various types of anomalies and the general challenges faced by anomaly detection tasks in different domains. The different types of learning used by anomaly detection models or methods are described and outlined, as well as the various performance measures employed to assess them. We also define financial fraud, outlining its history and the various acts of fraud that fall under that classification. Figures and statistics relating to financial fraud losses are detailed to justify the need to use anomaly detection techniques to detect fraudulent financial activity. The different types of fraud are also summarized, highlighting how they are committed and the current measures in place to prevent them. Most importantly, we present a critical review of the research literature on financial fraud detection, with the focus on papers published from 2002 to 2020. An emphasis is placed on evaluating and comparing the limitations, challenges, and entire range of performance measures utilized to provide an informed commentary on current, state-of-the-art research findings.

The techniques covered from the surveyed literature span deeply researched models like support vector machines (SVM), decision trees (DT), random forests (RF), hidden Markov models (HMM), multilayered perceptron networks (MLP) and more. Uniquely from other surveys, we cover novel research papers that employ models not previously studied for detecting financial fraud. The most prominent of these are deep learning architectures such as convolutional neural networks (CNN), autoencoders (AE) and generative adversarial networks (GAN). We conclude with a summary of this review's key findings and provide suggestions for the direction of future research efforts.

3. Problem description and formulation

3.1. What are Anomalies?

Many authors have proposed varying definitions for an anomaly; however, there has not been a universally adopted definition. Exact definitions of an anomaly depend on assumptions regarding the structure of the data and the application in consideration. However, definitions exist that are considered general to most if not all cases, regardless of the setting or application. Of those definitions, the most widely recognized is by Hawkins, who defines the concept of an anomaly or an outlier in this case, as follows: "*An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism*" (Hawkins, 1980). This definition refers to data from a statistics-based intuition, where normal data follows a generating mechanism and anomalies are samples or instances which deviate from this mechanism. Thus, anomalies often relay useful information about a system's abnormal characteristics that are impacting the generating mechanism (Aggarwal, 2013). For the remainder of this paper, we adopt Hawkins' definition of the concept of an anomaly.

A two-dimensional graph demonstrating the concept of different types of anomalies is illustrated in Fig. 1. As can be seen from this figure, the data elements form two norm al regions denoted by N_1 and N_2, as those are the regions where most of the events lie. Observations that are further away from most of the other observations, either individually or as a small collective, like points o_1, o_2 and the region O_3, are anomalies.

Anomaly detection bears similarities to noise removal, which deals with unwanted noise in data, but the two are distinct from each other

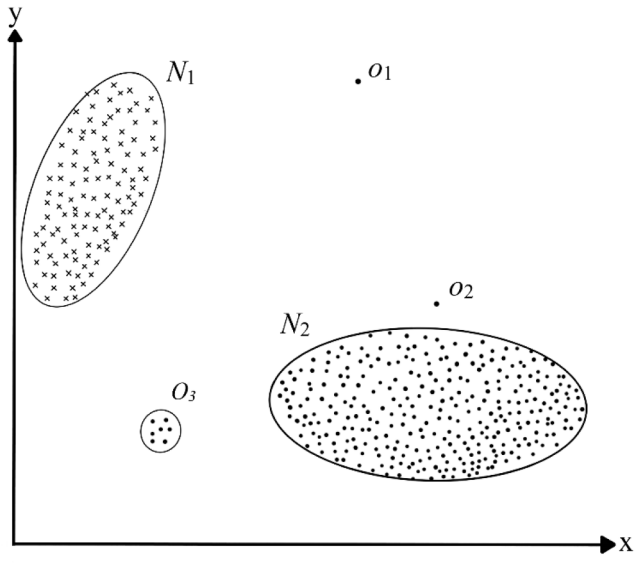

Fig. 1. Graphical visualization of anomalous data in a simple two-dimensional representation.

(Chandola et al., 2009). In real-world applications, the data may be affected by a significant amount of noise, which may not be of interest to the analyst, but acts as a hindrance during the data analysis stage. It is usually only significantly interesting deviations that are of interest (Aggarwal, 2013). The detrimental effect of noise on data analysis drives the need for noise removal, as it removes any unwanted objects before data analysis (Xiong, Pandey, Steinbach, & Kumar, 2006). Novelty detection is another topic also related to but distinct from anomaly detection. Instead, these types of techniques detect previously unobserved or novel patterns in the data, and generally, the main distinction from anomaly detection being that novel patterns are typically incorporated into a model after detection (Markou & Singh, 2003).

Anomalies occur for various reasons such as human error, instrument faults, natural deviations in populations, fraudulent activity, behavioural changes of systems or faults within a system (Hodge & Austin, 2004). How the anomaly detection system deals with the anomaly is dependent on the application area. For example, suppose the anomaly indicates a typographical error entered by a data-entry clerk. In that case, a simple notification to the clerk to correct the error will help restore the anomalous data to a normal entry. Anomalous data from instrument readings can be simply be deleted once identified. Furthermore, anomaly detection systems in critical environments such as intrusion monitoring or fraud detection systems must be able to detect anomalies immediately and in real-time, with a suitable alarm to allow for intervention (Hodge & Austin, 2004).

3.2. Types of Anomalies

Anomalies can be classified into three different categories, and the type of anomaly being dealt with is a crucial aspect of consideration for any anomaly detection technique.

Point anomalies are the simplest type of anomaly and are the focus of most research applying anomaly detection techniques. They are characterized as individual instances of data or events significantly different from the rest of the data. In general terms, they are points that do not lie within the established normal or separating boundary, such as the points o_1 and o_2 in. To demonstrate with a real-life example, consider credit card usage data for individuals, defined with only one feature: the purchase amount. When compared, any transaction that is greater than the normal spending range for that individual would be considered a point anomaly.

Contextual anomalies are instances of data that are anomalous in a specific context and not if otherwise. The data set structure induces the notion of the context for these types of anomalies, and the problem formulation strictly requires its specification. Two attributes are used to define each data instance, contextual and behavioural attributes. Contextual attributes determine the context for that instance, such as the time in time-series data. Behavioural attributes define non-contextual characteristics, such as the purchase amount in a credit card dataset. The anomalous behaviour is, in turn, determined using the behavioural attributes in a specific context. For instance, a contextual attribute in credit card fraud would be the time of purchase. Assuming an individual has a weekly spending bill of a hundred dollars, except for Christmas week, where it is a thousand dollars. A purchase of a thousand dollars in an average week in May would be considered a contextual anomaly because it does not conform to the typical behaviour in the context of time, even though spending the same amount in the week of Christmas would be considered normal (Chandola et al., 2009).

Finally, collective anomalies are a collection of related instances that are anomalous with respect to the entire data set. The individual events or data instances within a collective anomaly may not necessarily be anomalies by themselves; however, their occurrence as a collection is anomalous. It is important to note that point anomalies can occur in any data set, and collective anomalies can only occur in data sets where there is a relation between data instances. Contextual attributes can only occur when there are contextual attributes in the data. Point or collective anomalies can even be contextual anomalies if observed with respect to a context (Chandola et al., 2009).

3.3. Challenges in Anomaly Detection

As defined earlier, an anomaly is an observation or event that does not conform to expected normal behaviour. Therefore, an intuitive approach is to define a region to represent the normal behaviour, such that observations not within the normal region's boundary are labelled as anomalies, an example of which can be seen in Fig. 1. However, this seemingly simple approach is made much more challenging due to the unique nature of anomaly detection problems, which present complexities distinct from most analytical and learning problems. These complexities are attributed to several factors.

It is very difficult to define a normal region or boundary to encompass all possibilities of normal behaviour, and usually, the boundary between normal and anomalous behaviour lacks precision. As a result of this, observations of either the normal or anomalous class near the boundary may be misclassified. Furthermore, anomalies that arise due to malicious activity are often changing and adapting, driven by adversaries of the anomaly detection system and their attempts to disguise anomalous events as normal, ultimately increasing the difficulty of detection. Generally, this is also the case in other domains where normal behaviour is continuously evolving, and the notion of normal behaviour now may not be sufficiently representative of future behaviours. Concept drift is the term used in most literature to refer to the phenomenon of underlying models changing over time (Gama, Žliobaitė, Pechenizkiy, & Bouchachia, 2014). Anomalies are also heterogeneous, implying that they are irregular, meaning a class of anomalies could demonstrate very different characteristics suggesting abnormality from another anomalous class (Pang, Shen, Cao, & van den Hengel, 2020).

Furthermore, the notion of an anomaly varies from different domains and applications. In the field of medicine, slight deviations from the baseline (such as fluctuations in body temperature) may be considered anomalous. In contrast, a similar degree of deviation in the financial domain, specifically in the stock market, might be considered normal (Chandola et al., 2009). For this reason, applying a technique that is developed for one domain may not be as straightforward to implement in another.

Another factor contributing to the complexity of anomaly detection is the lack of labelled data for training and validation of models due to several reasons. One such reason is privacy concerns when dealing with sensitive data or the cost of labelling data if it must be performed by humans (Domingues, Filippone, Michiardi, & Zouaoui, 2018). Anomalous instances are also rare in occurrence, contrasted by normal instances, which often account for an overwhelming majority of the data (Pang et al., 2020; Boukerche, Zheng, & Alfandi, 2020). In such cases, standard classifier anomaly detection techniques tend to ignore the small classes due to being overwhelmed by the larger ones (Chawla, Japkowicz, & Kotcz, 2004). In normal data instances, the presence of noise, often acting similarly to anomalies, also introduces a challenge by making it more difficult to deduce clear boundaries or decision rules within the data set.

In low-dimensional spaces, anomalies often display prominent abnormal features or characteristics. However, they become hidden and indiscernible in high-dimensional spaces, a longstanding and extensively researched problem known as the curse of dimensionality (Zimek et al., 2012). A straightforward solution is to reduce the dimensionality of the data to a lower space spanned by a smaller subset of the original or newly constructed features such as in subspace and feature selection-based methods (Keller, Muller, & Bohm, 2012; Lazarevic & Kumar, 2005). It is essential to identify intricate feature interactions in high-dimensional data, but it is still a significant challenge for anomaly detection. Besides, guaranteeing information preservation in the new feature space for relevant methods is vital to accurate downstream anomaly detection. However, it is made challenging due to the

heterogeneous nature of anomalies (Pang et al., 2020).

Whether or not an anomaly detection technique can be applied to a problem is also influenced by the nature of the values of features in the input data. Numerical and categorical or symbolic data both require different statistical models when using different statistical or machine learning techniques. In some cases, it may be necessary to convert or encode symbolic data into numerical data by using various techniques. Varying distance measures are necessary when using distance-based techniques, depending on the nature of the input. Some techniques are only capable of processing image data, usually represented by matrices of real numbers flattened to a vector. There can even be a relationship between data instances, such as in spatial, temporal and graph data. This is another consideration that must be factored into the design stage to ensure the applicability, robustness and performance of the technique being used for that specific application.

3.4. Data Labels and Output

In any data set, the label associated with an instance of data denotes whether the instance is anomalous or normal. However, it is essential to note that obtaining accurately labelled data that is representative of all the types of behaviours is often challenging and costly. Labelling data requires substantial time and effort as it is usually done manually by an expert in the application domain (Chandola et al., 2009; Aggarwal, 2013). Furthermore, it is much more challenging to obtain a set of labelled data that encompasses all different types of possible anomalous behaviour compared to obtaining labelled normal behaviour. Due to the dynamic nature of anomalies, as described in the challenges with anomaly detection, anomalous behaviour can change, resulting in new anomalies, for which there may not be any labelled data.

The availability of labelled data is a significant consideration when deciding on the appropriate anomaly detection method to employ. Three categories that a method can be categorized in are supervised, semi-supervised or unsupervised anomaly detection.

Supervised anomaly detection techniques work under the assumption that the data set used consists of labelled instances that fall under either a normal or anomalous class. Most approaches under this category construct a predictive model for the normal and anomalous classes, and new unseen data can be classified by comparing it against the determined model. A significant issue with supervised anomaly detection, which has been briefly discussed previously, is that the anomalous class is usually rare in occurrence compared to the normal class. It is also challenging to obtain accurate labels representative of the anomalous class. This is known as the class imbalance problem, and in practical applications, the ratio between the classes can be from 1 to 100 or as drastic as 1 to 10,000 (Chawla et al., 2004). Various literature has emerged over the last two decades addressing this issue (Chawla et al., 2004; Phua, Alahakoon, & Lee, 2004).

Semi-supervised anomaly detection techniques assume that the only instances in the data set that are labelled are the ones belonging to the normal class. Because of this, they are more applicable. A model is constructed only for the normal class and not the anomalous class, unlike supervised anomaly detection. The test set of the data is then compared against the model to identify anomalous instances. This is useful in cases where it may be difficult or expensive to model an anomaly, such as a fault in large vehicles such as aircraft and spacecraft (Fujimaki, Yairi, & Machida, 2005). There is a very limited range of techniques and literature designed to work with data sets where only the anomalous class is available, as it is challenging to retrieve data that is representative of all anomalous behaviour.

Unsupervised anomaly detection is the most widely applicable of the three categories as the techniques do not require any labels in the data set. An implicit assumption is made by unsupervised methods that anomalous events are far less frequent than normal events in the test set of the data; otherwise, these techniques' false alarm rates will be higher than what expected. It is common practice to adapt semi-supervised methods to an unsupervised anomaly detection problem using an unlabelled sample of the train set of the data. An assumption is necessary in this case: that the test set has very few anomalous instances, and the model learned is robust to those few anomalous instances.

It may seem straightforward enough that the output of an anomaly detection technique can be a label of either normal or anomalous for each respective instance. However, scoring techniques exist, which involve assigning a score to each instance that corresponds to a degree of anomalousness to produce an ordered list of anomaly scores. The choices are then to consider a few of the highest scores if the ratio of anomalous instances is known or assign a cut-off score such as an application-specific threshold for labelling the most relevant instances as anomalous. Binary classification techniques do not provide the same flexibility directly; however, each technique's parameter choices allow for a degree of indirect control.

3.5. Performance Measures

The anomaly detection problem is often, but not exclusively, considered a classification task. An effective way of evaluating the performance of classification models is by inspecting their confusion matrix. The confusion matrix describes the difference between a data set's ground truth and the model's predictions. There are other metrics that are more concise and that can provide more specific information into the performance of a classifier. The *precision* is the accuracy of the positive predictions. The *recall* of a classifier, also known as sensitivity, is the proportion of positive instances that are correctly detected by the classifier. The *accuracy* is the number of correct overall predictions made by the model. The *specificity* of a classifier is a ratio between the correctly classified negative samples and the total number of negative samples. While one may, intuitively, try to maximize both recall and precision, there is an inverse relationship between the two values. Forcing a higher precision may result in a lower recall, and vice versa. This is known as the precision/recall trade-off. A better measure to maximize instead is the F-score (also known as the F_1-score), which is the harmonic mean of the precision and recall.

By tweaking the value of the decision threshold for a classifier, it is possible to monitor the performance change in terms of the trade-off between specific measures like recall and precision. This is known as parametric evaluation, which is an evaluation of all the possible confusion matrices that are produced from varying the decision threshold. And then plotting two curves known as the Receiver Operating Characteristic (ROC) curve and the precision-recall curve (Lucas & Jurgovsky, 2020). A way of comparing classifiers is by measuring the area under the ROC curve (AUC), where perfect classifiers will have an AUC equal to 1 and purely random classifiers have an AUC equal to 0.5. The axes are both proportional to the class they represent, and as such, the ROC curve is insensitive to imbalanced data sets (Lucas & Jurgovsky, 2020). The precision-recall curve (PRC) plots the precision on the y-axis and the recall on the x-axis. The recall is a proportion of the positive class, representing the ratio of positive classes identified. The precision is based on both the positive and negative classes. Consequently, imbalanced data sets affect precision by making classification more challenging. Similarly, the area under the precision-recall curve (AUPRC) can be determined to compare the performance of different classifiers.

Although the mentioned performance measures are the most frequently and popularly used for evaluating classifiers, there are many more, and the list of metrics discussed here is not exhaustive. Clustering algorithms that group similar data points based on relevant features employ various distance measures such as the mean squared error (MSE), Euclidean distance, Manhattan distance and others to quantify the similarity or dissimilarity between samples or observations. These unsupervised techniques divide a set of data into meaningful subsets or groups, where entities within a group are similar, and entities of different groups are dissimilar (Sabau, 2012).

4. Financial fraud

Financial fraud has garnered much more attention in the past decade due to the potential consequences of undetected anomalies to the industry and to everyday life. These crimes can vary in nature and have the effect of possibly destabilizing economies, increasing the cost of living and impacting the consumer's sense of security (Syeda, Zhang, & Pan, 2002). There is no universally accepted definition for financial fraud; however, we adopt the definition from the Association of Certified Fraud Examiners (ACFE) as "any intentional or deliberate act to deprive another of property or money by guile, deception, or other unfair means" (Association of Certified Fraud Examiners, 2004).

With the rapid expansion and advancement in modern technology such as the Internet, hardwire devices like phones and laptops and social media, there has also been an increase in fraudulent activity (Kou, Lu, Sirwongwattana, & Huang, 2004). This has resulted in billions of dollars of losses to business, which has consequentially motivated extensive efforts towards exploring anomaly detection techniques for the detection of fraud. It is imperative to be able to identify fraud as quickly as it occurs, as there is a cost incurred over time by undetected cases (Association of Certified Fraud Examiners, 2020). It is not just individual fraudsters and perpetrators who take advantage of these advances in technology. Well-developed organized perpetrators and crime communities have also been investing in expanding and evolving their techniques (Bhattacharyya et al., 2011). Therefore, it is imperative to continuously evolve and improve these fraud detection techniques as fraudsters are always adapting and innovating strategies and methods to breach them (West & Bhattacharya, 2016).

It is reported by the Internet Crime Complaint Center (IC3) that 467,361 complaints of internet crimes were reported in the US, with reported losses exceeding $3.5 billion, which is an increase of nearly 30% from the previous year (Center, 2019). These losses involved various acts such as compromising business e-mails to conduct unauthorized transfers of funds, cases of identity fraud involving fraudulent cheques and credit card applications and much more. In the ACFE's 2020 Report to the Nations, investigations have led to the estimates that globally, organizations lose 5% of revenue to fraud annually and that the typical fraud case goes undetected for 14 months, costing an average of 8,300 dollars per month (Association of Certified Fraud Examiners, 2020). Table 1 presents the reported complaints and total loss reported annually by the IC3 from the year 2015 to 2019. It is evident from Table 1 that the total losses reported have been increasing at an accelerating rate each year since 2017, which further highlights that fraudulent efforts persist in adapting and evolving to circumvent systems in place to prevent them.

Financial fraud has been classified under a framework by the Federal Bureau of Investigations (FBI). The crimes have been characterized by deceit, concealment, or violation of trust regardless of whether accompanied by the threat of violence or physical force (Bureau, 2011). Various types of fraud exist under that classification; one such example is bank fraud, which consists of credit card fraud, money laundering and mortgage fraud. Insurance fraud is another common type of financial fraud that can involve a variety of claims covering crops, healthcare, automobiles, and others. Other types of financial fraud exist, such as securities and commodities fraud and insider trading. However, in this survey, we focus mainly on covering the application of anomaly detection techniques to credit and debit card fraud, as well as insurance fraud. The survey also covers the recent advances in the few papers found covering anti-money laundering research. We note that there is a lack of available research literature on other types of financial fraud such as mortgage fraud, commodities & securities fraud, and insider trading. As such, the types of fraud were beyond the scope of this survey paper.

Table 1
Annual complaints received and total loss reported by IC3 (Center, 2019).

Year	Complaints received	Total losses (USD)
2015	288,012	1,070,700,000
2016	298,728	1,450,700,000
2017	301,580	1,418,700,000
2018	351,937	2,706,400,000
2019	467,361	3,500,000,000

4.1. Credit Card Fraud

The widespread use of credit cards as the primary method of transacting is a striking example of the digitalization of everyday life and society over the past couple of decades. This adoption also presents the problem of credit card fraud, which involves sophisticated strategies and techniques employed for the theft of money and assets. According to (Report, 2020), card transactions' fraudulent losses reached $28.65 billion in 2020, with the US accounting for a third of that gross loss with $9.62 billion. As credit cards gain popularity and their use is becoming even more prevalent, there is also a concern about the increase in fraudulent efforts (Bhattacharyya et al., 2011). There are significant ramifications associated with undetected credit card fraud, as the crimes have even been exploited for funding organized crime, narcotics trafficking and even financing terrorists (Bhattacharyya et al., 2011).

There are two ways of classifying credit card fraud, either as application or behavioural fraud, which are sometimes also referred to as offline and online fraud, respectively (Kou, Lu, Sirwongwattana, & Huang, 2004; Bolton & Hand, 2001). Application fraud is often associated with identity fraud, as it usually involves fraudsters attempting to issue new cards from credit companies using other people's private information. Behavioural fraud is comprised of four different acts, which involve mail theft, stolen or lost cards, counterfeit cards, 'cardholder not present' fraud or bankruptcy fraud (Bhattacharyya et al., 2011; Bolton & Hand, 2001). Mail theft fraud primarily involves fraudsters intercepting other people's mail to obtain physical credit cards or personal banking information, which they then use to commit application fraud. Stolen or lost credit card related fraud typically occurs when physical cards are stolen from individuals either with or without their knowledge or when fraudsters find or gain access to lost cards. Bankruptcy fraud is considered one of the most challenging types of fraud to detect and generally entails people using a credit card with no intention of ever paying back the balance and then filing for personal bankruptcy, as defined by Ghosh and Reilly in 1994 (Ghosh & Reilly, 1994).

With the rapid digitization of everyday life, more information is available online that is subsequently vulnerable to attack or exploitation by criminals and fraudsters. Fraudsters rely on these vulnerabilities to obtain credit card information through illegal means, which they could then implement into a fake or counterfeit card. All the mentioned methods can also be involved in 'cardholder not present' frauds, which only require the credit card details to conduct transactions remotely, such as by mail, phone, or the Internet. There are many ways that the information necessary to commit these types of fraudulent acts

is obtained, like online 'phishing' scams to trick unaware individuals, or by intrusions of company's networks and computer systems, or even by physical devices known as 'skimmers' installed on the card readers of tampered ATMs to steal people's card information (Bhattacharyya et al., 2011; Quah & Sriganesh, 2008).

It has been noted by Bolton and Hand that there has been a dearth of published literature in the study of credit card detection in earlier years, which was attributed to two reasons (Bolton & Hand, 2002). First of which was that the exchange of ideas in fraud detection is usually very limited as it may be counterintuitive to describe the detection techniques with detail to the public domain, which may provide the knowledge and information required to devise new tools and techniques to bypass these detection techniques (Bolton & Hand, 2002). They also state that data sets of these types of fraudulent data are often kept private and censored in the rare instance they are made public (Bolton &

Hand, 2002). As previously discussed, there is also a cost for undetected fraud over time, which means that the value of fraud detection is a function of time (Bhattacharyya et al., 2011). The sooner the detection of fraudulent activity, the less the potential losses by individuals and companies. This is especially important to keep in mind as the typical fraudster has been known to exploit credit cards by spending as much as possible in as little time as possible until the fraud is detected and the card is deactivated (Bolton & Hand, 2002).

Most credit card fraud detection strategies involve creating a profiler for cardholders by analyzing individual spending behaviours and patterns (Abdallah, Maarof, & Zainal, 2016). Chandola *et al.* classify profile-based approaches into two types: either a "by-owner" approach or a "by-operation" approach (Chandola et al., 2009). The "by-owner" approach profiles each credit card user based on their usage and spending history, comparing new transactions to the user's profile and flagging the ones inconsistent from the profile as anomalies.

The disadvantage of this approach is that it is typically costly as it requires a central data repository, which must be queried every time a user makes a transaction (Chandola et al., 2009). "By-operation" approaches detect fraudulent transactions from transactions occurring at specific geographic locations that are unlikely to be frequented by the genuine cardholder. The type of anomaly detected by both approaches are contextual anomalies, the contexts being the user for "by-owner" approaches and the geographic location for "by-operation" approaches. When constructing fraud detection systems or models using credit card datasets, the samples of transactions are described by an initial set of raw features that describe or depict information about each sample. From the inspection of the surveyed literature, it has been apparent that regardless of the study, the raw features describing datasets are often quite similar. After investigation, it has been found this is due to standards set by the International Financial Reporting Standards (IFRS) foundation that govern and regulate financial reporting practices that credit card issuers and financial institutions must adhere to (Institute, 2011). The raw features common to most datasets used in the literature surveyed in this paper are summarized in Table 2.

4.2. Insurance Fraud

Insurance fraud is another type of financial fraud that involves some sort of trickery or deception committed throughout any point of the claim process by a claimant, healthcare provider, employee in the insurance company or even agents and brokers (Ngai et al., 2011). The primary industries targeted by fraudulent insurance claims are the healthcare and automobile insurance industry; however, crop and home insurance fraud also occurs, although there is a lack of literature on both (Abdallah et al., 2016). It is estimated that the total cost of insurance fraud in the US exceeds $80 billion annually and is ultimately passed on to consumers in the form of higher insurance premiums (Bureau, 2011).

Table 2
Summary of typical raw features in credit card datasets (Bahnsen et al., 2016).

Feature	Description
Transaction ID	Transaction identification number
Time	Date and time of the transaction
Account number	Identification number of the customer
Card number	Identification of the card
Transaction type	Internet, ATM or POS
Entry mode	Chip and pin or magnetic stripe
Amount	Amount of transaction
Merchant code	Identification of the merchant type
Merchant group	Merchant group identification
Country	Country of transaction
Country 2	Country of residence
Type of card	Visa debit, MasterCard, American Express
Gender	Gender of the cardholder
Age	Age of the cardholder
Bank	Issuer bank of the card

Of all the different types of fraud that fall under this category, automobile insurance fraud detection has attracted the most attention (Ngai et al., 2011). According to a study conducted on behalf of the Insurance Bureau of Canada by KPMG, an international accounting organization, the cost of fraudulent auto-insurance claims in the Canadian province of Ontario exceeded $1.6 billion in the year 2012 (KPMG, 2012). Furthermore, elements of suspected fraud have been reportedly found in 21% to 36% of automobile insurance claims, and less than 3% of those suspected fraudulent claims end up being prosecuted (Derrig, 2002; Viaene, Derrig, Baesens, & Dedene, 2002).

Automobile insurance claims generally involve a contract between the insurance company and the insured individual or organization to cover the relevant costs of theft or accidental damage of a vehicle. Fraudulent claims can be perpetrated by individual fraudsters; one way this is achieved is by means of deception during the claim process. Evidence also exists of organized groups collaborating together to commit insurance fraud, usually by staging or faking accidents, where in some cases an accident may not have even occurred; instead, the vehicles were transported to the scene (Šubelj, Furlan, & Bajec, 2011). Regardless, most fraudulent cases are opportunistic frauds in that they are not planned, and instead, the opportunity arising from an accident is seized by an individual by exaggerating the damages or statements made in a claim (Šubelj et al., 2011). Some of the schemes involved in automobile insurance fraud have been identified and defined by the ACFE, the most common of which are described in Table 3 (Association of Certified Fraud Examiners, 2019).

In modern society, healthcare has become a significant concern that is tangled with political, social, and economic issues. The public demand for high-quality medical services and the technology necessary to provide them is met with a substantial financial cost. Also, the availability of government-sponsored healthcare insurance systems is critical for many low-income individuals and families depending on them for support to pay for the continually increasing costs of drugs and treatment (Yang & Hwang, 2006). A growing problem has been the abuse and exploitation of healthcare insurance systems by fraudsters for benefits they or someone else may not be entitled to. Healthcare providers have also been found to exploit the system for financial gain through practices inconsistent with sound fiscal, business, or medical practices. This results in unnecessary costs or reimbursements for services not medically necessary or that fail to adhere to professionally recognized standards

Table 3
Descriptions of different types of automobile insurance fraud schemes (Association of Certified Fraud Examiners, 2019).

Type of Fraud	Description of Fraud
Ditching	Disposing or abandoning a vehicle to obtain claim payout from insurance policy.
Past posting	Obtaining insurance and filing a claim for a prior accident that occurred while uninsured.
Vehicle repair	Billing the price of brand-new parts when old parts were used for repairs, sometimes involving collusion between adjusters and body repair shops.
Vehicle smuggling	Involves the purchase of new vehicles that are shipped to foreign ports and reported stolen in claim applications.
Phantom vehicles	Use of legal ownership documents of a car as basis for insurance policy issuance to collect claims, even though the car may not really exist.
Staged accidents	Schemes in which the occurrence of accidents is planned, predetermined, or fabricated and never occurred.
Vehicle Identification Number (VIN) switch	Involves the sale of a wrecked vehicle and reporting it as being repaired. Instead, the VIN plate is switched with that of a stolen vehicle of the same make and model.
Rental car fraud	A person can perpetrate several schemes without owning a car by using a rental car. The most prevalent involve property damage, bodily injury, and export fraud.

for healthcare (Yang & Hwang, 2006). It has been reported by the National Health Care Anti-Fraud Association (NHCAA) that the total losses due to fraudulent healthcare insurance claims in the United States are estimated, conservatively, to have exceeded $100 billion in 2018, approximately 3% of the total expenditure on healthcare that year which was $3.6 trillion (Association, 2018). However, some government and law enforcement agencies estimate the losses to reach as high as 360 $ billion. Sparrow (Sparrow, 2000) was among the first to classify the various types of fraudster behaviours in healthcare insurance into two categories: "hit-and-run" or "steal a little, all the time." With "hit-and-run" fraudsters, they simply submit as many fraudulent claims as possible, disappearing shortly after receiving payment for them. Fraudsters who "steal a little, all the time" try to ensure their actions go undetected so that they can continually make fraudulent claims or billings over long periods of time. The various types of fraudulent activities that can take place to defraud healthcare insurance policies are outlined and described by the FBI (Bureau, 2011) in Table 4.

Traditionally, detecting fraudulent insurance claims relied heavily on manual auditing and inspection by experts in the field, which can be costly and inefficient, especially since these types of claims must be detected before they are paid out. As a result of this time sensitivity, the recognition and attention gained by techniques that fall under the umbrella of machine learning and data mining for detecting insurance fraud have been steadily increasing in recent years. With the constant research and developments in computational capacity and power, these types of techniques demonstrate the potential to detect fraudulent cases in much less time and possibly greater accuracy than manual inspection, subsequently resulting in significantly reduced financial losses. This ultimately translates to a decrease in costs and overall greater potential profits.

4.3. Money Laundering

The process used by criminal or terrorist individuals and organizations to legitimize or "clean" the proceeds of their crimes and disguise their origins is known as money laundering. These crimes are often linked with organized crime syndicates, drug trafficking, sex trafficking, and the financing of terrorists (West & Bhattacharya, 2016). Money laundering is a global problem that incurs significant costs to the world economy due to its destructive effects on national economies. Furthermore, it can jeopardize the stability of and increase the operational risk of financial institutions as money laundering often motivates and fosters corruption in both the public and private sectors (Schott, 2004). This makes countries rampant with money laundering and corruption less attractive to foreign investors, perpetuating the cycle of inequality faced by their citizens (Schott, 2004).

Behavioural patterns of money laundering activity and structural network features are essential to anti-money laundering research. However, traditional research has focused on legislative considerations and compliance requirements and has been methodologically limited to incident identification, avoidance detection and suspicion surveillance (Gao & Ye, 2007). Thus, investigations are generally performed manually, which is time-consuming, resource-intensive, tedious. Therefore, applying anomaly detection techniques may reduce the time to process the voluminous datasets associated with these crimes and even lead to improvements in detection effectiveness and decrease false alarm rates (Gao & Ye, 2007). We note, however, a general lack of available literature studying applying anomaly detection techniques to anti-money laundering in published literature.

Table 4
Descriptions of different types of healthcare insurance fraud schemes (Bureau, 2011).

Type of fraud	Description of fraud
Billing for services not rendered	Involves providers billing for services not rendered or services previously claimed.
Upcoding of services	Practice involving providers submitting claims with reimbursement values greater than the value of the services provided.
Upcoding of items	Billing practice by providers where value of items claimed is greater than value of actual items provided.
Duplicate claims	Services or items that have more than one claim filed for the same thing, changing a portion of the claim such as the date
Unbundling	Practice of submitting bills in a fragmented fashion to maximize reimbursement value for services that may have been billed at a reduced cost together
Excessive services	Involves the provision of medical services or items which are more than a patient's actual needs
Medically unnecessary services	Services not justified by a patient's medical condition or diagnosis.
Kickbacks	The offering, soliciting, or accepting of money or something of value in exchange for the referral of a patient for healthcare services or items

4.4. Other Types of Fraud

Another type of fraud encountered in the financial domain is securities & commodities fraud. This type of fraud can involve a variety of different schemes, the FBI describing some as pyramid schemes, Ponzi schemes, prime bank schemes, high yield investment fraud, advance fee fraud, hedge fund fraud, commodities fraud, foreign exchange fraud, market manipulation, broker embezzlement and late-day trading (Bureau, 2011). With the continuing integration of capital markets, unprecedented opportunities have been created for businesses to raise or access capital and for investors to diversify their portfolios. The increased opportunities also come with a commensurate rise in fraud risk, as such driving the need for research to detect these types of frauds.

Mortgage fraud is defined by the FBI as a material misstatement, misrepresentation or omission by a debtor at any point of the process in which the information is relied upon by an underwriter or lender to obtain a loan (Bureau, 2011). Furthermore, corporate fraud involves the falsification of financial information, self-dealing by corporate insiders or obstruction of justice designed to conceal any of these types of criminal conducts. Finally, mass marketing fraud is a general term for crimes exploiting mass-communications media (Bureau, 2011).

5. Anomaly detection for fraud

Anomaly detection has a significant role in financial fraud detection and is used to identify and extract information from vast data quantities (Ngai et al., 2011). There have been significant amounts of literature applying statistical methods, as well as artificial intelligence and machine learning techniques to approach credit card and insurance fraud detection, the majority of which focus on the latter two. Also, most research papers have begun to shift their focus to unsupervised techniques, many of which addressing imbalanced datasets and the lack of available data.

Frequently used models for credit card fraud are decision trees (DT), support vector machines (SVM), logistic regression (LR), *k*-means clustering, *k*-nearest neighbours (*k*NN) and more (Sahin & Duman, 2011). These techniques can be used individually, or an ensemble technique can be used by using more than one algorithm, which could result in better and more accurate detection. One such example of a popular ensemble learning method in anomaly detection is the RF algorithm. Deep learning anomaly detection techniques have emerged and gained prominence in the last few years, demonstrating significantly better performance than other techniques in addressing real-world problems. These include neural network (NN) architectures of various types, such as convolutional neural networks (CNN), long short-term memory networks (LSTM) and more. Other deep learning architectures gaining more prominence in recent literature are autoencoders (AE) and generative adversarial networks (GANs) (Fiore, De Santis, Perla, & Zanetti, 2019; Wang et al., 2020).

The range of techniques to detect insurance fraud has been found to be more limited than those for credit card fraud. In recent years, much of the research efforts to detect financial fraud seem to have been diverted to focus on the latter. We note that this is a recent change in trend, as it was shown by Ngai *et al.* that the most prominent area of research for financial fraud detection techniques was insurance fraud (Ngai et al., 2011). Similar to credit card fraud, some of the earlier and most popular techniques applied to detect insurance fraud are naïve Bayes (NB) classifiers, LR, DT, SVM and more (Viaene et al., 2002). In recent years, deep learning models like MLPs have been gaining increased research attention for this task, and research efforts have even explored text mining and natural language processing (Wang & Xu, 2018).

In this section, we classify and review the techniques in the surveyed published literature that have been applied to financial fraud detection in recent years based on their learning approach. The four approaches covered are supervised, unsupervised and semi-supervised methods, and briefly, graph-based methods.

5.1. Supervised Methods

i. Support Vector Machine (SVM)

SVMs are promising non-parametric techniques based on statistical learning theory in machine learning that have shown use in classification tasks (Cortes & Vapnik, 1995). These algorithms are especially suitable for binary classification problems like fraud detection due to their unique properties and features. A summary of the published literature employing SVMs surveyed in this paper applied to credit card and insurance fraud detection can be seen in Table 5.

SVMs work by mapping the input space into a higher dimensional feature space to find an optimal separating hyperplane. They can achieve this without introducing any further computational complexity (Cortes & Vapnik, 1995). The ability of SVMs to work with many features has made them attractive for detection tasks involving a highly imbalanced data set due to their ability to extract relevant and important features. Kernel functions are the mechanism behind how this is made possible, thanks to the nature of SVMs possessing the property of kernel representation. Another property of SVMs is margin optimization, which results from the criteria that select the hyperplane, which minimizes overfitting by maximizing the margin of separation from the two classes.

With an input space X and a higher dimensional space H, a kernel function k is defined as $k(x_1, x_2) = \langle(\Phi(x_1), \Phi(x_2)\rangle$, where $\Phi : X \rightarrow H$ is a mapping transforming the input space into a higher-dimensional space. The choice of kernel function is dependent on the application and nature of the task, and the most frequently used are polynomial functions, radial basis functions and linear functions.

Sahin and Duman (Sahin & Duman, 2011) applied SVMs to detect fraudulent credit card transactions, noting the highly imbalanced nature of the data sets involved in their experimentation. To overcome this, the authors use stratified sampling to under-sample the data set's legitimate cases to a meaningful number (Sahin & Duman, 2011). This consisted of preprocessing the data, which first involves determining the most successful features in discriminating fraudulent and legitimate transactions. Stratified samples are then formed of legitimate records using these variables, then subsequently combined with the fraudulent records. Different choices of kernel functions were used to observe the variation in performance. Using data obtained from a national bank, the authors compared the proposed method with various types of DTs such as C&RT, C5.0 and CHAID (Sahin & Duman, 2011). Results showed that the DT models outperformed the SVM models over the test set with accuracies, but the opposite was true when compared with the training set accuracy (Sahin & Duman, 2011). The C5.0 had the overall best performance with a train and test set accuracy of 99.15% and 94.52%, respectively. The SVM models showed identical performance despite the choice of kernel function, with train and test set accuracies of 98.75% and 93.08%, respectively (Sahin & Duman, 2011). This difference in performance between the two sets of data led to conclusions in the literature that SVMs tend to overfit the training data. However, as the amount of training data was increased, this overfitting behaviour became less remarkable, and the performances become comparable (Sahin & Duman, 2011). The authors suggest exploring other techniques such as MLPs and expanding the performance measures to provide more detailed comparisons.

Similar studies by Bhattacharyya *et al.* were published, implementing SVMs, LR and RF to detect fraudulent credit card transactions for comparison (Bhattacharyya et al., 2011). RF is a supervised ensemble method of DT for classification, which addresses the problems of instability and reliability in DT not previously discussed by Sahin and Duman (Bhattacharyya et al., 2011; Sahin & Duman, 2011). RF build ensembles of DTs combined with bagging and are noted to be the ensemble of choice for DTs (Bhattacharyya et al., 2011). Using a data set obtained from an international credit card company, the authors derive custom attributes that address the high dimensionality and heterogeneity issues with credit card data. The authors also suggest random undersampling of the data, specifically, the majority class, as it is generally better than other sampling approaches (Bhattacharyya et al., 2011; Van Hulse, Khoshgoftaar, & Napolitano, 2007). The data set's anomalous labels totalled 2,420 and were split into two subsets of 1,237 and 1,183 transactions, using the first set to populate four training sets with normal labels and the second set for testing. The four training sets were populated with varying amounts of legitimate transactions to create varying fraud rates of 15, 10, 5 and 2 percent for cross-validation and comparison. Experimentation results lead to suggestions by the authors that RF showed overall better performance. However, SVM outperformed logistic regression on precision and F-score measures, the latter of which is the harmonic mean of precision and recall (Bhattacharyya et al., 2011). The accuracy of the SVM was 93.8%, trailing logistic regression at 94.7% and RF at 96.2%. The SVM had the lowest recall of the three techniques of 52.4%, compared to 72.7% by RF.

Table 5
Summary of published literature on SVM-based fraud detection.

Year	Reference	Type of fraud	Method	Comments
2011	(Sahin & Duman, 2011)	Credit card	SVM	SVM with stratified sampling overfit the data and outperformed by DT.
2011	(Bhattacharyya et al., 2011)	Credit card	SVM	LR outperforms SVM. As fraud rate decreases, results become comparable.
2011	(Lu & Ju, 2011)	Credit card	ICW-SVM	ICW-SVM superior to SVM and DT, and computationally more efficient.
2013	(Hejazi & Singh, 2013)	Credit card	OCSVM	One-class SVM outperforms SVM in imbalanced data sets.
2020	(Rtayli & Enneya, 2020)	Credit card	RF and SVM	RF-SVM ensemble accuracy comparable to but less than LOF-IF, however, demonstrated the highest AUC.
2012	(Tao, Zhixin, & Xiaodong, 2012)	Auto-insurance	DFSVM	DFSVM outperforms vanilla SVM in terms of F-score, recall and precision.
2015	(Sundarkumar & Ravi, 2015)	Auto-insurance	*k*RNN-OCSVM [a]	Notable increase in AUC and recall for SVM model, with loss in precision.
2016	(Sundarkumar, Ravi, & Siddeshwar, 2015)	Auto-insurance	OCSVM [a]	*k*RNN identified to slightly limit overall performance, therefore eliminated.

[a] The denoted methods proposed are SVM-based undersampling techniques augmented with a fraud detection system rather than actual classifiers.

Similar to previous findings, the performance of the SVM became comparable as the data set increased in size, and the fraud rates were lower (Bhattacharyya et al., 2011). Furthermore, all three techniques had notably better performance identifying and correctly classifying normal transactions than they did with the fraudulent class (Bhattacharyya et al., 2011). This is apparent from the range of specificity values witnessed, the lowest of which was 97.9% from the LR, then 98.4% by the SVM, the highest by RF of 98.7%. Overall, RF showed better performance than other techniques across all performance measures, capturing more fraudulent cases with fewer false positives. The authors also note the attractiveness of RFs due to their computational efficiency and simplicity of implementation by having only two adjustable parameters (Bhattacharyya et al., 2011).

Finally, the literature concludes by stating that these results were achieved without any parameter tuning of the SVM, which can have a significant effect on its performance. As such, suggestions for future research suggest further investigation of parameter tuning as well as examining differences in fraudulent behaviour among different types of fraud, for example between stolen and counterfeit cards, to derive attributes capturing more comprehensive representations of fraud (Bhattacharyya et al., 2011).

Lu and Ju also proposed an Imbalance Class Weighted SVM (ICW-SVM) for credit card fraud detection, addressing the issue of large scale and dimensionality by utilizing principal components analysis (PCA), a dimensionality reduction technique (Lu & Ju, 2011). This involves extracting the principal components that capture the distribution of the original features of the data, retaining the key features of information. The proposed method uses a Gaussian kernel function to map the support vector. The ICW-SVM then handles the imbalanced data issue by allowing for the adjustment of weights of the normal and fraudulent classes, which alters the hyperplane position as needed (Lu & Ju, 2011). This eliminates the need to use sampling techniques such as SMOTE, random undersampling and others to balance the data set.

For experimentation, the authors use an imbalanced credit card data set from a commercial bank in China, where there is a total of 29 features and 42,928 transactions, the fraudulent class only accounting for 1.57% of all transactions (Lu & Ju, 2011). Following the PCA transformation of the data set, five key features are derived, with the content of each outlined in Table 6 (Lu & Ju, 2011). The total contribution of all five principal components captured 87.87% of the data, with a loss of approximately 12% of the data. The descriptions of feature combinations provided in this work are something rarely ever disclosed in published literature and can be considered a significant contribution by this paper. Addressing the previously mentioned suggestions of Bhattacharyya *et al.* (Bhattacharyya et al., 2011), parameters for the proposed ICW-SVM were found empirically by experimenting with variations in each setting (Lu & Ju, 2011). The ICW-SVM was compared against the standard SVM and DT, and it was demonstrated to be superior in performance and shown to have the highest overall accuracy of the three algorithms at 91.28%. The ICW-SVM also had the highest recall of 85.25%, a significant difference from 53.55% by SVM and 46.98% by DTs (Lu & Ju, 2011). The authors also show that the ICW-SVM is much more computationally efficient with the expansion of the scale of data sets when PCA was performed. The proposed method with PCA executed almost four times faster when compared to the same method without using PCA, sacrificing only 1.4% accuracy to achieve 90.9% (Lu & Ju, 2011). The authors infer that the proposed model is useful and would achieve even better performance in an environment with real data due to its scalability. Further studies were suggested to determine how to effectively select the correct parameters are kernel functions to enhance the proposed technique's performance.

Table 6
Structure of derived attributed from credit card data (Lu & Ju, 2011).

Key features	Content
Negative information of account (X_1)	Default frequency, unpaid credit balance of each cycle
Situation of card holding (X_2)	Overdraft frequency, overdraft limit, maximum number of days overdue
Trading frequency (X_3)	Date of consumption, shopping frequency, store code, number of average consumptions
Ratio of number of transactions (X_4)	Daily transactions/largest transactions number in history
Basic customer information (X_5)	Age, education, occupation, income, housing conditions, industry prospects

Although considered an unsupervised anomaly detection technique, one-class SVMs (OCSVM) are a kernel-based method of SVMs that were proposed by Schölkopf *et al.* (Schölkopf, Williamson, Smola, Shawe-Taylor, & Platt, 2000). They are only trained on the normal instances of data, e.g. the legitimate transactions, to learn the boundary around these points. Any points that do not lie within the boundary are classified as anomalies. In OCSVMs, the input data is mapped into a feature space using a kernel function to find a separating hyperplane, but the main difference from regular SVMs is that the separating hyperplane is between the data points and the origin (Schölkopf et al., 2000). As the name suggests, the hyperplane is optimized to separate the data into one class (e.g. the legitimate transactions), such that all samples not grouped into that class are considered part of the anomalous class (e.g. the fraudulent transactions). An example graphical representation of the separating hyperplane in two-dimensional space based on (Schölkopf et al., 2000) can be seen in Fig. 2. This approach to OCSVM is most prevalent due to its simple implementation and application; however, other approaches exist (Hejazi & Singh, 2013). Tax and Duin describe a hypersphere boundary between anomalies and the normal class as an alternative to a hyperplane in (Tax & Duin, 2001).

Hejazi and Singh proposed an OCSVM approach to classifying credit card fraud using various types of kernel functions such as linear, polynomial and radial basis functions (Hejazi & Singh, 2013). The authors use a German credit card data set from the University of California, Irvine (UCI) repository, which has a total of 1,000 samples and 20 features, 300 of the samples belonging to the anomalous class (University of California, 2000). The literature compares binary classification SVMs balanced with various sampling techniques to the proposed OCSVM with an imbalanced data set. The parameters of the OCSVM were determined empirically and outlined in the literature to observe the effects on the model's performance.

Of the SVM models trained on balanced data sets, both random sampling and spread subsampling had the highest test accuracy of 88.99%, compared to the imbalanced data set with 83.44%. However, the SVM with an imbalanced data set had the best test set accuracy of

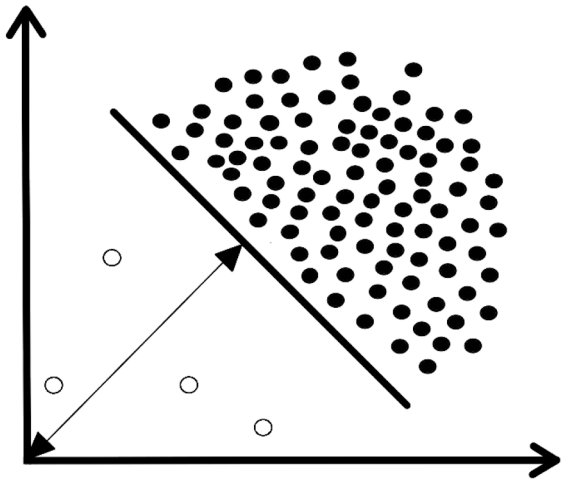

Fig. 2. Two-dimensional graphical illustration of how a separating boundary is determined by OCSVM methods.

77%, compared to 68% by random sampling coming in second. This led to conclusions that the while showing slight improvements in performance with balanced data, SVMs tend to overfit and generalize poorly (Hejazi & Singh, 2013). On the other hand, when compared with the proposed OCSVM, it was shown that OCSVMs with imbalanced data sets significantly outperform regular SVMs with balanced data sets, which tends to overfit the training data (Sahin & Duman, 2011; Hejazi & Singh, 2013). The linear kernel function OCSVM had the highest train set accuracy of 90.16% and a test set accuracy of 91%. The polynomial kernel function of degree three had the second-highest accuracy of 90%, but the highest test set accuracy of 92%. The radial basis function kernel demonstrated comparable performance in train and test set accuracy of 89.66% and 91%, respectively. Overall, the proposed OCSVM model by Hejazi and Singh demonstrated superior performance compared to previous SVM approaches in published literature, generalizing well to unseen data instead of overfitting the training set (Hejazi & Singh, 2013). The computational efficiency of the OCSVM is also preferred, demonstrating to be at least two times faster than SVMs, and executed in less than 0.1 s on average regardless of the type of kernel function used. Most importantly, it is made evident in the literature that parameter tuning and optimization are critical to the performance of the OCSVM, which, if neglected, can result in severe deterioration in the accuracy of the model (Hejazi & Singh, 2013).

More recent studies published in 2020 by Rtayli and Enneya propose an enhanced credit card fraud detection method based on RF as a feature selection algorithm coupled with an SVM to detect anomalous transactions (Rtayli & Enneya, 2020). The RF algorithm selects relevant features from the data to increase the subsequent SVM model's overall performance, which is tasked with classifying the transactions as legitimate or fraudulent. The extraction of non-redundant and robust features is done by utilizing an importance score to select the extremely discriminative features, which helps reduce the dimensionality of the data (Rtayli & Enneya, 2020). Using the metrics of accuracy, recall and area under the curve (AUC), this technique was compared with other methods such as DT, as well as local outlier factor (LOF) and isolation forest (IF), which are unsupervised techniques further discussed in later sections. The authors used a public dataset of European credit card transactions retrieved from Kaggle (Machine Learning Group. (2017), 2017), an online data science community. It consisted of 284,807 total transactions, of which 492 were fraudulent. Results showed that the accuracy of the proposed RF and SVM model had an accuracy of 95.18%, which was less than the other models. Moreover, the proposed model vastly outperformed the three models with 87% recall and IF coming in second with 34%. The AUC of the proposed model was 91%, which also shown to be superior to LOF with 52%, IF with 67% and DT with 50%. Overall it was concluded that the SVM based on RF feature selection had good accuracy, alerted less false positives and was most robust at detecting fraudulent transactions (Rtayli & Enneya, 2020).

For the detection of fraudulent automobile insurance claims, Tao *et al.* proposed a dual membership fuzzy SVM (DFSVM) (Tao, Zhixin, & Xiaodong, 2012). Instead of the binary 0 or 1 classification of SVMs, in this technique, the model assigns each sample with two membership values that denote the probability of belonging to each respective class. This results in twice the amount of training samples that the model makes full use of, helping to overcome the overfitting problem of SVMs (Tao, Zhixin, & Xiaodong, 2012). The choice of the kernel function in the paper was the radial basis function (RBF); however, no discussion was provided on the reasoning behind the choice of kernel. The proposed model was compared against a regular SVM, using automobile insurance data of cases in Beijing. A total of 800 samples were used, with half of the cases being legitimate and the other half being fraudulent. We note that the literature introduces a bias by choosing the number of samples of each case in such a way that the dataset is balanced without the use of proper sampling methods or techniques. The categorical features of the dataset were transformed into numerical encodings, as necessary for this type of model. Hyperparameter tuning of the models was achieved by using a grid parameter search and cross-validation techniques. Experimentation results showed that the proposed DFSVM had improved performance in comparison to previous SVM methods, as evident from F-scores of 91.02% and 87.99%, respectively (Tao, Zhixin, & Xiaodong, 2012). The recall and precision of the DFSVM were also the highest reported, at 91.31% and 90.73%, respectively. The authors conclude by reaffirming the improvements of introducing fuzzy class memberships over previous techniques in previously published literature.

Sundarkumar and Ravi proposed a novel hybrid undersampling method based on OCSVM and reverse *k*-nearest neighbours (*k*RNN) (Sundarkumar & Ravi, 2015). The authors apply the proposed technique to improve the performance of classifiers in detecting fraudulent automobile insurance claims by addressing the problem of imbalanced class distributions. The literature makes use of a public dataset from an automobile insurance company with 15,420 samples, 923 of which were fraudulent, accounting for 6% of the total samples and highlighting the highly unbalanced nature of the dataset. Preprocessing of the data was necessary in order to remove irrelevant features and extract more relevant features from the ones already present. Of the original 31 features in the dataset, only 24 remained, which included some derived features. The methodology consisted of extracting a test set of 20% of the original data by stratified random sampling, which is left untouched so as to validate the efficiency of the proposed model (Sundarkumar & Ravi, 2015). The remaining 80% of the dataset is then assigned as the train set. The next step involves eliminating outliers from the majority class of the train set using *k*RNN and extracting the support vectors of the resulting set using OCSVM. The choice of using SVM is justified by their ability to efficiently perform row dimensionality reduction, implicitly accomplishing undersampling (Sundarkumar & Ravi, 2015). The minority samples are merged with the reduced majority samples from the proposed technique, which are subsequently used to train a classifier with the 10-fold cross-validation method.

To observe the performance of their novel hybrid undersampling approach, the authors trained and compared several classifiers such as LR, MLP, DT and SVM on the merged training set. It was shown that all classifiers displayed improved overall performance, with higher AUC scores than when compared against the same models trained on the original unbalanced dataset (Sundarkumar & Ravi, 2015). The SVM classifier showed the best overall performance, with an AUC of 75.14%, which is a 12% increase from an SVM trained on an original dataset. The DT had an AUC of 74.71%, coming in second place in terms of overall performance. The performance of the SVM and DT were found to be quite similar, with recalls of 91.89% and 90.74%, respectively. This demonstrates the proposed model's ability to detect and accurately classify fraudulent efficiently. The increase in recall of both models is met with a compromise in specificity, which decreased significantly for the DT to 58.69% from 99.83%. The SVM's specificity decreased less drastically from 63.2% to 58.39%. This indicates that with the proposed hybrid undersampling technique, the tradeoff for increased detection of fraudulent cases is with a corresponding increase in the false alarm rate.

Finally, the authors perform a *t*-test to determine the statistical significance of the SVM's performance. The literature finds that all the models are statistically significantly different with respect to the SVM's recall, except for the DT. Consequently, the authors suggest that the DT is most suitable for the proposed novel technique due to it being computationally faster and, more importantly, the fact that it yields 'if-then' decision rules allowing for interpretability of knowledge extracted (Sundarkumar & Ravi, 2015). From the findings of this research, the authors suggest further effort should be allocated towards determining whether the presence of *k*RNN has an influence on the proposed model. If not, then it is hypothesized that OCSVM by itself can undersample the majority class with comparable predictive performance and significantly improved computational speed (Sundarkumar & Ravi, 2015).

As a result, Sundarkumar *et al.* (Sundarkumar, Ravi, & Siddeshwar, 2015) explored employing just the OCSVM to undersample the majority class to determine whether or not the presence of *k*RNN in the

methodology had any effect on performance. In their work, the authors used the same dataset and methodology employed by Sundarkumar and Ravi in (Sundarkumar & Ravi, 2015). Instead, the OCSVM is chosen to handle outlier detection and removal in the majority class and under-sampling based on the properties of support vectors (Sundarkumar & Ravi, 2015). The proposed technique was compared against the results from the previous work. Experimentation showed that the proposed methodology led to slight improvements in the AUC as well as the recall for both the DT and SVM model. These findings led to conclusions that the *k*RNN has no significant effect on classification performance (Sundarkumar, Ravi, & Siddeshwar, 2015). The paper, however, does not investigate or determine the time and economic savings as a result of these findings, which could be an avenue of future exploration.

ii. Neural Networks (NN)

Neural networks are computational algorithms that mimic the working principle of the human brain using elements known as neurons (Ngai et al., 2011). Each neuron has a weight and bias associated with its connections, and the network structure generally consists of an arbitrary number of neurons placed within three main layers: the input, hidden, or output layer. The networks are usually densely connected, meaning input neurons are connected to every neuron in the hidden layer, which are then also connected to every neuron in the output layer.

In this section, we are primarily concerned with feedforward neural networks, also known as multilayered perceptron networks (MLP), which are the simplest type of neural network. These types of networks propagate inputs forward in one direction from the input layer to the output layer (Michelucci, 2018). A graphical illustration of a densely connected NN can be seen in Fig. 3. Each neuron of the network considers its inputs as the outputs of the neurons from the previous layer. The output of each neuron, except in the input layer, must undergo an activation function before it can be an input for the next layer. The choice of activation functions depends on the application, but the most commonly used are either the sigmoid function or the rectified linear unit. Neural networks that fall under supervised learning must learn from the dataset using a learning algorithm. One of the most popularly used algorithms is called the backpropagation of error, which involves calculating the error with respect to the training data. This error is then propagated backwards through the network, correcting the neurons' weights and biases accordingly using gradient descent (Maes & Tuyls, 2002). A summary of the papers reviewed implementing MLPs for fraud detection can be seen in Table 7.

One of the first implementations of MLPs for credit card fraud detection is described by Maes *et al.* in 1993 (Maes & Tuyls, 2002). The authors implement a standard NN trained by standard backpropagation on a labelled transaction dataset. Of the two experiments conducted, the first involved investigating the effects of preprocessing the dataset. Ten features were observed to provide the most substantial results following a correlation analysis, which meant removing one feature (Maes & Tuyls, 2002). This led to the MLP's best result, where for 70% of the true positives, there were only 15% false positives. The second experiment highlighted the importance of parameter tuning, which influences the learning process. The authors note that by decreasing the learning rate at certain intervals, improvements can be made in speed and efficiency (Maes & Tuyls, 2002). Results show that the MLP produces good results; however, it is outperformed by Bayesian networks. Also, in terms of the training time required, the MLP was considerably faster than the Bayesian networks. Future suggestions in the literature suggest pruning algorithms to get rid of neurons and connections practically not used during training and the use of variations such as radial basis networks and in the error function of backpropagation for weight optimization (Maes & Tuyls, 2002).

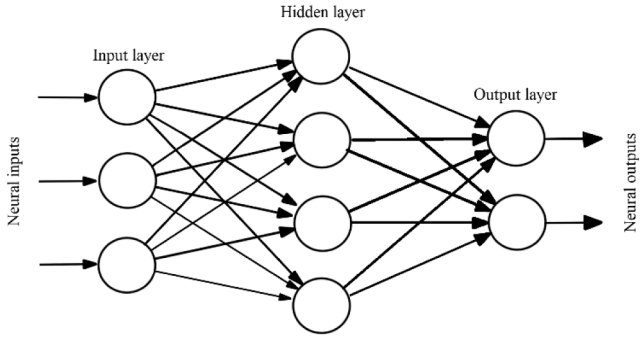

Fig. 3. Labelled visual representation of a densely connected multilayered perceptron.

Table 7
Summary of Published Literature on NN-Based Credit Card Fraud Detection.

Year	Reference	Type of Fraud	Method	Method proposed
1993	Maes *et al.* (Maes & Tuyls, 2002)	Credit card	MLP	MLP trained on preprocessed dataset produced good results but was outperformed by a Bayesian network. Adaptive learning rate proved to be beneficial.
1994	Ghosh and Reilly (Ghosh & Reilly, 1994)	Credit card	MLP	MLP resulted in 20 to 40 percent decrease in economic losses.
1997	Aleskerov *et al.* (Aleskerov et al., 1997)	Credit card	MLP	Using momentum during training improved MLP performance, detecting 85 percent of fraud cases.
2011	Patidar and Sharma (Patidar & Sharma, 2011)	Credit card	GA-designed MLP	Theoretical research proposing GA to address lack of guidelines for selecting number and size of hidden layers to use in a network.
2014	Khan *et al.* (Khan, Akhtar, & Qureshi, 2014)	Credit card	SA-trained MLP	Model achieves high detection rate at cost of increased false positives and is computationally expensive.
2015	Behera and Panigrahi (Behera & Panigrahi, 2015)	Credit card	FCM-MLP	FCM for sample filtering, then MLP trained with SCG to classify suspicious achieved 94 percent accuracy, only 6 percent false alarm rate.
2018	Wang *et al.* (Wang, et al., 2018)	Credit card	WOA-trained MLP	Model generalizes well and addresses problems of NNs overfitting with F-score of 98.4 percent.
2018	Gómez *et al.* (Gómez et al., 2018)	Credit card	MLP ensemble	Ensemble of MLP filters reduce effects of imbalanced data, improve classification performance of classifier.
1992	He *et al.* (He et al., 1997)	Health insurance	MLP	MLP had poor accuracy on its own, which improved when SOM clustering implemented prior to training.
2005	Viaene *et al.* (Viaene et al., 2005)	Auto-insurance	BNN	BNN had greatest AUC out of SVM and DT, but SVM had higher accuracy.
2011	Xu *et al.* (Xu, Wang, Zhang, & Yang, 2011)	Auto-insurance	MLP ensemble	One-step secant and gradient descent with momentum MLPs outperformed resilient backpropagation MLP.
2018	Wang and Xu (Wang & Xu, 2018)	Auto-insurance	LDA-MLP	LDA enabled text mining of documents, improving F-score of MLP by 7.2% and recall by 9.6%.

Ghosh and Reilly implemented and trained an MLP using transaction data from Mellon Bank (Ghosh & Reilly, 1994). The network was trained on a sample of legitimate and fraudulent accounts, followed by a blind test of the trained model on a separate, unsampled, holdout set of transactions. The authors showed that the model could achieve a loss reduction ranging from 20% to 40%, resulting in the installation of the model for use in Mellon Bank's credit card fraud detection systems (Ghosh & Reilly, 1994). Aleskerov *et al.* proposed CARDWATCH to provide a graphical user interface (GUI) for a credit card fraud detection system based on neural networks (Aleskerov, Freisleben, & Rao, 1997). The authors train an MLP on synthetic data representing customer transactions using backpropagation with momentum as the learning algorithm. The momentum term improves efficiency by moving the correction of weights in the direction of the last weight update (Aleskerov et al., 1997). Experimental results showed that the model produced strong results, detecting 100% of the normal transactions and 85% of fraudulent transactions.

Various research papers have been published that have proposed variations of standard NNs that have been used in preceding literature. In the case of credit card fraud, Patidar and Sharma propose and discuss using genetic algorithms (GA) to make decisions about the topology of a NN, which refers to the way neurons should be interconnected, the most common way being fully connected and three-layered (Patidar & Sharma, 2011). This is an attempt to address a problem with neural networks, which is the lack of clear guidelines on setting those parameters of a network that play a critical role in the learning and performance of NNs (Patidar & Sharma, 2011). Although the proposed research seems promising, the authors do not analyze the feasibility of this with experimentation.

Khan *et al.* proposed using simulated annealing (SA), a probabilistic technique based on a thermodynamic heat treatment process, as a training algorithm for NNs (Khan, Akhtar, & Qureshi, 2014). The authors justify this technique to address the issue of standard backpropagation getting stuck at local minima and is computationally less expensive than other *meta*-heuristic algorithms such as GA (Khan, Akhtar, & Qureshi, 2014). Using a dataset from the UCI repository, the authors design an MLP consisting of 20 input neurons, one for each feature of the dataset, and two outputs representing a binary classifier, with an arbitrary 50 neurons in the hidden layer (University of California, 2000). The model took two days to complete training on 75% of the original dataset, leading to a 1% training error. The experimentation shows that the model produces good detection accuracy, identifying 92% of the fraudulent cases and 85% of the legitimate cases. This indicates that there is a trade-off between detecting more of the fraudulent cases and the false positive rate (Khan, Akhtar, & Qureshi, 2014). One issue that was not addressed was the significant amount of time elapsed until the model completed training, which suggests that it might be more computationally expensive than anticipated by the authors.

The Whale Optimization Algorithm (WOA) was recently proposed in 2018 as a learning algorithm for an MLP by Wang *et al.* aimed at solving the problems of slow convergence to local minima and the low stability of backpropagation techniques (Wang, Wang, Ye, Yan, Cai, & Pan, 2018). The WOA is a novel *meta*-heuristic proposed by Mirjalili and Lewis in 2016, which the authors show in their literature that the WOA is very competitive compared to state-of-the-art *meta*-heuristic algorithms (Mirjalili & Lewis, 2016). The authors used the European data set, which consisted of 284,807 transactions, of which 492 were fraudulent. PCA was used to find a lower-dimensional representation of the data, protecting customer privacy with 8 principal components. Experimental results show that using the WOA to update an MLP's weight led to outstanding performance, with the model achieving an F-score of 98.04%, accurately detecting 97.83% of fraudulent transactions and 96.40% of normal transactions (Wang, et al., 2018). Compared to models trained with other *meta*-heuristic algorithms such as GA, the WOA outperformed by every measure. The authors also note that the proposed technique demonstrates superior generalization capabilities

A hybrid approach to MLP credit card fraud detection by Behera and Panigrahi involves implementing a fuzzy clustering detection model (Behera & Panigrahi, 2015). Fuzzy *c*-means (FCM) or fuzzy clustering is a form of clustering that allows each data point to belong to more than one cluster, which has been proposed by Bezdek *et al.* (Bezdek, Ehrlich, & Full, 1984). The authors propose a two-step model, where transactions are initially clustered by FCM based on two features: transaction amount and items purchased. A suspicion score is then calculated based on the Euclidean distance of the transaction from the centroid of the cluster (Behera & Panigrahi, 2015). The suspicion score is then compared against experimentally determined upper and lower thresholds. If it is in the lower threshold, it is a normal transaction; if it is in the upper threshold, it is fraudulent, and if it is between the two thresholds, the transaction is considered suspicious, and the MLP is applied to classify it further appropriately (Behera & Panigrahi, 2015). The authors use Scaled Conjugate Gradient (SCG) as a supervised learning algorithm for the MLP, noting its faster execution time and eliminating the problem of poor behaviour and convergence on large scale datasets (Møller, 1993). Five hidden layers were used for the MLP, which was noted to increase the performance at the cost of increasing the computational time for training. Results showed that the proposed method yielded a true positive rate of 93.90% and a false positive or false alarm rate of just 6.10% (Behera & Panigrahi, 2015). Future suggestions in the literature suggest considering more features by the system to improve the model.

Gómez *et al.* proposed using a set of two consecutive filters in cascade, which consist of an ensemble of NNs, and a final MLP as a classifier (Gómez, Arévalo, Paredes, & Nin, 2018). The purpose of the filters is to classify reject as many legitimate transactions as possible while conserving as best as possible the number of fraudulent transactions. The remaining transactions are then further classified as legitimate or genuine by the last neural network. The authors show that the imbalanced data set used reduced from a ratio of 5000:1 genuine to fraudulent transactions to 420:1 by the first filter and close to 100:1 by the second filter (Gómez et al., 2018). The final MLP's hidden units had ReLU activation functions and softmax output units to provide binary classifications. Backpropagation with stochastic gradient descent as the learning algorithm for the last MLP, applying batch normalization to increase convergence speed during training (Ioffe & Szegedy, 2015). The Value Detection Rate (VDR) was used as a measure of performance for the filters, which is the ratio of fraudulent cases still present after filtering by each model, and results show that a VDR of 93.7% and 66.4% were achieved after the first and second filter respectively. The area under the ROC curve was also inspected, with results showing an average of 86.8% on test data over three months. The recall of the final neural network was shown to be 86.6%, and the authors note that the ratio is a good compromise to avoid saturating back-end alert systems with false positives (Gómez et al., 2018). Overall, the results of the proposed model were deemed promising, and the authors suggest exploring other types of deep learning architectures such as LSTM to exploit inherent temporal information related to fraud patterns and spending behaviour.

The earliest applications of NNs for insurance fraud were published by He *et al.* in 1992, where the authors trained an MLP to classify the practice profile of a sample of general practitioners (He, Wang, Graco, & Hawkins, 1997). The authors propose training the MLP on a dataset of 1,500 general practitioner profiles randomly selected from an Australian population sample. The dataset consisted of 28 features selected by expert consultants in the field. The experts also labelled each practitioner's profile on a scale of 1 to 4, '1′ indicating a high likelihood of abnormal practice, '4′ indicating standard practice, and '2′ and '3′ signifying practice profiles somewhere in between (He et al., 1997). An example of one of the features includes the proportion of initial-to-subsequent consultations, in which practicing practitioners generally have a lower proportion. The dataset was assumed to be balanced, with an equal distribution of classes, and thus was split into two halves for training and testing.

The authors choose an MLP configuration of 28–15-4, signifying 28 neurons in the input layer for each feature, 15 neurons in the hidden layer, and four neurons in the output layer (He et al., 1997). An adaptive learning rate was used, and a weight decay term was added to the loss function of the network to avoid over-training issues. The accuracy of the model's predictions was calculated to determine its classification performance. Results from the proposed MLP were shown to be unsatisfactory, with an accuracy of 63.80% on the train set and 59.87% on the test set. The poor performance of the proposed model is attributed to the degree of uncertainty and inconsistencies in labelling by human experts, especially when there are more than two classes (He et al., 1997). To test this hypothesis, a SOM was implemented and used to cluster the 1,500 practitioner profiles based on the similarity of input features. When setting the number of clusters to 20, the authors observe that the first five group's members are classified mostly as '1′ or abnormal, while the last five groups are mostly '4′ or normal. As for all the clusters in between, there was no clear majority in any of the groupings. Based on this, the authors suggest that the intermediate classifications of '2' and '3' are difficult to distinguish. Therefore, to combat this, the authors propose an MLP using two-class classification instead, using the same methodology as already established. Accordingly, profiles classified with intermediate labels of '3' or '4' were reclassified into classes '1' and '4', respectively. As expected, improved accuracy was observed in both the train and test accuracies of the two-class MLP proposed, achieving 88.40% and 80.93%, respectively (He et al., 1997).

Viaene *et al.* explored the explicative capabilities of NNs trained using a Bayesian learning approach for the detection of fraudulent automobile insurance claims (Viaene, Dedene, & Derrig, 2005). Bayesian NNs (BNN) apply the process of Bayesian inference to find the predictive distribution for the labels of a "test" case, given the inputs for that case as well as the input and label of the training cases (Neal, 1996). This approach is based on MacKay's (MacKay, 1992) evidence framework, which optimizes an automatic relevance determination (ARD) regularized objective function (Viaene et al., 2005; Neal, 1996). This objective function allows for the determination of the relative importance of each input feature to the trained model. In Bayesian learning, with all prior explicit assumptions, the weights and hyperparameters of a network are determined using Bayesian inference to map prior assumptions into posterior knowledge after observing the training data (Viaene et al., 2002). The benefits of BNNs with ARD are that they allow for the inclusion of a large number of potentially relevant features without the detrimental effects generally associated with high-dimensional datasets (Viaene et al., 2005). This eliminates the need to remove irrelevant inputs *a priori,* as they are dealt directly with by the ARD's regularization parameter scheme.

In their work, Viaene *et al.* conducted a baseline benchmark study, empirically evaluating the performances of the DT, SVM and proposed BNN models in detecting fraudulent automobile insurance claims (Viaene et al., 2002; Viaene et al., 2005). A dataset consisting of 1,399 claims made in the state of Massachusetts in 1992 was used. The information in the data was collected by the Automobile Insurers Bureau of Massachusetts and tracked 25 binary features as well as 12 continuous, categorical features considered relevant to fraud investigators and adjusters. The literature shows that the BNN consistently outperformed the other models in terms of the AUC score (Viaene et al., 2005). The AUC of the proposed BNN reached up to 88.49%, compared to only 84.30% and 86.52% by the DT and SVM, respectively. In terms of accuracy, however, the SVM performed best by reaching up to 91.49%. Both the proposed BNN and DT still achieved a comparable accuracy of 91.21%, a negligible difference with the SVM. One important note is that the authors do not mention or discuss the distribution of fraudulent cases in the data and whether it is balanced in nature.

An ensemble of MLPs was proposed by Xu *et al.* using rough set theory-based dimensionality reduction techniques for preprocessing the original data (Xu, Wang, Zhang, & Yang, 2011). Rough set methods have demonstrated the ability to significantly reduce pattern dimensionality, proving to be viable for the front end of NNs, a topic that has been extensively reviewed by Thangavel and Pethalakshmi (Thangavel & Pethalakshmi, 2009). In their research, Xu *et al.* (Xu, Wang, Zhang, & Yang, 2011) use the rough set reduction technique to generate multiple subsets of reductions of the training data. The dataset used in the literature is the same one used in the research proposed by Sundarkumar and Ravi in (Sundarkumar & Ravi, 2015), which consisted of 15,420 samples of automobile claims from an insurance company. The authors further extend their methodology by employing the random subspace method, also known as feature or attribute bagging, to randomly select a generated reduction to train each of the MLP classifiers in an ensemble. The predictions of each MLP are then aggregated using voting or weighted voting strategies, and the final output of the model is then compared against the test set.

Several different MLPs were experimented with and compared to validate the performance of the proposed method, the first of which was trained using gradient descent with momentum and an adaptive learning rate (Xu, Wang, Zhang, & Yang, 2011). One-step secant backpropagation and resilient backpropagation were the two other types of MLPs used. The accuracies and ROC curves for all the models were compared as single classifiers and as an ensemble of two classifiers as outlined in the research methodology. As expected, an improvement in accuracy was achieved by all three models. The highest accuracy achieved of the single classifiers was 83.1% by the one-step secant MLP, followed by 81.9% for the gradient descent MLP and 76.7% with the resilient backpropagation MLP (Xu, Wang, Zhang, & Yang, 2011). The highest accuracy recorded in the ensemble configurations was 88.7%, achieved by both the gradient descent with momentum and the one-step secant MLP. Rather than evaluating the AUC of each model, the authors simply provide visual comparisons of the corresponding ROC curves for each of the models in both their single and ensemble configurations. No discussion is present in the research paper on the ROCs provided for each of the proposed techniques. However, we note that by visual inspection, the one-step secant MLP's ROC tends closer to the top left corner of the graph than the gradient descent MLP. This suggests that the one-step secant MLP has a higher recall value, meaning more efficient detection of fraudulent claims, with fewer false alarms. Future research efforts put forward by the authors involve exploring other types of ensemble classification models, other types of fraud, as well as applying the proposed system for online detection of fraud (Xu, Wang, Zhang, & Yang, 2011).

Studies by Wang and Xu leveraged deep learning and text mining to propose a novel technique for the detection of automobile insurance fraud (Wang & Xu, 2018). Most of the previous studies on insurance fraud detection examine numeric or categorical factors such as the time of claim submission or the make, model, or colour of the insured car. However, textual information in insurance claims has rarely been studied or analyzed to detect fraud. Therefore, the authors propose using Latent Dirichlet Allocation (LDA) based text analytics to extract hidden textual features in the written descriptions provided in claims by claimants (Wang & Xu, 2018). An MLP is subsequently trained on the labelled data with the extracted features to learn to detect fraudulent claims. In natural language processing, LDA is a generative probabilistic model proposed by Blei *et al.,* which uses a Bayesian model to extract and model topics in a text as a probability distribution (Blei, Ng, & Jordan, 2003). Each topic is considered independent of one another, can also be viewed as a probability distribution of the many words in a document (Wang & Xu, 2018). A perplexity term is introduced in the literature to help determine the appropriate number of topics. This term measures the prediction ability of a probability distribution, with more appropriate probability distributions having a relatively low perplexity value (Wang & Xu, 2018).

The authors use a real-world dataset from an automobile insurance company, with 37,082 labelled samples. There are 415 fraudulent claims in the dataset or 1.12%, highlighting the imbalanced nature of the dataset. Consequently, oversampling of the majority is proposed

with SMOTE to result in a dataset of 1,660 samples for each of the legitimate and fraudulent classes. A total of 10 features described each sample, one of which is a textual attribute, which will be processed using LDA to extract key hidden topics and augmenting them as categorical features into the dataset. With the goal of combatting overfitting, 10-fold cross-validation training of 10 different LDA with different numbers of topic groups was carried out. The lowest perplexity is achieved by the model with five topics, each representative of certain words from expert auditors' descriptions in the claim—the most significant words in each topic outlined in Table 8.

In the first topic, the collection of words describe liabilities of accidents, the driver of the car and the third party involved. The second topic describes how the incident was addressed by individuals, as some may report incidents to the police, and others may accept compensation from a private party with pleasure (Wang & Xu, 2018). Topic 3 is a description of the scene of the incident and whether there were any witnesses or police intervention. Driving behaviour is primarily distributed in the key words derived from topic 4, according to descriptions or accounts of suspicious behaviour based on the human expert's individual experience. Information about any damage or personal injury resulting from an incident is found in topic 5. The authors hope that the MLP trained on these added features will be able to extract the experience of human experts in the field.

The MLP architecture consisted of seven hidden layers, which was determined empirically, as it was observed that the amount of computation increased without any improvement in performance past seven hidden layers (Wang & Xu, 2018). Through manual tuning, the number of nodes in each of the layers was chosen as 8, 8, 10, 7, 8, 6 and 4. ReLU activations functions were used in the neurons to avoid the vanishing gradient problem, and a dropout probability of 0.2 for each neuron was employed to help curb overfitting. Furthermore, the MLP made us of an adaptive learning rate, starting at 0.5 and decreasing as the epoch increases. The proposed model was compared against RF and SVM and found to have a recall of 91%, 8% higher than the RF, and 22.8% higher than that of the SVM. This was reasoned to be due to deep learning models being able to capture and understand the abstract experiences extracted by the LDA (Wang & Xu, 2018). The MLP model with LDA extracted features had an accuracy and precision of 91.4% and 91.7%, respectively, the highest of the three models by a margin of at least 5%. The apparent overall superior performance of the proposed model is further supported with an F-score of 91.3%, compared to 81.4% by the RF and 76.2% by the SVM.

A comparative analysis was also conducted in the author's studies, comparing each of the three models with and without the proposed LDA-based text mining and feature extraction method. For the MLP without LDA, the optimal structure was determined to be with seven hidden layers of 6, 7, 9, 4, 5, 4 and 4 neurons in each layer, using ReLU activation functions similar to the previous experimentation. Results showed that the MLP's overall performance without LDA suffered, with the F-score decreasing by 7.2% to 94.1%. This decrease in performance was also significant in terms of recall, which was 81.4%, a decrease of 9.6% from the proposed model using LDA. Overall, the MLP model with LDA proved to be the best detector of fraudulent claims. The authors note that the analysis of the text description derived from human experts may inevitably involve subjective ideas or characteristics. Thus, the literature suggests it may be fruitful to explore ways of eliminating these implicit biases to observe the effect and allow the model to generalize better (Wang & Xu, 2018).

Table 8
Keys words of five topics extracted by LDA in (Wang & Xu, 2018).

Topic	Key words
1	Scene, driver, third-party, liability
2	Report, treatment, compensation
3	Policeman, right, collision
4	Drive, right, collision
5	Front, back, glass, no-injury, impaired

iii. Convolutional Neural Networks (CNN)

The CNN is another type of deep learning architecture that was first introduced in the 1990 s, which is characterized as a network with many layers categorized into the input, pooling, fully connected, and output layers (LeCun & Bengio, 1998). They are named after the fact that the convolutional layer performs a mathematical operation known as convolution on the input. Fig. 4 displays a representation of the structure of a CNN's layers. CNNs have been predominantly used for image-driven pattern recognition tasks due to the essential nature of the input, which is usually in matrix form (O'Shea & Nash, 2015). However, they have been demonstrated to be applicable in other fields and domains by manipulating the structure of the input data.

The various layers of a CNN's architecture, as illustrated in Fig. 4, are responsible for carrying out different operations. As briefly mentioned, the convolution operation is performed in the convolutional layer, which preserves the spatial relationship of the input to extract derived features. Different filters are used for the convolution operation, acting as feature detectors. The pooling layer reduces the dimensionality of the feature maps produced from the convolutional layer using subsampling techniques, retaining the most critical information. Pooling allows for the input to be more manageable, reducing the number of computations and parameters in the network, which helps to control overfitting (Krizhevsky, Sutskever, & Hinton, 2012). The fully connected and output layer is a traditional densely connected feedforward NN or MLP whose input is the output from the convolutional and pooling layer and serves to classify the data. The relevant works involving CNN-based approaches can be seen in Table 9.

CNNs were first proposed for the task of credit card fraud detection by Fu *et al.* in 2016 to capture intrinsic patterns of fraudulent behaviours from labelled data (Fu, Cheng, Tu, & Zhang, 2016). The authors propose this method in response to the overfitting behaviour of MLPs. A labelled credit card dataset from a commercial bank is used and adapted by the authors, who propose a method of feature transformations to capture temporal representations in the data. This involves partitioning features of the data into several groups having different features over different time windows, by which two features of the same type by different time windows have strong relationships and, as such, will be close in position within a feature matrix (Fu, Cheng, Tu, & Zhang, 2016). New features were derived from the raw data, such as the average transaction amount, differences between a current amount and the average amount, as well as other features generated for the fixed time window. A novel feature is proposed by the authors to describe the relationship between a user's transaction amount and the total transaction amount over a period, known as trading entropy, which was implemented into the model (Fu, Cheng, Tu, & Zhang, 2016).

The original one-dimensional feature vectors were reshaped into a feature matrix in which the rows represent different features, and the

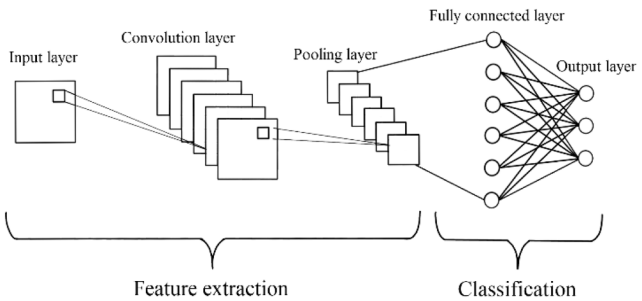

Fig. 4. Schematic representation of the layers of a CNN and the associated functions.

Table 9
Summary of published literature on CNN and LSTM-based fraud detection.

Year	Reference	Type of fraud	Method	Comments
2016	Fu *et al.* (Fu, Cheng, Tu, & Zhang, 2016)	Credit Card	CNN	CNN achieved F-score of 0.33 and outperformed MLPs.
2017	Heryadi and Warnars (Heryadi & Warnars, 2017)	Credit Card	CNN-LSTM	CNN's short-term and LSTM's long-term abilities combined to capture temporal relations. Best AUC achieved of 77%.
2018	Zhang *et al.* (Zhang et al., 2018)	Credit Card	CNN	CNN achieved recall of 94% and precision of 91%, outperforming MLP, but was considerably slower in training.
2009	Wiese and Omlin (Wiese & Omlin, 2009)	Credit Card	LSTM	LSTM outperformed SVMs, as well as the MLP proposed by Maes *et al.* in (Maes & Tuyls, 2002).
2018	Jurgovsky *et al.* (Jurgovsky et al., 2018)	Credit Card	LSTM	LSTM with feature aggregation strategy performed similarly to RF but detected different fraud behaviours. Combination of models suggested.

columns represent different time windows. A heatmap was generated to illustrate the correlations with some examples of fraudulent and legitimate transactions. Discussion of the experimentation results was limited but showed that the CNN model with trading entropy achieved an F-score of 0.33, which was considered superior compared to other techniques such as MLPs, SVMs and RFs (Fu, Cheng, Tu, & Zhang, 2016).

Further studies to evaluate the performance of CNNs for predicting fraudulent transactions were published by Heryadi and Warnars, comparing their performance to LSTMs and a hybrid CNN-LSTM model (Heryadi & Warnars, 2017). The hybrid model attempts to combine the short-term and long-term ability of the CNN and LSTM models, respectively, in capturing temporal relations. The input data is first processed with a CNN, and then the output is fed to the LSTM to be classified. Card transaction data from a bank was treated with undersampling due to the imbalanced nature, and custom features were constructed to capture short and long-term relationships of spending patterns.

The constructed features consisted of vectors of 50 elements, each representing historical transactions of a cardholder over a month, such as the daily transaction amount, average transaction amount over 2 consecutive days and the minimum and maximum amount over the entire month (Heryadi & Warnars, 2017). Principal components of the feature sets were determined to reduce the dimensions using PCA, with experimental trials comparing the performance using 20, 30 and 40 principal components. The authors note that the AUC was the most suitable metric for identifying the best performing model, which was experimentally determined to be the CNN model in all trials where the number of principal components is varied (Heryadi & Warnars, 2017). The highest AUC of 77% was achieved with the CNN with 30 principal components, and the authors note that the CNN had the smallest accuracy score of all the models. Inferences can be made that the CNN model compromises minimizing the number of false positives to catch more fraudulent cases. The authors interpret the results of the CNN's superiority to be due to its strength in capturing short-term trends, corresponding to the short-term nature in which fraudsters commit fraudulent transactions through credit cards (Heryadi & Warnars, 2017).

Studies conducted by Zhang *et al.* on employing CNNs to detect fraudulent transactions produced results that support previous findings of the efficacy of this technique in this application domain (Zhang, Zhou, Zhang, Wang, & Wang, 2018). The authors note, however, that their proposed model can achieve better performance than existing CNN models by only using the raw features from transaction data for training. Transaction data provided by a commercial bank consisted of 5 million transactions over 6 months, with 62 dimensions and positive samples exceeded the fraudulent ones by approximately 33 times. The sequential continuity of the data was ensured by using one-month batches of data for experimentation. Feature engineering methods and statistical analysis of raw data helped determine the significant characteristics for the fraud detection model from the data set, which resulted in the reduction to an 8-dimensional input for direct input to the model. The authors implement a feature sequencing layer prior to the input layer to arrange the features into a one-dimensional vector feature, processed by a one-dimensional convolution kernel feature vector in the convolutional layer. In Fig. 5, a 1×2 convolution kernel is depicted based on (Zhang et al., 2018), showing the transformation process, which generates derivative features from two adjacent features.

The CNN model proposed in this literature was compared against an MLP trained with backpropagation, as well as the CNN model by Fu *et al.* in (Fu, Cheng, Tu, & Zhang, 2016), which uses generated features as opposed to raw data as input to the model. The models were trained and evaluated with the same data sets, except the proposed CNN model was cross-validated over 10 different feature sequencings for the most optimal effect. Results showed that the proposed CNN model had the best performance with a precision of 91%, recall of 94%. This was an increase of 26% and 2%, respectively, compared with the implementation of the CNN model proposed by Fu *et al.* (Fu, Cheng, Tu, & Zhang, 2016; Zhang et al., 2018). In terms of computational efficiency, the MLP was significantly faster than the two CNN models. A single feature arrangement had a training time of 65 s, and the entire model took 752 s to train with all the feature arrangements. The traditional CNN took 352 s to complete the training without the feature derivative work; however, the authors note that considering the time for feature processing would result in the traditional model being considered slower in performance (Zhang et al., 2018). Finally, the effect of the feature arrangements on the classification performance is highlighted, as it was shown was the model produced the best results with the 8th sequencing determined, in which the authors note computational capability is the limiting factor for decreasing the time for training (Zhang et al., 2018). The authors conclude by suggesting the exploration of LSTM networks to capture sequential characteristics more efficiently, as well as further exploration of the influence of characteristic combinations on the model by varying the types of convolution kernel used.

iv. Long Short-Term Memory Networks

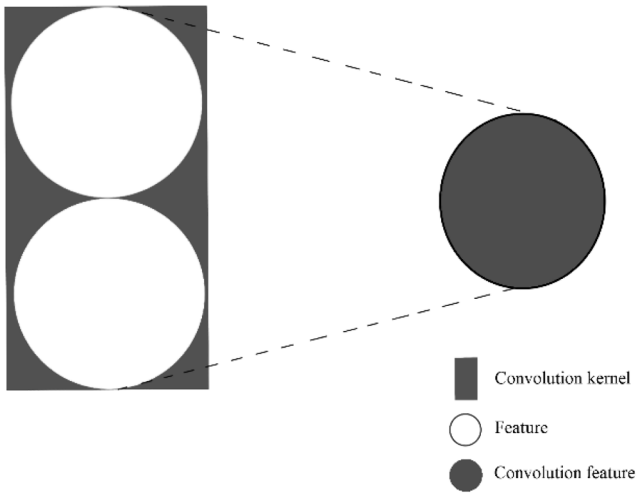

Fig. 5. The convolution process of a 1×2 convolution kernel between two feature.

Long Short-Term Memory networks (LSTM) are an extension of recurrent neural networks (RNN), a form of deep neural network primarily used for time series data proposed by Hochreiter and Schmidhuber in 1997 (Hochreiter & Schmidhuber, 1997). Each neuron in an LSTM is a cell with 'memory' that can store information, maintaining its own state, in contrast to RNNs that merely take the current input from their previous hidden state to output a new hidden state. The improved memory capacity of LSTMs is thanks to the introduction of input and output "gates" into the cell, which were shortly followed by the introduction of the forget gate by Gers *et al.* (Gers, Schmidhuber, & Cummins, 2000). For a thorough review of LSTMs and the different variants, we refer the reader to a recent survey paper by Yu *et al.* (Yu, Si, Hu, & Zhang, 2019).

LSTMs currently constitute the state-of-the-art in many real-world applications such as text, writing and speech recognition, as well as natural language processing. They are also well-known for addressing the vanishing gradient problem that is generally associated with RNNs (Hochreiter & Schmidhuber, 1997). Only very recently have LSTMs been applied to detect fraud, of the early studies implementing already discussed were by Heryadi and Warnars (Heryadi & Warnars, 2017). They have been recognized by other authors, such as Gómez *et al.* (Gómez et al., 2018), and have even been suggested to show promise in detecting fraudulent credit card transactions or insurance claims. Despite their promise, their implementation in research is rare, and there is an apparent lack of available literature studying their performance to detect financial fraud. For a sumary of the relevant literature on LSTM-based methods see Table 9.

Wiese and Omlin (Wiese & Omlin, 2009) are among the first to explore developing a credit card fraud detection model with LSTMs. The authors cite that current fraud detection measures in place are plagued by misclassifications and that their usefulness is hampered by high false-positive rates (Wiese & Omlin, 2009). The literature proposes that by using the LSTM to analyze sequences of transactions as a whole instead of as individuals, the system will be able to capture more temporal relationships and behaviours and ultimately become more robust to minor fluctuations or shifts in legitimate spending behaviour (Wiese & Omlin, 2009). In the studies, the standard gates such as the input, output, and forget gates are used in the LSTM's architecture. Standard preprocessing of the data was conducted, which involved encoding the symbolic features, such as the transaction data into numerical values and standard manual feature selection. Furthermore, a statistical value known as the transaction velocity was calculated and used as an additional input feature. The transaction velocity, in a fraud context, is calculated by the number of transactions the took place in an account during a pre-specified timeframe (Wiese & Omlin, 2009). Different velocities can be calculated by grouping certain merchants into one velocity calculation.

In their studies, Wiese and Omlin compared the performance of the proposed LSTM credit card fraud detection model against another that employed SVMs (Wiese & Omlin, 2009). A dataset of credit card transactions consisting of 30,876 transactions was used, with the fraudulent class accounting for less than 0.1% of the total cases. The dataset was reordered based on the transaction date and time. Half of the dataset was set aside for training, the other half for testing. The architecture decided upon for the LSTM included was determined through empirical analysis and consisted of two memory blocks with two memory cells each, densely connected to the input and output neurons (Wiese & Omlin, 2009). A total of 30 trials were conducted with 100 training epochs, which was also empirically chosen since it resulted in the best generalization performance. During the training of the model, when a misclassification error occurred, the training on the current transaction sequence was restarted, and this process was repeated until all the transactions of a sequence were correctly classified (Wiese & Omlin, 2009). Then, a new sequence is randomly selected from the training set for the next iterations, and at the end of the training cycle, the predictions of the model for both the training and testing set were recorded to draw up ROC curves for an AUC comparison.

From the 30 trials conducted, the average AUC of the proposed model on the training and testing set was 97.57% and 98.22%, respectively, demonstrating remarkably impressive separation between the classes (Wiese & Omlin, 2009). As expected by the authors, the proposed model outperformed the SVM, which was considered a remarkable feat as the SVM also performed exceptionally well. The SVM achieved an AUC of 89.32% on the testing set. However, the literature suggests that it is possible that the kernel and hyperparameters determined were not the most optimal and required more thorough tuning (Wiese & Omlin, 2009). Furthermore, the training speed of the LSTM was slower than the SVM, but once trained, it had a much higher classification rate of 2,690 transactions per second, compared to 213 transactions per second by the SVM. Wiese and Omlin (Wiese & Omlin, 2009) compared the LSTM fraud detection model against the MLP technique proposed by Maes *et al.* (Maes & Tuyls, 2002) and found that their model was superior in performance. The MLP proposed by Maes *et al.* reported a recall of 68% at a false alarm rate of 10% and a recall of 74% at a false alarm rate of 15%, while the LSTM by Wiese and Omlin had average recalls of 98.9% and 99.45% at 10% and 15% false alarm rates, respectively (Maes & Tuyls, 2002; Wiese & Omlin, 2009).

A detailed discussion is provided on the limitations of the model, as well as future suggestions for avenues of research to explore for improved fraud detection performance. First, it is noted that the LSTM was applied to a set of dissimilar time series of variable length, which is considered a gross overcomplication, and as such future efforts can be allocated towards using LSTMs to model singular time series (Wiese & Omlin, 2009). Also, the topology of the network used in the studies was quite simple, with only two blocks of two memory cells each, which was attributed to Occam's razor, stated simply as "the simplest explanation is usually the right one." However, the authors mention that this leads to fewer weights, opening up possibilities that may allow for storing these weights on chips of next-generation payment cards (Wiese & Omlin, 2009). The possible benefit of such an implementation is that point of sale terminals would be able to conduct initial fraud detection using the weights stored on the chip, forwarding only transactions with high degrees of fraud to the host system easing some of their burden (Wiese & Omlin, 2009). Essentially, this enables the possibility for true online fraud detection systems, where the detection of fraud may even equate to its prevention. Finally, Wiese and Omlin also suggest investigating unsupervised methods, citing that in many cases where fraud detection systems are implemented in banks or countries relatively new to these techniques, there is usually a lack of records or data identifying the fraudulent cases (Wiese & Omlin, 2009). This eliminates the possibility of using supervised learning without resorting to human experts for manual labelling, which is both very costly and time inefficient.

Jurgovsky *et al.* (Jurgovsky et al., 2018) proposed an LSTM model for detecting credit card fraud, but instead of modelling transaction sequences like in previous research, the authors make use of a novel feature aggregation methodology explicitly proposed for credit card detection models by Bahnsen *et al.* in (Bahnsen, Aouada, Stojanovic, & Ottersten, 2016). The feature aggregation strategy involves grouping transactions made during the last specified period, initially by card or account number, followed by the type of transaction, the merchant group, country or others, and then calculating the number of transactions or the total amount spent in all those transactions (Bahnsen et al., 2016).

Bahnsen *et al.* note that the aggregated features still have information missing that they do not capture, and as such, take the methodology a step further by deriving time features (Bahnsen et al., 2016). This involves analyzing the time of the transaction, the reasoning being that customers are expected to make transactions at similar hours. An issue arises when analyzing time features, specifically when looking at the mean of certain transactions times, which is that it is easy to make the mistake of using the arithmetic mean. The arithmetic mean does not take into account the periodic behaviour of time features. For example, the

arithmetic mean of four transaction times made at 2:00, 3:00, 22:00 and 23:00 is 12:30, which is counterintuitive as there were no transactions made close to or at that time (Bahnsen et al., 2016). To overcome this limitation, the authors propose modelling the time of transactions as periodic variables using the von Mises distribution, a method of statistical analysis for circular data (Fisher, 1993). This allows for the determination of a period mean, which is a more realistic representation of the transaction times than the arithmetic mean.

The studies by Jurgovsky *et al.* used the previously described feature aggregation strategy to train an LSTM model and compare its performance against an RF model in detecting credit card fraud (Jurgovsky et al., 2018). Two different datasets of credit card transactions were used, one consisting of online e-commerce transactions and the other of face-to-face or point of sale (POS) transactions. The size of both datasets was considerable, both consisting of millions of samples. More specifically, the e-commerce dataset has 2.9 million instances in the training set, 600 thousand instances in the validation set and 3.3 million samples in the test set. The face-to-face dataset has 4.3 million training set samples, 700 thousand samples in the validation set and 4.7 million samples in the test set. In the e-commerce dataset, the number of fraudulent samples comprised only 0.08% of the entire dataset, and similarly, in the face-to-face dataset, they were only 0.065%. Two transaction sequences were constructed for comparison, either of length 5 (short) or length 10 (long). The performance of both the RF and LSTM models were observed and recorded with both lengths of transaction sequences using the feature aggregation methodology, as well as with only the raw features.

Several metrics were employed in the literature to compare the performance of both classifiers, as well as to observe the differences in types of fraud detected in every experiment. The choice of these metrics is guided by two criteria: robustness against the imbalanced class and attention to business-specific details and interests (Jurgovsky et al., 2018). The first was the AUPRC, noted for its benefits and robustness in imbalanced settings in fraud detection. More specifically, for being sensitive to the number of false positives and capturing the effect of the large numbers of genuine samples on the performance of the classifiers (Jurgovsky et al., 2018). (Jurgovsky et al., 2018). Furthermore, the Jaccard Index was used to investigate the qualitative differences of the two approaches, highlighting the similarity of classifiers based on the type of fraud detected in the form of a confusion matrix.

Based on the results from the experimentation, there were no clear distinctions in performance between both the RF and proposed LSTM models as both performed comparably to each other. However, there was a clear improvement for both classifiers in performance when the aggregated features proposed in the studies were used. In the face-to-face transaction dataset, the LSTM improved from 20% to 24.6%, and the RF from 13.8% to 24.1% using the case of the short sequences (Jurgovsky et al., 2018). Using longer sequences in training resulted in slightly weaker performance with the LSTM model, with an AUPRC of 23.6% with the aggregated features, outperformed by the RF at 24.2%. The e-commerce dataset showed a similar trend, but with better overall performance than the face-to-face dataset, where the LSTM and RF with long sequences had AUPRCs of 40.2% and 40.4%, respectively (Jurgovsky et al., 2018). These AUPRCs are a significant increase than the case without the aggregated features, further demonstrating the viability of the methodology. The most interesting results were observed using the Jaccard Index, which presented a heatmap comparing the degree of similarity of the types of fraud detected by each type of model. In terms of the types of frauds detected by each of the two models when compared with themselves, there was a consistency in the similarity of the types of fraud detected. However, this property was slightly more substantial in the RF models, as the LSTMs show slightly more variation. When the LSTM and RF models were then compared with each other, it was apparent that the LSTM captured a more comprehensive range of different types of frauds. Further, it was shown that the two models detect very different types of fraudulent card behaviour (Bahnsen et al., 2016).

It is postulated by Jurgovsky *et al.* that the combination of two models working together in a fraud detection system may lead to even better results due to the different types of frauds detected by both models, as well as the overall more exhaustive range of types of fraudulent transactions that would be detected (Jurgovsky et al., 2018). Several observations are also detailed in the study in regards to the architecture of the LSTM, which although specifics of are not provided in the study, the authors mention the model is prone to overfitting when there are only a few nodes (Jurgovsky et al., 2018). This was combatted by incrementally increasing the number of nodes, which produces the optimal results when combined with an ADAM optimizer instead of SGD.

v. Naïve Bayes

The naïve Bayes (NB) classifier is a popular supervised model and is often considered as the simplest form of Bayesian network classifier (Friedman, Geiger, & Goldzsmidt, 1997). The model learns the conditional probabilities of each attribute in the training set, given the class label. Classification is then done by applying Bayes' theorem to compute the probability of the label given the particular instance's features or attributes (Friedman et al., 1997). An assumption is necessary for the feasibility of this computation: that all the features in the dataset are independent of the value of the class labels. Considering input data $x = (x_1, \cdots, x_n) \in \mathbb{R}^n$ and target labels y, the structural assumption simplifies the model to:

$$p(x|y) = \prod_{m=1}^{n} p(x_m|y) \tag{1}$$

The assumption of the independence of the predictors given the class is, for many domains, arguably somewhat restrictive. Nonetheless, the NB classifier has often been reported to demonstrate surprisingly good performance based on an extensive history of empirical studies (Viaene et al., 2002; Viaene, Derrig, & Dedene, 2004). This was true even in cases where the underlying assumption of independence was considered unrealistic (Friedman et al., 1997).

Studies by Viaene *et al.* (Viaene et al., 2004) proposed NB classifiers to predict fraudulent automobile insurance claims, similar to their previous research in (Viaene et al., 2002). The authors propose boosting the model using the AdaBoost algorithm proposed by Freund and Schapire (Freund & Schapire, 1996). The basic idea of boosting is the construction of classifiers in sequence, trained on various versions of the original training set. The various training sets are formed by resampling or reweighting, where any data instances poorly classified by one of the classifiers in the sequence have a more significant weighting in the next model. Upon termination of training after a fixed number of iterations, the classifiers are then combined using a majority voting scheme. In AdaBoost, each classifier's vote is adaptively weighted according to its quality of classification (Ridgeway, Madigan, & Richardson, 1998). The concept of perturbing the data in stages allows the NB model to focus incrementally on the regions of data more difficult to learn (Viaene et al., 2004). The proposed model is extended further with a weight of evidence voting formulation for the case of NB with AdaBoost, as proposed by Ridgeway *et al.* (Ridgeway, Madigan, & Richardson, 1998). The weights of evidence voting facilitates more interpretable results and explanations that are otherwise eliminated by AdaBoost whilst retaining the predictive performance of the NB classifier (Ridgeway, Madigan, & Richardson, 1998).

In the literature, the Massachusetts automobile claim data set was used, which consisted of 1,399 samples of data. Two-thirds of the dataset was used for training the proposed naïve Bayes AdaBoosted weights of error (ABWOE) classifier, which consisted of a sequence of 25 base classifiers. The model was compared against a simple NB classifier, an AdaBoosted classifier (AB), as well as a model using majority voting

instead of the proposed method. The evaluation of the models was based on the criteria of accuracy, AUC, and a logarithmic score ($\overline{L}$) such that $\{\overline{L} \in [0, \infty)\}$, zero being optimal (Viaene et al., 2004). The proposed ABWOE classifier outperformed the rest of the models for all performance measures, with 84.43% accuracy, 89.19% AUC and the lowest $\overline{L}$ of 0.3697. The proposed voting framework is confirmed to be superior, as the model with majority voting had an AUC of 50%, indicating no discrimination capacity, and output predictions are random guesses. The plain NB had an accuracy of 83.03% and AUC of 88.56%, slightly shy of the ABWOE's performance. The AB model's accuracy and AUC, 84.41% and 89.08%, respectively. These results are comparable to that of the ABWOE, but the proposed model was ultimately considered superior with a $\overline{L}$ less than half that of the 0.8789 achieved by the AB. In sum, the authors recognize the ABWOE scoring framework's accessibility, interpretability and general applicability to be prime advantages of the model, allowing for flexible human expert interaction and tuning (Viaene et al., 2004).

A similar study conducted by Phua *et al.* studied the effects of using a hybrid stacking-bagging training approach using an NB, C4.5 DT and MLP (Phua et al., 2004). The purpose of the research is to address the issues of skewed or imbalanced associated with the poor performance of classifiers. Bagging makes use of base classifiers such as NB, fitting them with randomly sampled subsets of the training data. Each base classifier then votes on the class of a sample, and the majority vote is chosen as the final classification. The difference between bagging and boosting is that in bagging, the models learn independently from each other in parallel, compared to in sequence with one another in boosting. Stacking takes this one step further by using the output of the base models in bagging to train the *meta*-model to select the best base classifiers and then combine these predictions with bagging (Phua et al., 2004).

In the literature, Phua *et al.* used a labelled automobile insurance claim dataset consisting of 11,338 samples, a fraud rate of just 6%, described by 25 categorical and six numerical features (Phua et al., 2004). Several other features are derived in order to help increase the predictive accuracy of the models. The low fraud rate of the dataset brings about the problems of imbalanced datasets, especially in high-dimensional feature spaces such as the one used in this study. The authors partition the genuine, non-fraud samples in the dataset into 11 sets of 923 instances. Then, these sets of genuine samples are merged with fraudulent samples, making 11 partitions of fraud to genuine claim ratios of 40:60, 50:50 or 30:70. Minority oversampling with replacement is the sampling approach used by the authors to distribute the partitions. By creating these 11 partitions, the problems associated with an imbalanced dataset are eliminated, and the time complexity is reduced for the learning models (Phua et al., 2004). Interestingly, a financial cost model is developed by the authors to observe the actual performance of a model in terms of cost savings. The proposed cost model follows two assumptions: all fraud alerts must be investigated, and the average cost per claim must be higher than the average cost per investigation (Phua et al., 2004). The authors use statistical approximations from the year 1996 for the average cost per claim at USD$2,640 and the average cost per investigation at USD$203 (Phua et al., 2004). The cost model is comprised of four components summarized in Table 10, which illustrates the higher cost associated with false alarms since they incur both investigation and claim costs.

Table 10
Components of cost model for insurance fraud detection in (Phua et al., 2004).

Type of claim	Cost
Detected fraud (DF)	Number of cases detected * Average cost per investigation
False Alarms (FAL)	Number of false alarms * (Average cost per investigation + Average cost per claim).
Undetected fraud (UDF)	Number of cases undetected * Average cost per claim
Genuine (G)	Number of genuine claims * Average cost per claim

According to the authors, there are two extremes for the performance of the model. At one extreme, no fraud detection takes place, and all claims are regarded as genuine (no action). At the other extreme, the model achieves perfect performance and detects all fraudulent claims with no false alarms (optimal scenario). Based on these extremes, the cost model is described by Phua *et al.* as follows (Phua, Alahakoon, & Lee, Minority report in fraud detection: classification of skewed data, 2004):

$$Savings = Noaction - [UDF + FAL + G + DF] \tag{2}$$

$$\%Saved = \frac{Savings}{Optimalscenario} * 100\% \tag{3}$$

Results of the experimentation with the proposed model were measured using the cost model developed and overall accuracy. When compared against other classification models using just a single classifier, an ensemble of classifiers or just bagging, the proposed method was deemed superior as it resulted in the most significant cost savings of $167,000 with an accuracy of just 60% (Phua et al., 2004). These financial savings are a 29.7% reduction in cost compared to a perfect classifier. Other models, such as one using just a C4.5 DT trained on an under-sampled 40:60 fraud to genuine partition, generated cost savings of $165,000 and accuracy of 60%, which is 0.3% less than the proposed stacking-bagging method. Surprisingly, some MLP models resulted in very little savings, the worst one even causing a loss of $6,000, even though it had an accuracy of 92% (Phua et al., 2004). An inverse relationship was observed between the cost savings and the accuracy of a model, which was attributed to the costs associated with an increased number of false alarms at the cost of detecting more fraudulent cases. As such, the importance of utilizing more important measures is apparent in this scenario.

The stacking-bagging model considered 33 classifiers trained on different partitions of the original dataset. From those classifiers, only the top 15 were bagged to produce the best predictions, of which nine were C4.5 models, four were MLPs and two NB classifiers. A limitation identified in this research is the simplicity of the cost, and the small size of the dataset, for which the authors suggest exploring SMOTE, as it has been shown to handle imbalance data issues better (Phua et al., 2004). The availability of more datasets would also allow for verification of these results.

vi. Other Supervised Methods

Mahmoudi and Duman proposed the Modified Fisher Discriminant Analysis (MFDA), an adaptation of Linear Discriminant Analysis (LDA), for a credit card fraud detection system (Mahmoudi & Duman, 2015). LDA is a supervised learning method that has been used for dimensionality reduction and has even been applied to classify data by determining decision boundaries or surfaces between regions that are linear functions of input vectors. The MDFA adjusts the Fisher criterion in LDA models by applying a weighted average on both classes, defining the weights to be the total available credit limit that is usable on each card (Mahmoudi & Duman, 2015). This weighting biases the linear discriminant towards the cases that are more financially beneficial to detect accurately, as opposed to merely maximizing the correct number of instances classified. Thus, instead of maximizing the number of correctly classified instances, the MDFA will ensure that the critical and cost-effective instances are classified correctly as best as possible (Mahmoudi & Duman, 2015).

The authors conducted experimentation with the proposed MDFA, comparing it with the performances of DT, MLP, NB and standard LDA models. A dataset provided by a Turkish bank was used in the studies, which consisted of 8,448 legitimate and 939 fraudulent transactions described by a total of 102 features. Feature selection was achieved using a DT model, using the selected features to train the rest of the models. The dataset was divided into three portions for cross-validation.

The literature's evaluation of the model's results is provided in an economically-oriented focus, rather than observing the actual prediction accuracy or performance. Instead of observing popular classifier performance measures such as the recall, precision or overall accuracy, the authors use a cost model to determine the percentage of cost savings achieved by each model (Mahmoudi & Duman, 2015). Furthermore, only the top 313 ranked fraudulent predictions of each model were considered in the paper's discussion.

It was shown that the proposed MDFA achieved the most significant overall savings increase of 90.8%, classifying 236 out of 313, or 75.3% of all samples correctly (Mahmoudi & Duman, 2015). MLPs showed the worst cost savings of all models, a still significant increase of 86.30%, classifying 73.5% of the fraudulent transactions correctly as fraudulent. The LDA detected 76.7% of the fraudulent cases correctly, more than the proposed MDFA. However, the LDA was inferior in terms of cost savings by almost 4% compared to the MDFA, with a savings increase of 87.15%. Overall, the authors find the results of the studies and proposed model to be favourable and direct future efforts towards investigating other types of linear discriminant functions such as the Linear Perceptron Discriminant (LPD) to develop iterative linear discriminants (Mahmoudi & Duman, 2015). Furthermore, the consideration of misclassification costs of genuine samples could be variable instead of fixed, limiting the number of false alarms and dissatisfaction amongst consumers, which could result in variable churns costs (Mahmoudi & Duman, 2015).

A detection system for fraudulent automobile insurance claims was proposed by Dhieb *et al.,* who proposed extreme gradient boosting (XGBoost) with DTs (Dhieb, Ghazzai, Besbes, & Massoud, 2020). XGBoost is a system proposed by Chen and Guestrin (Chen & Guestrin, 2016) specifically designed to optimize memory usage and exploit the computing power of modern hardware, resulting in increased execution speeds and performance (Dhieb et al., 2020). In the proposed methodology, boosting is used to build sequential sub-trees from an original DT, where each sub-tree reduces the error of the previous sub-tree. This results in an update of the residuals of the cost function employed and reducing the error.

The XGBoost DT model proposed in the literature was evaluated against a standard DT, NB classifier and *k*NN. The dataset used comprised of more than 64,000 automobile insurance claims, and the authors do not mention the ratio of fraudulent to legitimate cases or whether the classes were balanced or not. Furthermore, other than a brief discussion of minor preprocessing steps to handle errors or missing values, no insight is provided into any feature selection or extraction techniques or steps. However, a confusion matrix was developed to display the correlations between each of the features in the dataset. Following training and testing, the results of the experiments showed that the proposed detection system had the highest score across all measures except training time (Dhieb et al., 2020). This included an accuracy of 99.25%, precision of 99.28%, recall of 99.2% and an F-score of 99.26%. The worst performance was displayed by the *k*NN algorithm, with an F-score of 0.255 (Dhieb et al., 2020). The DT model came was the second-best model, with an F-score of 89.2%; however, it exceeded the XGBoost in terms of computational efficiency with a training time of 0.471 s, less than half of 9.95 s by the XGBoost (Dhieb et al., 2020). The NB classifier also performed poorly, with an F-score of 42.5%, and as such, the authors conclude the research in favour of their proposed model.

5.2. Unsupervised Methods

i. Isolation Forest (IF)

A novel unsupervised approach to anomaly detection known as IF was proposed by Liu *et al.* in 2008 to explore the concept of explicitly isolating anomalies (Liu, Ting, & Zhou, 2008). The concept of isolation, which according to the authors, has not been studied in current literature, enables the method to exploit sub-sampling to create an algorithm with a low constant linear time complexity and a low memory requirement (Liu, Ting, & Zhou, 2008). IF is also capable of scaling up to handle massive data sets and high-dimensional problems with many irrelevant features.

Isolating samples involves building ensembles of Isolation Trees (iTrees) to separate them from other samples (Liu, Ting, & Zhou, 2008). Partitions are generated by selecting a random feature and then arbitrarily determining a split value between that feature's maximum and minimum values. This partitioning is done recursively, which can be represented by the iTree structure, and the number of partitions required to isolate a point is equivalent to the path length from the root node to a terminating node (Liu, Ting, & Zhou, 2008). The premise is that anomalies are easy to isolate and require fewer partitions. Conversely, normal observations are difficult to isolate and require more partitions.

Two input parameters are required by the IF algorithm: the sub-sampling size ψ and the number of trees t. The sub-sampling size ψ controls the size of the training data, and the authors find empirically that 2^8 or 256 generally provides enough detail across a wide range of data. The literature shows that there is no need to increase ψ further as it could cause increases in processing time and memory size without no increase in detection performance (Liu, Ting, & Zhou, 2008). The number of trees controls the size of the ensemble, and the authors show in the literature that the path lengths usually converge well before $t = 100$ (Liu, Ting, & Zhou, 2008).

Efforts to apply the IF algorithm to detect credit card fraud have only emerged in recent years. In 2018, the IF was proposed to detect fraudulent transactions in credit cards by Ounacer *et al.* (Ounacer, Ait El Bour, Oubrahim, Ghoumari, & Azzouazi, 2018). The authors aim to address the issues of supervised learning and demonstrate high accuracy and detection performance in real-time. The training phase of the process involves building the IF model using iTrees with subsamples of the training set, and the testing phase then passes each sample through the model to calculate the number of splits required across all trees to isolate that observation and return an anomaly score between 0 and 1, where 1 indicates to fraud, and the threshold is set at 0.5 to classify samples (Ounacer et al., 2018). The proposed method was compared with other unsupervised models such as OCSVM, LOF and *k*-means clustering using the European credit card data set containing 284,807 transactions with just 492 fraudulent transactions. For privacy reasons, all input features except for time and amount were transformed using PCA, and the features used for the model consisted of the 28 principal components obtained. Results showed that the IF had the best performance with the highest accuracy and AUC of 95.12% and 91.68%, respectively (Ounacer et al., 2018). Coming in second is the *k*-means clustering algorithm with 90.12% and 51.91% for accuracy and AUC, respectively. The authors conclude by highlighting the capability of the IF algorithm in detecting fraudulent transactions and suggest exploring the implementation of the model for online, big-data processing architectures.

According to Stripling *et al.*, it has been reported among 69% of industry experts that there is a belief of an increase in workers' compensation fraud (Stripling, Baesens, Chizi, & vanden Broucke, 2018). Workers' compensation insurance policies cover the costs the arise when employees sustain injuries or illnesses while on the job. The authors propose using a model that utilizes the IF algorithm to compute an anomaly score from features in the training set to create a new feature. The computed anomaly score for each sample is augmented into a new training set, and the features used to compute the anomaly score are omitted from this set. This newly formed training set is then used to train a classifier, with the goal of improving its performance. According to the literature, the selection of nominal attributes that should undergo the IF transformation is driven by expert human knowledge to obtain meaningful scores (Stripling et al., 2018).

In their work, Stripling *et al.* empirically evaluate the performance of their proposed technique on a real-world dataset of workers' compensation claims received from an anonymous European organization

(Stripling et al., 2018). The dataset has 9,572 labelled samples with 23 features, only 3 of which are described due to confidentiality reasons: the type of injury, the policyholder's industry sector and the registered duration of incapacity (Stripling et al., 2018). Some classifier models chosen for comparison were LR, DT, RF, linear kernel SVM and radial basis function kernel SVM. Each of the model's AUC scores was compared with and without the proposed anomaly scores computed by the IF. The highest AUC was achieved by the linear SVM without the anomaly scores, at 87.72%, compared to 80.75% with the IF anomaly scores (Stripling et al., 2018). It was noticed that there was around a 3–5% decrease in AUC of all the models with the IF anomaly scores. However, the authors note that this was not necessarily the only criterion that should be considered. When the results of the model were presented to investigators from the company providing the dataset, it found higher appreciation among the investigators as it uncovered fraudulent claims not previously detected by human experts. The investigators confirmed that the model demonstrated high detection accuracy and practicality, especially with cases that previously remain undetected (Stripling et al., 2018).

The authors conclude by highlighting the benefits of the model, mainly since it works in an unsupervised fashion. This is especially useful in cases where labels may be difficult or expensive to obtain for each sample in a dataset. Future suggestions to improve the model included exploration of techniques to automate the feature selection process. The scalability of the model is also an avenue for further investigation and studies

ii. Self-Organizing Maps (SOM)

The SOM is a type of neural network employing unsupervised learning that configures the neurons of the network according to the input data's topological structure (Kohonen, 1990). This process is known as self-organization, which iteratively tunes the weights of neurons to approximate the input data, resulting in the clustering and profiling of the data set (Quah & Sriganesh, 2008; Kohonen, 1990). The neurons in a SOM are arranged in a matrix structure that maps inputs from a high-dimensional space to the two-dimensional array of neurons (Quah & Sriganesh, 2008). This mapping is designed to model similar input vectors as neurons that are closer together in the resulting matrix, providing a visualization of the input.

A variety of distance measures can be used during the iterative training process to group the nodes, such as the Euclidean distance, Manhattan distance, Chebychev distance and more (Zaslavsky & Strizhak, 2006). After training, the data in the data set gets classified into legitimate or fraudulent sets by self-organization, and any new transaction thereafter also undergoes the same process before being passed into the SOM. These new transactions are classified as either legitimate or fraudulent based on whether they are within a specific threshold value (Quah & Sriganesh, 2008). Rather than having just binary classifications for fraudulent and legitimate labels, SOM methods can also have more than one cluster representing each class.

Employing SOM as a technique for detecting fraudulent credit card transactions was first proposed by Zaslavsky and Strizhak (Zaslavsky & Strizhak, 2006). The authors apply the SOM to create a model of the typical behaviour of cardholders, analyzing deviations to identify suspicious transactions. In the literature, the initial step is to create a cardholder profile, which is a model representing generalized patterns of transactions executed historically by the cardholder. This model is trained by the SOM based on the set of transactions and can identify typical transactions of a cardholder (Zaslavsky & Strizhak, 2006). The performance of the SOM model was examined on several cardholder records with various characteristics of behaviours to explore the constructed model's dependence on the transaction similarity degree. The authors showed that the accuracy of the method increased with the increase of the size of the data set, especially in the more imbalanced credit cards examined (Zaslavsky & Strizhak, 2006).

Furthermore, two-dimensional maps were built to illustrate the clustering of cardholder behaviour, in which the clusters in this map are defined by three types: typical ATM transactions, the typical point of sale (POS) transactions and rare or anomalous transactions. Discussion on the quantitative performance of the method provided by the authors is limited; however, the methodology in the literature is recognized to be in the early stages of research. The noted advantages of the proposed method are that it is not dependent on statistical assumptions about the data distribution and deals well with noisy data. Furthermore, the model allows for modification of the model with the influx of new transactions, requiring no a priori information other than a set of transactions performed by the cardholder (Zaslavsky & Strizhak, 2006).

Quah and Sriganesh also proposed using SOMs as a technique to help in the detection of fraud by employing them to unearth hidden patterns between various features or attributes within the data (Quah & Sriganesh, 2008). Furthermore, the authors utilize the SOM as a pre-classifier, a means of filtering out the number of transactions that need to be sent for review in a real-time detection system, reducing processing time, cost and complexity (Quah & Sriganesh, 2008). The proposed SOM model outputs clusters deciphered from input vectors that sketch out each cardholder's profile, deriving spending and behavioural patterns. Transactions in a dataset are compared against the profile of the cardholder and clustered based on their Euclidean distance from that cluster. Upon the grouping of samples into appropriate clusters, a parameter controlling a radius around the centroid is designated to set and control the threshold for labelling transactions as within that cluster or not (Quah & Sriganesh, 2008). For a particular cardholder, a dense may indicate that many of his transactions are of one particular type, and a sparse cluster could mean that very few transactions of another particular type (Quah & Sriganesh, 2008). As such, transactions in the dense clusters, within a determined threshold value, are filtered out, which can aid a system to more easily identify transactions requiring further review (Quah & Sriganesh, 2008).

The model was implemented using a dataset from a bank in Singapore. However, according to the authors, the studies conducted are at an early stage of research, just like the SOM proposed by Zaslavsky and Strizhak in (Zaslavsky & Strizhak, 2006). Only the clustering capabilities of the method were demonstrated in the author's work, using visual representations of the clusters formed as well as by inspecting the similarity of features in the same cluster. It was discussed that it is possible to further extend the proposed research by implementing a feedforward NN, using the output of the proposed SOM model as the training input (Quah & Sriganesh, 2008). We note that there is also a lack of quantifiable results or metrics in the literature regarding the model's filtering accuracy. Regardless, the authors mention that when implemented into and trialled in a bank's architecture, the algorithm executes very quickly, requiring less than a minute to execute (Quah & Sriganesh, 2008). Limitations of the proposed model, as identified by the authors, is its sensitivity to the choice of hyperparameters, such as the number of neurons, the similarity measure used, as well as the choice of cluster centers (Quah & Sriganesh, 2008).

Extending upon the work of previous literature, Olszewski implemented a SOM model into an actual fraud detection system on the basis of a threshold-type binary classification algorithm that was proposed (Olszewski, 2014). Initially, user accounts are represented by data matrices assembling a sequence of records. The user account's features are then visualized with the SOM, and the centroid of the visualized features on the grid is computed. This results in a single 2-dimensional point that can visualize the entire account. The classification method utilized in the paper is presented graphically in Fig. 6 which displays the individual accounts on a SOM grid based on (Olszewski, 2014).

The classification threshold is set according to the value of dissimilarity between the centroid of the entire SOM grid (as in Fig. 6) and the SOM neuron corresponding to the maximal value in the U-matrix corresponding to that SOM (Olszewski, 2014). The U-matrix is a graphical representation of a SOM, where each entry corresponds to a neuron in

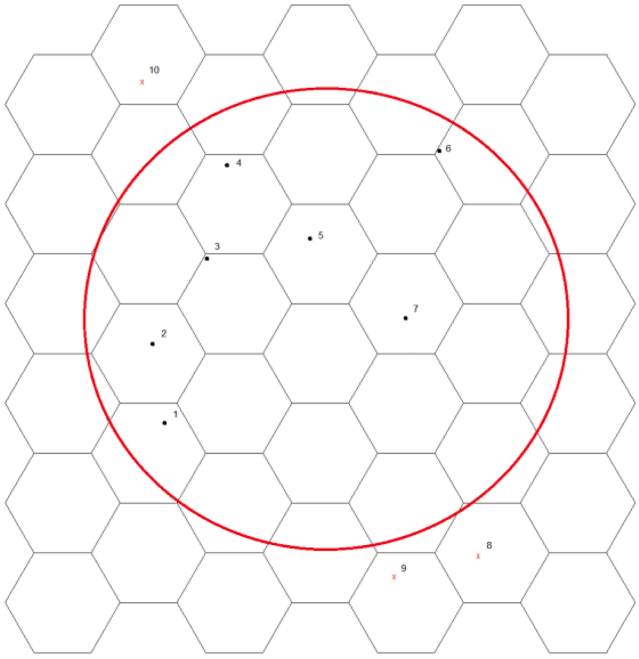

Fig. 6. Graphical illustration of the threshold-type binary classification method for a SOM.

the SOM grid. The value of each entry is the average dissimilarity between the neuron and the neighbouring neurons (Olszewski, 2014). A sequence of high values in a U-matrix, consequently, represent borderlines that separate the clusters of data on the SOM grid (Olszewski, 2014).

Olszewski evaluated the proposed model on a credit card fraud dataset consisting of 10,000 accounts of selected credit card holders from Poland (Olszewski, 2014). Only three features were used: the amount of money spent in a transaction, the location of the transaction, and the time of the transaction. Among all of the accounts, only 100 were fraudulent, representing 1% of the total dataset. The model's detection performance was compared against a standard SOM-based clustering model, a Gaussian mixture (GM) based model, and a growing hierarchical SOM (GHSOM), which was proposed by Huang *et al.* in (Huang, Tsaih, & Yu, 2014). The GHSOM approach was developed to discover topological patterns of fraudulent financial reporting, where a classification rule was also presented by the authors. The difference in the GHSOM is that the focus is mainly on creating data clusters rather san utilizing SOM visualization (Olszewski, 2014; Huang et al., 2014).

Results from the experimentation showed that the proposed SOM-based model with the detection threshold performed the best of all the models, achieving perfect classification with perfect scores of 100% across all measures of precision, recall, accuracy and F-score (Olszewski, 2014). The closest results are achieved by the SOM clustering-based method, with an F-score of 86.12%, recall of 90%, accuracy of 85.5% and precision of 82.57%. The GHSOM-based method and the GMM-based method had F-scores of 80% and 69.77%, respectively. As such, the authors concluded that based on the experimental results achieved, the proposed detection threshold proposed with the SOM model demonstrates clear superiority over the three other models. The model has a perfect fraud detection rate, with zero false alarms. The authors suggest exploring other detection methods to observe the flexibility of the model and adapt it to other applications (Olszewski, 2014). We highlight the significance of these results, as, beyond our knowledge, a perfect detection rate in a credit card fraud setting has not been observed in any other literature we have surveyed.

iii. Autoencoders

An autoencoder (AE) is a type of unsupervised deep learning network symmetric in structure with fewer nodes in the middle layers. It has a section that encodes inputs into a lower-dimensional representation and another section that decodes or reconstructs that input again (Boukerche et al., 2020). The goal of training an AE is to learn a reduced encoding of data efficiently and then reconstruct it. As illustrated in Fig. 7, the input layer passes the input data to the hidden layer, where the lower-dimensional encoding is learned. Then, the encoding is passed from the hidden layer to the output layer, where it is decoded and reconstructed as much as possible. The number of hidden layers in an AE is arbitrary, with the condition that for each part of the network, e.g. the encoder, each subsequent hidden layer must have fewer neurons than the previous layer. This architecture imposes a bottleneck in the network, restricting the amount of information that can traverse through and in turn forcing a compressed knowledge of the original input (Kucharski, Kłeczek, Jaworek-Korjakowska, Dyduch, & Gorgon, 2020)

The premise of detecting anomalies with AEs is that anomalies are more challenging to reconstruct by a trained AE than normal instances. As such, a reconstruction error can be determined for each sample of a data set using an AE, which can subsequently be used as an anomaly score (Boukerche et al., 2020; Chalapathy & Chawla, 2019). Since anomalies are difficult to reconstruct; they will have larger reconstruction errors that can be used to identify and detect them.

There have been several types of regularized AEs introduced in recent literature capable of learning richer and more expressive representations of input features. An example of such is sparse autoencoders (SAE), which encourage sparsity in a layer's neurons during training by keeping the top K-most active units (Makhzani & Frey, 2014). Denoising autoencoders (DAE) are trained to reconstruct a "repaired" input from a corrupted version of it, which is done by first corrupting the initial input with a stochastic mapping (Vincent, Larochelle, Lajoie, Bengio, & Manzagol, 2010). Contractive autoencoders (CAE) go even further by learning feature representations robust to small variations of instances around their neighbours by adding the Frobenius norm of the Jacobian matrix of the encoder's activations as a penalty term (Rifai, Vincent, Muller, Glorot, & Bengio, 2011). Variational autoencoders (VAE) prevent overfitting by introducing regularization into the representation space by encoding data samples using a prior distribution over the latent space (Pang et al., 2020; Doersch, 2016). Also, VAEs make use of a reconstruction probability rather than a reconstruction error, which is

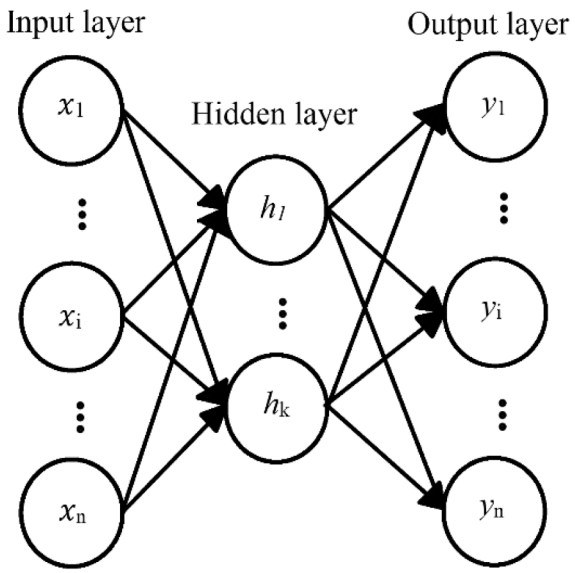

Fig. 7. Schematic of an autoencoder network's basic, symmetric architecture.

the probability that a data point can be generated from the latent variable drawn from the approximate posterior distribution (An & Cho, 2015). We refer the reader to Table 11 for a summary on autoencoder-based fraud detection approaches.

Kazemi and Zarrabi proposed using AEs in a system to detect and prevent fraudulent credit card transactions before they are processed (Kazemi & Zarrabi, 2017). The authors suggest training an AE to extract the most relevant features and then add a softmax layer to classify the network's output. The effect of the layer size is investigated in the literature by comparing the results of two AEs with different configurations. The first AE has 20 neurons in the input layer, then 15, 10 and 5 neurons in the next hidden layers of the first symmetrical half of the network. The second AE has 20 neurons in the input layer and 30, 50, and 100 neurons in the subsequent layers. The two AE models were compared with a SOM on a German credit card data set with 1,000 samples. The literature, however, makes no mention of the distance measure used for the reconstruction error, and the literature's qualitative results are limited but showed that the AE with the more extensive architecture had the best predictive performance with an accuracy of 84.1%. Nevertheless, the SOM outperformed the AE with smaller hidden layers by 0.8%, with an accuracy of 82.4%. (Kazemi & Zarrabi, 2017).

Table 11
Summary of published literature on autoencoder-based fraud detection.

Year	Reference	Type of fraud	Method	Comments
2017	Kazemi and Zarrabi (Kazemi & Zarrabi, 2017)	Credit card	AE	AE accuracy of 81.6% was outperformed by SOM accuracy of 82.4%
2018	Pumsirirat and Yan (Pumsirirat & Yan, 2018)	Credit card	AE	AE was superior to RBM, but performed poorly with small dataset size
2018	Sweers *et al.* (Sweers, Heskes, & Krijthe, 2018)	Credit card	VAE	AE with deeper architecture performed the best in terms of recall of 93.8% compared to VAE, both models had identical precision scores.
2018	Renström and Holmsten (Renström & Holmsten, 2018)	Credit card	Stacked AEs	Stacked AE and VAE models outperformed single AE model with a recall of 99% but had slightly lower precisions.
2019	Jiang *et al.* (Jiang, Zhang, & Zou, 2019)	Credit card	DAE-MLP[a]	DAE used to remove noise from input and use output to train MLP classifier. Outperformed MLP classifier trained on raw input.
2020	Misra *et al.* (Misra et al., 2020)	Credit card	AE-MLP[a]	AE using only the encoder for feature extraction, with the output used to train a classifier. AE-MLP classifier outperformed AE in (Pumsirirat & Yan, 2018)
2020	Tingfei *et al.* (Tingfei et al., 2020)	Credit card	VAE-MLP[b]	VAE to oversample minority class outperformed SMOTE and GAN oversampling when training MLP classifier.
2016	Paula *et al.* (Paula, Ladeira, Carvalho, & Marzagão, 2016)	Money laundering	AE	AE was able to detect fraudulent cases previously identified by domain experts

[a] The denoted methods are AE-based feature extraction or data preprocessing techniques implemented in conjunction with a classifier.

[b] The denoted methods are AE-based oversampling techniques augmenting generated data to a classifier's training set.

Pumsirirat and Yan conducted a more comprehensive study proposing AEs to detect fraud in the same data set as the previously mentioned study and two others (Pumsirirat & Yan, 2018). An Australian data set was also used, with 690 samples, and the familiar European credit card data set with 284,807 samples. Unlike the previous literature, the data sets in this study were preprocessed and transformed using PCA. The AE was designed with 21 input neurons and 16, 8 and 4 neurons in each layer of the network's encoder part with hyperbolic tangent activation functions, and the mean squared error (MSE) was used as a measure of reconstruction error (Pumsirirat & Yan, 2018). The restricted Boltzmann machine (RBM) is a shallow network that can learn the probability distributions of its inputs, which was also implemented for comparison. Results showed that both methods showed poor performance on the German data set, with the AE having an AUC of 43.76%, slightly lower than the RBM with 45.62%. The performance did not improve much with the Australian data set; however, the AE performed better here, with an AUC of 54.83% compared to 52.38% for the RBM. A significant jump in the AUC of both models was demonstrated on the European data set, which is the largest of the three data sets. The AE proved to be superior at 96.03%, compared to 95.05%. Thus, for large data sets, the authors conclude that both the AE and RBM model demonstrated the ability to excel in the task of detecting fraudulent credit card transactions in large data sets (Pumsirirat & Yan, 2018).

Sweers *et al.* expanded on previous literature by proposing a VAE model to detect fraudulent transactions on the European credit card data set and compared their performance with regular AEs (Sweers, Heskes, & Krijthe, 2018). Similarly, the data set was transformed using PCA to reduce the dimensionality and veil private information, citing confidentiality concerns. The authors implemented four AE and four VAE architectures of varying structural complexity. The first two models of the AE and the VAE have one hidden layer containing and either 2 or 10 neurons in that layer. The second two models of each type of AE have several hidden layers 'stacked' with varying numbers of neurons in each layer and are thus referred to as the stacked AE and the stacked VAE models. After encoding the input with the probabilistic VAE encoder, which results in a posterior distribution, the reconstruction probability is calculated. Samples are then drawn from a normal distribution, which is reparametrized with the posterior distribution's mean and standard deviation. The samples are then decoded by the probabilistic VAE, resulting in a prior distribution whose mean and standard deviation was then used to calculate the reconstruction probability (Sweers, Heskes, & Krijthe, 2018). It was shown that the stacked AE model with larger hidden layers had the best performance in terms of a recall of 93.8%, compared to 91.3% by the stacked VAE (Sweers, Heskes, & Krijthe, 2018). The precision of both models was identical at 0.009, which seems alarmingly low and suggests that the model produces many false positives and is not addressed or discussed by the authors. However, the stacked VAE was shown to be superior to both the simple AE and simple VAE architectures. It was stated in the literature that both models exhibited relatively similar high performance in detecting fraudulent transactions in credit card data; however, we comment that more in-depth discussion on the precision of the models is still necessary until such conclusions can be made.

More recent efforts by Misra *et al.* (Misra, Thakur, Ghosh, & Saha, 2020) implemented the method proposed by Pumsirirat and Yan (Pumsirirat & Yan, 2018) using the same methodology, but with a slight variation to the approach. The authors first train the AE with the feature set transformed with PCA and then only use the encoding part of the network to extract representative features of the data (Misra et al., 2020). Next, a classifier is trained on the encoded features in a supervised manner with the labels of the transactions. Upon testing, the test set is encoded using the AE and then passed to the classifier for classification. The authors experimented with three different models, such as MLPs, *k*NN and LR, and found the MLP achieved the best performance. That model was subsequently compared against the model in (Pumsirirat & Yan, 2018) and demonstrated to be more effective overall in

detecting credit card fraud with an F-score of 82.65% and an accuracy of 99.94%, compared to 8.9% and 97.05%, respectively, by the inferior method. The authors conclude that the proposed method maintains a degree of balance between precision and recall and propose tuning the technique to handle stream data in the future by training on batches of transactions (Misra et al., 2020). The authors note, however, the vital but challenging task of periodically retraining the model to keep up with evolving patterns of fraud.

Renström and Holmsten proposed a novel architecture of stacked AEs to further develop on previous works (Renström & Holmsten, 2018). Three AEs with one hidden layer were trained separately in a series configuration, with the output and the classification error of the prior AE acting as the input of the next AE. The proposed model used the MSE measure in the reconstruction error and was compared with a single AE and a VAE on the same data set as the previous study, employing the same methodology of feature transformation using PCA (Renström & Holmsten, 2018). It was shown that the proposed stacked AE method yielded an improved recall of 99% in comparison to the single AE at 94%. This improved recall value indicates that only 1% of fraudulent transactions were undetected, with the VAE showing similar results to the stacked AE. The single AE had a higher precision of 83% than the proposed stacked AE with 78% (Renström & Holmsten, 2018). We note that the results and discussion provided in the author's work are much more detailed and in-depth than that produced by Sweers *et al.* (Sweers, Heskes, & Krijthe, 2018), mainly by providing a clear picture of the precision values achieved by the models.

Tingfei *et al.* also proposed VAEs for the task of credit card fraud detection; however, the authors suggest its use to address the imbalanced data problem as an oversampling method rather than to classify instances of data (Tingfei, Guangquan, & Kuihua, 2020). The VAE is used to generate and inject data resembling the minority class into the training set. Then, an MLP with one hidden layer and ReLU activation functions in the neurons is trained using the data set in a supervised manner. Two other methods for oversampling the minority class are also implemented for comparison using the same procedure as the proposed method. The first is the synthetic minority over-sampling technique (SMOTE), proposed to address the problem of imbalanced data sets for classification tasks by Chawla *et al.* (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). Rather than oversampling by replacement, SMOTE creates synthetic examples along line segments joining any or all the k minority class nearest neighbours (Chawla et al., 2002). The other method involves employing GANs to oversample the minority class through an adversarial involving two competing networks, which will be discussed further in later sections. The authors also use the European credit card data set to experiment with the models, with all features transformed using PCA except for the purchase amount and time, which were instead standardized and normalized to new eigenvalues, which were then shown to be related to the eigenvalues of the transformed features (Tingfei et al., 2020).

The number of injected minority samples was varied to investigate the effects on performance. When the number of injected cases into the model reached 175, the performance of the model was significantly improved in all five measures, and the best results were achieved by all three models (Tingfei et al., 2020). The precision, F-score, specificity, and accuracy of the proposed VAE method demonstrated to be the best of the other techniques. The accuracy of the VAE was 93%, which is 2% higher than the optimal values of SMOTE and GANs at 91% (Tingfei et al., 2020). The F-score, which is considered a better measure of a classifier's performance, was 88% for the proposed VAE method, compared to 86.15% and 86.55% for the GAN and SMOTE techniques, respectively. Furthermore, the authors found through investigation that as the number of injected samples into the training set reached 100 times the number of actual minority cases, all three models' performances suffered significantly, with the VAE being affected the least (Tingfei et al., 2020). It is hypothesized in the literature that the reason behind this drop in performance could be attributed to the difference in diversity of the minority cases. Finally, it was also shown that although the proposed model had the best recall, it was barely affected and had the smallest differences in value as the ratio of minority classes generated was increased (Tingfei et al., 2020).

DAEs are another type of autoencoder network that were proposed by Jiang *et al.* to remove the noise from corrupt input data to reconstruct the undisturbed input as best as possible and use the output to improve the performance of an MLP classifier (Jiang, Zhang, & Zou, 2019). The European data set was used with PCA transformed features, which is initially oversampled using the SMOTE technique to address the class imbalance. Then, the data is intentionally corrupted by adding Gaussian noise to the data within the. This is then used to train the DAE to learn to undo this corruption rather than merely reconstructing the input by minimizing the MSE between the DAE's output and the original uncorrupted data (Jiang, Zhang, & Zou, 2019). The reconstructed samples are then used to train a supervised MLP classifier with cross-entropy loss function and softmax output. Experimental results indicated that the proposed method achieved superior performance achieving a recall of 84% and an accuracy of 97.93% in comparison to an MLP classifier without oversampling or denoising, which only had a recall of approximately 20% for the same accuracy value (Jiang, Zhang, & Zou, 2019). The results in this literature demonstrate the highest accuracy out of the reviewed literature; however, further analysis and exploration of the results produced by the methodology is recommended to confirm the findings.

Paula *et al.* proposed using an AE trained on databases of foreign trade in Brazil with the objective of identifying organizations or companies involved in the export business and showing signs of divergence from regular patterns of behaviour (Paula, Ladeira, Carvalho, & Marzagão, 2016). The original dataset used consisted of 80 features, and by using gradient boosted machines were filtered down to 18 features that are able to explain 80% of the variability of exported volumes by companies. The AE network was designed with 18 input and output neurons and three hidden layers with a 6–3-6 neuron structure as the bottleneck. By simple observation, the network was chosen to be trained with 50 epochs to avoid over or underfitting and had ReLU activation functions in the neurons (Paula, Ladeira, Carvalho, & Marzagão, 2016). The model was evaluated with an MSE, measuring the difference between the estimator and what was estimated. The higher the MSE, the more likely a sample was anomalous in relation to the patterns found in the data by the proposed model. The MSE values were organized in ascending order a plotted on a graph for visual inspection, which indicated an evident change in behaviour in the last 20 records. The authors mention that these samples were forwarded to third party experts in export fraud. The system was considered efficient by the experts, who indicated that it identified fraud cases that were already known by experts. Therefore, with further research, we believe that the application of these newer deep learning architectures to anti-money laundering systems shows excellent opportunities for future efforts.

iv. Other Unsupervised Methods

An unsupervised spectral ranking method for anomalies (SRA) was proposed by Nian *et al.* and applied to detect fraud in automobile insurance claims (Nian, Zhang, Tayal, Coleman, & Li, 2016). In their work, the authors extend spectral clustering algorithms, demonstrating that spectral optimization based on a defined Laplacian matrix can be viewed as a relaxation of unsupervised SVM. Using the magnitudes of the components of eigenvectors from the Laplacian matrix, it is then possible to approximate the degree of support for the optimal binary class separation function. From this, an anomaly score can be yielded instead of cluster groupings (Nian et al., 2016). The technique models data as an undirected graph using vertices to represent data instances and an adjacency matrix, where the elements specify the similarity between the different vertices (Nian et al., 2016).

The public automobile insurance dataset was used, as seen in other

literature, which consisted of 15,420 samples with a 6% fraud rate, 25 categorical and six numerical features. The labels were discarded and used only for evaluation of the model during testing. The model was compared against an OCSVM and LOF designed for the same detection task. The only quantitative performance measure for evaluation used was the AUC, and the discussion provided by the authors mainly covered the effect of varying the type of kernel used. Regardless, the proposed SRA demonstrated the highest AUC of 74% out of the three models, compared to 59% by the OCSVM and 69% by LOF (Nian et al., 2016). The results also showed that the model's performance was sensitive to the hyperparameters used and thus required careful tuning. Furthermore, visual representations of the clusters formed by the principal eigenvectors of the model were inspected to observe the complexities of the separating structures constructed. We note that a more thorough discussion of the various other relevant performance measures is needed; however, the proposed SRA demonstrated to be applicable to detect fraudulent automobile insurance claims.

The ARIMA model is a technique used for time series that is a generalization of the autoregressive moving average model (ARMA). The ARMA model is comprised of the autoregressive (AR) and the moving average (MA) models. The AR model assumes that there is a dependent linear relationship between an observation and a specific number of lagged or previous observations, and an error term. In contrast, the MA model makes use of the dependency between observations and the residual errors that are a result of applying the model to lagging observations (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020).

In time series analysis, the central assumption is that the series is stationary, meaning it has zero mean, and the variance is constant over time. However, in most practical situations, this is generally not the case (Adhikari & Agrawal, 2013). The solution to this is the ARIMA model, a prevalent and simple model, which facilitates the differencing of data points in time series so that they are made stationary (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020; Adhikari & Agrawal, 2013). In the detection of fraudulent activity, time series are a useful tool when dealing with aggregated features produced by aggregation, which is a method of deriving new features from the data that might be more useful to the model (Pozzolo, Caelen, Le Borgne, Waterschoot, & Bontempi, 2014).

The application of ARIMA models for detecting fraudulent credit card transactions has only been recently implemented by Moschini *et al.* in (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020). In their work, the authors propose the model to address the unbalanced nature of data sets available, as well as the lack of adaptability by many models to consider changing spending behaviours and patterns over time. The model is unsupervised, and the only source of information to the model is from the spending behaviour of a cardholder. The model is initially calibrated on the daily number of legitimate transactions to learn the spending behaviours of customers. Then, rolling windows of time are used to predict fraudulent transactions from the test set. Fraudulent transactions were flagged based on their calculated standard score (also known as a Z-score) being more significant than a threshold (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020).

Using a dataset provided by a fraud-prevention company known as NetGuardians SA containing information about credit card transactions of 24 cardholders from the period of June 2017 to February 2019, the ARIMA model was implemented and compared against other models such as box plots, LOF, IF and *k*-means clustering. The authors comment on the highly imbalanced nature of the data set, with the proportion of fraud consisting of only 0.76% of all transactions (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020). Experimental results show that the ARIMA model displayed the highest precision and F-score of 50% and 55.56%, respectively. The *k*-means algorithm, however, was the best performer in terms of recall. Overall, the worst model was shown to be LOF, with a precision of 8.4% and an F-score of 14.04%; however, the authors note that this model is designed to be useful with multidimensional data sets, unlike in this methodology (Breunig, Kriegel, Ng, & Sander, 2000). Box plots demonstrated the best performance overall with an F-score of 72.22%, although the authors note that the advantage of the ARIMA model is that it is based on the concept of modelling customer behaviour.

Overall, ARIMA models for credit card fraud detection have been shown by Moschini *et al.* to perform better when there is a significant amount of fraudulent activity within the same day (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020). They also reduce the number of false-positive alerts in comparison to other benchmark models and take into account the dynamic spending behaviour of customers by utilizing rolling windows. The authors' major identified problem is that ARIMA models operate under the assumption that the observations in a data set are equally spaced in time, which does not hold in this study as transactions were unequally spaced. It is suggested in the literature that future research efforts will be directed towards using more advanced approaches such as the continuous-time autoregressive moving average (CARMA) process to address this issue (Moschini, Houssou, Bovay, & Robert-Nicoud, 2020).

5.3. Semi-Supervised Methods

i. Hidden Markov Models (HMM)

HMMs are stochastic processes that are used to model much more complicated processes than a traditional Markov model. They are comprised of a finite set of states that are governed by a set of transition probabilities. Each state also has an associated probability distribution responsible for generating an outcome or observation. The outcome is known in an HMM, but it is the state that is unknown or hidden (Rabiner, 1989).

An HMM is characterized by several things, the first of which is the number of states in the model. For each state, there are also several distinct outcomes or observations that correspond to the modelled system's physical output. A state probability matrix or state transition matrix gives the probabilities of transitioning from one state to another in a single step, which can be represented graphically by a state transition diagram, as shown in Fig. 8. The hidden states are represented by x_i, each mapping to an observable outcome y_i with probability b_{ij}. The state transition probabilities a_{ij} represent the probabilities of moving from one hidden state to another (Toledo & Katz, 2009). Rabiner in (Rabiner, 1989) provides a comprehensive tutorial on HMMs.

A credit card fraud detection system was proposed by Srivastava *et al.*

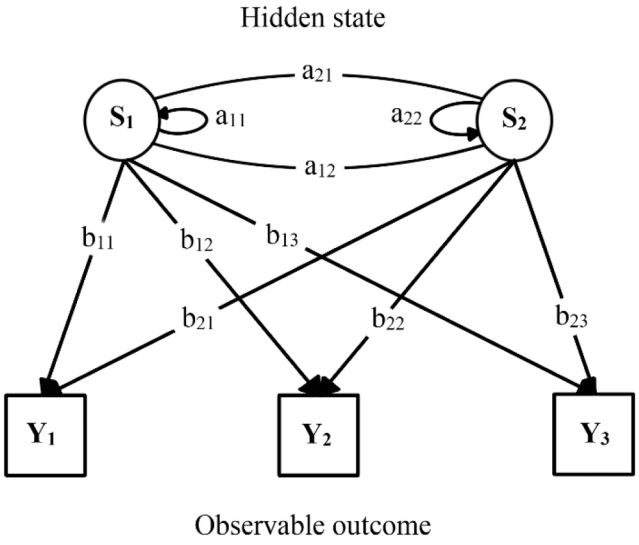

Fig. 8. An example of a hidden Markov model state transition diagram.

to model the sequence of operations in credit card transactions and detect fraudulent activity using multinomial HMMs (Srivastava, Kundu, Sural, & Majumdar, 2008). The normal transactions of an individual are used to train an HMM for each cardholder. Incoming transactions for individuals are compared with their respective models, and any transaction not accepted by the HMM with sufficiently high probability is subsequently considered fraudulent. The authors begin by quantizing the purchase values into price ranges, configurable based on cardholder spending habits (Srivastava et al., 2008). The *k*-means clustering algorithm is used in the literature to determine the low, medium, and high spending ranges, with the number of clusters fixed *a priori* to three. Then, a state transition matrix is constructed from the types of purchases, with the reason being that cardholders typically make purchases depending on their need to procure different types of items over a period of time. This results in the generation of sequences of transaction amounts, forming part of the hidden state of the HMM.

A simulated data set was generated and used to train and evaluate the model, with the authors noting the difficulties of obtaining data from institutions that are hesitant to share (Srivastava et al., 2008). Sequences of transactions of different lengths were generated and used to train HMMs with varying numbers of states. It was shown that as the length of the sequence and the number of states were increased, the classifying performance of the model also increased. However, this comes along with the cost of higher computational complexity. Similarly, the model demonstrated that as the sequence length increased, so did the false positives rate. Based on various trials, the authors settle on choosing a sequence length of 100 and 5 states. Experiments were carried out by varying the number of fraudulent transactions intermixed with a sequence of genuine transactions. Results showed that the proposed method outperformed methods proposed in previous literature. The HMM achieved an accuracy of close to 80% on average over a wide variation of the input data and was demonstrated to be scalable for handling large volumes of transactions (Srivastava et al., 2008). The importance of having accurate spending profiles is highlighted in the studies, with the model showing significant degradation in performance with low quality or a lack of information.

Bhusari and Patil also verified the performance of HMM for detecting credit card fraud by recreating the model proposed by Srivastava *et al.* and achieving similar results on a simulated data set (Bhusari & Patil, 2016). Studies by Dhok employed a similar approach and methodology to apply the HMM model to a real-world credit card transaction data set (Dhok, 2012). The results are consistent with the previous literature results, demonstrating the feasibility and performance of the HMM for this application. Robinson and Aria extended previous authors' work with success to a prepaid credit card processing and detection system to detect fraud in real-time for merchants (Robinson & Aria, 2018).

A supervised approach of using HMM models for credit card fraud detection was proposed by Lucas *et al.* by creating sequential features to describe temporal dependencies between cardholder transactions (Lucas et al., 2020). The authors suggest using an RF classifier to detect fraudulent transactions based on the sequential features generated by their proposed HMM technique. The sequence of credit card transactions is modelled from three perspectives in the literature's framework.

The first involves comparing the likelihood of transaction sequences with legitimate historical transaction sequences and sequences with at least one fraudulent transaction. Both comparisons are necessary because it is not sufficient for fraudulent behaviour to be far from legitimate, but it is also expected to be relatively close to risky behaviour (Lucas & Jurgovsky, 2020). Secondly, the features are created to describe the transaction sequences of cardholders and merchant terminals, which have been shown to have a positive effect on the detection efficiency (Lucas et al., 2020). Finally, the authors considered the time elapsed between two transactions and the transaction amount as the primary signal for constructing the features. Combining these perspectives results in eight sets of sequences produced from the data's training set for which there is an HMM trained on each sequence (Lucas et al., 2020). A likelihood value is associated with a transaction by each HMM based on the previous sequences of transactions. These values are used as additional features for the RF classifier.

In the literature, the authors' proposed automated feature engineering methods compared with using an RF classifier without the HMM-based features with a data set from an industry partner (Lucas et al., 2020). Experiments showed that the AUC of the precision-recall curve in the novel method increased by 18.1% for transactions made face-to-face and 9.3% for e-commerce transactions. It was also demonstrated that the method's feature engineering strategy could be relevant for various other classifiers due to its robustness to the choice of hyperparameters for feature construction (Lucas et al., 2020). Future works suggested by the authors involve combining the predictions of an LSTM network with the HMM feature-based RF classifier's predictions to extend upon the work of Jurgovsky *et al.* (Jurgovsky et al., 2018).

ii. Generative Adversarial Networks (GAN)

GANs are a type of deep learning framework proposed by Goodfellow *et al.* in 2014 that consists of two networks in competition with each other (Goodfellow et al., 2014). The first is a generative model G to capture the distribution of the training data. Its adversary is a discriminative model D that determines the probability of a sample coming from the training data instead of G. The objective in training G is to maximize the chance of inducing D, which is simply a classifier, to make mistakes distinguishing the data (Goodfellow et al., 2014). The two models in a GAN are deep learning architectures that learn a representation of the original input. Increasing the number of layers or the size of layers in the network can help it learn deeper and more abstract representations (Bengio, Courville, & Vincent, 2013).

As illustrated in Fig. 9, the generative model's input is random noise z, which it then transforms with a function and then produces examples of the real data. The discriminator then learns to better distinguish between the real and generated examples by minimizing its prediction errors, and the generator tries maximizing the error, resulting in a competition formalized as a minimax game in (6):

$$\min_{\theta_G}\max_{\theta_D}(\mathrm{E}_{x\ p_d}[\log D(x)] + \mathrm{E}_{z\ p_z}[\log(1 - D(G(z)))]) \tag{7}$$

where θ_G and θ_D are the parameters of the generator and discriminator networks, respectively, p_d is the data distribution and p_z is the prior distribution of the generative network (Goodfellow et al., 2014). Although GANs are unsupervised learning algorithms, they use a supervised loss as part of the training. In most financial fraud applications

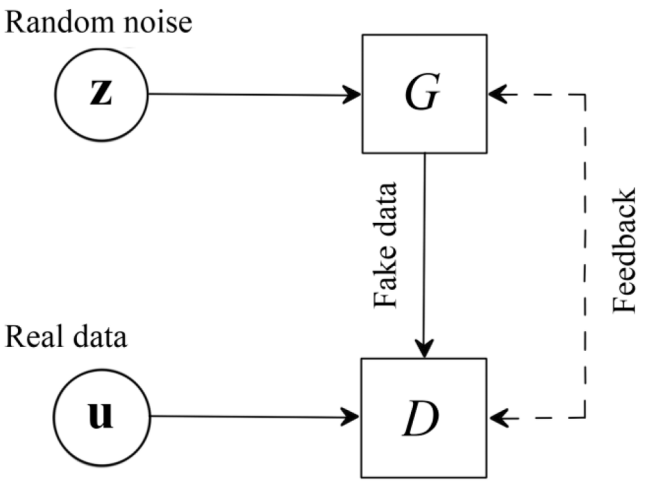

Fig. 9. Schematic of a GAN's generator G, accepting random noise z as input and outputting generated examples to the discriminator D. The discriminator distinguishes the generated examples by G from the real data u.

to date, GANs have been used in a semi-supervised fashion as a method of oversampling for data augmentation, which is the reasoning behind their classification under semi-supervised in this paper.

Mirza and Osindero proposed extending GANs to a conditional model, known as conditional GANs (CGAN), by conditioning the generator and discriminator on some extra information (Mirza & Osindero, 2014). This extra information could be of any kind,

such as class labels or data from other modalities, which is fed to both networks as an additional input layer. An outline of the surveyed works involving GANs for fraud detection can be seen in Table 12.

Table 12
Summary of Published Literature on GAN-Based Fraud Detection.

Year	Reference	Type of fraud	Method	Comments
2018	Chen *et al.* (Chen, Shen, & Ali, 2018)	Credit card	SAE-GAN	GAN trained on SAE-learned features from majority class has improved F-score and precision, but with a decrease in recall. SAE-GAN outperforms OCSVM and OCGP in terms of F-score.
2019	Tanaka and Aranha (Tanaka & Aranha, 2019)	Credit card	GAN-DT[a]	DT trained with minority class GAN-based oversampling had slightly higher precision, but lower recall than when using SMOTE or ADASYN.
2019	Fiore *et al.* (Fiore et al., 2019)	Credit card	GAN-MLP[a]	MLP trained with GAN-based oversampling had improved recall, and proposed model also outperformed SMOTE in terms of recall, but had slightly lower specificity.
2019	Ba (Ba, 2019)	Credit card	WCGAN-LR[a]	LR with WCGAN-based oversampling had most balanced performance with higher F-score and AUC than GAN, CGAN, SMOTE and ADAYSN. However, WCGAN's recall of 64.2% was significantly inferior to ADAYSN's at 90%.
2019	Zheng *et al.* (Zheng et al., 2019)	Credit card	AE-OCAN	Complementary GAN generator trained on AE-learned representations of genuine transactions; discriminator is proposed as OCAN. Proposed model performs better in F-score, precision and accuracy than OCSVM but outperformed in recall.
2020	Charitou *et al.* (Charitou et al., 2020)	Money laundering	SAE-GAN	SAE features extracted from entire train set, then used to train generator of GAN to produce complementary samples. Samples generated are augmented into training set, and discriminator is trained to classify samples. Proposed model outperformed LR, SVM, MLP and RF with either ADASYN or SMOTE in terms of F-score, accuracy and precision. RF with ADASYN, however, outperformed in terms of recall.

[a] The denoted methods are GAN-based oversampling techniques augmenting generated data to a classifier's training set.

Up until recent years, GANs for anomaly detection have been mainly applied to problems in which the nature of the input data is in the form of an image, as surveyed by Di Mattia *et al.* (Di Mattia, Galeone, De Simoni, & Ghelfi, 2019). Among the first papers applying them to detect fraudulent credit card transactions was published by Chen *et al.,* who propose using an SAE to obtain representations of normal transactions to train a GAN (Chen, Shen, & Ali, 2018). The discriminator is trained to distinguish real genuine transactions from faked genuine transactions, which are produced by the generator trained on the normal representations learned and output by the SAE. The SAE extends upon the idea of original AEs by incorporating a sparse penalty term to the reconstruction error, resulting in more robust features by adding constraints for concise expression of the input data (Zhang, Cheng, Liu, & Liu, 2018). This induced sparsity results in a greater average activation value by limiting the neurons with low, undesired activation values to zero.

The European data set is used for training and testing in the proposed method's literature (Chen, Shen, & Ali, 2018). The training set consisted of 5,000 normal transactions, and 492 transactions of each fraudulent and normal class were used for testing. The authors used a novel technique to visualize high-dimensional data known as t-distributed stochastic neighbour embedding (t-SNE), proposed by van der Maaten and Hinton in (van der Maaten & Hinton, 2008), to visualize the test data distribution. Several experiments were held to demonstrate the effectiveness of the proposed model, the first of which involved varying the size of the hidden layer of the SAE to observe the effect on the overall model's predictive performance. A positive trend was apparent between the hidden layer's size when it was varied from 30 to 60 neurons and the model's precision (Chen, Shen, & Ali, 2018). However, the recall and F-score began to decrease when the number of neurons increased past 50. The second experiment involved comparing the performance of the proposed model against state-of-the-art one-class methods such as the OCSVM proposed by Tax and Duin in (Tax & Duin, 2001), as well as one-class Gaussian processes (OCGP), which was proposed by Kremmler *et al.* (Kremmler, Rodner, Wacker, & Denzler, 2013). The proposed method had the highest F-score and precision out of the three models, 87.36% and 97.59%, respectively, with the OCGP coming in second. However, in terms of recall, the OCSVM showed the best performance with 95.11%. The proposed SAE-GAN came in second by a significant difference of 15.74% at 79.37%. Finally, the GAN's discriminator was trialled with and without the SAE features. A performance improvement was proven in the proposed technique, with an 11% increase in precision from 87.41% to 97.59% and a 4.6% rise in F-score from 83.47% to 87.36%. The increase in precision and F-score came at a slight cost of the recall, which decreased by 1.12% to 79.37% from the model without the SAE (Chen, Shen, & Ali, 2018).

The high precision of the proposed SAE-GAN model is attributed to, by the authors, the fact that since the model is trained on normal transactions to distinguish them better, it is less likely to misclassify normal transactions (Chen, Shen, & Ali, 2018). The authors justify the low recall of the proposed model due to the increase in the SAE output's dimensionality, resulting in overfitting of the data because of information redundancies (Chen, Shen, & Ali, 2018). Overall, the literature demonstrated that the proposed GAN model trained with SAE features demonstrated to be superior to a standalone GAN in detecting credit card fraud. However, the authors suggest further research to address the problem of poor stability and F-score convergence of the model to improve its performance (Chen, Shen, & Ali, 2018).

Followed shortly after were efforts by Tanaka and Aranha to develop a method of employing GANs in the task of credit card fraud detection to improve the performance of a classifier (Tanaka & Aranha, 2019). The authors use GANs to generate artificial training data for machine learning tasks, which is extremely useful for classification models based on imbalanced data sets. The ability of GANs to create artificial data is compared to techniques like adaptive synthetic (ADASYN) sampling (He, Bai, Garcia, & Li, 2008) and SMOTE (Chawla et al., 2002). A noted

benefit of artificially created data is that it is also useful when dealing with data that contains sensitive information (e.g. financial information, medical information and imaging)) (Tanaka & Aranha, 2019). In the literature, the proposed GAN's generator is trained to mimic and produce samples of the minority class when fed with a noise vector, and the discriminator must learn to distinguish between real samples and the fake samples produced by the generator. As a result, the generator will learn to output samples that are progressively more resembling the original data.

In the literature, using the European data set, the experimental procedure is outlined by the authors as follows (Tanaka & Aranha, 2019): first, the target minority class (i.e. fraudulent transactions) is separated, and several GANs with a varying number and size of hidden layers are trained with only that subset of the data. Two GANs have one hidden layer, with either 128 or 256 neurons, and two other GANs have two hidden layers with either 128 and 256, or 256 and 512 neurons in the first and second hidden layer, respectively. The activation functions in the neurons were chosen to be the leaky ReLU. The GANs were used to generate and augment new samples into the data set until balanced. A DT classifier was trained using the newly balanced data set. The proposed method was compared against other DT models using SMOTE, ADASYN and one without any oversampling. Finally, the models' performances were all compared on a balanced and imbalanced test set.

The literature showed that all the models displayed much better results on the balanced test set than the imbalanced test set, which the authors mention makes sense since the classifier has an easier time identifying minority class labels (Tanaka & Aranha, 2019). Furthermore, for all the methods with oversampling, there was an improvement of at least 30% in the recall score, compared to 56.5% by the simple DT model. SMOTE and ADASYN had the highest recall of 86.1%, and the GAN with the smaller two hidden layers came close with a recall of 82%. That said, the authors warn that more importance should be given to the results from the imbalanced test set, as it cannot be expected that test data will be balanced in actual applications (Tanaka & Aranha, 2019). With that in mind, negligibly better accuracy and precision was achieved by the GAN than SMOTE and ADASYN, but with a worse recall value. The importance of these metrics depends on the application domain, and in the case of detecting fraud, a high recall score is considered paramount, as it indicates a higher amount of detected fraud. Consequently, the authors suggest that ADASYN and SMOTE should be preferred over their proposed method (Tanaka & Aranha, 2019). Even so, the literature concludes that with further work and research, it is reasonable to expect comparable future results to the other two techniques used (Tanaka & Aranha, 2019).

Fiore *et al.* conducted even more extensive studies using GANs to oversample the minority class in imbalanced data sets and improve a classifier's performance (Fiore et al., 2019). The author's choice of the classifier was an MLP, and an emphasis was put on the design and tuning stage of the experimentation. The importance of finding optimal values for hyperparameters is highlighted in the research to be the critical aspect of maximizing the performance of the proposed method (Fiore et al., 2019). As such, using the European credit card data set, the MLP classifier was trained on a training set of two-thirds of the original data to determine the set of optimal hyperparameters that facilitate the best performance on the test set. Then, the proposed GAN is trained on the fraudulent samples only, and the original data is balanced with samples from the generator. Another MLP is trained on the data balanced and augmented with fraud samples from the GAN, using the first-trained MLP's hyperparameters.

To start the tuning process of the networks, the authors compare various configurations and structures of classifiers. Since too many layers would complicate training and cause overfitting, while too few layers might hinder the network's ability to build representations at decent levels of abstractions, as a reasonable trade-off, the authors chose to test networks with only two or three hidden layers (Fiore et al., 2019). The number of neurons in the hidden layers was determined empirically by being varied from 20 to 40, except for the generator's first layer, which seemed to require a larger number of units (around 100) (Fiore et al., 2019). Three different activation functions were explored, including the sigmoid, ReLU and hyperbolic tangent functions. The literature found the best performing classifier to be a network with two hidden layers, 30 ReLU units, 30 sigmoid units, and two softmax units in the output layer to output the final classification.

The literature showed that when the number of generated samples was augmented into the training set increased, the recall improved appreciably compared to the imbalanced training set (Fiore et al., 2019). The recall improved from 70.23% in the imbalanced dataset to a high of 73.03% recall with 3,150 generated samples. This improvement was counterbalanced by a negligible decrease in the specificity of 0.004% from 99.9998%, corresponding to the increase in false positives (Fiore et al., 2019). A comparison was also carried out with SMOTE, the significance of which is noted to be due to its similarity to the framework by the authors. The recall of the proposed GAN method is generally higher than the SMOTE, while the specificity is higher than the author's framework. However, this comparison further corroborates the superior performance of the proposed GAN with the highest F-score of 81.90% compared to SMOTE, with an F-score of 81.78% (Fiore et al., 2019).

The training of the GAN is identified as the costly portion of the computational complexity, which is circumvented by training on a small portion of the training data. The authors conclude that while the proposed method may be capable of identifying similar frauds, it can be expected to be mostly ineffective in spotting frauds that are novel, where there is no information to generalize upon (Fiore et al., 2019). In the hope of improving upon the performance of the proposed research, ensemble methods are suggested as an avenue of exploration for future work; however, the authors suggest the directions for future study are manifold.

The Wasserstein GAN (WGAN) was proposed by Arovsky *et al.,* which used the Earth Mover (EM) distance as the measure of error, and does not require a careful design of the network architecture as the typical GAN (Arjovsky, Chintala, & Bottou, 2017). Ba proposed implementing the WGAN instead of regular GANs, citing them as more stable in training and capable of producing more realistic fraudulent transactions when trained on the minority class (Ba, 2019). Similar to Tanaka and Aranha's work in (Tanaka & Aranha, 2019), the WGAN was used to oversample the minority class and balance the data set. This data set was then used to improve the performance of an LR classifier. The architecture of the proposed WGAN consisted of only one hidden layer, and the hyperparameters of the network, such as the learning rate, drop-out rate, and the number of neurons, were found using a random search algorithm. The literature also extends the WGAN to develop a conditional WGAN (WCGAN) to examine any effects or improvements in performance compared to the WGAN. The authors used the European data set in their methodology, training the model on 80% of the fraudulent samples with mini-batches of 64 samples. Adam was used as the optimization algorithm, which is a stochastic optimization method suited for problems that are large in terms of data and dimension (Kingma & Ba, 2014). The F-score of the proposed WGAN and WCGAN was compared against the typical GAN and CGAN, as well as the sampling techniques SMOTE and ADASYN. Results demonstrated that the WCGAN was the most balanced classifier with an F-score of 71%. However, the recall of the model, 64.2%, is relatively low regardless of the F-score, compared to the recall of ADASYN with 90.1%. This may make the ADASYN seem more advantageous than the WCGAN; however, it only had a precision of 1.8%, meaning it was too prone to false alarms. The AUC of the WCGAN was 94.8%, the highest of all the models. Overall, it was concluded that the proposed WCGAN produced a balance of precision and recall that resulted in the best overall performance with the highest F-score (Ba, 2019).

A one-class adversarial net (OCAN) was proposed by Zheng *et al.* to detect fraudulent credit card transactions by using a complementary GAN trained on representations of data learned by an AE (Zheng, Yuan,

Wu, Li, & Lu, 2019). In the literature, the AE is responsible for learning the representation of legitimate transactions in the hidden space from the raw features. Then, the GAN is trained with a generator that produces complementary samples in the legitimate transactions' low-density area instead of matching the representation's distribution like a regular GAN. Fig. 10 illustrates the difference in output between the generators of the regular and complementary GANs based on (Zheng et al., 2019). The discriminator is subsequently trained to distinguish between the generated complementary samples and the legitimate samples based on a probability distribution threshold. Since fraudulent behaviour is complementary to that of genuine behaviour, it is expected that the proposed method should be able to differentiate between the two classes (Zheng et al., 2019).

The performance of the method proposed by the literature was evaluated using the European data set against other one-class techniques such as one-class nearest neighbours (OCNN) and OCSVM. The effect of learning a representation of the data using an AE is also investigated by evaluating each model with only the raw features transformed using PCA. Using a training set of 700 legitimate transactions and a test set of 490 fraudulent and 490 legitimate transactions, the OCAN proposed by Zheng *et al.* displayed the best performance out of all the other models both in terms the precision, F-score and accuracy, and came second for recall behind OCSVM (Zheng et al., 2019). The F-score achieved by the proposed method was 86.56%, an increase of 2.4% from the OCAN with just raw features. However, the precision of the OCAN with just raw features was higher, at 97.55%, compared to the proposed OCAN with 90.67% precision. In terms of recall, the OCSVM was more favourable with 95.09% instead of 83.2% of the OCAN with learned representation. Also, the AUC of the corresponding ROC was calculated to be 97.5% for the proposed OCAN, a minor increase from 96.45% with just raw features. Overall, the authors conclude that the proposed method's results are auspicious, particularly for detecting credit card fraud compared to previous traditional techniques (Zheng et al., 2019).

An anti-money laundering system using GANs was proposed by Charitou *et al.,* training the model using the robust features extracted from the training set by an SAE (Charitou, Garcez, & Dragicevic, 2020). The structure of the proposed framework consists of first using an SAE with two encoding and two decoding layers. The SAE projects the input into a higher dimension and reconstructs it again from the sparse representation. This mapping seeks to increase the distance between the positive and negative class samples (Charitou et al., 2020). Next, the data representations extracted from the SAE are used as the input to the GAN, which adopts a complementary generator that tries to match the data representations and generate new complementary samples (Charitou et al., 2020). Together with the real representations of the data, the generated samples are then used to train the discriminator model, which learns to distinguish and detect fraudulent cases.

The authors partner with a company in the UK to improve their current system, using a dataset provided by the company. It consisted of 4,700 samples, 1,200 of which were flagged for potential money laundering activity, accounting for more than 25% of the whole dataset (Charitou et al., 2020). The data described anonymized gambling activity from the company's legal online gambling services, and the cases of known fraud or money laundering were labelled in the dataset. Specifics regarding preprocessing of the data or any feature selection or extraction steps were not mentioned in the research. The proposed model's performance was compared with various other models such as LR, RF and MLP using different sampling techniques like SMOTE and ADASYN. The proposed SAE-GAN model demonstrated the best overall performance with the highest F-score of 89.85% (Charitou et al., 2020). It also had the highest recorded accuracy of 94.37%. The recall of the RF model with ADASYN oversampling demonstrated the highest recall of 95.69% compared to 93.08% by the proposed model, with an F-score slightly lower at 88.62% (Charitou et al., 2020). The basic RF model showed the best precision at 87.81%, slightly better than the SAE-GAN at 86.72%.

Overall, the authors state that the model was considered a significant improvement from the previous measures in place for money laundering detection by the company. The proposed SAE-GAN's F-score yielded an increase of 3.64% from the current detection system in place and a 0.52% increase compared to the other methods evaluated (Charitou et al., 2020). The versatility of the framework is further emphasized for supervised settings. It is suggested that testing different sparse coding methods may further improve the model's results, which the authors will attempt to implement into a more extensive framework for future experimentation with a GAN designed to generate synthetic data (Charitou et al., 2020).

iii. Other Semi-supervised Methods

A novel hybrid approach using GA and unsupervised FCM clustering (GAFCM) to detect fraudulent automobile insurance claims was proposed Subudhi and Panigrahi (Subudhi & Panigrahi, 2020). The approach proposed in this study is similar to that in the studies by Behera and Panigrahi in (Behera & Panigrahi, 2015), in which the majority (genuine) samples of data were initially clustered by the FCM to identify and remove outliers, resulting in a balanced dataset when augmented with the rest of the minority samples. A supervised learning model was then trained on the balanced dataset to further classify the samples into either the genuine or fraudulent class with improved

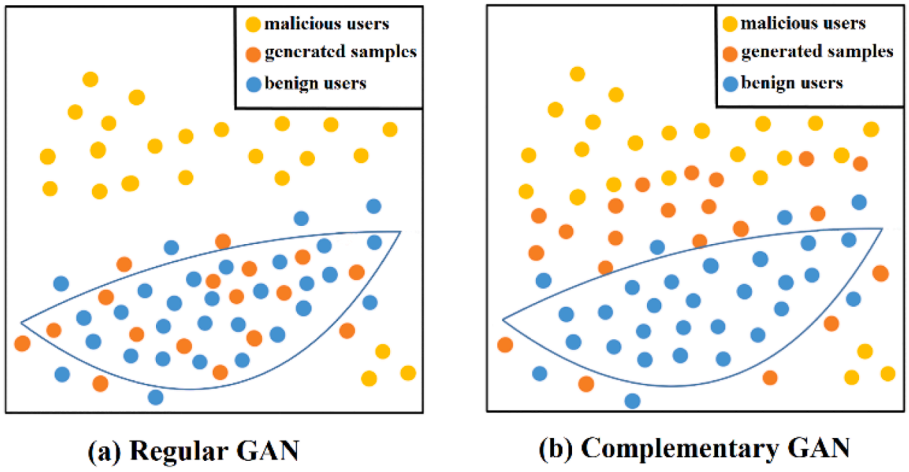

Fig. 10. Two examples of outputs produced by generators of (a) a regular GAN and (b) a complementary GAN. The dotted blue line indicates the high-density region of benign transactions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

performance. In this study, the authors extend upon the work of Behera and Panigrahi (Behera & Panigrahi, 2015) by using GA to optimize the cluster centers of the FCM algorithm during training to observe for improvements in detection performance and experimenting with automobile insurance claims instead of credit card transactions (Subudhi & Panigrahi, 2020). The classification of the overall proposed method as semi-supervised is due to the unsupervised FCM using only the majority samples, even if unlabelled, to perform the outlier detection.

It is postulated in the literature that using GA to train the FCM algorithm has the benefits of increasing its robustness, allowing it to conduct a more extensive search for the optimal cluster centers and reducing the chances of getting stuck at a local optima (Subudhi & Panigrahi, 2020). A dataset consisting of 15,420 automobile insurance claims was used, with only 923 fraudulent claims, accounting for less than 6% of the total number of claims. The authors highlight one of the significant issues faced by researchers in this field, noting the lack of availability of public datasets other than the one used in their study. A quarter of the training set was set aside for testing, and the rest for training. No discussion is provided on whether any preprocessing or feature selection measures took place. The genuine majority samples in the training set were separated, and the FCM was trained on those samples using 10-fold cross-validation. A total of 4,773 genuine instances were removed by the GAFCM from the original training set of 10,627 samples (Subudhi & Panigrahi, 2020).

The balanced training set was used to train two subsequent detection models, an FCM and GAFCM, which were then compared with two identical FCM and GAFCM models that were trained with the original imbalanced training set. An evident improvement in sensitivity, specificity and accuracy of both models was observed when using the balanced dataset. The GAFCM performed the best overall, however, with 66.67% sensitivity, 86.95% specificity and 84.34% accuracy (Subudhi & Panigrahi, 2020). These values represent increases by 5.13%, 2.16% and 1.12%, respectively, from the GAFCM model trained with an imbalanced dataset. The FCM with an imbalanced dataset had a sensitivity of 59.22%, a specificity of 84.49% and accuracy of 81.97%, which is a significant improvement from the unbalanced case but still inferior to the proposed GAFCM. (Subudhi & Panigrahi, 2020).

To further verify the improved performance of the hybrid GAFCM technique proposed over the FCM proposed by Behera and Panigrahi, the balanced datasets produced by each model were used to train a supervised learner. Several models' performances were analyzed, which included DT, SVM and MLP. Results from this experimentation further solidify the effectiveness of the proposed GAFCM technique, showing overall improvements across the board in all measures for each of the three models (Subudhi & Panigrahi, 2020). Of the three models using the GAFCM balanced dataset, the SVM displayed the highest sensitivity, specificity, and accuracy of 83.21%, 88.45% and 87.02%, respectively. These values are clear improvements from 64.34% accuracy, 72.23% specificity and 70.15% accuracy by the SVM model with FCM outlier detection (Subudhi & Panigrahi, 2020). Thus, the technique outlined by Subudhi and Panigrahi (Subudhi & Panigrahi, 2020) is considered favourable over previous methods, performance better as both a standalone classifier and a data undersampling method to address the issues associated with imbalanced datasets for supervised classifiers. We note that more extensive studies are suggested to address certain aspects not covered by the authors, such as whether feature selection or preprocessing may yield further incremental improvements.

5.4. Graph-based Methods

The popularity of graph-based anomaly detection techniques for fraud detection has been on the rise, especially when it may be beneficial to analyze the connectivity patterns in large networks. These techniques have proven to be especially useful in anti-money laundering and insurance fraud detection settings, where there are often multiple entities or organizations that can be linked to fraud. A recent and thorough review of graph-based fraud detection techniques is provided by Pourhabibi *et al.* in (Pourhabibi et al., 2020). However, it is noteworthy to highlight some of the critical papers directly relating to this survey.

Yang and Hwang, for example, proposed a process-mining framework to detect healthcare insurance fraud or abuse by utilizing the concept of clinical pathways (Yang & Hwang, 2006). The application of clinical pathways aims to have medical staff performing care services in the right order to enable best practice without rework or wasting resources. These pathways are typically driven by physician orders, and once they are created, they can be viewed as algorithms of decisions to be made and the care to be provided to a given patient (Yang & Hwang, 2006). In their work, the authors outline a framework involving the mining of frequent patterns from clinical instances to facilitate an automatic and systematic construction of systems to detect healthcare fraud. The model was evaluated with a real-life dataset, with empirical results demonstrating the model's efficiency and its capability in identifying fraud cases not previously identified by domain experts or a manually constructed detection model (Yang & Hwang, 2006).

An expert system using social network analysis to detect fraudulent automobile insurance claims was proposed by Šubelj *et al.* in (Šubelj et al., 2011). In automobile insurance fraud cases, organized collaborators often work together to perpetrate these crimes, which sometimes consist of drivers, chiropractors, repair garages, mechanics, lawyers and others (Šubelj et al., 2011). As such, the authors represent data in networks to observe the relationships between different individuals or groups. Detection of suspicious entities or collaborators is then achieved by employing a novel assessment algorithm known as the Iterative Assessment Algorithm (IAA) (Šubelj et al., 2011). The results showed that the proposed system was able to detect fraudulent cases efficiently and that the appropriate data representation was vital and is very applicable in practice as it does not require a labelled dataset (Šubelj et al., 2011). The system instead allows for the imputation of a domain expert's knowledge, which can thus be adapted to newer types of fraud as they are identified. The benefit of the system is that it does not require large amounts of data, with the only challenges being that it relies on user-defined thresholds or parameters that must be refined.

Branting *et al.* explored using graph analytics to detect fraudulent healthcare claims and activity (Branting, Reeder, Gold, & Champney, 2016). The authors apply various groups of network algorithms; one such group calculates the behavioural similarities to known fraudulent and genuine healthcare providers with respect to measurable activities such as healthcare procedures and drug prescriptions (Branting, Reeder, Gold, & Champney, 2016). Another group of algorithms estimates the propagation of risk from healthcare providers through geospatial collocation, such as shared practice locations or other addresses (Branting, Reeder, Gold, & Champney, 2016). The proposed model was evaluated empirically on various datasets, demonstrating an F-score and AUC of up to 91.9% and 96%, respectively. It was also shown that the most predictive features were based on the collocation-based risk propagation algorithms (Branting, Reeder, Gold, & Champney, 2016).

In the context of anti-money laundering, a social network analysis algorithm (SNA) was proposed by Dreżewski *et al.* (Dreżewski, Sepielak, & Filipkowski, 2015). In their paper, the authors note that offenders often create sophisticated organizational structures that can be identified. Furthermore, it is possible to detect the roles of each member within that network, which is all possible using SNA. The SNA module builds networks from the data by assigning roles to nodes and then analyzes the connections between nodes to find the proximity between entities, acting almost like a clustering algorithm (Dreżewski et al., 2015). The authors use various data sources such as bank statements and national court registers and compare the roles assigned to nodes in networks of the two different domains. This analysis revealed crucial information about the true leaders of fraud and the vulnerabilities within the network. A node proximity module was used, which provides knowledge on the different bank accounts possessed by the same individual. The correctness of the roles assigned to members of the network

was verified using the module to find the nodes with the same roles in different networks (Dreżewski et al., 2015). Overall, the proposed was found to be useful for automated analysis of criminal networks and the identification of the patterns of interaction between offenders and the assigned roles of members within a criminal group (Dreżewski et al., 2015). This effectiveness is partly credited to the model making use of multiple data domains.

Colladon and Remondi (Colladon & Remondi, 2017) also explored the use of SNA not only to detect but also to attempt to prevent money laundering. The authors proposed a new approach to sort and map relational data, and based on certain network metrics, a predictive model is presented to assess the risk profiles of clients. In their studies, Colladon and Remondi applied the model on a factoring company's central database, which contained information about the financial operations linked to the factoring business as well as other useful information about company clients (Colladon & Remondi, 2017). The factoring business involves businesses or clients selling their accounts receivable or invoices to a third party at a discounted rate, which can occur for many reasons, such as to satisfy short-term liquidity needs. The SNA model proposed is used to predict the risk profiles associated with each of the clients involved in the business. The analysis was focused on the economic transactions and money transfers made from debtors to the factor. Four kinds of relational graphs were plotted: the first to take into account operational risks associated with the activities of the economic sector; the second to consider the risks associated with particular geographic locations; the third to study the transaction amounts; and the fourth to identify potentially dangerous links between different organizations with the same owner or representatives (Colladon & Remondi, 2017). These graphs were filtered to focus on transactions associated with a higher risk, and the risk factors were separated in the analysis to assess their importance and contributions better.

Through a visual analysis of the graphs, it was possible to identify clear clusters of subjects that were involved in court trials. Based on this, an alarm is suggested to inspect all nodes within a cluster as soon as any of them are involved in suspicious or illicit operations. Furthermore, the analysis of the four different network graphs gave evidence of the importance of studying the centrality of transactions within a network, as clients with higher risk profiles are usually less central within the network (Colladon & Remondi, 2017). The proposed model was demonstrated to be useful and effective in preventing money laundering when applied to a factoring company's database. However, the authors identify some limitations to be addressed by future studies, one being that a more significant sample size is required for analysis to determine whether or not the model may generalize well in practice (Colladon & Remondi, 2017). Another limitation is that it may be worthwhile to study the effects of additional control variables that were not present in these studies, such as the age and size of companies involved in the financial operations (Colladon & Remondi, 2017). Finally, the authors suggest that the metrics proposed in their studies should be combined with other tools that are based, for instance, on machine learning algorithms or combining data at even a national level to allow for quicker detection of suspicious nodes within the network with the aim of preventing money laundering (Colladon & Remondi, 2017).

6. Concluding remarks and suggestions

In this survey, the different ways that the problem of anomaly detection has been formulated in literature were outlined and discussed. We unified the notion of the anomaly to provide a clear theoretical understanding of the problem at hand. The methodology behind this research was driven by the mission that a comprehensive review on anomaly detection techniques should facilitate for the reader not only to be informed of the motivations behind using particular models but also of their advantages and limitations when applied to a specific area of fraud. We achieved this by detailing a comparative analysis of the various approaches implemented in each application.

The significance of detecting fraud and its detrimental effects on the financial economy was highlighted in this paper, along with the associated challenges of applying anomaly detection techniques to combat this continually growing problem. The main areas explored in this survey were credit card fraud, insurance fraud and money laundering. It was made evident that the challenges faced varied significantly based on the different fraud applications. For instance, the ability of systems to detect fraud in real-time is crucial for credit card frauds but may not be as prudent in insurance fraud detection systems. Furthermore, we showed that there is no single universally applicable anomaly detection technique or approach for all the different types of financial fraud outlined in this survey. An evident lack of publicly available datasets, labelled or not, was identified as a significant limitation in this field. We also attribute the aforementioned reason to the dearth of research on other types of financial fraud. More importantly, the imbalanced nature of datasets due to the rare occurrence of fraudulent cases was emphasized as one of, if not the most critical considerations that must be factored in during the design stage of any fraud detection system or model.

From the surveyed literature, a clear shift in trend is apparent, with most of the recent research adopting unsupervised and semi-supervised models as opposed to supervised models. We identify this to be as a result of labelled data being much harder to come by in practice due to the tediousness and cost associated with manual labelling. Even when datasets are labelled, it is often the case that not all instances of fraud have been detected. Fortunately, unsupervised and semi-supervised methods have been observed to detect frauds previously unnoticed by domain experts or detection measures that were in place. Furthermore, increased attention has been given to developing sophisticated feature selection and extraction methodologies prior to training a model. These practices have been proven to improve the effectiveness of fraud detection systems by reducing the computational complexity and increasing detection accuracy.

Generative models, which are unsupervised methods, have been shown in this survey to be especially popular in recent literature and have been applied in a variety of fraud detection systems for credit card and insurance fraud, and even anti-money laundering. These models can learn deeper and more complex representations of raw features from a training set's latent space. GANs and the various types of different AE networks are examples of generative models applied in this area. These models have been used as classifiers and have each demonstrated different strengths as either feature extraction or oversampling techniques. For example, we showed that features extracted from a training set by AEs significantly improve the performance of supervised classifiers such as the MLP. For oversampling the minority class, a semi-supervised approach addressing the imbalanced class issues associated with this research, both GANs and VAEs have proven to be superior in creating more realistic samples that capture a broader representation of data distributions for data augmentation than traditional oversampling approaches like SMOTE and ADASYN. These approaches have also proven to be preferable over those that involve undersampling the majority class, such as random undersampling, stratified sampling or even clustering algorithms dedicated to outlier detection and removal.

Other deep learning architectures that have demonstrated effectiveness and are increasingly popular in recent years, especially to detect credit card fraud, are CNNs and LSTMs. CNNs can capture short-term temporal relations and behaviours of cardholders and are less likely to overfit the data than an MLP. LSTMs, on the other hand, capture the longer-term temporal behaviours and identify cases of fraud not otherwise detected by other models when implemented in conjunction with them. These types of networks are heavily dependent on appropriate feature transformation strategies to adapt the input data into a form familiar to those types of models. Due to these transformations, LSTMs and CNNs detect frauds as contextual anomalies, which is considered a recent development in this field, as most models are often oriented to detect point anomalies.

However, the limitations of deep learning models are that they require much more careful design and tuning compared to simpler models like SVM and RF, as they are rather sensitive to the choice of hyperparameters and the architecture structure. Furthermore, the quality and amount of data required by these models is generally relatively high. As deep learning models often operate in a black-box fashion, the processes involved to arrive at their prediction may require intricate tools and strategies to aid in their interpretability. These types of networks are also very heavily dependent on appropriate feature transformation strategies to adapt the input data into a form familiar to those types of models. Furthermore, the choice of learning algorithm or loss measure used to optimize these models impacts their overall performance, which must also be factored into consideration during the design stage.

Finally, we bring attention to the matter of interpretability of fraud detection models, which allow for explainable results that may assist in applications requiring human interaction or for the inference of valuable insights to support decision-making. Specifically, it has been observed from the surveyed literature that there is a significant dearth in terms of discussion on this matter, especially when it can considered to be a critical tool for researchers and industries in developing deeper understandings of the workings of their proposed models and learning or unearthing patterns from data. For example, when dealing with simpler models like LR, DT, RF and NB, these models are intrinsically equipped with mechanisms to probabilistically determine the impact and evidence of individual components on the overall model. However, these models have been proven to be inferior in detection performance to more recent, complex deep learning methods, which are generally black-box models, and consequently have the most to benefit from being equipped with techniques to aid in the interpretability of their results. As such, we posit that several opportunities exist for future researcher in this regard, particularly with two specific models that have displayed promising performance in the research literature for explaining deeper more complex models: local interpretable model-agnostic explanations (LIME) (Ribeiro, Singh, & Guestrin, 2016) and Shapley additive explanations (SHAP) (Lundberg, 2017).

The most promising directions for future research, in our opinion, involve investigating the performance of detection models that incorporate both the oversampling and discriminative powers of generative models with the ability of LSTMs and CNNs to capture long and short-term temporal relations in data and ultimately result in a system that is more robust and efficient in detecting fraudulent cases. Further, with these findings, it may be worthwhile exploring the adaptation of the techniques explored in this survey to the less researched areas of fraud such as securities & commodities fraud, mortgage fraud, insider trading and others.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

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

Adewumi, A. O., & Akinyelu, A. A. (2018). A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management, 8*, 937–953.

Adhikari, R., & Agrawal, R. K. (2013). An Introductory Study on Time Series Modeling and Forecasting. *Retrieved from*.

Aggarwal, C. C. (2013). *Outlier analysis*. New York, NY: Springer.

Aleskerov, E., Freisleben, B., & Rao, B. (1997). *CARDWATCH: A neural network based databased mining system for credit card fraud detection*. New York City, NY: IEEE/IAFE Computational Intelligence for Financial Engineering.

American Institute of CPAs. (2011). *International financial reporting standards (IFRS)*. American Institute of CPAs.

An, J., & Cho, S. (2015). *Variational Autoencoder based Anomaly Detection using Reconstruction Probability*. SNU Data Mining Center.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein Generative Adversarial Networks. *34th International Conference on Machine Learning*. Sydney, Australia. Retrieved from arXiv:1701.07875.

Association of Certified Fraud Examiners. (2004). *What Is Fraud?* Retrieved Dec. 2, 2020, from https://www.acfe.com/fraud-101.aspx.

Association of Certified Fraud Examiners. (2019). *Insurance Fraud Manual*. Retrieved Dec. 29, 2020, from https://www.acfe.com/uploadedfiles/acfe_website/content/documents/insurance-fraud-handbook.pdf.

Association of Certified Fraud Examiners. (2020). *Report to the Nations*. Retrieved Dec. 5, 2020, from https://acfepublic.s3-us-west-2.amazonaws.com/2020-Report-to-the-Nations.pdf.

Ba, H. (2019). Improving Detection of Credit Card Fraudulent Transactions using Generative Adversarial Networks. Retrieved from arXiv:1907.03355.

Bahnsen, A. C., Aouada, D., Stojanovic, A., & Ottersten, B. (2016). Feature engineering strategies for credit card fraud detection. *Expert Systems with Applications, 51*, 134–142.

Behera, T. K., & Panigrahi, S. (2015). Credit Card Fraud Detection: A Hybrid Approach Using Fuzzy Clustering & Neural Network. *2nd International Conference on Advances in Computing and Communication Engineering*. Dehradun, India.

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(8), 1798–1828.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences, 10*(2–3), 191–203.

Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems, 50*, 602–613.

Bhusari, V., & Patil, S. (2016). Study of Hidden Markov Model in Credit Card Fraudulent Detection. *World Conference on Futuristic Trends in Research and Innovation for Social Welfare*. Coimbatore, India.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Bolton, R. J., & Hand, D. J. (2001). Unsupervised profiling methods for fraud detection. *Credit Scoring and Credit Control Conference VII*. Edinburgh, UK.

Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science, 17*(3), 235–255.

Boukerche, A., Zheng, L., & Alfandi, O. (2020). Outlier Detection: Methods, Models, and Classification. *ACM Computing Surveys, 53*(3), Article 55.

Branting, L. K., Reeder, F., Gold, J., & Champney, T. (2016). Graph Analytics for Healthcare Fraud Risk Estimation. *IEEE/ACM International Conference on Advances in Social Network Analysis and Mining (ASONAM)*. San Francisco, CA.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: identifying density-based local outliers. *ACM SIGMOD International Conference on Management of Data*. Dallas, TX.

Chalapathy, R., & Chawla, S. (2019). *Deep Learning for Anomaly Detection: A Survey*. Retrieved from arXiv:1901.03407.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys, 41*(3), 1–58.

Charitou, C., Garcez, A. d., & Dragicevic, S. (2020). Semi-supervised GANs for Fraud Detection. *International Joint Conference on Neural Networks*. Glasgow, UK.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chawla, N. V., Japkowicz, N., & Kotcz, A. (2004). Editorial: Special issue on learning from imbalanced data sets. *ACM SIGKDD Explorations Newsletter, 6*(1), 1–6.

Chen, J., Shen, Y., & Ali, R. (2018). Credit Card Fraud Detection Using Sparse Autoencoder and Generative Adversarial Network. *IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. Vancouver, BC.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco, CA.

Colladon, A. F., & Remondi, E. (2017). Using social network analysis to prevent money laundering. *Expert Systems with Applications, 67*, 49–58.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297.

Delamaire, L., Abdou, H., & Pointon, J. (2000). Credit card fraud and detection techniques: A review. *Banks and Bank Systems, 4*(2), 57–68.

Derrig, R. A. (2002). Insurance fraud. *The Journal of Risk and Insurance, 69*(3), 271–287.

Dhieb, N., Ghazzai, H., Besbes, H., & Massoud, Y. (2020). A secure AI-driven architecture for automated insurance systems: Fraud detection and risk measurement. *IEEE Access, 8*, 58546–58558.

Dhok, S. S. (2012). Credit card fraud detection using hidden markov model. *International Journal of Soft Computing and Engineering, 2*(1), 88–92.

Di Mattia, F., Galeone, P., De Simoni, M., & Ghelfi, E. (2019). *A Survey on GANs for Anomaly Detection*. Retrieved from arXiv:1906.11632.

Doersch, C. (2016). *Tutorial on Variational Autoencoders*. Retrieved from arXiv: 1606.05908.

Domingues, R., Filippone, M., Michiardi, P., & Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition, 74*, 406–421.

Dreżewski, R., Sepielak, J., & Filipkowski, W. (2015). The application of social network analysis algorithms in a system supporting money laundering detection. *Information Sciences, 295*, 18–32.

Federal Bureau of Investigation. (2011). *Financial Crimes Report to the Public.* Retrieved Dec. 13, 2020, from https://www.fbi.gov/file-repository/stats-services-publications-financial-crimes-report-2010-2011-financial-crimes-report-2010-2011.pdf/view.

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

Fisher, N. I. (1993). *Statistical analysis of circular data.* Cambridge, UK: Cambridge University Press.

Freund, Y., & Schapire, R. E. (1996). Experiments with a New Boosting Algorithm. *13th International Conference on Machine Learning.* San Francisco, CA.

Friedman, N., Geiger, D., & Goldzsmidt, M. (1997). Bayesian network classifiers. *Machine Learning, 29*(2–3), 131–163.

Fu, K., Cheng, D., Tu, Y., & Zhang, L. (2016). Credit Card Fraud Detection Using Convolutional Neural Networks. *13th Conference on Neural Information Processing Systems.* Barcelona, Spain.

Fujimaki, R., Yairi, T., & Machida, K. (2005). An approach to spacecraft anomaly detection problem using kernel feature space. *11th ACM SIGKDD international conference on Knowledge discovery in data mining.* Chicago, IL.

Gama, J., Žliobaitė, I., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys, 46*(4), Article 44.

Gao, Z., & Ye, M. (2007). A framework for data mining-based anti-money laundering research. *Journal of Money Laundering Control, 10*(2), 170–179.

Gers, F., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation, 12*(10), 2451–2471.

Ghosh, S., & Reilly, D. L. (1994). Credit Card Fraud Detection with a Neural-Network. *27th International Conference on System Sciences*.

Gómez, J. A., Arévalo, J., Paredes, R., & Nin, J. (2018). End-to-end neural network architecture for fraud scoring in card payments. *Pattern Recognition Letters, 105*, 175–181.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems, 27*, 2672–2680.

Gupta, M., Gao, J., Aggarwal, C. C., & Han, J. (2014). Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering, 26*(9), 2250–2267.

Han, C., Rundo, L., Murao, K., Noguchi, T., Shimahara, Y., Milacski, Z. A., . . . Satoh, S. (2020). *MADGAN: unsupervised Medical Anomaly Detection GAN using multiple adjacent brain MRI slice reconstruction.* Retrieved from arXiv:2007.13559v2.

Hawkins, D. M. (1980). *Identification of outliers.* Heidelberg, Germany: Springer.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *IEEE International Joint Conference on Neural Networks.* Hong Kong, China.

He, H., Wang, J., Graco, W., & Hawkins, S. (1997). Application of neural networks to detection of medical fraud. *Expert Systems with Applications, 13*(4), 329–336.

Hejazi, M., & Singh, Y. P. (2013). One-class support vector machines approach to anomaly detection. *Applied Artificial Intelligence, 27*(5), 351–366.

Heryadi, Y., & Warnars, H. L. (2017). Learning temporal representation of transaction amount for fraudulent transaction recognition using CNN, Stacked LSTM, and CNN-LSTM. *IEEE International Conference on Cybernetics and Computational Intelligence.* Phuket, Thailand.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Hodge, V. J., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review, 22*(2), 85–126.

Huang, S.-Y., Tsaih, R.-H., & Yu, F. (2014). Topological pattern discovery and feature extraction for fraudulent financial reporting. *Expert Systems with Applications, 9*(41), 4360–4372.

Internet Crime Complaint Center. (2019). *2019 Internet Crime Report.* Retrieved Dec. 2, 2020, from https://pdf.ic3.gov/2019_IC3Report.pdf.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *32nd International Conference on Machine Learning.* Lille, France.

Jiang, P., Zhang, J., & Zou, J. (2019). *Credit Card Fraud Detection Using Autoencoder Neural Network.* Retrieved from arXiv:1908.11553.

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.-E., He-Guelton, L., & Caelen, O. (2018). Sequence classification for credit-card fraud detection. *Expert Systems with Applications, 100*, 234–245.

Kazemi, Z., & Zarrabi, H. (2017). Using deep networks for fraud detection in the credit card transactions. *IEEE 4th International Conference on Knowledge-Based Engineering and Innovation.* Tehran, Iran.

Keller, F., Muller, E., & Bohm, K. (2012). HiCS: High Contrast Subspaces for Density-Based Outlier Ranking. *IEEE 28th International Conference on Data Engineering.* Washington, DC.

Khan, A. U., Akhtar, N., & Qureshi, M. N. (2014). Real-Time Credit-Card Fraud Detection using Artificial Neural Network Tuned by Simulated Annealing Algorithm. *International Conference on Recent Trends in Information, Telecommunications and Computing.* Chandigarh, India.

Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization.* Retrieved from arXiv:1412.6980.

Kirkos, E., Spathis, C., & Manolopoulos, Y. (2007). Data Mining techniques for the detection of fraudulent financial statements. *Expert Systems with Applications, 32*(4), 995–1003.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE, 78*(9), 1464–1480.

Kou, Y., Lu, C.-T., Sirwongwattana, S., & Huang, Y.-P. (2004). Survey of Fraud Detection Techniques. *IEEE International Conference on Networking, Sensing and Control.* Taipei, Taiwan.

KPMG. (2012). *Auto Insurance Fraud in Ontario.* Retrieved Dec. 29, 2020, from https://107feb26-9cf5-48a4-8339-7e0bceae4740.filesusr.com/ugd/21ffb0_83e3830e4b694e6ea658216ad80caca4.pdf.

Kremmler, M., Rodner, E., Wacker, E.-S., & Denzler, J. (2013). One-class classification with Gaussian processes. *Pattern Recognition, 46*(12), 3507–3518.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Networks. *25th Advances in Neural Information Processing Systems.* Grenada, Spain.

Kucharski, D., Kłeczek, P., Jaworek-Korjakowska, J., Dyduch, G., & Gorgon, M. (2020). Semi-supervised nests of melanocytes segmentation method using convolutional autoencoders. *Sensors, 20*(6).

Lazarevic, A., & Kumar, V. (2005). Feature bagging for outlier detection. *11th International Conference on Knowledge Discovery in Data Mining.* Chicago, IL.

LeCun, Y., & Bengio, Y. (1998). Convolutional networks for images, speech, and time series. *In The handbook of brain theory and neural networks*.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. *8th IEEE International Conference on Data Mining.* Beijing, China.

Lu, Q., & Ju, C. (2011). Research on credit card fraud detection model based on class weighted support vector machine. *Journal of Convergence Information Technology, 6*(1), 62–68.

Lucas, Y., & Jurgovsky, J. (2020). *Credit card fraud detection using machine learning: A survey.* Retrieved from arXiv:2010.06479.

Lucas, Y., Portier, P.-E., Laporte, L., He-Guelton, L., Caelen, O., Granitzer, M., & Calabretto, S. (2020). Towards automated feature engineering for credit card fraud detection using multi-perspective HMMs. *Future Generation Computer Systems, 102*, 393–402.

Lundberg, S. M. (2017). A Unified Approach to Interpreting Model Predictions. *31st Conference on Neural Information Processing Systems.* Long Beach, CA.

Machine Learning Group. (2017). *Credit Card Fraud Detection (European data set).* Retrieved Dec. 10, 2020, from https://www.kaggle.com/mlg-ulb/creditcardfraud.

MacKay, D. J. (1992). The evidence framework applied to classification networks. *Neural Computation, 4*(5), 720–736.

Maes, S., & Tuyls, K. (2002). *Credit card fraud detection using bayesian and neural networks.* Havana, Cuba: First International NAISO Congress on Neuro Fuzzy Technologies.

Mahmoudi, N., & Duman, E. (2015). Detecting credit card fraud by modified fisher discriminant analysis. *Expert Systems with Applications, 42*(5), 2510–2516.

Makhzani, A., & Frey, B. (2014). *k-Sparse Autoencoders.* Retrieved from https://arxiv.org/abs/1312.5663.

Markou, M., & Singh, S. (2003). Novelty detection: A review—part 1: Statistical approaches. *Signal Processing, 83*(12), 2481–2497.

Michelucci, U. (2018). Feedforward Neural Networks. In *Applied Deep Learning: A Case-Based Approach to Understanding Deep Neural Networks* (pp. 83–136). Berkeley, CA: Apress.

Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software, 95*, 51–67.

Mirza, M., & Osindero, S. (2014). *Conditional Generative Adversarial Nets.* Retrieved from arXiv:1411.1784.

Misra, S., Thakur, S., Ghosh, M., & Saha, S. K. (2020). An autoencoder based model for detecting fraudulent credit card transaction. *Procedia Computer Science, 167*, 254–262.

Møller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks, 6*(4), 525–533.

Moschini, G., Houssou, R., Bovay, J., & Robert-Nicoud, S. (2020). *Anomaly and Fraud Detection in Credit Card Transactions Using the ARIMA Model.* Retrieved from arXiv:2009.07578.

National Health Care Anti-Fraud Association. (2018). *The Challenge of Health Care Fraud.* Retrieved Dec. 30, 2020, from https://www.nhcaa.org/resources/health-care-anti-fraud-resources/the-challenge-of-health-care-fraud/.

Neal, R. M. (1996). *Bayesian learning for neural networks.* New York: Springer.

Ngai, E., Hu, Y., Wong, Y., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems, 50*(3), 559–569.

Nian, K., Zhang, H., Tayal, A., Coleman, T., & Li, Y. (2016). Auto insurance fraud detection using unsupervised spectral ranking for anomaly. *The Journal of Finance and Data Science, 2*(1), 58–75.

Olszewski, D. (2014). Fraud detection using self-organizing map visualizing the user profiles. *Knowledge-Based Systems, 70*, 324–334.

O'Shea, K., & Nash, R. (2015). *An Introduction to Convolutional Neural Networks.* Retrieved from arXiv:1511.08458.

Ounacer, S., Ait El Bour, H., Oubrahim, Y., Ghoumari, M. Y., & Azzouazi, M. (2018). Using Isolation Forest in anomaly detection: The case of credit card transactions. *Periodicals of Engineering and Natural Sciences, 6*(2), 394–400.

Pang, G., Shen, C., Cao, L., & van den Hengel, A. (2020). Deep learning for anomaly detection: A review. *ACM Computing Surveys, 1*(1).

Patidar, R., & Sharma, L. (2011). Credit card fraud detection using neural network. *International Journal of Soft Computing and Engineering, 1*, 32–38.

Paula, E. L., Ladeira, M., Carvalho, R. N., & Marzagão, T. (2016). Deep Learning Anomaly Detection as Support Fraud Investigation in Brazilian Exports and Anti-Money Laundering. *15th IEEE International Conference on Machine Learning and Applications.* Anaheim, CA.

Phua, C., Alahakoon, D., & Lee, V. (2004). June). Minority report in fraud detection: Classification of skewed data. *ACM SIGKDD Explorations Newsletter, 6*(1), 50–59.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). *A Comprehensive Survey of Data Mining-based Fraud Detection Research.* Retrieved from arXiv:1009.6119.

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

Pozzolo, A. D., Caelen, O., Le Borgne, Y.-A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. *Expert Systems with Applications, 41*(10), 4915–4928.

Pumsirirat, A., & Yan, L. (2018). Credit card fraud detection using deep learning based on auto-encoder and restricted boltzmann machine. *International Journal of Advanced Computer Science and Applications, 9*(1), 18–25.

Quah, J. T., & Sriganesh, M. (2008). Real-time credit card fraud detection using computational intelligence. *Expert Systems with Applications, 35*(4), 1721–1732.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE, 77*(2), 257–286.

Renström, M., & Holmsten, T. (2018). *Fraud Detection on Unlabeled Data with Unsupervised Machine Learning.* Dissertation.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?" Explaining the Predictions of Any Classifier. *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data.* San Francisco, CA.

Ridgeway, G., Madigan, D., & Richardson, T. (1998). Interpretable Boosted Naïve Bayes Classification. *4th International Conference on Knowledge Discovery and Data.* New York, NY.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive Auto-Encoders: Explicit Invariance During Feature Extraction. *28th International Conference on Machine Learning*.

Robinson, W. N., & Aria, A. (2018). Sequential fraud detection for prepaid cards using hidden Markov model divergence. *Expert Systems with Applications, 91*, 235–251.

Rtayli, N., & Enneya, N. (2020). Selection Features and Support Vector Machine for Credit Card Risk Identification. *13th International Conference Interdisciplinarity in Engineering.* Targu Mures, Romania.

Sabau, A. S. (2012). Survey of clustering based financial fraud detection research. *Informatica Economica, 16*(1), 110–122.

Sahin, Y., & Duman, E. (2011). Detecting Credit Card Fraud by Decision Trees and Support Vector Machines. *International MultiConferences of Engineers and Computer Scientists. Hong Kong*.

Sánchez, D., Vila, M., Cerda, L., & Serrano, J. (2009). Association rules applied to credit card fraud detection. *Expert Systems with Applications, 36*(2), 3630–3640.

Saradjian, M. R., & Akhoondzadeh, M. (2011). Thermal anomalies detection before strong earthquakes (M >6.0) using interquartile, wavelet and Kalman filter methods. *Natural Hazards and Earth System Sciences, 11*(4), 1099–1108.

Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., & Platt, J. C. (2000). *Support vector method for novelty detection*. Denver CO: Advances in Neural Information Processing Systems.

Schott, P. A. (2004). *Reference guide to anti-money laundering and combating the financing of terrorism*. Washington, DC: The World Bank.

Singh, K., & Upadhyaya, S. (2012). Outlier detection: Applications and techniques. *International Journal of Computer Science, 9*(1), 307–323.

Sparrow, M. K. (2000). *License to steal: How fraud bleeds America's health care system*. Denver, CO: Westview Press.

Spence, C., Parra, L. C., & Sajda, P. (2001). *Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model*. Kauai, HI: IEEE Workshop on Mathematical Methods in Biomedical Image Analysis.

Srivastava, A., Kundu, A., Sural, S., & Majumdar, A. K. (2008). Credit card fraud detection using hidden markov model. *IEEE Transactions on Dependable and Secure Computing, 5*(1), 37–48.

Stripling, E., Baesens, B., Chizi, B., & vanden Broucke, S. (2018). Isolation-based conditional anomaly detection on mixed-attribute data to uncover workers' compensation fraud. *Decision Support Systems, 11*, 13–26.

Šubelj, L., Furlan, Š., & Bajec, M. (2011). An expert system for detecting automobile insurance fraud using social network analysis. *Expert Systems with Applications, 38*(1), 1039–1052.

Subudhi, S., & Panigrahi, S. (2020). Use of optimized Fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection. *Journal of King Saud University - Computer and Information Sciences, 32*, 568–575.

Sundarkumar, G. G., & Ravi, V. (2015). A novel hybrid undersampling method for mining unbalanced datasets in banking and insurance. *Engineering Applications of Artificial Intelligence, 37*, 368–377.

Sundarkumar, G. G., Ravi, V., & Siddeshwar, V. (2015). One-class support vector machine based undersampling: Application to churn prediction and insurance fraud detection. *IEEE International Conference on Computational Intelligence and Computing Research.* Madurai, India.

Sweers, T., Heskes, T., & Krijthe, J. (2018). *Autoencoding Credit Card Fraud.* Retrieved from https://www.cs.ru.nl/bachelors-theses/2018/Tom_Sweers__4584325__Autoencoding_credit_card_fraude.pdf.

Syeda, M., Zhang, Y.-Q., & Pan, Y. (2002). *Parallel granular neural networks for fast credit card fraud detection*. Honolulu, HI: IEEE World Congress on Computational Intelligence.

Tanaka, F. H., & Aranha, C. (2019). *Data Augmentation Using GANs.* Retrieved from arXiv: 1904.09135.

Tao, H., Zhixin, L., & Xiaodong, S. (2012). Insurance Fraud Identification Research Based on Fuzzy Support Vector Machine with Dual Membership. *International Conference on Information Management, Innovation Management and Industrial Engineering.* Sanya, China.

Tax, D. M., & Duin, R. P. (2001). Uniform object generation for optimizing one-class classifiers. *Journal of Machine Learning Research, 2*, 155–173.

Thangavel, K., & Pethalakshmi, A. (2009). Dimensionality reduction based on rough set theory: A review. *Applied Soft Computing, 9*(1), 1–12.

The Nilson Report. (2020). *Card Fraud Losses Reach $28.65 Billion.* Retrieved Dec. 22, 2020, from https://nilsonreport.com/mention/1313/1link/.

Tingfei, H., Guangquan, C., & Kuihua, H. (2020). Using variational auto encoding in credit card fraud detection. *IEEE Access, 8*, 149841–149853.

Toledo, T., & Katz, R. (2009). State dependence in lane-changing models. *Transportation Research Record: Journal of the Transportation Research Board, 2124*(1), 81–88.

University of California, Irvine. (2000). *Statlog (German Credit Data) Data Set.* Retrieved Dec. 23, 2020, from https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data).

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research, 9*(86), 2579–2605.

Van Hulse, J., Khoshgoftaar, T. M., & Napolitano, A. (2007). Experimental Perspectives on Learning from Imbalanced Data. *24th International Conference on Machine Learning.* Corvalis, OR.

Viaene, S., Dedene, G., & Derrig, R. A. (2005). Auto claim fraud detection using Bayesian learning neural networks. *Expert Systems with Applications, 29*(3), 653–666.

Viaene, S., Derrig, R. A., & Dedene, G. (2004). A case study of applying boosting naive bayes to claim fraud diagnosis. *IEEE Transactions on Knowledge and Data Engineering, 16*(5), 612–620.

Viaene, S., Derrig, R. A., Baesens, B., & Dedene, G. (2002). A comparison of state-of-the-art classification techniques for expert automobile insurance claim fraud detection. *The Journal of Risk and Insurance, 69*(3), 373–421.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research, 11*, 3371–3408.

Wang, C., Wang, Y., Ye, Z., Yan, L., Cai, W., & Pan, S. (2018). Credit Card Fraud Detection Based on Whale Algorithm Optimized BP Neural Network. *13th International Conference on Computer Science & Education.* Colombo, Sri Lanka.

Wang, X., Du, Y., Lin, S., Cui, P., Shen, Y., & Yang, Y. (2020). adVAE: A self-adversarial variational autoencoder with Gaussian anomaly prior knowledge for anomaly detection. *Knowledge-Based Systems, 190*.

Wang, Y., & Xu, W. (2018). Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud. *Decision Support Systems, 105*, 87–95.

West, J., & Bhattacharya, M. (2016). Intelligent financial fraud detection: A comprehensive review. *Computers & Security, 57*, 47–66.

Wiese, B., & Omlin, C. (2009). Credit Card Transactions, Fraud Detection, and Machine Learning: Modelling Time Series with LSTM Recurrent Neural Networks. In *Innovations in Neural Information Paradigms and Applications* (pp. 231–268). Heidelberg, Germany: Springer.

Xiong, H., Pandey, G., Steinbach, M., & Kumar, V. (2006). Enhancing data analysis with noise removal. *IEEE Transactions on Knowledge and Data Engineering, 18*(3), 304–319.

Xu, W., Wang, S., Zhang, D., & Yang, B. (2011). Random Rough Subspace based Neural Network Ensemble for Insurance Fraud Detection. *4th International Joint Conference on Computational Sciences and Optimization.* Yunnan, China.

Yang, W.-S., & Hwang, S.-Y. (2006). A process-mining framework for the detection of healthcare fraud and abuse. *Expert Systems with Applications, 31*(1), 56–68.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation, 31*(7), 1235–1270.

Zareapoor, M., R, S. K., & Alam, M. A. (2012). Analysis of credit card fraud detection techniques: based on certain design criteria. *International Journal of Computer Applications, 52*(3), 35-42.

Zaslavsky, V., & Strizhak, A. (2006). Credit card fraud detection using self-organizing maps. *Information and Security, 18*, 48–63.

Zhang, C., Cheng, X., Liu, J., & Liu, G. (2018). Deep Sparse Autoencoder for Feature Extraction and Diagnosis of Locomotive Adhesion Status. *Journal of Control Science and Engineering, 2018*.

Zhang, Z., Zhou, X., Zhang, X., Wang, L., & Wang, P. (2018). *A model based on convolutional neural network for online transaction fraud detection.* Security and Communication Networks.

Zheng, P., Yuan, S., Wu, X., Li, J., & Lu, A. (2019). One-class adversarial nets for fraud detection. *AAAI, 33*(1), 1286–1293.

Zimek, A., Schubert, E., & Kriegel, H.-P. (2012). A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining, 5*(5), 363–387.