Engineering Applications of Artificial Intelligence 76 (2018) 130–157

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Survey paper

How Artificial Intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark

Nick F. Ryman-Tubb [a,*], Paul Krause [b], Wolfgang Garn [c]

[a] Room 28MS02, The Rik Medlik Building, University of Surrey, Stag Hill, Guildford GU2 7XH, UK
[b] Department of Computer Science, University of Surrey, Guildford, UK
[c] Business Analytics Group, Department of Business Transformation, University of Surrey, Guildford, UK

ARTICLE INFO

Keywords:
Fraud detection
Financial crime
AI
Machine learning
Payments card
Cyber-crime
Translational research

ABSTRACT

The core goal of this paper is to identify guidance on how the research community can better transition their research into payment card fraud detection towards a transformation away from the current unacceptable levels of payment card fraud. Payment card fraud is a serious and long-term threat to society (Ryman-Tubb and d'Avila Garcez, 2010) with an economic impact forecast to be $416bn in 2017 (see Appendix A).[1] The proceeds of this fraud are known to finance terrorism, arms and drug crime. Until recently the patterns of fraud (*fraud vectors*) have slowly evolved and the criminals *modus operandi* (MO) has remained unsophisticated. Disruptive technologies such as smartphones, mobile payments, cloud computing and contactless payments have emerged almost simultaneously with large-scale data breaches. This has led to a growth in new fraud vectors, so that the existing methods for detection are becoming less effective. This in turn makes further research in this domain important. In this context, a timely survey of published methods for payment card fraud detection is presented with the focus on methods that use AI and machine learning. The purpose of the survey is to consistently benchmark payment card fraud detection methods for industry using transactional volumes in 2017. This benchmark will show that only eight methods have a practical performance to be deployed in industry despite the body of research. The key challenges in the application of artificial intelligence and machine learning to fraud detection are discerned. Future directions are discussed and it is suggested that a cognitive computing approach is a promising research direction while encouraging industry data philanthropy.

1. Introduction

For the first time, fraud detection works are all consistently benchmarked and ranked contemporaneously using industry volumes from 2017. This industry benchmark and survey indicates that despite the academic validity of the research surveyed, its impact on the payment card industry has been minimal. Additional evaluation metrics to explicate the business impact of each fraud detection approach are identified. These show that whilst a fraud detection algorithm may perform well in terms of standard academic measures of accuracy, they can fail to address the broader business context. It is argued that it is important to broaden the evaluation criteria in this way in order to transition this programme of research into a level of technical readiness that is required for impact and to attract the interest of industry (Campolo et al., 2017). This need to meet the challenges of industry is increasingly being recognised globally. For example, the UK Government Industrial Strategy White Paper, specifically highlights the need for funding to "*help service industries to identify how the application of these technologies can transform their operations*" (UK-Government, 2017).

Cashless payments can be made to purchase services/goods using a payment card without the need for physical banknotes. Payment card fraud is the criminal act of deception using a physical (plastic) card or Card-Holder Data (CHD) without the knowledge of the genuine cardholder (Ryman-Tubb and Krause, 2011). CHD is vulnerable to being compromised by criminals who use it to undertake fraud so as to be monetised. A fraud vector consists of a specific sequence of operations to undertake payment card fraud that have been subsequently recognised or detected by law enforcement or fraud experts and reported. There are a wide range of fraud vectors discussed in detail in Shen et al. (2007).

* Corresponding author.
E-mail address: n.ryman-tubb@surrey.ac.uk (N.F. Ryman-Tubb).

[1] A prefix of $ indicates the USA Dollar (USD) value for that variable. 1m = One million ($1x10^6$), 1bn = One billion ($1x10^9$) and 1tn = One trillion ($1x10^{12}$). Appendix A details terms, abbreviations, sources and computation of industry data used. Plotted points and values may contain errors due to the uncertainties in industry figures; error-bars are omitted. Where tables are sorted this is indicated.

https://doi.org/10.1016/j.engappai.2018.07.008
Received 17 November 2017; Received in revised form 28 June 2018; Accepted 24 July 2018

Since the launch of general payments cards in 1950s, fraud vectors have become established over time and became well-known to the industry. Until recently, criminal methods have changed only slowly (Mann, 2006b) which may partly explain the lack of research impetus. Until the 1970s every transaction was processed using paper documents that were physically posted (Evans and Schmalensee, 2005). With the development of the magnetic stripe to store CHD that could be automatically read by terminals, the process could be automated (Svigals, 2012). It was at this point that early research started to focus on the simple automation of detecting fraud and to devise new methods using rules (Parker, 1976). It was not until 1994 that the earliest significant work (Ghosh and Reilly, 1994) was published in this domain.

It will be demonstrated in Sections 2 and 3 that from the earliest work, only a small improvement has been made by the research community, bringing limited impact on the reduction of payment card fraud detection. It is discussed in Section 4 that some of this earliest work is ranked in the top quartile of all works. It is then identified in Section 5 that there is a gap in research into improved systemic methods to manage fraud and future directions are suggested. The following sections outline the contact of payment card fraud, the research challenges. Industry metrics are proposed so that the effectiveness of each method is determined and can then be usefully ranked in a benchmark. Thus, the "state of the art" in fraud detection methods is established.

1.1. The growth of payments and payment card fraud

It is important to review the background of payment card fraud so that the motivation to devise methods to tackle the problem can be understood in context. It is argued here that the economic health, day-to-day government social and cultural existence of citizen's is threatened by the continued growth in payment card fraud and yet research has made slow progress in terms of impact. Society is now a cyber-society dependent on the continued availability, accuracy and confidentiality of information stored, processed and communicated by computers. Businesses and citizens all benefit from this infrastructure and the rapid advancement of cyber-technology including the ability to make rapid secure payments. If fraud reaches a point where security or an economy is sufficiently threatened, trust in these systems will be damaged and their use endangered.

Unfortunately, general society perceive payment card fraud as a minor crime where its effects are mitigated by their issuer refunding any personal fraud; the individual impact to the victim of fraud is softened. There is a common belief that (1) payment fraud only affects banks, big business and government and (2) that the fraud is undertaken by individuals and typically by "bedroom hackers" (Castle, 2008). However, it has been identified that criminal enterprises and Organised Crime Groups (OCGs) use payment card fraud to fund their activities including arms, drugs and terrorism (Financial-Fraud-Action-UK, 2014). The activities of these criminals include violence and murder (Everett, 2003; Jacobson, 2010)—individual acts of fraud have a human cost. In 2017, it is forecast that there will be 349 bn payment card transactions with Card Expenditure Volume ($\$CEV$) at $26.3 tn with direct fraud losses ($\$fraud$) at $24 bn; it is here calculated that the economic impact is a minimum of $416 bn (Appendix A). Fig. 1 shows the exponential growth of $\$CEV$ and $\$fraud$. In 2017, it is forecast that for the first time $\$fraud$ will grow more rapidly than $\$CEV$. As argued in Ryman-Tubb (2011), the same technology that has enabled cashless payments is fuelling exponential growth in payment card fraud.

1.2. Payment card transaction process

There are multiple participants that are involved when a cashless transaction takes place (see Fig. 2). When a merchant wishes to take payment from a cardholder's payment card, then the details of that transaction are passed to the merchant's acquirer. The acquirer then requests authorisation from the cardholder's card issuer and the transaction is approved or declined. This decision is then passed back to the merchant to complete the transaction. If the transaction is *authorised* then the sale is completed and the goods are taken or dispatched.

1.3. Fraud Management System (FMS)

To determine if a payment card transaction is authorised, a number of processes are undertaken, one of which includes the FMS. The FMS receives a transaction, makes a decision using some form of classifier and returns this as part of the authorisation process. If the transaction is determined to be suspicious it is typically blocked or declined and a *fraud ticket* is created. This fraud ticket contains sufficient information for a human *reviewer* to understand the transaction and then make a decision. In most organisations, a team of reviewers check fraud tickets and an investigation is undertaken that might include contacting the cardholder or merchant.

1.4. Major challenges in real-world fraud detection

The timely understanding and detection of fraud vectors is fundamental to reducing the growing payment card fraud problem. The complex scientific and industry challenges of detecting payment card fraud through the use of AI and machine learning have been identified in this survey and each is discussed in the following sections. Specific applications in the near future and research directions are discussed in Section 5.

1.4.1. Transparent decisions

It is argued that an important factor limiting the impact of research is that the majority of published methods are *black-boxes* where their workings are mysterious; the inputs and its decision on fraud can be observed but how one becomes the other is opaque. They cannot easily explain their decisions or reasoning so that humans cannot understand the new emerging fraud vectors. However, industry considers that it is only the timely understanding of new fraud vectors that will allow improved prevention methods to be put in place. For fraud practitioners, it is argued that comprehensible classifiers are essential to guide them towards a particular type of investigation and towards creating prevention that is more effective.

> "*Gaps in knowledge, putative and real, have powerful implications as do the uses that are made of them. Alan Greenspan, once the most powerful central banker in the world, claimed that today's markets are driven by an 'unredeemably opaque' version of Adam Smith's 'invisible hand' and that no one (including regulators) can ever get more than a glimpse at the internal workings of the simplest of modern financial systems".* (Pasquale, 2015).

1.4.2. Cost of fraud detection to the payments industry

If academic research is to have a greater industry impact then it is argued that researchers need to understand that costs are a key motivation within the payments industry. For example, in practice most FMS produce a large volume of $AlertD$ that must be matched against available and costly human review resource and so the issue of prioritisation requires attention. It is argued that only if the various costs are taken into account that a more effective FMS can be created (Hand et al., 2008). The output of a fraud detection system requires human reviewers to investigate alerts generated. There is an operational cost for such a process — with the number of reviewers, experts and the required IT being a significant proportion (typically 30% of the value of fraud write-offs in 2017). An illustration of the size of a review team is given in Appendix A.

The accuracy of a fraud detection model can be set so as to detect all fraud but this will have a resultant uneconomical increase in the operational cost to detect the fraud, as $AlertD$ becomes unrealistic. Therefore, a commercial decision must be made between these costs and the impact and savings by detecting fraud (Bose, 2006). This is further complicated as "disturbing good customers" by contacting them about an alerted transaction that is not fraud does not inspire customer confidence; implying to the innocent customer that there is the suspicion of fraud is likely detrimental to good relations (Leonard, 1993). Few methods take this into account.

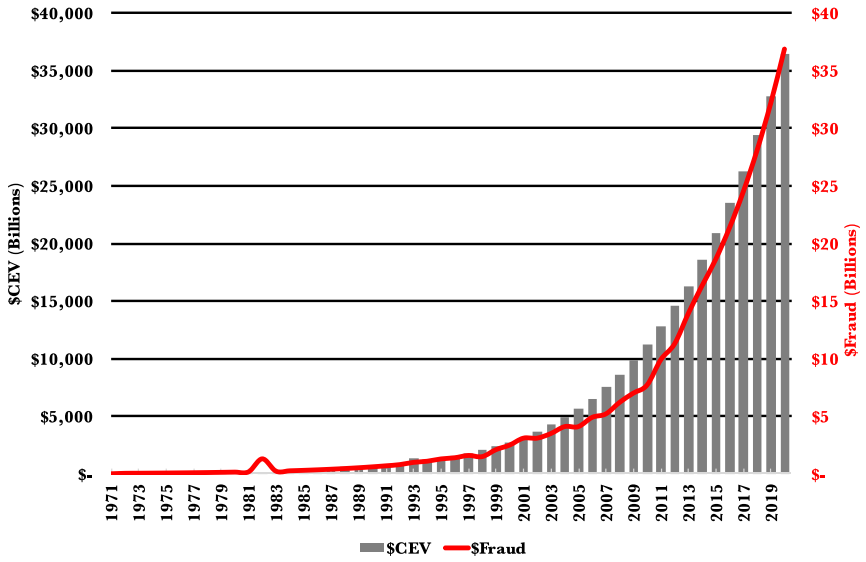

Fig. 1. Worldwide payment card volume and fraud write-off by value (source: Appendix A).

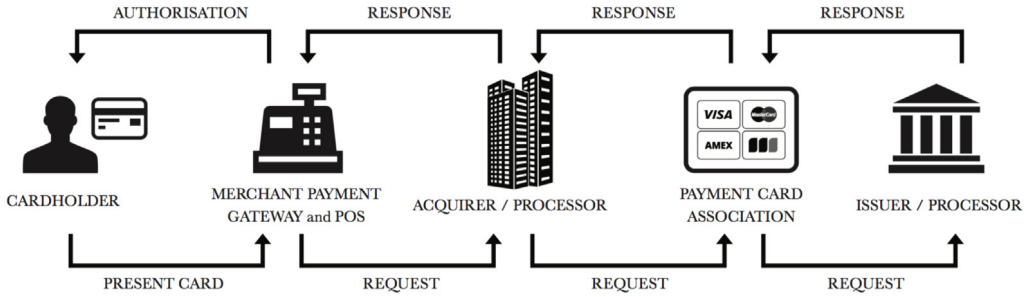

Fig. 2. Payment card authorisation process.

1.4.3. Lack of large-scale and sensitive real-world datasets

Researchers have reported that the exchange of ideas in fraud detection and specifically in payment card fraud detection is severely limited due to security and privacy concerns, especially following publicised data breaches. Even when datasets are available from industry the results are censored making it difficult to assess the work as a whole, for example Sahin et al. (2013). Some researchers in this survey have had to use synthetic datasets that try to replicate real-world data (Lopez-Rojas and Axelsson, 2014). As the profiles of genuine and fraudulent behaviours change over time, synthetic data may be insufficiently rich. Therefore, there is no reason to suspect that results cited will necessarily be the same when scaled using real-world data. It is argued that any dataset with fewer than c.1 m records must be considered "small" in the context of financial transaction datasets and around 75% of the surveyed studies used small datasets. Therefore, the reported results in the survey may be unreliable when scaled to larger datasets. This highlights the inability of the research community to realistically demonstrate impact to industry.

The data held on each transaction including the CHD, the cardholder and merchant is sensitive. It is straightforward to use this data to perpetrate fraud. This makes it difficult for the payment processors to provide data for researchers to assess new detection methods. There are methods of obfuscation that could be used while maintaining the relationships within the data but this process requires the data-holder to be assured that the original data could not be recreated or imputed (Shokri, 2015). There are laws in different jurisdictions that forbid such data from leaving their borders as well as data protection, e.g. the EU General Data Protection Regulation (GDPR) (European-Union, 2016) and other laws that make this process increasingly difficult (Yuen, 2008). It is for this reason that those that hold such large-scale real-world datasets are reluctant or unable to make them available for research where the results can be subsequently published to the wider research community.

It is necessary to understand that the data available to an FMS depends on which payment participant has deployed the system. A merchant only has data on the transactions that have occurred at their firm and does not have information on other transactions that have been undertaken by a particular cardholder. The issuer only has data on the transactions that have been undertaken on their issued card by the cardholder and has no information on any transactions that have been carried out by other means by their customer on the products or services purchased. The acquirer typically only has the transactional information from the merchant along with information they keep on their merchants such as the original application data and statistics on their transactions over a period. Data is spread among many different interconnected computer systems. This is a considerable challenge to the research community.

1.4.4. Fraud model metrics

The fraud detection problem is defined as determining if a payment transaction is genuine and so authorised or suspicious (potential fraud) and so blocked/passed for review. It is expected that fraud vectors and criminal MO follows certain patterns that have similarities (Turvey, 2011). There is a sequence of events or arrangement of transactions that is undertaken by the criminals for a particular fraud vector. Since reviewers have reported recognisable fraud vectors then it is argued that each fraud vector has some common attributes. It follows that automated methods may be able to recognise such common attributes to

discriminate transactions. There are generally three types of classifier: (1) Rules, (2) Supervised classifier, (3) Anomaly classifier. In the surveyed literature, a dataset is used to evaluate performance, as defined in Stanford-Research-Institute (2008). In some methods, a stratified k-folded cross-validation approach is taken which aims to provide results that are indicative of performance on a more generalised and independent dataset, discussed in Japkowicz and Shah (2011).

To measure the performance of a classifier typically a confusion matrix is used, a detailed discussion is given in Sokolova and Lapalme (2009). This is used to evaluate the performance of a two-class model based on the classifier decision at a fixed threshold θ and that of the known class label. True Positive (TP) is defined as a fraud transaction was expected and was correctly classified by the decision system. In some published work, this is defined as a True Negative (TN) and where this is the case the reported figures are converted to the definition given here. Various metrics are presented in the surveyed studies based on their reported confusion matrix. $accuracy$ in is often presented as a measure of performance but it is known to not be a reliable metric when the dataset is unbalanced as in real-world fraud datasets (Provost et al., 1998a). The precision of fraud transactions is the number of fraud transactions correctly identified out of the total number of identified fraud transactions. The false-positive rate is the number of genuine transactions that were wrongly identified as fraud out of all known genuine transactions.

The $F\text{-}score$ is a single metric that indicates how many fraud transactions are correctly classified and how many are missed. It does not include TN which in this domain is important, as FPR is a key metric for real-world fraud detection. The $F\text{-}score$ is especially biased when there is a large class imbalance and so does not provide a useful comparison metric. It is included here only because it is reported in many of the surveyed studies.

In some studies, a Receiver Operating Characteristic curve (ROC) is used to determine other performance metrics as it indicates how well the classifier is able to be *specific* and *sensitive* simultaneously over a range of measurements, e.g. Provost et al. (1998b) and Vuk and Curk (2006). ROC space is insensitive to class imbalance and so does not take into account the class ratio. Consequently, selecting the threshold/operating point θ as the "optimal" trade-off between cost of failing to detect positives versus the cost of raising false alarms does not necessarily represent the real-world "best" point. A third dimension that is sensitive to class imbalances will yield different points as a slice in ROC space. This enables the characterisation of the classifier over different class distributions, e.g., a business may wish to reduce false alerts while eschewing precision of fraud detection. This is an important real-world decision point.

An improved single measure is the Matthews Correlation Coefficient (MCC) (Matthews, 1975), Eq. (1), which is a single measure that can be used in highly unbalanced data as it takes into account true/false positives and true/false negatives. MCC is a correlation coefficient between the observed and predicted binary class with a value $[-1, +1]$. A positive coefficient of $+1$ represents a perfect prediction, 0 no better than the "coin flip" classifier and <0 indicates a worse performance than the "coin flip" classifier. This paper calculates the MCC for all the surveyed methods.

$$MCC = \frac{(TP.TN) - (FP.FN)}{\sqrt{(TP + \mathrm{F}P).(TP + FN).(TN + FP).(TN + FN)}} \tag{1}$$

1.4.5. Practitioner metrics

The surveyed studies all apply methods to the real-world problem of fraud detection. To provide meaningful performance measures in the real-world a set of *practitioner metrics* are used in the payments industry (Ryman-Tubb, 2011). Within payment transactions there are specific *entities* that relate to one another that are a one-to-many relationship linked by a common key: (1) a single transaction at a date/time, (2) a set of transactions for a unique payment card normally sorted in ascending order by the date/time of each transaction so that when a single transaction is alerted the entire card is considered as being alerted, (3) a set of cards that belong to a unique account number, such as multiple payment cards issued to a single business. For a specific entity, a set of business metrics can be calculated and these may be given as a graph (similar to ROC) or as a trade-off table allowing a specific performance to be selected by the business. The metrics typically include the number of entity alerts produced each day as a range plotted against (1) %fraud entity detected, (2) %amount saved following the first alerted transaction, (3) entity FPR shown as the number of incorrect entity alerted, (4) entity TPR shown as the number of correct entity alerted, (5) numeric score from the classifier. These metrics can be tabulated against a range of thresholds θ, allowing the business to select θ by balancing the real-world entity metrics against reviewer resource. Unfortunately, few surveyed studies provide sufficient results to calculate these common practitioner metrics.

1.4.6. Real-world benchmark metric

It is next proposed that to understand if the method can be practically scaled to the real-world, its effectiveness must be determined contemporaneously using industry statistics. Therefore, an important real-world performance measure is proposed as the number of alerts per day, denoted $AlertD$ and is given in Eq. (6). As industry statistics for issuers can be determined (see Appendix A) figures have been calculated for an average "large issuer in 2017" (*Tier-1*) in Table 1. These are used in the benchmark in Section 3 to recalculate indicative performance in terms of $AlertD$ as if the method is deployed in an FMS today. Tier-1 transactions are unbalanced with an average RGF of 5000. The results from the surveyed work use a range of datasets, which have a different RGF. Therefore, when re-calculating the performance, the method may not maintain the same performance so the greater the difference in RGF the less confidence in the results for the re-calculation. It is not known if this is significant and so some caution must be taken when drawing conclusions on the ranking of methods. In those surveyed studies where TPR and FPR is stated or these can be estimated, then from Eqs. (2)–(5), using P and N from Table 1, $AlertD$ in Eq. (6) is calculated. This paper ranks the surveyed methods using $AlertD$.

$$TP' = P_{tier1} \cdot TPR \tag{2}$$

$$TN' = N_{tier1} \cdot (1 - FPR) \tag{3}$$

$$FP' = N_{tier1} \cdot FPR \tag{4}$$

$$FN' = P_{tier1} \cdot (1 - \mathrm{TP}\mathrm{R}) \tag{5}$$

$$AlertD = TP' + FP' \tag{6}$$

1.4.7. Class imbalance

There is a large class imbalance, so that the Ratio of Genuine to Fraud (RGF) transactions, Eq. (7), in real-world transactional datasets is high; there are considerably fewer fraud transactions compared to genuine transactions making the problem of classifying them nontrivial. The FPR has the greatest adverse effect on real-world performance of an FMS, as with high transaction volumes and unbalanced RGF, misclassified transactions will consist of mostly genuine transactions and so any misclassification due to a high FPR will generate disproportionately higher $AlertD$ that need to be manually reviewed by a human. The proportion of $AlertD$ that contain fraud transactions is a key metric, denoted A/f in Eq. (8) and summarised in Table 14. In industry, human reviewers tend to mistrust and can ignore alerts and information from the FMS if it generates too many false alarms. Bar-Hillel (1980) describes this as the human "base-rate fallacy".

$$RGF = P/N \tag{7}$$

$$A/f = AlertD/TP' \tag{8}$$

1.4.8. Concept drift and disruptive industry technologies

The detection of fraud is nonstationary as fraud vectors change over time and thus when a fixed FMS is put in place the effectiveness is

Table 1
Calculated worldwide Tier-1 issuer statistics per day (see Appendix A).

Tier-1 issuer per day		1971	1982	1993	2017
Number genuine transactions	$\#N_{tier1}$	7 k	40 k	560 k	5.7 m
Number fraud transactions	$\#P_{tier1}$	6	300	200	1,150
Fraud write-off	$\$Fraud\,D_{tier1}$	$330	$35 k	$27 k	$400 k
Ratio of Genuine to Fraud	RGF	700	200	100	5000

reduced over time. Fraud vectors are also reflexive due to the criminals responding to the system subsequently to alter their MO. Therefore, it is argued that concept drift within the data available is significant and that FMS approaches that do not take this into account will become less effective and so losses and operational costs will significantly increase. It is important to consider that some innovation can undermine existing products, businesses or entire industries — through a disruptive event (Cortez, 2014). There is a disruptive event in the payment industry that is creating unknown fraud vectors that are changing at a more rapid rate than has been seen since the introduction of payment cards (Choo et al., 2007). This event is due to the reported exponential growth in (1) smartphone, e-commerce and m-commerce, (2) contactless payments, (3) shifts in fraud liability, (4) e-wallet, (5) Near Field Communications (NFC), (6) cyber-crime including large data breaches, (7) commoditised high power and cloud computing, (8) virtual currencies, (9) micro payments (Appendix B). As crime migrates due to these technologies, it will do so more rapidly than in the past due to innovative technology. Traditional forms of payment fraud are giving way to criminals who are highly computer literate and who are living in an age of a high-tech communication with a technology driven lifestyle and with prolific use of social media. More sophisticated and subtle fraud vectors are emerging as criminals have started to use Artificial Intelligence and machine learning for offensive purposes (Dvorsky, 2017). This is likely to have a substantial impact on financial fraud and the compromise of secure systems world-wide. When a civilisation is at a point of crisis it is only then it seems forced to make changes. In the 7th century BC, Pittacus of Mytilene is attributed to the aphorism, "*necessity is the mother of invention*" ("*Ανάγκα και θεοί πείθονται*"). It is argued that a crisis may then influence those in the payments industry, governments and lawmakers to make changes to fund and recognise the significant impact of research that will bring about new prevention and detection methods.

1.4.9. Latency in verification of fraud

There is a latency between the point of a human review or when a customer reports a suspicious transaction and it being determined to be fraudulent. This latency can be over days or even months while the case is investigated. The datasets used therefore contain this latency with respect to the marked classes. In the real-world, this means that the data available to the FMS from which to train a classifier is already dated and needs to be given consideration in fraud detection classifiers.

1.4.10. Real-time data stream

The loss due to fraud is incurred at the moment of the transaction for issuers and merchants. Therefore, to be effective, fraud needs to be detected in real-time. A real-time FMS is illustrated in Fig. 3. It receives a transaction and then makes a decision as part of the authorisation flow and returns this decision to accept/block/decline/refer the transaction as a response message. Real-time functionality is particularly important where a card transaction can be stopped during authorisation based on the output of a fraud decision process. A transaction occurs at a specific time and is part of some sequence and can therefore be considered a stream of data. The temporal and sequential nature of transactions is known to reviewers to contain important information for the detection of fraud.

The above challenges have in part contributed to the slow progress in improved and transparent detection methods making the research area interesting. The remainder of this paper is organised as follows: Section 2 describes survey methodology. Section 3, a survey of methods is presented. Section 4 discusses the survey findings and Section 5 proposes future research directions and Section 6 concludes the paper. Appendix A provides details on industry statistics used for benchmark calculations and Appendix B summaries disruptive technology forecasts.

2. Survey methodology

The core goal of this paper is to identify and provide guidance on how the research community can better transition their research into industry. Thus, this survey will establish that since the earliest work only small real-world improvements have been made, leading to limited industry engagement. It is therefore necessary to provide insight into these earliest works to understand research progression. It is not the intention of this survey to discuss historical aspects of these works but to examine the techniques that are among the top ranked (see Table 14).

A complete survey of key published work, focused in the domain of fraud detection for payment cards has been undertaken. This work extends and consolidates other surveys without using secondary citations, into a consistent single review and provides an industry specific benchmark and uniquely uses real-world metrics. Google "Scholar" and the IEEE "Xplore Digital Library" were mostly used to search a large selection of indexed studies using search terms such as "payment fraud", "fraud detection", "credit card fraud" and "payments". Literature survey studies and general subject descriptive papers and books were a useful source of further references but were excluded from the actual survey. The papers in the survey include research on the application of Artificial Intelligence and machine learning techniques to the problem of detecting fraud in payments. The papers are examined with respect to their novelty, publication year, methods, algorithms, results and implementations.

Yufeng et al. (2004) offers a literature survey of techniques used for general fraud detection from 1991 to 2002 including payment card fraud detection. Phua et al. (2010) provides a survey of data mining methods for general fraud detection methods covering 1994–2004. Sethi and Gera (2014) discuss general methods focused on credit card fraud detection with a summary of common fraud vectors. A short literature review in Ryman-Tubb (2011) discursively summarises the earlier methods. Danenas (2015) provides a useful survey of patents in financial fraud detection over the period 1998 to 2013. A survey on anomaly detection methods that include fraud detection is given in Ahmed et al. (2016). A short survey covering fraud detection techniques over 2005–2015 with an emphasis on machine learning is given in Adewumi and Akinyelu (2016). Abdallah et al. (2016) review a range of fraud detection applications, including payment card fraud, telecommunications, insurance and online auctions. This survey differs from these in its comprehensiveness and its use of a consistent set of evaluation criteria that are informed by the real-world needs of the payment card industry.

2.1. Distribution of surveyed papers

Using the criteria above, there are around 695 key published works dated between 1990 and 2018 that are identified and evaluated from academic journals and conference proceedings; their distribution is given in Fig. 4. The early works were driven by the transformation of electronic computing into a utility enabling artificial intelligence and machine learning and the growth in payment card usage and so fraud. While there is significant growth in academic interest over the

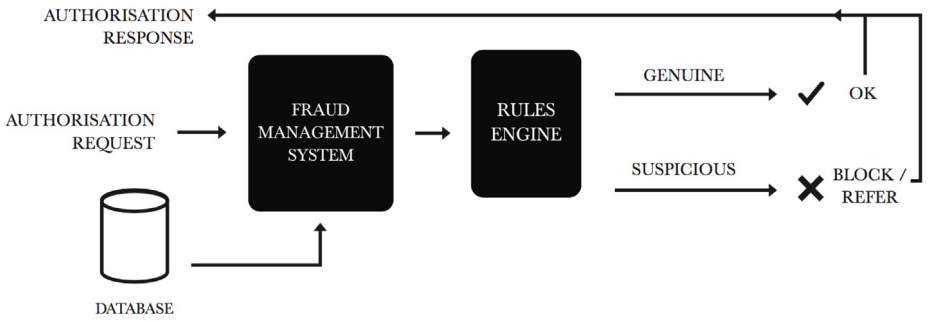

Fig. 3. Real-time FMS.

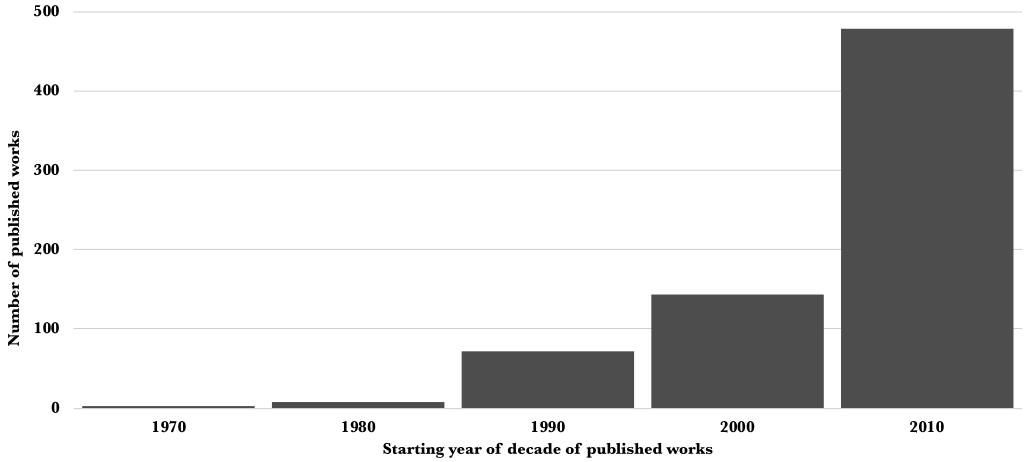

Fig. 4. Distribution of published works (1970–2018).

Table 2
Payment card fraud detection ontology.

Section	Method	Description
3.1	Expert systems/Decision Tree	Based on human-readable symbolic representations of knowledge sometimes called Knowledge Based Systems (KBS). Expert systems are the most established AI technique used in fraud detection. AI includes symbolic approaches: rules, Decision Trees (DT) and Case Based Reasoning (CBR).
3.2	Supervised neural network	Creates a model by inferring a function from training data with inputs and associated (labelled) outputs. This model is used to classify $\{genuine, fraud\}$ classes. Supervised neural networks and their derivatives are used extensively in fraud detection.
3.3	Unsupervised neural networks & clustering	Creates a model by topographically representing input data so that data with similar properties are placed at nearby locations so the input data is therefore meaningfully clustered. Unsupervised neural networks and their derivatives are typically used to detect unusual or anomalies in transactions for fraud detection.
3.4	Bayesian network	Creates a probabilistic model by inferring conditional dependencies from data.
3.5	Evolutionary algorithms	Used as a search method to find an optimised set of functions that can classify fraud using a heuristic algorithm that mimics aspects of biological natural selection. This includes Artificial Immune System (AIS) models that are inspired by aspects of the biological immune system.
3.6	Hidden Markov Model (HMM)	A statistical model of the probability of sequences of events.
3.7	Support Vector Machine (SVM)	Creates a classifier from training data with inputs and associated outputs by creating single separating hyperplanes between two classes.
3.8	Eclectic and hybrid	A range of novel methods where the main classification method is not listed above.

decades, it will be shown that there is only a gradual improvement in effectiveness. From this body of work, only 51 works have published results in a form that can be usefully compared and benchmarked. Those earlier works that are highly ranked (Table 14) using current real-world transaction volumes are reviewed in detail. The body of work forms a proposed ontology given in Table 2 where each has a taxonomy and this is described in each section. It will be seen that each of these detection methodologies has different strengths and weaknesses.

3. Survey of methods

The purpose of the survey is to consistently benchmark and rank payment card fraud detection methods, as if they were implemented

in 2017. A total of 51 methods are ranked. This is to help determine why there has been limited progress in terms of real-world performance. In the following sections an overview of each of the key methods is presented. Results are recalculated and tabulated in a consistent format using $AlertD$ in Eq. (6) and are summarised at the end of this survey in Table 14.

3.1. Expert system/Decesion tree

In this paper, expert system is used to mean any symbolic AI that is based on human-representations of knowledge. Probably the earliest form of modern AI is the expert system with early deployments starting in the 1970s (Feigenbaum, 1977). Expert systems were designed to solve real-world problems that could not be easily specified for conventional software. Human experts are used to create a base of knowledge that is encoded typically using symbolic rules. This knowledge base can then be queried to produce a result using reasoning. In more recent implementations rules can be created using machine learning methods such as inductive learning that extract rules directly from a dataset (Al-Khatib, 2012). Expert systems create a transparent solution that can lead to better human understanding — an important industry requirement.

3.1.1. Rule based

In Shao et al. (1995) it was reported that expert systems are widely adopted throughout the payments industry and this remains true today. Human payment fraud experts create rules that aim to capture their knowledge on fraud vectors. As described in Dazeley (2006) this is an expensive and time-consuming task and requires experts in their field. Small sets of well-written rules are transparent as to their operation and easy to understand. As the external environment changes (1.4.8), the fixed rules need to be updated. For many problems, such as fraud detection, these changes may need to adapt to new payment and criminal methods. A large rule-base is difficult to understand and to maintain. As it grows the impact of newer rules is difficult to determine and more computing power is required to evaluate.

Leonard (1993) proposes an early expert system for the detection of credit card fraud where a Canadian bank provided a dataset consisting of 12,709 transactions. The RGF of 21 in the Canadian Bank dataset is unusually low even for that decade (see Table 1). An unrealistic 496 k $AlertD$ would have been created with the human review team having to review 611 alerts (A/f) to find a single fraud.

A more sophisticated expert system is proposed in Vatsa et al. (2009) and builds on the work in Liu and Li (2002) to use game theory combined with an expert system shown in Fig. 5.

In game theory, there are two players (1) the fraudster and (2) the FMS. Both are opponents that wish to win by maximising their gain; there are two parties with conflicting goals. Thus, the fraudster attempts various fraud vectors and the FMS must minimise any loss by detecting them as early as possible. It has been observed that criminals continue to use a stolen CHD until it is blocked. A typical fraud process is to try low value transactions at lower risk location in the belief that this is unlikely to trigger the FMS to block the stolen CHD. They have a belief about how the FMS operates as an "opponent". The criminal.v. FMS can be considered as playing according to a Nash Equilibrium (Rosenthal, 1973). In this case, to improve the performance of the FMS it needs to adapt by predicting the next "move" of the criminal. It can only do this if it takes into account feedback from the real-world so that repeated "moves" can be made where the criminal modifies their behaviour to avoid being blocked. As the FMS gathers information on each transaction made, its belief is adjusted using machine learning so as to improve the likelihood of "winning". A synthetic dataset was used and the results are reported as the number of fraudulent transactions correctly alerted that improve over nine rounds where P_f improves from 45% to 70%. Results are re-plotted in Fig. 6 and demonstrate how the FMS progresses by updating its strategy per move so that more fraudulent transactions are correctly detected. When the results from

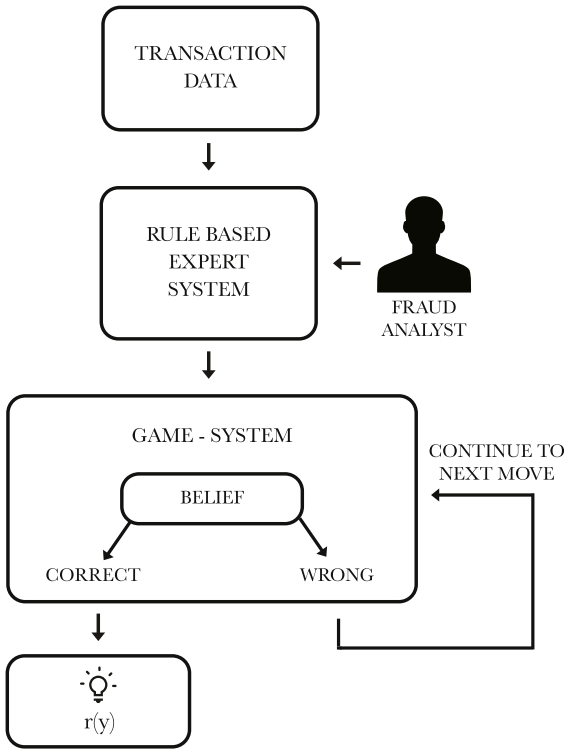

Fig. 5. Game theoretic fraud detector diagram (Vatsa et al., 2009).

the work are recalculated this gives an impracticable $AlertD$ 1.7 m due to poor FPR of 30%. It is argued that this is likely to be the result of the initial rules used, as the earlier Leonard (1993) had an FPR of 8.65% so that adding the game theoretic method to such an expert system should improve overall performance over time but this was not presented here.

Ranking the lowest in this benchmark, HaratiNik et al. (2012) defines a method of using fuzzy rules. Terms are defined, e.g. (1) high, (2) average, (3) low, as the fields within a fuzzy rule. These terms are allocated to values using a membership function that is typically Gaussian. The output of the rule, typically in the range [0,1] is calculated by combining these terms using their membership. Using synthetic data, the results indicate a TPR of 91.6% but a poor FPR 77.5% and using Tier-1 volumes the $AlertD$ is 4.4 m. This is worse than a random coin-flip.

Ranking the highest in this benchmark, Correia et al. (2015) describe how a set of 14 rules were created through a manual trial-and-error method looking for specific patterns that indicate known fraud vectors. An open source software tool PROTON is used to implement the proposed method (IBM, 2015). This used an Event Processing Network (EPN) that had Event Processing Agents (EPA) for each written rule. An uncertainty measure was used in the transactions as the value of the fields may be imprecise and so each field was accompanied by a derived PDF so that a certainty output was calculated by the EPN for each new transaction. If an EPA is triggered it will generate a certainty score based on the sum of PDF in the fields in the transactions. A real-world dataset was used that had 5.6 bn transactions covering 2009 to 2011 with a RGF of 2000 and 27 fields. For evaluation 7 of the 27 fields were manually selected. Setting θ to 70%, the results reported 80% of the fraudulent transactions were detected with a FPR of just 0.02% (this is an excellent result but it is not clearly stated if this is the FPR). When recalculated, it generates just 2060 $AlertD$, missing 20% of the fraudulent transactions. The evaluation methodology used with such a large dataset is not stated. The creation of the EPAs is manual so that the thresholds, events and size of time windows that form the parameters are determined through trial-and-error experiments. The results are impressive but it is suggested

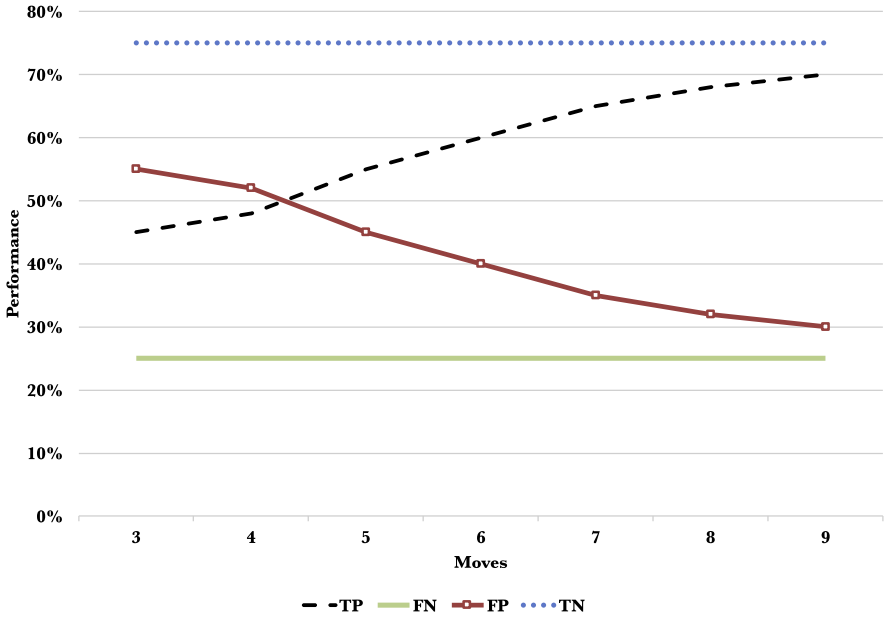

Fig. 6. Graph of game theoretic fraud detector performance after each "move" (Vatsa et al., 2009).

that the trial-and-error approach has likely overfitted the dataset and given differing real-world transactional and cardholder data may not perform in a similar way. Therefore, it is possible that there will be a poor generalisability of these results when used with different datasets.

Table 3 summarises the surveyed expert systems work and re-calculated results. It can be seen that Correia et al. (2015) produces the lowest *AlertD* and is ranked the highest performance out of all studies included in the benchmark. It may be that the fraud vectors in this dataset were sufficiently unique to be mostly linearly separable from genuine transactions using the simple rules. This is counterintuitive, as it is expected that the fraudsters will attempt fraud that looks similar to that of a genuine transaction and so there is likely to be some overlap between the two classes based solely on the transactional dataset. The other methods in the table are all seen to be impractical.

3.1.2. Decision Tree (DT) / random forest

A DT is a graphical representation of a tree used to make a decision and is described in detail in Morgan and Sonquist (1963). A DT is created as a classifier using inductive learning by creating a tree structure which attempts to separate the classes into mutually exclusive subgroups at each node in the tree (Quinlan, 1986). A benefit of this method is that the generated DT can be viewed as rules in a similar form to that in expert systems. These are English-like and are easily understood by traversing a path from the root to a classification leaf. There are many well-known algorithms that can induce a DT from training vectors, e.g. Quinlan (2007) and Cohen (1995). Previous work has compared DT learning to neural network methods (surveyed in 3.2) and indicate that when performance and generalisation are important then a neural network outperforms in most cases (Fisher and McKusick, 1989). This would suggest that the DT is not a likely candidate for real-world payment card. However, as will be seen some of the surveyed works demonstrate otherwise. A newer DT approach has emerged that addresses the problems of overfitting in the earlier DT algorithms by combining different DTs using subsets of the training dataset and random selection — known as a random forest. Early work (Breiman, 1996) used "bagging" where each DT is generated from a random selection from examples in the training dataset. In newer methods, features are randomly selected from a random subset of all possible features for the node split and a threshold selected according to an information gain criteria (Geurts et al., 2006). Many different DTs are created which may be individually weak predictors but together cover more of the search space. This ensemble of DTs is then averaged for performance.

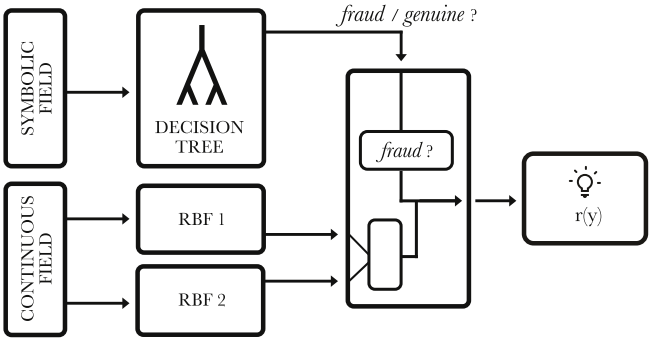

Fig. 7. Diagram of the proposed RBF and Decision Tree approach (Brause et al., 1999).

Stolfo et al. (1997) compares four algorithms for generating a decision tree: ID3, CART, RIPPERk and BAYES/CN2. These were tested with 0.5 m transactions with 30 fields provided by a financial services firm; no details of the dataset are given due to confidentiality. Taking the best results of 80% (TPR) and 13% (FPR) then *AlertD* can be calculated as 745 k which is an impractical volume to review.

Chan et al. (1999) builds on this work to add a financial cost to the classifications during the creation of the DT classifier. For each transaction, there is an associated value of that transaction and therefore the associated cost of either correct detection or misclassification. These cost values are used in an algorithm called *Adacost* originally proposed in Fan et al. (1999). A metaclassifier method is used where multiple DTs are created using a range of algorithms and are then combined to produce the highest *accuracy*. The same 0.5 m transactions were used. The results are given in terms of the cost savings with no other metrics and so the method cannot be included in the benchmark.

Despite being an early work, Brause et al. (1999) is ranked at 8 in this benchmark. Here, rules are generated by a DT using a real-world dataset of 548,708 with an RGF of 93 (realistic for the period, see Table 1), supplied by an unnamed bank. Two separate models are created as illustrated in Fig. 7. The RBF model is described in 3.2.2.

Table 3
Summary of expert system methods surveyed for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%MISS
Correia et al. (2015),	1	0.596	2,060	2	0.020	80.00	20.00
Leonard (1993)	32	0.031	495,620	611	8.650	70.76	29.24
Vatsa et al. (2009)	47	0.013	1,716,904	1,499	30.000	70.00	25.00
HaratiNik et al. (2012)	51	−0.002	4,434,313	3,871	77.500	91.60	8.40

The fields in the dataset are separated so that, (1) a DT is induced from for the symbolic fields, (2) an RBF is trained on the numeric fields. For the fraud classification output the decision from the RBF overrides the DT in the case where fraud is indicated. The 747 rules generate a reported TPR 90.91% and FPR 0.27. When recalculated for Tier-1, $AlertD$ is 16 k which ranks the work 5 using industry statistics. The work notes the high computational complexity of the rule-induction method. This is a surprising result as typically the generalisation of the rules is poor as larger rulesets reduce efficacy and confidence; with 747 rules and 5850 fraud examples. It is suggested that this approach may not generalise well as almost each example of a pattern of fraud is explicitly represented in an individual rule. For transactions where the fraud patterns are more complex and contain overlapping or contradictions, then this approach is likely to perform poorly. This highlights the difficultly of consistently comparing research methods that use different datasets.

In Sahin and Duman (2011b) the performance of: (1) CART, (2) C5.0 and (3) CHAIR, DT algorithms are assessed using a real-world dataset. This database had a sparse RGF of 22,500 with just 978 fraudulent transactions out of 22 m records. The only performance measure is given as $accuracy$ of between 86.79%–94.69%. Given the highly unbalanced dataset this measure cannot be used to determine the efficacy of the classifier. This work is continued Sahin et al. (2013) and is similar to Chan et al. (1999) that uses the monetary cost of a misclassification within the splitting criteria. The same dataset is used but results are provided using a proposed measure called "Saved Loss Rate" and so again cannot be compared to other work. The work states that the cost-based methods outperform the previous method but do not provide evidence that this is so.

Minegishi and Niimi (2011) use an on-line DT where the DT is generated as new marked transactions arrive at the FMS ("stream" based) as an alternative to requiring all the data to be present in a single $TRAIN$ dataset. To impute the DT the Very Fast DT (VFDT) algorithm from Domingos and Hulten (2000) was selected. Here, the Hoeffding bound to the information gain is used as the splitting criteria. The dataset comprised 124 fields with 47,091 credit card transactions which is re-sampled to an RGF of 9. 94.93% of the fraud transactions were correctly identified but with a poor FPR of 41.53% using 106 rules generated. Over 2 m $AlertsD$ are generated, indicating the DT overfitting of data that contains noise.

Detecting anomalous transactions is proposed in Kokkinaki (1997) and uses a modified DT that is used to essentially store a list of habits at each node of typical cardholder behaviour. If a new transaction for a cardholder does not match one of the habits in the DT then the transaction is atypical and marked as suspicious. No experiment of the method was undertaken and no method for updating cardholder behaviour is provided. It is argued in the work that such a system would need to be able to store and rapidly recall and evaluate a DT for each individual cardholder; for a Tier-1 issuer this is likely to be impractical, given the number of cardholders and the number of transactions (in Table 1). This approach is common to many anomaly detection methods. The idea is novel and could usefully be explored further.

A further anomaly method using a modified DT is proposed in Jianyun et al. (2006). This work uses the (Han et al., 2000) DT algorithm to extract the associations between the fields, importantly over a certain time period of transactions. Each cardholders profile over a period generates a new DT. The DT is then used on a new transaction to indicate a level of match/anomaly. The level indicates how close the new transaction is to the learnt cardholders' normal behaviour. Synthetic data is used to evaluate the method and the results are given in terms of cost savings and so cannot be compared to other work. The reported results indicate (except for one dataset) that there is no significant difference between this method and that of the standard C4.5 DT algorithm.

Fadaei Noghani and Moattar (2017) use a feature selection and then random forest DT approach. Features are selected from the fields using three described measures (1) Chi-Squared, (2) "ReliefF" that determines volubility and (3) information gain. The highest-ranking features for each method are used to generate a subset dataset. This subset is then classified using a C4.5 DT and the accuracy determined. If a feature decreases the accuracy of this classifier then the next highest-ranking field is added and the process repeated. The process is designed to create a set of features while removing those that are less important. This method may remove important information where there is a weak but important correlation between the fields. Once a dataset has been created it is used to create a random forest. A public dataset of 29,104 transactions with RGF 26 was used in the experiment. The results are only reported on P_f and F-score and so FPR cannot be determined. A F-score of 0.9996 is reported with 27 trees in the forest which ranks the work 1 if this measure were to be used as a benchmark (calculated in Table 14). However, as discussed this is not a useful comparison in the real-world, as TN and therefore FPR is not given. The focus on feature selection is useful — although in this case it is unclear if information is being lost as a known limitation of the splitting approach.

Dal Pozzolo et al. (2017) present an important paper that reflects many of the key challenges identified in 1.4 and is ranked at 7 in the industry benchmark and is therefore reviewed in detail. In particular it is the seminal work that investigates the impact of concept drift in this domain in the real-world (1.4.8). It carefully considers many of the challenges discussed in 1.4; class imbalance, the latency in verification of fraud by reviewers/customers, the measuring of real-world performance to balance misclassification against precision to generate manageable $AlertD$ and is tested on a large real-world dataset. The work proposes a real-world measure based on normalised card alerts per day NCP_k, that is the proportion of cards correctly alerted out of all cards reviewed, as $1/f$ in Eq. (8). It is noted that reviewer resource available is limited and so this measure is a key real-world metric. The work proposes the use of two classifiers: (1) is trained on a marked transactional dataset following fraud having been reported and investigated that occurs some considerable time after the event, denoted "delayed-samples". It is argued that the majority of transactions that are authorised each day are not labelled for a considerable period and so performance will suffer where concept drift is prevalent. It notes that this classifier is the most common throughout literature and that in the real-world it is only re-trained on an occasional batch basis. (2) is trained daily on a dataset that is the result of investigations following the alerts raised that day, denoted "feedbacks". Feedbacks have Sample Selection Bias (SSB) as they are not representative of the underlying distribution. Typically approaches to correct for this use a weighting and this may reduce the impact of such feedbacks in a single classifier. The work distinguishes between (1) having a large class imbalance skewed towards genuine transactions and (2) where the balance depends upon the detection performance of the FMS and will be skewed towards fraud transactions. The approach then aggregates the output of (1) and (2) by a variable that weights their posterior probability contribution. Various parameters are proposed that vary the length of time in days from which

the $TRAIN$ datasets are created. The $TRAIN$ dataset for (1) is created using random undersampling of the genuine class while retaining all the fraud class. A random forest of 100 DTs is used as the underlying classifier such that each tree is trained on a randomly selected set of genuine transactions but the same fraud examples. Two different approaches are tested. A real-world dataset of 75 m transactions over 3-years was provided by a bank, with 51 fields, split into two datasets with 2013 RGF 415 and 2014–2015 at 525. Measures that include the proposed real-world metric, precision and AUC are used and the various configurations ranked and compared. Each experiment uses 10-crossfold validation. The results show that the highest performing configuration is that which combines both (1) and (2) and it is noted that (2) has a significant impact on precision, suggesting the stream of transactions is nonstationary. The results do not provide TPR and FPR but the proposed NCP_k. To provide a benchmark in this survey, a confusion matrix has been estimated using their Table 4, based on the 2013 dataset and selecting the highest-ranking classifier. From Table 1, this dataset had 21,830,330 transactions over a 136-day period, giving 160,517 transactions a day with fraud at 0.19%. There are $160,517 \times 0.19\% = 305$ fraud transactions each day. If an assumption is made that there is an average of 2 transactions /day, then there are 80,259 cards/day and 152 cards/day contain fraud (P). The results in the work are given where $AlertD$ is set at 300 cards a day for review. NCP_k is given as 0.48 and so the correctly alerted number of cards a day (TP) can be calculated as $300 \times 0.48 = 144$ and so FP is 156. FN can then be calculated as $P - TP = 8$. The total number of cards with only genuine transactions per day (N) is given by $80,259 - P = 80,106$. TN is then given by $N - FP = 79,950$. Based on $AlertD$ at 300, MCC is calculated from the confusion matrix as 0.672, A/f as 2, with just 5.6% of fraud cards missed a day. If this is recalculated for Tier-1 (noting that the above calculated results are for cards and not transactions but assuming the proportions remain similar) then $AlertD$ is 12 k placing the work in the top quartile in the benchmark. These figures are wide estimates and reviewing the performance in context it is suggested that this approach may be ranked higher in this benchmark. The view of transactions as streams and the move away from the focus on the individual classifier by simply selecting an established type here, indicates that research is moving away from the earlier emphasis to that of a more encompassing approach. The results further compare a single classifier (2) and try to compensate for SSB introduced but concludes that this is ineffective using importance weighting — likely to be due to the interaction between the dataset and the feedback of the reviewers.

The surveyed DT methods with benchmark figures are given in Table 4. Two methods are highly ranked despite the DT method typically being sensitive to noise in the data. A benefit of the approach is good explainability but the surveyed methods create a large number of rules each with many antecedents which makes their interpretation difficult. It might be argued that the dataset may have included relatively few differing fraud vectors and so the DT was able to reasonability generalise. However, with the discussed rapid changes in the payments industry (1.4.8) it is not known if this method would continue to perform. Therefore, caution must be taken when considering the approach for future improved implementations.

3.1.3. Case-Based Reasoning (CBR)

A CBR system determines a weighting for each field in a fraud "case" typically using a stochastic hill-climbing algorithm to find the best combinations of field weights. The CBR determines a degree of match between the new transaction and the previous cases stored. If a similar case is found, then an alert is generated. This alert is then analysed by the review team and if determined to be correct then this new case is added to the stored cases — Fig. 8. However, a large number of differing examples of fraud are required for accurate operation, since the system poorly generalises. The number of patterns to distinguish fraud from a genuine transaction is large and so the intrinsic dimensionality of the model grows so that the number of fraud examples will grow exponentially. For a typical Tier-1 issuer, if each fraud transaction were detected they would add to the case-base that would become impractical.

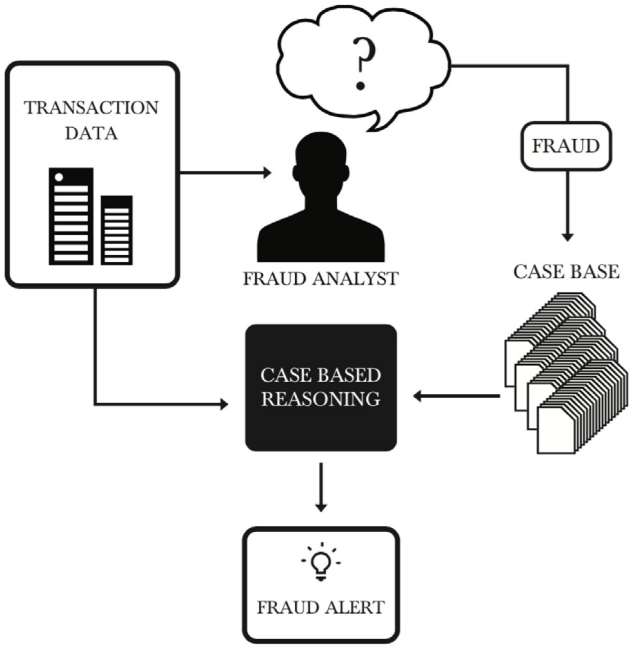

Fig. 8. Diagram of Case-Based Reasoning (CBR) FMS (Wheeler and Aitken, 2000).

In Wheeler and Aitken (2000) a CBR method is applied to fraud detection in applications for credit loans. The real-world dataset consisted of 128 fields, 4000 records with an RGF of 23. The $TEST$ dataset consisted 680 records with an RGF of 6. The work updates the standard CBR method to use four different algorithms to search for matches each of which reports a confidence.

The work concluded that the multi-algorithm method is capable of "high accuracy" but the published results do not support this claim. Recalculated results give $AlertsD$ 1.3 m which is considerably worse than the human written rules described in the earlier Leonard (1993). The benchmark results for this CBR method is given in Table 5.

3.2. Supervised neural networks

Supervised neural networks are constructed from a number of simple neurons interconnected by connections (synapses) each of which has an associated weight to form a network — discussed in Bishop (1995).

Tafti (1990) is the earliest notable work on the use of a neural network explicitly for the detection of payment card fraud and is included here for completeness. A real-world dataset from Chase Manhattan Bank of 1000 records were sampled from 100,000 records and used to train a neural network. This was undertaken using an off-the-shelf educational software tool for experimenting with a range of neural network algorithms, called "NeuralWorks Professional". No details of the results or the neural network architecture chosen are given. The work likely informed the research community on the importance of this domain and the research challenges.

3.2.1. Probabilistic-Restricted Coulomb Energy (P-RCE) neural network

Ghosh and Reilly (1994) is the seminal work using machine learning in payment card fraud detection and is ranked at 2 in this benchmark — despite the age of this original research. This was undertaken at an early point in the application of neural networks and uses a local function Probabilistic-Restricted Coulomb Energy (P-RCE) that is similar to RBF described in 3.2.2. In this work the dataset only had a few different

Table 4
Comparison of Decision Tree methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Dal Pozzolo et al. (2017)	7	0.289	12,222	11	0.195	94.43	5.57
Brause et al. (1999)	8	0.239	16,486	14	0.270	90.91	0.63
Stolfo et al. (1997)	38	0.028	744,561	650	13.000	80.00	20.00
Minegishi and Niimi (2011)	48	0.015	2,376,971	2,186	41.534	94.93	5.07

Table 5
CBR method results for fraud classification.

Work	Rank ↓	MCC ↓	AlertD	A/F	%FPR	%TPR	%Miss
Wheeler and Aitken (2000)	46	0.010	1,259,048	1,099	22.000	50.00	50.00

fraud vectors — this is likely to be as criminals at that time in the 1990s continued to use the same fraud methods. 20 input fields were created from the 50 fields in the dataset which was provided by Mellon Bank in the USA using a manual pre-processing step that is not described. The dataset consisted 450 k credit card transactions with RGF 30. Two modelling datasets were created: (1) train data for a specified period of transactions, (2) test data for a period following the train data. This is a reasonable approach to test the model generalisation. As discussed, industry typically match the number of alerts generated to the capacity of their team of reviewers. In this work, $AlertD$ was set to 50 by selecting an appropriate threshold θ. This is compared to the 750 a day from the existing rule-based system at the bank that only detected one fraud correctly per week. The trained model had a TPR 60% and FPR 0.09% that when recalculated, gives $AlertD$ of 6 k — only a few of the following 24 years of research match these results. The P-RCE results are interesting; as if the P_f is accepted at this level then this method has an excellent FPR. It is this algorithm that is used by at least one vendor in their FMS products and remains in use today (ACI-Worldwide, 2017).

3.2.2. Radial Basis Function (RBF) neural network

The only method to propose an RBF is Hanagandi et al. (1996); it is included here only for completeness. A pre-processing approach was used on 36 input fields and although not stated this appears to be Principal Component Analysis (PCA) which generated five components. These were used as input fields to the model which was then trained. No results are presented but the work claims, "*The result obtained by this technique was better than ANN [Artificial Neural Network] with back-propagation... however it was not the best of all the modelling methods applied to the problem*".

3.2.3. Multi-Layer Perceptron (MLP) neural network/deep learning

Rumelhart et al. (1986) proposed a back-propagation algorithm as a method of training an MLP that was to become widely adopted for a three-layer structure. Since then an entire research field has developed for all aspects of neural network architectures and training and it is not necessary to detail these.

The earliest MLP work Aleskerov et al. (1997) uses a synthetic dataset with 7 input fields as: (1) 323 records for train and (2) 112 for test, with an unrealistic RGF of 1. Results indicate a TPR 85%and FPR 13.48%. However, given the very small and lack of a real-world dataset, the general performance of the approach could not be determined.

Dorronsoro et al. (1997) used a variant on the back-propagation algorithm that minimised the ratio of the determinants of in-class and outside-class variances with respect to linear projections of the class target (Fisher and McKusick, 1989). The card scheme Visa provided a real-world dataset but neither the number of records or fields is stated. If a threshold is chosen to optimise TPR, then this results in a TPR of 73% with FPR 14%. Calculating this for the benchmark produces an unmanageable $AlertD$, ranking the work 40.

Richardson (1997) is again early work and yet is placed 6 in the benchmark — although caution is required when interpreting the results. 61 input fields were used in an MLP model that were derived from the original dataset of 5 m records (the RGF was not stated). Derived fields included information on a prior period transaction, for example a moving average calculated over a specific period for the same cardholder. A TPR of 61.41% and a low FPR of 0.13% was reported for a specific threshold. When recalculated for the benchmark, this would generate 8 k $AlertD$ while missing almost 40% of the known fraud. However, the work notes that the threshold value can be changed so that the missed fraud is reduced but that this will increase $AlertD$ which is a commercial decision. The top quartile ranking for such a basic approach is unexpected given the considerable progress made in the field of machine learning to improve neural network and other classifiers over the years to date. This may in part indicate the difficulty is comparing methods which are not tested using the same or similar datasets. However, a large, real-world dataset was used and so there is no reason to suspect the result not to be indicative that sometimes a straightforward approach yields good results.

A more recent work Tsung-Nan (2007) is motivated to both reduce the dimensionality of the MLP as a known limitation of neural networks in general and use sequence of transactions linked to a cardholder. The time-based sequence of transactions was converted into a single dimension using grey incidence analysis and Depster–Shafer algorithm to fuse the values (Dempster, 2008). No results were given but the work is included here as a novel pre-processing method that could perhaps have further future consideration.

In Guo and Li (2008) a synthetic dataset is pre-processed so that a confidence value is calculated for each field. This value is determined applying a PDF on continuous fields and a simple probability based on the total frequency of a discrete field. This pre-processed data is used to train a standard MLP using a (slow and superseded) back-propagation learning algorithm. The best results are given as FPR 8% and with 95% TPR. When these results are recalculated they indicate level of 459 k $AlertD$ with 5% of the fraudulent transactions missed. The performance remains lower than the much earlier Ghosh and Reilly (1994) in 3.2.1, although it is unclear if this pre-processing method would provide improved results on real-world data. The use of a confidence measure is an important area of future discussion.

In Ise et al. (2009) transactional data is treated as a stream of data. The work automatically constructed new derived fields from 53 original input fields from real-world dataset that had over 1 m transactions a day using time-oriented information contraction methods. The dataset was marked with a RGF of 263. The best features for classification performance are chosen from all the generated features by a novel stepwise procedure. Fraud experts manually created additional new derived fields. An MLP was then trained and evaluated. The results are presented as graphs and show that in nine cases out of ten, the existing method was the same or better and so it was concluded that the selection of features had reduced generalisation but no metrics are given.

In Sahin and Duman (2011a) 13 classification methods are compared that include MLP neural network and logistic regression. A real-world dataset of 22 m records was used with a RGF of 22,495. The fields were manually pre-processed by grouping together symbols that resulted in 20 input fields. The work used stratified sampling to under sample the $genuine$ records rather than the more often used oversampling of $fraud$ records. The results demonstrated that the neural network classifiers

outperform those of the linear regression but notes that the performance of all the models decreases the larger class imbalance. The work used a commercial software package called "SPSS Clementine" to generate the results. The reported results put the basic neural network classifier at 9 in the benchmark.

Lee (2013) propose a cardholder behavioural modelling method using a complex autoregressive network. This model learns time-based transactions so that the output classification depends linearly on its own previous values and on a stochastic term. The model is tested on a small dataset of 200 transactions obtained from a public source but no information is given on the number of frauds. The *accuracy* is given as 80%, which infers that $TP + TN = 160$. As the RGF is not known statistics cannot be calculated without making an assumption. If it is assumed here that the RGF is 5 then a TPR of 44% and FPR of 12% is calculated. Despite the complexity of the proposed solution these results are poor and would result in over 698 k $AlertD$. This may be a result of the small and possibly poor-quality dataset used but it is not known.

Mishra and Dash (2014) propose a method that projects low dimensional space to high dimensional space using Chebyshev orthogonal functions, called Chebyshev Functional Link Neural Network (CFLANN). Two small public datasets were used to assess the method. These datasets are for credit scoring on loans and not fraud detection as stated in the work. The results are compared to an MLP with an *accuracy* of 86% (89%). No other measures are given and so it is impossible to understand the overall performance of this method except it appears to be worse than the reported MLP.

Mahmoudi and Duman (2015) propose a method that uses the Fisher discriminant function in a similar method to the Minerva algorithm (Dorronsoro et al., 1997). This method notes that cost of FP is higher than FN due to the unbalanced dataset, so a modified Fisher discriminant function is proposed which makes the standard function more sensitive to FP. This modification introduces a weighted average into the objective when training an MLP. Unusually this weight is calculated by taking the available amount of credit available to the cardholder at the time of a transaction over the average credit available to all cardholders. A retail bank in Turkey provided a small real-world dataset of 8448 genuine transactions (RGF of 9). A number of experiments are performed reporting a FPR of 8.32% and fraud detection performance of just 25%. These results are recalculated, to give 476 k $AlertD$ with 75% of the fraudulent transactions missed. By weighting the objective function, the FPR is not sufficiently reduced and is at the expense of correctly detecting fraud.

Zakaryazad and Duman (2016) propose a Profit based Neural Network (PNN) as modified MLP that has a multiplier applied to the error function during training based on a measure of cost. Here, cost is a measure of importance of the classification/misclassification as previously discussed. A range of variants on the modified error function for the neural network are described and experiments undertaken to compare these along with a standard MLP, DT and Bayes classifier. Two real-world datasets were used from a Turkish bank, (1) with 9388 transactions with RGF 9 and 102 fields, and (2) with 5960 transactions with RGF 5 and 46 attributes. The first dataset has a high dimensionality and its size is likely to generate a poor model. No cross-validation was used, although each of the experiments was run ten times with different random weights and the same number of training epochs selected which is also likely to lower the performance of the models — as training is typically stopped when some measure of error is reached. For (1) the PNN that used the log to calculate each example's profit outperformed all other methods with TPR of 65% and FPR 1.989%, that would generate $AlertD$ 115 k. For (2) the basic PNN that just uses a simple multiplier was ranked second in their benchmark with TPR of 54% and FPR 11.2%, that would generate $AlertD$ 644 k. The authors calculate the cost savings made by each method and it is noted that when using this measure, the proposed PNN methods outperform the standard approaches. The selection of the cost values and the thresholds has a significant impact on the models. It can be appreciated that the results vary depending upon the dataset used, here a small dataset with very low RGF. As discussed this makes determining the performance of research methods difficult to quantify to those in industry and therefore their impact cannot be determined.

Charleonnan (2016) propose a method that uses three neural classifiers, where each are considered weak classifiers, (1) MLP, (2) RBF and (3) Bayes. The approach is motivated by the highly unbalanced nature of the datasets in fraud. An initial distribution of classes is initialised randomly so that the genuine class is undersampled and the fraud class is oversampled. This re-balanced dataset is then used to train an MLP. Once trained, the output of the MLP is then used to update the distribution of classes based on misclassification errors. A new training dataset is created and the process is repeated until some error measure is reached. The entire process is then repeated for the RBF and Bayes classifiers. Once each weak classifier has been trained the output of each is combined by taking the majority vote for the recognised class. A small public dataset of cardholders from Taiwan bank was used to evaluate the results. There were 25 k records with an unrealistic RGF of 3.5 and 23 fields. An assumption is made that those cardholders who did not pay their balance were fraudulent. The work reports that the proposed method outperforms other methods that are compared in terms of *accuracy*. A graph is provided where TPR is approximately 51% (which therefore misses almost half the fraud cases) and FPR can be calculated as 19%. A low FPR is needed in the real-world so as to reduce false alerts by the majority genuine transactions. In this case $AlertD$ is calculated to be over 1 m placing it in the bottom quartile of the benchmark. While a well-motivated approach, it does not consider how the re-balancing of the dataset impacts the results when used with large datasets.

Addressing the industry driven need for transparent systems and ranked at 3 in the industry benchmark, Ryman-Tubb and d' Avila Garcez (2010), Ryman-Tubb and Krause (2011) and Ryman-Tubb (2016) propose the Sparse Oracle-based Adaptive Rule (SOAR) Extraction method that extracts knowledge in the form of association rules from a neural network trained on a real-world, large-scale transactional dataset to detect payment card fraud. It is noted that the purpose of the work is not to create an improved fraud detector but to show that fraud rules can be extracted from a black box classifier so as to be understood by reviewers. The work used a real-world dataset supplied by a large issuer of 171 m records covering 122 days with a RGF of 165,515. A 1% random sample of the genuine class was taken and all 1033 fraud examples were selected. This was sampled and pre-processed to create the datasets. In the most recent work, the previous MLP fraud detector is replaced using an advanced deep learning MLP with regularisation approaches to reduce overfitting. SOAR was used to extract rules by filtering the output of the neural network so that the rules were only extracted based on high confidence classifications from the neural network. 11 high confidence rules were extracted. The real-world dataset was used and results based on (1) transactions with a TPR of 75.56% and FPR of 0.09% and (2) cardholders with a TPR of 91.78% and a FPR of 0.17%. When the transaction results are recalculated, it generates 5927 $AlertD$ while missing 24% of the fraudulent transactions. The extracted rules have been able to distinguish most of the genuine transactions. (See Fig. 9.)

3.2.4. Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) has four processing layers and was originally created for image recognition, described in LeCun et al. (1998) and Huang et al. (2016).

Fu et al. (2016) propose a method of two key components, (1) deriving meaningful features prior to training and (2) the use of a convolutional neural network (CNN) to detect transactional fraud. The work highlights the importance of pre-processing data so as to capture cardholder behaviour that occurs over time. Standard statistical aggregation is used for fields such as value, average value, number of transactions, etc., over selected time periods and new derived fields are

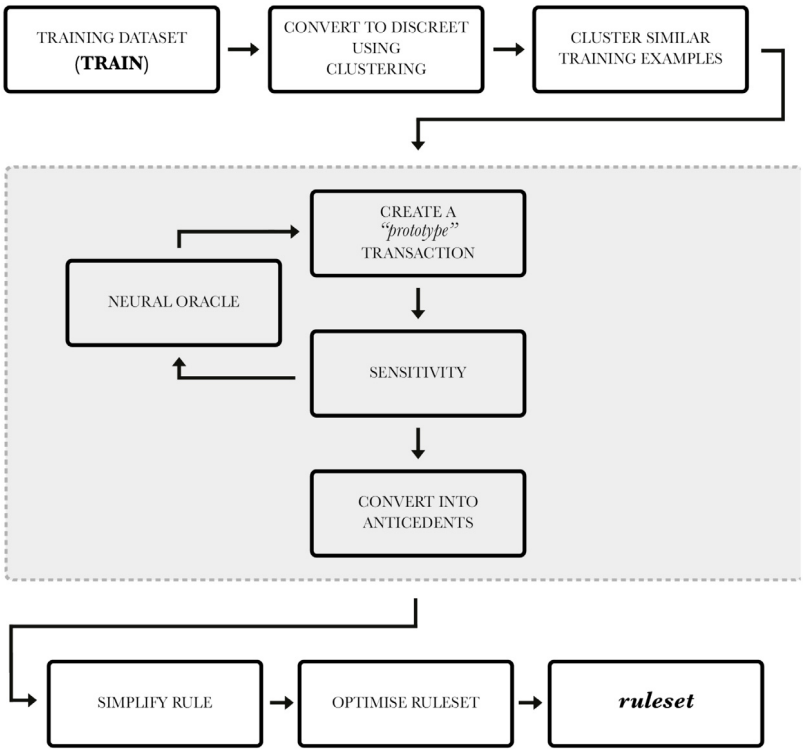

Fig. 9. Diagram of SOAR (Ryman-Tubb and d' Avila Garcez, 2010).

constructed. A novel derived field is proposed called trading entropy. Trading entropy is calculated for each new transaction based on the type of merchants (i.e. electrical, food, etc.) and proportion of total value spent at each of those merchants by a cardholder prior to the current transaction. Using information theory (Shannon, 1948) a measure of entropy is calculated using this value such that a transaction that differs significantly from previously has greater "information". The higher the value the more unusual the transaction for a particular merchant category and the work notes that this correlates to a higher probability of fraud. This encapsulates the sequential nature of transactions. A real-world dataset provided by a Chinese bank of 260 m transactions with a sparse RGF 65,000. It is suggested that this sparse rate is a reflection of how credit cards are used within China. Due to this low rate, the work uses a method to create additional synthetic fraudulent transactions that can then be included for training. KNN (see 3.3.2) was used to cluster all the fraud examples and then generated a new fraud by choosing two from within a cluster. The genuine class was randomly under sampled. Experiments were undertaken for various class balances. The records were then transformed into a matrix suitable for the CNN. This matrix consisted of the fields/trading entropy as rows and their statistical aggregation over differing time periods for the columns. Once this pre-processing has been completed it was split into $TRAIN$ being 11-months of data and $TEST$ being the following 1-month. No details on the size of these datasets or number of fields is given. Segmenting the dataset by transaction date/time is a good approach as it reflects how an FMS would be used in the real-world. The CNN was trained using an unspecified method. The results were compared with three other methods, (1) MLP, (2) SVM, (3) Random Forest DT but no details are given on these. The results are only reported in terms of the $F\text{-}score$ and are given by way of a graph and so can only be estimated. On average the proposed CNN method outperforms the other methods, with the best at 0.33, compared (1) 0.29, (2) 0.26, (3) 0.30. As discussed, the $F\text{-}score$ is not a useful measure in this domain. However, it has been calculated for surveyed methods and is given in Table 14 which indicates that these results are in the top quartile. As $AlertD$ cannot be calculated, the method is not included in the benchmark table but it appears to be promising. In particular the generation of meaningful derived fields and the use of sequence/time is an important aspect of fraud detection that is a growing research interest.

Table 6 is a summary of supervised neural network methods sorted by rank. The supervised neural network methods have generally favourable performance placing five methods in the top quartile, when ranked against other methods.

3.3. Unsupervised neural networks and clustering

In general, unsupervised neural networks learn the relationship between the input fields so as to form clusters where each cluster groups together similar inputs (Hartigan, 1975).

3.3.1. Self-Organising Map (SOM)

The Self-Organising Map (SOM) was created as a biological representation of sensory neurons creating maps and is described in detail in Kohonen (1984).

In Zaslavsky and Strizhak (2006) and Quah and Sriganesh (2007) cluster the fields from a transactional dataset using a SOM. First, new fields are derived so as to capture temporal relationships, e.g., total volume of transactions and average transaction amount on a specific card (1) for the day, (2) over five days, etc. The SOM is then trained using these derived fields to a point where it is considered to have converged. A tiny synthetic dataset was used of 100 records with 10 different types of fraud. Each pre-processed transaction is then processed by the SOM that outputs the Best matching Unit (BMU). This BMU is then recorded against a specific cardholder, so that the profile of each cardholder is generalised. When a new transaction is processed, a threshold is used on the BMU and the result is then compared to the stored cardholder profile. If this differs then it is alerted as a potential fraud. The results indicated a TPR of 65.75% with an FPR of 3.45% that would generate over 198 k $AlertD$ using the benchmark. The performance using such a small dataset may not be indicative or scale to the real-world.

Olszewski et al. (2013) proposes using a SOM to detect fraudulent telecommunications accounts by looking for anomalies in a user's

Table 6
Summary of neural network methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Ghosh and Reilly (1994)	2	0.180	5,614	5	0.090	40.00	12.24
Ryman-Tubb (2016)	3	0.332	5,927	7	0.001	75.56	24.44
Richardson (1997)	4	0.230	8,140	7	0.130	61.41	38.59
Sahin and Duman (2011a)	9	0.134	53,684	51	0.920	92.29	7.71
Zakaryazad and Duman (2016)	12	0.064	114,542	100	1.989	65.06	34.94
Brause et al. (1999)	23	0.038	275,291	385	37.600	95.20	37.60
Guo and Li (2008)	30	0.045	458,715	422	8.000	95.00	5.00
Mahmoudi and Duman (2015)	31	0.009	476,306	1,654	8.321	25.13	74.87
Lee (2013)	37	0.014	698,112	1,371	12.195	44.44	55.56
Aleskerov et al. (1997)	39	0.030	771,765	674	13.475	85.00	15.00
Dorronsoro et al. (1997)	40	0.024	801,684	959	14.000	73.00	27.00
Charleonnan (2016)	43	0.012	1,087,449	949	19.000	51.00	49.00

account using a proposed threshold setting method. A centroid is placed over the account on the SOM two-dimensional map and a dissimilarity measure calculated. A minuscule real-world dataset of 100 accounts was used from a Polish telecoms company with a RGF of 9 and results presented on a ROC chart. If θ is set for a TPR of 90% then this generates a FPR of 20%. This performance has been re-calculated for Tier-1 issuer and with such a poor FPR this method would produce over 1 m $AlertD$. It is therefore impractical as a classifier. The author continues their work in Olszewski (2014) using real-world credit card fraud dataset of 10,000 account transactions from the Warsaw region in Poland with a RGF of 1000. The results reported an unlikely "perfect" fraud detection rate of 100% with a FPR of 0% and so the highest possible fraud detection performance was achieved. This performance may be due to small number of transactions that was used, which may represent a single common fraud vector within the city of Warsaw and so the pattern could be easily separated from the others. This method is unlikely to scale to produce such a perfect classifier with other datasets and is so not included in the benchmark.

3.3.2. KNN and other clustering

K-Nearest Neighbour (KNN) is an early clustering method that is typically based on the Euclidean distance between a data record and those in $TRAIN$ and requires the number of clusters to be set (Fix and Hodges Jr, 1951), a good description is given in Bishop (2006b). A comparison can be made to the P-RCE algorithm (see 3.2.1) which shares many common features. The following studies are included in the survey here as an example of KNN in fraud detection but none provide promising real-world results.

Wen-Fang and Na (2009) also use KNN with a real-world dataset from a China bank of 16,584 transactions with an RGF of 10. Each transaction has 51 fields which are manually pre-processed to a dataset with 28 fields. If the cluster for a new transaction differs from what is expected by a specified threshold it is determined to be anomalous and generates an alert. The highest TPR is quoted as 89.4% but no other metrics are given and so the work cannot be usefully benchmarked.

In Sherly and Nedunchezhian (2010) the KNN approach is extended by combining it with a DT that is created using past examples of fraud. If the anomaly detection indicates a suspicious transaction then the DT is then used to help reduce false positives. Experiments used synthetic data but no details of size and RGF is provided, the best TPR is 85% and FPR 10%. When recalculated this generates $AlertD$ of 573 k — an unrealistic level but the work shows promise in terms of combining different approaches with the aim to reduce FPR rather than focus on TPR (see 3.8), an important industry impact factor.

Tasoulis et al. (2008) propose an interesting adaptive method that uses clustering on a stream of data without the assumption that all the $TRAIN$ data is available to calculate the clusters. Various published stream clustering algorithms are explored. A real-world dataset was used with 77 fields and these are pre-processed but no size of the dataset is given. It is noted that many of these fields are categorical and so methods to group and reduce these are used. Experiments are run as though transactions are processed by the system as they occurred. From the presented graphs, the TPR appears to average around 65% and the FPR 20% but it is unclear. The results are interesting as the system takes into account the order that transactions occur. Setting the system to a slower adaption generates a more stable FPR and an improved TPR. This work is important among the survey as it adapts to a stream of transactions (see 1.4.10). The results are not included in the benchmark as they cannot be sufficiently determined from the graphs.

Juszczak et al. (2008) provides an excellent introduction to the problems of payment card detection with a focus on the difficulties of the data, its distribution and characteristics. It discusses feature extraction using derived fields and importantly the encoding of the temporal and sequential dimension of transactions, such as to calculate "global features" by producing derived fields that encapsulate behaviour over time. A focus is given to one-class classification where it is assumed that none or almost no fraud transactions are available and so the objective is to classify only genuine transactions and reject all others. A range of methods are experimentally tested — including KNN. Two real-world datasets have been used, (1) 2.4 m records with an RGF 40, (2) 600 k records with RGF 35. The authors previously proposed a metric for fraud detection (Hand et al., 2008) using a modified ROC curve based on costs. No other common metrics are provided and so the work cannot be compared to the body of work. The work reports that their KNN methods do not outperform SVM approaches (see 3.7). The work concludes that it can usefully identify new types of fraud rather than focus on classification performance.

Weston et al. (2008) propose a peer group method. In this method, information from other cardholder accounts is leveraged by finding those accounts that are similar. A general profile is then created over time, called a *peer group* that tracks similar behaviour. The idea is to provide more robust anomaly detection than clustering on individual transactions. A UK bank provided a dataset of 50,000 accounts covering a 4-month period. The first 3-months of the data were filtered to contain only genuine transactions. The final month contained 4159 accounts with an RGF 17 and was used to evaluate the performance of the system. A range of experiments were undertaken and the results evaluated on a daily basis rather than over the entire dataset. Graphs are given that plot $AlertD$ against the number of fraudulent accounts missed as a proportion of the number of fraudulent accounts — differing from a standard ROC. As expected, performance is shown to reduce the smaller the initial period. It is seen that a larger peer group improves performance — likely to be due to generalisation. The results are given in terms of $AlertD$ and the number of frauds missed (FN) but these do not indicate TPR or FPR and so the work cannot be benchmarked.

Krivko (2010) propose an anomaly detector trained only on genuine transactions. These transactions have derived fields added that calculate aggregated statistics over various time windows. Any new transaction that is a set distance from those in the anomaly detector is considered to be an outlier and therefore suspicious. It is noted that this method alone would lead to a large number of genuine transactions being misclassified, resulting in a poor performance. Manually selecting different characteristics, the accounts are divided into ten groups one of which is allocated to a cardholder along with decision boundary parameters.

An anomaly detector is used and the output filtered using the specified group. A real-world dataset with 76 fields for each of the 189 m transactions generated by 618.712 debit cardholders. Each field was pre-processed to encode this in a form suitable for the anomaly detector. The natural dataset was sampled to a total of 11,555 cardholders with fraud examples sub-sampled from the total set of frauds giving an RGF of 7.4. This rebalancing was necessary with such a small sample. The results from the experiments were adjusted to represent the actual RGF in the natural dataset. The best results are a poor TPR 27.6% and FPR 11.4%; recalculated this would generate $AlertD$ of 650 k while missing over 70% of the know fraud. The authors note that this method compares well with the existing deployed expert system as it (1) detected fraud earlier and (2) generated substantial savings. This is perhaps an indication of the poor performance of many deployed FMS giving considerable scope for even simplistic research methods to be deployed.

Lesot and d'Allonnes (2012) propose a method to create profiles using a fuzzy hierarchical clustering method. The weighting is calculated using a fuzzy matching approach. This process is again iterative and stops when the positions of the clusters stabilise. A few clusters that summarise the data can be more easily understood so that a balance between the number of clusters and their density needs to be determined. This impacts the computational processing time and the work discusses a number of methods to reduce this complexity. A dataset of 959 k online fraudulent transactions is used and 156 clusters are created that vary in size between 150 k and 1.5 k representative transactions. This work does not propose to use these profiles to detect fraud but the general profiles can be used by experts to better understand fraud vectors.

Kültür and Çağlayan (2017) propose a clustering method for all the transactions for each cardholder, as in earlier works. A range of methods that do not require an assumption to be made as to the number of initial clusters, are tested. The work considers "special event" times in the real-world, such as public holidays where it is known spending behaviour changes. The method creates two profiles per cardholder, (1) for regular dates and (2) for a range of differing holiday periods. The work notes that some account holders have multiple payment cards (multi-card) and in this case the transactions for all their cards are considered as one cardholder. A dataset of 150,957 transactions was supplied by a Turkish bank that were for 105 cardholders. A $TRAIN$ dataset was extracted using only the genuine transactions of 150,227 and a $TEST$ dataset of 767 transactions with both classes and an RGF 21 was extracted. The best results, here considered as the lowest FPR, were where holidays were included and for multi-card. In this case FPR of 18.22% and TPR 97.10% that would generate an unmanageable 1 m+$AlertD$. The work is interesting as it considers real-world aspects of cardholder behaviour. However, it is unlikely to scale with larger issuers, that have an average of c.50 m active cardholders (ValuePenguin, 2017), all of which would need to be stored and processed. No method of updating the individual cardholder profiles is given and many clustering approaches are considered computationally intensive.

Table 7 is a summary of unsupervised neural network methods used for payment card fraud detection excluding Ogwueleka (2011) and Olszewski (2014). The methods used as a classifier alone performs poorly in performance compared to others surveyed, especially the much earlier works.

3.4. Bayesian network

A Bayesian network is a network of nodes that are connected to form a directed acyclic graph. The Bayesian network describes the joint probability distributions over a set of arbitrary inputs and the dependence between the variables, a good description is given in Bishop (2006a).

In Maes et al. (2002) the Bayesian network was created using the STAGE algorithm (Boyan and Moore, 1998) which uses a metric that measures the best fit against the complexity of the topology. Results are given for a dataset that is not described with just four fields, TPR 68% of the fraudulent transactions are correctly classified and 10% FPR. When re-calculated, 573 k $AlertD$ is generated making this an impractical approach.

Panigrahi et al. (2009) propose a hybrid fraud detection method using a Bayes classifier based on individual cardholder transactions that use Dempster–Shafer theory (Shafer, 1976) to combine the results of each fraud detector into an overall belief (genuine, fraud or suspicious). The system has three detector components: (1) A rule-based system, (2) An anomaly detector, (3) A Bayesian behavioural model that uses historic marked transactions for each individual cardholder. These transactions are first processed to calculate the frequency that the payment card is used through measuring a transaction gap (time) over successive eight-hour time windows. A posterior probability is calculated on a new transaction using the Bayes rule for either genuine or fraud cases and that which has the highest probability is chosen as the output. A Dempster–Shafer Adder (DSA) is used to combine evidences from the detector components and compute an overall belief value for each transaction. For each transaction, the detector components contribute their independent observations about the behaviour of the transaction — see Fig. 10. DSA assumes a Universe of Discourse that is a set of mutually exclusive and exhaustive possibilities: (1) the hypothesis that the transaction is not fraud, (2) the hypothesis that the transaction is fraud, (3) the universe hypothesis that the transaction is suspicious.

A synthetic dataset is generated using a method similar to that in the earlier CARDWATCH (Aleskerov et al., 1997) with an improved method for creating realistic sequences of cardholder transactions. The best results are given as 98% of fraudulent transactions correctly classified and 4% misclassified. These results are recalculated, to give 230 k $AlertD$ with 2% of the fraudulent transactions missed. $AlertD$ is four times fewer alerts generated than the CARDWATCH method. It is stated that the proposed method exhibits a substantial reduction in false alarms without compromising the detection rate.

Bahnsen et al. (2013) propose a cost-based method that takes into account the costs to a business of the correct and incorrect classifications from an FMS (discussed in 1.4.2). A Bayesian network is used as the classifier. A European card processing company with 80 m transactions provided a 2012 real-world dataset each with 27 fields and the RGF of 4000. A manual process was used to select attributes that were considered useful and these were then used to derive 260 fields that aimed to capture behaviour over time (such as average spend over 30 days). A subset of this data was used with 750 k transactions which was adjusted to have a RGF of 214. The results here are taken from a graph and shown the TPR of 80% and FPR around 2%. When re-calculated, 115 k $AlertD$ would be generated. Table 8 is a summary of Bayesian network methods used for payment card fraud detection and indicates that the Bayesian network methods provide no practical improvement in performance.

3.5. Evolutionary computing

Evolutionary algorithms are search algorithms based on the mechanics of biological natural selection and natural genetics.

3.5.1. Genetic algorithms

A description of genetic algorithms is given in Holland (1973). Real-world optimisation problems are often NP hard and genetic algorithms have been found to be an efficient hyperspace search approach. Like many approaches, including gradient descent, the algorithms aim to ignore local optima and find the global optimum(s) but the approach is computationally expensive. While the concept is simple to understand, the algorithms require a high degree of expertise in encoding the problem and the evaluation of the fitness function.

In Bentley et al. (2000) a genetic algorithm is used to find fuzzy rules to classify the data. The data consisted 4000 real-world credit card transactions covering January to December in 1995 with 96 fields

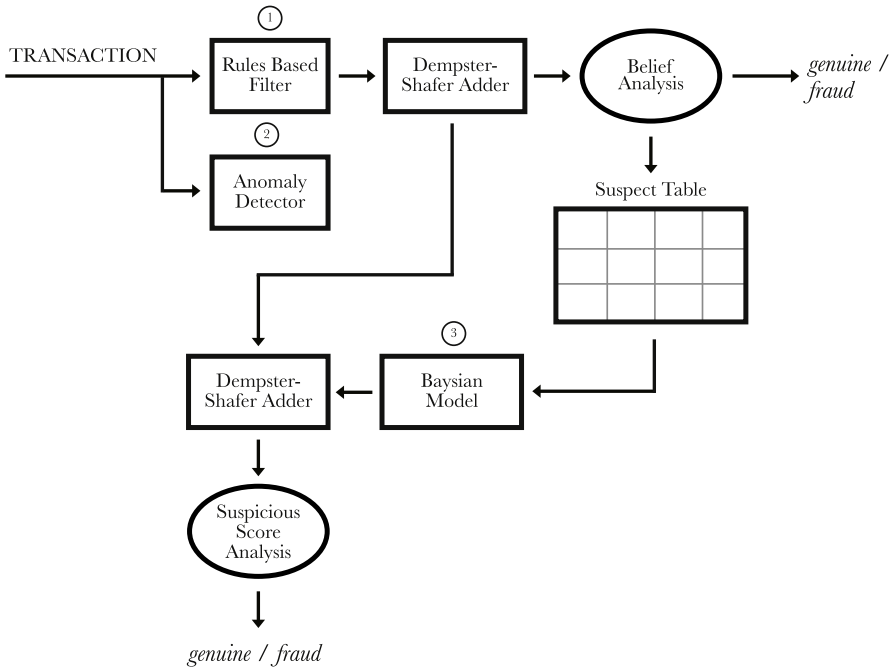

Fig. 10. Diagram of hybrid FMS method using Dempster–Shafer Adder (Panigrahi et al., 2009).

Table 7
Summary of unsupervised neural network methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Zaslavsky and Strizhak (2006)	16	0.048	198,105	173	3.450	65.75	34.25
Sherly and Nedunchezhian (2010)	35	0.035	573,008	589	10.000	85.00	15.00
Krivko (2010)	36	0.007	652,435	2,064	11.400	27.60	72.40
Kültür and Çağlayan (2017)	42	0.029	1,043,358	938	18.220	97.10	97.10
Olszewski et al. (2013)	45	0.025	1,145,099	1,000	20.000	90.00	10.00

Table 8
Summary of Bayesian network methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Bahnsen et al. (2013)	13	0.079	115,323	101	2.000	80.00	20.00
Panigrahi et al. (2009)	19	0.068	229,936	205	4.000	98.00	2.00
Maes et al. (2002)	34	0.027	572,813	500	10.000	68.00	32.00

of which 34 fields were removed as being irrelevant. The $TRAIN$ dataset consisted 66 records used for cross-validation using 3-folds to provide meaningful results on how the model might generalise to an independent dataset with a low RGF of 4. A multi-objective genetic algorithm is then used to determine which of the rules will survive and then have children. The $TEST$ dataset was used and three experiments were undertaken with different membership functions. The best results detected 100% of the fraudulent transactions with FPR of 5.79%. The results from the work are recalculated, to give 332 k $AlertD$, which is worse than the results in earlier work. The work reports that the best ruleset had three rules: (1) IS LOW (field57 ∨ field 50) (2) IS MEDIUM(field56) (3) (field 56 ∨ field 56). It can be seen that field56 dominates these rules. The work notes that the *rulesets* completely change depending on the experimental setup and so the method is not consistent. The initial random selection of transactions for the datasets has a significant impact on the results. It appears that the genetic algorithm is strongly overfitting the problem and creating a ruleset that produces the best results on the specific dataset but generalises poorly. The selection of the input fields is important as a single strong variable may indicate that there is an error in the dataset. It is unlikely in the real-world that a variable is such a single strong indicator of fraud.

Ozcelik et al. (2010) and Duman and Ozcelik (2011) propose a method that uses an evaluated confusion matrix to calculate a single cost/profit to the business that is used as a measure of fitness. The genetic algorithm is similar to that previously discussed except that fuzzy operators are not used. A real-world dataset of 1050 fraudulent transactions was used but no figure is given for the number of genuine transactions. It was stated that the algorithm, "*took several weeks to observe a convergence*" and so it was necessary to reduce the number of genuine transactions. The best results report that the method had a FPR of 35% reported as higher than the existing deployed solution with a claimed 89% saving. Savings are assumed to be the total value calculated using the proposed method over that of the existing solution but no detail is given. As the results are not presented they cannot be compared and so it is not known if this method presents any improvement over other methods discussed.

3.5.2. Artificial Immune System (AIS)

An Artificial Immune System (AIS) attempts to model certain aspects of what is understood as to how biological organisms defend against molecular foreign attack, described in de Castro and Timmis (2002). AIS appears to be a popular more recent method for fraud detection perhaps motivated by its complexity.

In Gadi et al. (2008) a real-world dataset from a large Brazilian bank (issuer) with 41,647 credit card transactions from between July 2004 and September 2004 with an RGF of 26. There were originally 33

fields, which were manually reduced to 10 fields each of which has a single digit value. The monetary gains and losses due to a classification were used as a fitness measure, where the cost is a fixed multiple. A commercial software tool was used (Waikato, 2010) which supported an implementation of AIS and the results are presented as the average saving for each method ranging from R$23.30 (AIS) to R$36.33 (Neural Network). The proposed AIS method has the least saving to the bank with the neural network having the most (a 55% difference). It is argued that the cost matrix used is naïve. As the results are not given in terms of comparable performance measures the work cannot be usefully compared with others and so is excluded from the benchmark.

Brabazon et al. (2010) use a real-world dataset provided by WebBiz with 21 fields and 4 m transactions from 462,279 unique customers with a realistic RGF of 738. This data was first pre-processed to remove any with missing values and to correct errors in the fields, including converting network IP addresses to country of origin. A randomly sampled subset of 50,000 transactions was taken with RGF of 238. Three AIS algorithms were then tested using: (1) Unmodified Negative Selection Algorithm, (2) Modified Negative Selection Algorithm, (3) Clonal Selection Algorithm. The AIS algorithms were implemented using an Euclidean distance where the Value Distance Metric (Stanfill and Waltz, 1986) is used to calculate the distance between fields that are nominal. The work finds that the Modified Negative Selection Algorithm has the best overall results where FPR is 4.06% and TPR is 96.55%. The results from the work are recalculated to give 233 k $AlertD$ due to the high FPR. The work notes that the results indicate the system is not a workable solution in the current form and suggest a better cost function and to use a combination of rules to filter fraud patterns that are evident.

Wong et al. (2012) based their work on a dataset provided by an Australian bank with 640,361 transaction records generated from 21,746 different cardholders with an RGF of 3904. The AIS algorithm was based on that proposed in Hofmeyr and Forrest (1999). The best results ("IV" in the work) indicate that 67.1% of the fraudulent transactions were identified with FPR 3.7%. The results from the work are recalculated to give 212 k $AlertD$ with 33% of the fraudulent transactions missed. The work compares performance to that of an FMS provided by the vendor Fair Isaac called "Falcon" in a case study for a Mexican bank. Falcon achieved a reported TPR of 80% with FPR of 0.194% and generated an excellent 10,560 $AlertD$ with 20% of the fraudulent transactions missed.

Soltani et al. (2012) proposed an AIS method for classification of specific cardholder behaviour using real-world data with 12 fields but no information is given on the size of the dataset. The best results indicate 100% of the fraudulent transactions were identified for each cardholder with FPR as 9.89%. When recalculated this generates $AlertD$ as 567 k. Each cardholder only had a small number of transactions in the dataset and so the detection of an unusual transaction was made relatively trivial, which may explain the 100% detection rate. It may be that this method will not scale to the real-world.

Hormozi et al. (2013) concentrate on implementing the AIS algorithm so that it can be processed in parallel on a cloud-computing platform and it is shown that processing in parallel reduces the compute time by at least 25x and this then allows the number of AIS detectors to be increased and this in turn improves detection rate. Using the same dataset as previously discussed in Gadi et al. (2008), the best results are TPR 75% with FPR 3.5% and recalculating using Tier-1 gives 198 k $AlertD$ with 25% of the fraudulent transactions missed.

Taklikar and Kulkarni (2015) repeat the method in Gadi et al. (2008) and use a synthetic dataset of just 50 transactions with 38 fraudulent transactions and 12 genuine transactions with an extremely unlikely RGF of 0.32 which does not reflect the sparse fraud examples in the real-world. The results report a fraud detection rate of 66% with very poor FPR of 50%. These results would result in over 2.8 m $AlertD$ with 34% of the fraudulent transactions missed. This is little better than a "coin-flip". It is only included here for completeness.

Halvaiee and Akbari (2014) extends the work in Gadi et al. (2008) and uses the same small dataset. A previously published extension of AIS was selected and its efficacy determined on the fraud detection domain using clonal selection (Watkins and Timmis, 2002). The method implements the algorithm using a cloud-based system that distributes the processing across a number of nodes using the Hadoop environment and a MapReduce approach to parallelise the processing. For the described method, a TPR of 51.84% and FPR of 1.8% is calculated for Tier-1 and $AlertD$ of 104 k. The work also reports on the use of the parallel processing that while improves the time-consuming task, it is noted that the overhead of the communication between the clusters makes the choice of the number of nodes and the splitting of the data complex.

3.5.3. Swarm/Bird optimisation

Elías et al. (2011) propose a Multi-objective Clustering (MOC) approach using Particle Swarm Optimisation. The work does not detail the objectives — it would seem that a simple objective to increase a single measure such as MCC would be sufficient. A general description of a multi-objective search algorithm is given but no experiments are presented. It is included here as creating clusters using such a method is novel and may have efficiency gains over other approaches.

Duman and Elikucuk (2013) proposed using a described Migrating Bird Optimisation (MBO) algorithm. A real-world dataset provided by Denzi Bank in Turkey was used with 22 m transactions where RGF is 22,294. An average TPR of 88.91% is reported with FPR of 6%. These results are recalculated to give 345 k $AlertD$ with 11% of the fraudulent transactions missed.

Table 9 is a summary of evolutionary computing methods used for payment card fraud detection. Comparing Table 9 with those previously discussed broadly indicates that the genetic algorithm method has promise but the FPR remains high for the typical volumes in this domain.

3.6. Hidden Markov Model (HMM)

An HMM is a statistical model based on the parametric probability distribution of observable features and are commonly used in temporal pattern recognition domains and is described in detail in Bishop (2006b).

The first work to propose an HMM in this domain, Srivastava et al. (2008) predicts temporal sequences based on individual cardholder transactions. The work does not address the requirement of adding a subsequent new transaction into the HMM sequence and it takes an empirical method to setting the sequence length and the number of states within the HMM. Transaction data is first quantised into a limited set of symbols that are determined using a K-means clustering algorithm. Experiments are based on synthetic data and reported a good TPR while maintaining a low FPR but no figures are given. The method is encouraging as proposes using temporal sequences for fraud detection. Computation increases linearly against the number of transaction sequences. The HMM algorithm is computationally complex and may not be sufficiently scalable to a deployable solution where a model is trained for each cardholder. No results are presented and so cannot be included in the benchmark.

In Chetcuti and Dingli (2008) cardholders are clustered based on their patterns of spending behaviour. For each cluster, the volume of transactions activated is then used as a state transition probably in an HMM by calculating this as a proportion of transactions activated in the other clusters. A real-world dataset was used for the experiments but the number of records is not stated. The best results are given as a TPR of 59% with FPR of 8% and 33% where no classification is given. The author notes that this is "*a very positive result*" but results give 458 k $AlertD$ — worse than the much earlier and less complex methods. This work is repeated in Bhusari and Patil (2011a, b), Dhok (2012), Mishra et al. (2013), Prasad (2013) and Khan et al. (2014) using a small synthetic dataset and fixed cardholder profiles, which depend on total spending value that are either set at fixed values or determined by clustering. Results state an improved TPR of 88% with the same FPR of

Table 9
Summary of genetic and AIS methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Halvaiee and Akbari (2014)	11	0.053	104,043	91	1.808	51.84	48.16
Hormozi et al. (2013)	17	0.056	198,335	230	3.452	75.28	24.72
Wong et al. (2012)	18	0.047	212,421	185	3.700	67.10	32.90
Brabazon et al. (2010)	20	0.066	233,352	204	4.060	96.55	3.45
Bentley et al. (2000)	26	0.057	332,353	290	5.790	100.00	0.00
Duman and Elikucuk (2013)	27	0.049	345,383	339	6.020	88.91	11.09
Soltani et al. (2012)	33	0.043	566,887	495	9.890	100.00	0.00
Taklikar and Kulkarni (2015)	50	0.004	2,860,924	2,498	50.000	65.79	34.21

8%. The authors note that the technique is useful and that it is scalable for handling large volumes but there is no evidence that this is so.

Patel and Kale (2012), Vaidya and Mohod (2012), Mule and Kulkarni (2014) and Thosani et al. (2014) use an HMM per cardholder to estimate the value of their next transaction in sequence. If the actual value of the transaction differs by a threshold then the cardholder is required to validate the transaction using two-stage verification. The verification approach is a well-motivated but no experimental results using a dataset are given and so the work cannot be compared.

Table 10 is a summary of HMM methods used for payment card fraud detection and comparing these results in Table 10 with other methods discussed position the HMM methods generally lower in performance than other simpler methods. In particular, the work has not been tested on large real-world datasets where it is expected that the complexity of the proposed methods will require higher computing power than other better performing methods.

3.7. Support Vector Machine (SVM)

A SVM is a classifier that was developed from the theory of Structural Risk Minimisation — a general description is given in Cortes and Vapnik (1995).

Chen et al., (2004, 2005) takes a novel method to the detection of fraud for a newly issued credit card where previous transactions do not exist. A questionnaire is given to the new customer to complete 105–120 questions. Examples of fraudulent transactions are collected (but no detail is given). In total 12,000 questionnaires were used to create SVM models for each of the individuals. A software tool called "mySVM" was used to create and train the SVMs. This is an interesting approach when data does not exist — such as for a new product. Only the $accuracy$ measure is reported with the best being 84% and so cannot be included in the benchmark.

Whitrow et al. (2009) propose a method of aggregating transactional data so that transaction information accumulated over time. A range of classifiers was used to assess the method one of these being an SVM using the RBF kernel. A real-world dataset of 175 m records that were generated by 16.8 m cardholders using POS or ATM terminals. This dataset contained 5946 fraudulent transactions and so an RGF 2824. This work attempts to take into account the costs associated with fraud by applying a cost matrix, here the cost of FP is given by a simplistic FP.$100. A cost of $2 per alert is used, $(TP + FP)$.$2 and the cost of a correct genuine classification is TN.$0. Results are only presented using total cost. For this reason, the work again cannot be compared. The work does indicate that the SVM has a similar performance to the other classifiers and that all the classifiers reduced the cost to the bank over that of using no classifier.

Dheepa and Dhanapal (2012) use Principal Component Analysis (PCA) to pre-process the input fields to reduce the fields of the $TRAIN$ dataset. A dataset of 576 genuine transactions and 15 fraudulent transactions was used (an RGF of 38). An SVM using the RBF kernel was trained and then tested using a 5-fold cross-validation. Results for this small dataset are reported as a TPR of 90% with FPR of 2.5%, which when recalculated give 144 k $AlertD$ with 10% of the fraudulent transactions missed placing it in the top quartile of the benchmark. It is not known if a system tested on such a small dataset with an unlikely RGF will scale to the real-world.

Table 11 is a summary of the one SVM method used for payment card fraud detection in the benchmark.

3.8. Eclectic

There are methods of payment card fraud detection where the main classifier cannot be categorised into the previous ontology. It is not necessary to fully detail each of these methods but they are typically more complex in terms of implementation. A summary of key work is next given and where possible performance is re-calculated to provide comparable industry benchmark measures.

As reported in Sahin et al. (2012) the neural network methods generally offer a more robust and accurate method for new or unexpected inputs whereas the symbolic inference methods are easy to understand and have exiting domain knowledge. Therefore, a method that integrates these two methods appears to offer a good hybrid solution. Zhaohao and Finnie (2004) propose a theoretical foundation for rules used for the detection of fraud. It reviews three different approaches: (1) Inference, (2) Knowledge-based and (3) hybrid of these. The work makes use of game theory discussed in Vatsa et al. (2009) and a set of general logical inference rules are proposed. This is an interesting approach but as no metrics are presented, again the method cannot be included in the benchmark.

The work in Cabral et al. (2006) is based on rough set theory, discussed in Pawlak (1991) and extends the work in Chiu and Tsai (2004). A real-world payment dataset was supplied by an electrical energy company based in Brazil. A small dataset of 38,551 records of genuine users and 1944 users marked as fraud (RGF of 20) was created after cleaning the original dataset for errors and missing records. The records are grouped by means of matching records based on their field values and then calculating a measure of support as a count similar to the method in the previously discussed Chiu and Tsai (2004). The unique records in the chosen set can then be represented in the form of rules. The best results detected just 30% of the fraudulent records correctly with a FPR of 4.1%. When these results are recalculated give 235 k $AlertD$ with 70% of the fraudulent transactions missed, ranking it in the bottom quartile of the benchmark.

As in the much earlier game theory work in Section 3.1.1, a fraudster wishes to maximise their gain as quickly as possible before the payment card is blocked (Kundu et al., 2006) based their approach on the assumption that this behaviour is unlikely to replicate that of the genuine cardholder. A hybrid method is proposed using two detectors: (1) Anomaly detection as the detection of unusual behaviour by a cardholder (more typically called a behavioural model), (2) Misuse detection models that use previously known patterns of fraud. A model is created for every cardholder based on the sequence of their genuine transactions. A novel algorithm, Basic Local Alignment Search Tool (BLAST) was devised that is able to establish a match between each model and the incoming sequence as it occurs. Synthetic data is generated to evaluate the performance of the system. A range of tests are reported with the best results detecting nearly 80% of the fraudulent records correctly with a FPR of around 18%. These results are recalculated generate 1 m $AlertD$ with 20% of the fraudulent transactions

Table 10
Summary of HMM methods for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Chetcuti and Dingli (2008)	28	0 .027	458 ,303	400	8.000	59 .00	41 .00
Bhusari and Patil (2011a)	29	0 .042	458 ,635	455	8.000	88 .00	12 .00

Table 11
Summary of SVM method for fraud classification.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Dheepa and Dhanapal (2012)	14	0.079	144,039	126	2.500	90.00	10.00

missed, due to the high FPR. The proposed method is complex and it is likely that the use of synthetic data and the random variations of individual cardholder behaviour lead to high misclassification of genuine transactions. Recently, more research is starting to consider the recognition of sequences within streams of transaction and this is a challenging area. Kundu et al. (2009) considerably advance their earlier work through proposing a hybrid method using two sequence alignment algorithms: (1) BLAST as previously discussed and (2) Sequence Search and Alignment by Hashing Algorithm (SSAHA) originally created to search large DNA databases (Ning et al., 2001). The two detectors are used and tested on synthetic data. A range of tests is reported by varying parameters in the proposed algorithms. The results compared to the previous work are given with the lowest FPR of 5% (18%) but detecting less than 70% (80%) of the fraudulent records correctly. The number of misclassifications has substantially reduced but at the expense of fraud classification performance. These results are recalculated to give 287 k $AlertD$ with 30% of the fraudulent transactions missed.

Wen-Fang and Na (2009) proposes an anomaly detection method where previous transactions are stored and a matrix calculated using a Euclidean distance measure between all the fields in all the previous genuine transactions and that of the new one — not dissimilar to a SOM. The distance between known genuine transactions and a new transaction is determined by summing the associated row in the matrix. A θ is set which if exceeded the new transaction is considered suspicious and generates an alarm. A small real-world dataset was supplied by a Chinese domestic commercial bank as cardholder transactions. There were 15,135 genuine transactions and 1449 fraudulent transactions, a RGF of 10 with 28 fields. The best results are given as 89.4% TPR. No other measures are given and so it is impossible to understand the overall performance of this method.

Ramaki et al. (2012) propose an ontology graph method previously discussed in Fang et al. (2007). The ontology graph is built using a dataset of genuine transactions using three concept descriptors: (1) Relationship between the classes, (2) Relationships between the transactions, (3) Relationships between (1) and (2). An algorithm is proposed to match a new transaction with that of the graph, in terms of calculating a distance matrix using the Euclidean distance measure between the fields in the new transaction and those in the ontology. This distance is used as an outlier measure and the higher the value the more unusual the transaction. A synthetic dataset of 5000 records is used with a reported 89.4% TPR and a FPR of 3%. These results are recalculated to give 173 k $AlertD$ with 11% of the fraudulent transactions missed.

Jha et al. (2012) propose a standard logistic regression, e.g. Crow (1960) fraud detector trained on transactions where these transactions have additional derived fields added that calculate aggregated statistics over specified periods. This aggregation method is proposed by many studies, e.g. Ise et al. (2009). A dataset from a Hong Kong bank was used with 49,858,600 credit card transactions over 13 months from January 2006 transactions generated by 1,167,757 credit cards. A logistical regression model was created. The results are given as 82.98% TPR with FPR of 4.52%. These results are recalculated to give 260 k $AlertD$ with 17% of the fraudulent transactions missed. It is interesting that such a well-established statistical modelling approach has results similar to many more complex machine learning methods in this survey and this serves to emphasise the lack of impactful research in this domain.

Ranking at 7 in this benchmark, Salazar et al. (2012) proposes a novel method using signal-processing techniques to create two fraud detectors: (1) A non-Gaussian mixture model, that is a non-Gaussian PDF is created by learning from a $TRAIN$ dataset — similar to that of a neural network, (2) a discriminant classifier that creates a quadratic hyperplane by assuming input data is normally distributed. Ordered statistical digital filters are used to fuse the output of the two probabilistic fraud classifiers. A dataset of 64 m transactions generated by 3 m credit card holders was provided by the Spanish bank, Banco Bilbao Vizcaya Argentaria. This was sampled into 10 m records containing just 2005 known examples of fraud and so an RGF of nearly 5000. A TPR of 60% with FPR of 0.2% is the best result chosen by varying the threshold. When results are recalculated, this would generate 12 k $AlertD$ with 40% of the fraudulent transactions missed. The high ranking is due to the low FPR which is key in such a sparse dataset.

Seeja and Zareapoor and Zareapoor and Shamsolmoali (2015) propose a simple frequent item-set data-mining method called "FraudMiner" based on transactions for each cardholder. For each class, transactions are matched based on their fields and then calculating a count of all those that are similar as a measure of frequency. The transaction with the highest frequency is then used as a single prototype representing the cardholders' behaviour and the other transactions are discarded. When a new transaction is processed, a matching algorithm is used which counts the number of fields that match in both cardholder prototypes. A decision is made that the transaction is fraudulent if the count is over a fixed θ. The FraudMiner is a simplification of the method in the previously discussed Chiu and Tsai (2004). A real-world dataset from an UCSD-FICO Data mining contest in 2009 is used of e-commerce transactions. A real-world transactional dataset covered a 98-day period, generated by 73,729 customers creating c. 100 k records, each with 20 fields. The method generated results where TPR is 80% and FPR of 20%. The work states that the FPR is a "low rate" but this is not so. These results are recalculated to give a poor 1.1 m $AlertD$ with 20% of the fraudulent transactions missed.

Ranked at 5, Carminati et al. (2014) proposes a semi-supervised method called "BankSealer" based on learning behavioural profiles for individual cardholders and then detecting outliers. Three fraud detectors are proposed each of which generate a score: (1) Global profiles are created in a similar method as discussed in Chiu and Tsai (2004) so that historic transactions are grouped into clusters. These clusters are then labelled as representing characteristics of spending behaviour. Each cardholder is associated with one of these profiles. (2) Temporal profiling that uses data aggregated over time for each cardholder and calculates the mean and variance of the numeric values to create a profile that is used to determine if the new transaction causes the mean and variance to change by more than a threshold. (3) A histogram of each cardholder's transactions is created and is used to compare with a new transaction. A large retail bank provided a real-world dataset for the period between April and June 2013 with 460,264 transactions that were unmarked, that is no frauds were known or reported. Based on their experience, fraud experts created three different types of attacks against online banking for the experiments. A number of experiments are performed with the best results being those where cardholders with less than three transactions were first removed reported a FPR of 0.19% and TPR of 98.26%. These results are recalculated to give 11,991

$AlertD$ with just 2% of the fraudulent transactions missed. These are excellent results but may be due to the use of human created fraud cases rather than real-world data with both classes marked. It is not known if the performance would remain similar if used with payment card datasets. However previous surveyed anomaly-based methods have performed poorly due to the high variability of individual cardholder behaviour.

Van Vlasselaer et al. (2015) highlight the importance of pre-processing fields. A novel method is proposed called APATE using a "network" of nodes that encapsulates the relationships between a transaction, cardholder and merchant and the time sequence of transactions. The method creates a distinct matrix for specific time intervals long-, medium- and short-term. Each matrix is calculated at the start of a time period (such as midnight) and creates an exposure score. The exposure score is calculated using Complex Network Analysis (CNA). The matrix represents nodes that are relationship scores between specific merchants, cardholders and transactions on the vertices, signifying a link between them within a constraint. Using a dataset that contains fraudulent transactions, the score values are initially set at the nodes to be labelled as fraud from the dataset. An iterative process denoted *influence propagation* is then undertaken that propagates the influence of labelled nodes across the network using the node scores so as to derive an updated score for all nodes, until a measure of convergence is reached. Each matrix is then updated when a new transaction is presented. Nine "exposure" scores are then calculated from these matrices, as {merchant, cardholder, transaction} x {long, medium, short}. The nine scores are then used as the inputs to the fraud detection classifier. For example, when different stolen CHD/Cards are used in a single merchant to undertake multiple frauds, this will generate a high exposure score. Similar linked merchants will now also have a propagated higher score. Experiments were undertaken using a dataset from a Belgian issuer with 3.3 m transactions and RGF 69. Various pre-processing steps were applied including the exposure scores. Three classifiers were chosen, (1) Random Forest DT, (2) MLP, (3) Logistic Regression and each was trained. Results are presented in a table by selecting a FPR of 1%, generates a TPR of 87.4% for the DT and when this is recalculated this gives $AlertD$ 58 k placing the work 10 in the benchmark. Although this is difficult to accurately determine, selecting FPR of 0.5% looks to generate a TPR of 50% which would generate $AlertD$ 29 k. This highlights the difficultly of comparing different studies, where the results depend upon the selection of a threshold — the value of which is not stated. The approach is complex and it is not known if it would scale to other issuers where with the volume of transactions and the number of cardholders is spread among a disparate number of merchant types.

Zanin et al. (2017) propose a similar method to Van Vlasselaer et al. (2015) called parenclitic network analysis. Again, new features are derived from both the features (fields) combined with the structure of correlations between entities and in this case just one network is created that uses both classes. For a transaction seven metrics are calculated from the network described in the paper. Once the network is built, these derived metrics are used as inputs to an MLP classifier. A dataset of 180 m transactions across 7 m cards covering a 1-year period was supplied by Spanish bank BBVA but the RGF is not stated. The results are presented as small ROC charts and so can only be estimated here. The best results are where both the derived network features and the original fields are used. In this case, if a FPR of 5% is selected then the TPR is c.40% which would generate $AlertD$ c. 290 k. The curve is weak in the low FPR region and so difficult to accurately determine these figures. While these results appear worse than the earlier work, they have been tested on a very large dataset. Given the differences in the datasets, it is not easy to compare the two. It is likely that adding features in this way improves the underlying classifier and this method is therefore an important contribution to improving fraud detection.

Saia (2017) propose a Discrete Wavelet Transformation (DWT) approach, generally described in Chui (1992). The approach considers all transactions as a sequence over time. The DWT algorithm is designed to (1) reduce the dimensionality of the time series using a linear transformation, (2) distribute the original time series over a separate time series so that information is distributed and wavelet coefficients generated. In the case of fraud detection, it is proposed to use only the genuine class transactions and that the second, time series is smaller and so an approximation that is computationally efficient to compare with new incoming transactions. A dataset of 284,807 transactions with RGF 578 was supplied by a European issuer. A 10-fold cross validation approach was taken so that a portion of genuine transaction was used as the $TRAIN$ dataset to generate the DTW prototype and a $TEST$ set containing both classes was used for evaluation. During evaluation, DTW is undertaken on each new transaction and compared using a cosine similarity measure with that of the prototype against a threshold. The results are only reported in terms of the $F\text{-}score$. The result for the DWT is 0.92 which is reported to be worse than a random forest DT that was used as a comparison with a F-score of 0.95 but noting that the DT approach required both classes to train. $F\text{-}score$ has been calculated for the studies in this benchmark then the DWT is ranked at the top of the benchmark. However, since F-score does not include TN as part of the metric considerable caution is needed, as discussed, the FPR is the key metric in the real-world. The method cannot usefully be included in the benchmark. The idea of viewing transactions as a time series and reducing the dimensionality is important.

The key studies in eclectic methods in payment fraud detection have been surveyed and are summarised in Table 12. It can be seen the eclectic methods cover a range of differing classification techniques and many propose hybrid/ensemble methods making use of multiple classifiers. Comparing results of the eclectic methods in Table 12 with those previously discussed, four are highly ranked and then others positioned widely.

Next a discussion of the survey is given followed by suggestions of future directions in this important applied research area.

4. Benchmark results and discussion of the survey

This survey has consistently benchmarked payment card fraud detection methods, as if they were implemented in an FMS in 2017. Focusing on AI and machine learning, methods for payment card fraud detection have been reviewed over a necessarily extensive period, from 1990 to 2017. Results using the proposed metrics can be compared for the first time using time industry statistics, the top ranked quartile are given in Table 13. The full results are given in Table 14.

While this survey has attempted to provide a benchmark, due to the different datasets used in each work, variation in the dataset size, fraud imbalance (RGF) and differing fields, dimensionality and complexity, they remain difficult to compare. Caution must be exercised when making conclusions on the efficacy of the fraud detection methods. There is a scarcity of research papers in this industry domain given the established impact of fraud on society. This may in part be explained by a legacy of those in the payments industry tacitly accepting that the cost of fraud as an acceptable write-off cost of business. As the uptake of payment cards grew so too did the profits of the banks. The fraud levels grew but were a disproportionately small portion of these profits (Evans and Schmalensee, 2005). The banks viewed the fraud write-off as similar to bad debt and therefore as a "cost of business" (Gates and Jacob, 2008). Despite the rapid change in computing technology and the growth of the Internet, fraud vectors have until recently slowly evolved and so current detection methods have been considered adequate by participators and FMS vendors. This may have led to limited motivation by industry to collaborate and fund further research into payment card fraud detection as the cost of fraud has become normative. This has had a significant impact on the research community.

An observation from the survey is that improving the performance of a classifier has generally been the focus of research rather than a systemic approach. In Table 14, the shaded entries highlight methods

Table 12
Summary of eclectic methods.

Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Carminati et al. (2014)	5	0 .303	11,991	11	0.190	98.00	2.00
Salazar et al. (2012)	6	0 .184	12 ,128	18	0.200	60.00	40.00
Van Vlasselaer et al. (2015)	10	0 .122	58 ,205	51	1.000	87.40	12.60
Ramaki et al. (2012)	15	0 .071	172 ,634	169	3.000	89.40	10.60
Cabral et al. (2006)	21	0 .018	234 ,878	683	4.100	30.00	70.00
Jha et al. (2012)	22	0 .053	259 ,510	273	4.520	82.98	17.02
Zanin et al. (2017)	24	0 .023	286 ,475	250	5.000	40.00	60.00
Kundu et al. (2009)	25	0 .042	286 ,819	358	5.000	70.00	30.00
Kundu et al. (2006)	41	0 .023	1,030,578	1,125	18.000	80.00	20.00
Seeja and Zareapoor (2014)	44	0 .021	1,144,985	1,249	20.000	80.00	20.00

Table 13
Top quartile, ranked by AlertD using 2017 Tier-1 industry statistics.

Descr	Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss
Expert	Correia et al. (2015),	1	0.596	2,060	2	0.020	80.00	20.00
Neural	Ghosh and Reilly (1994).	2	0.180	5,614	5	0.090	40.00	12.24
Neural	Ryman-Tubb (2016)	3	0.332	5,927	7	0.001	75.56	24.44
Neural	Richardson (1997)	4	0.230	8,140	7	0.130	61.41	38.59
Eclectic	Carminati et al. (2014)	5	0.303	11,991	11	0.190	98.00	2.00
Eclectic	Salazar et al. (2012)	6	0.184	12,128	18	0.200	60.00	40.00
DT	Dal Pozzolo et al. (2017)	7	0.289	12,222	11	0.195	94.43	5.57
DT	Brause et al. (1999)	8	0.239	16,486	14	0.270	90.91	0.63
Neural	Sahin and Duman (2011a)	9	0.134	53,684	51	0.920	92.29	7.71
Eclectic	Van Vlasselaer et al. (2015)	10	0.122	58,205	51	1.000	87.40	12.60
AIS	Halvaiee and Akbari (2014)	11	0.053	104,043	91	1.808	51.84	48.16
Neural	Zakaryazad and Duman (2016)	12	0.064	114,542	100	1.989	65.06	34.94

The shaded entries highlight methods with less than 20,000 alerts per day.

with less than 20,000 alerts per day that is argued to be manageable. The top-ranking method uses human written rules, three methods are based on neural networks, two use decision tree/random forest and one uses a semi-supervised method based on cluster profiling. While neural networks dominate as a classifier, there is not sufficient evidence to make a firm conclusion. As discussed, the variations in the differing datasets are likely to impact performance. This benchmark provides a guide as to those methods that have potential to achieve a low FPR while maintaining an acceptable TPR.

With the seminal work in 1994, computing power was around 420,000× more expensive than today (Appendix A). The earlier methods surveyed were likely constrained by the computing power available at that time and some authors mention this constraint or it is implied. However, this is less significant in 2018 and yet little research has made use of such advances. It is argued that computing power and payment fraud and its detection using machine learning are implicitly linked.

Some methods set the various hyper-parameters, features, sampling, etc. through a series of experiments so as to manually optimise the presented results. Therefore, there may be some doubt or bias introduced in the interpretation of such results. Results are often stated in such a way that they cannot be compared with other published work or readily reproduced. It is important for empirical experiments to avoid "*phenomenological adjustment of constants*" so as to fit the objectives of the experiment; even if unintentionally (Feynman et al., 1992).

It is conspicuous from the survey that many studies use small datasets. Most classifier methods are sensitive to grouped but random patterns in each subclass when these are near the decision boundary. This may be the case when there is only a small volume of fraud vector examples so that a subclass has a large fraction of these random patterns. If a particular subclass is placed in the search space that is distant from other subclasses and the dimensionality is high, then a large number of training records for that subclass will be required. However, in these small datasets this is not the case and so the classifier will generalise poorly — especially on newer data collected after the model has been created (as this will reflect changing crime behaviours). It is argued that in this case the method is likely to over fit to the random patterns common to members of that subclass and the resulting classifier will not therefore adequately capture the fraud knowledge domain. The examples in the dataset may overlap as criminals aim to make their transactions appear legitimate or the data has been incorrectly marked in the dataset. The cost of misclassification might outweigh the cost of the value of that transaction.

Fraud is typically carried out repeatedly using the same CHD/payment card until it is blocked. It is therefore important that these sequence frauds that occur over a time period are detected as early as possible. There are only a few methods that describe this issue and these use statistics that are aggregated over time to improve their performance. It is suggested that more advanced time series modelling approaches may yield better real-world performance.

5. Future directions and applications in the near future

Using the benchmark in the survey, FMS approaches are arguably already becoming less effective. If fraud detection technologies do not keep pace then businesses and individuals will continue to lose money from loss of their goods/services, charge-backs and fines, their reputation and in some cases business failure. Criminals will continue to gain funding with a wide societal impact. To be effective, fraud needs to be detected in real-time and deployed using a commodity hardware environment that can be easily maintained. To help law enforcement a clear evidential case needs to be presented with the reasons behind the fraud alert. This survey indicates that there is a considerable range of sub problems that need to be further researched. The methods surveyed have a wider application to other areas of financial crime including: anti-money laundering, tax, insurance, social security, on-line services and telecommunications services. In this context, some open research areas and possible future directions are proposed below.

5.1. Industry datasets and data philanthropy

The lack of large, real-world datasets in the field of fraud for the academic community hampers the research into practical new approaches to detection. It is suggested that there should be an aim to facilitate cooperation between researchers and the commercial world to make such datasets publicly available with permission, where these have been sufficiently obfuscated to overcome security and data protection

Table 14
Surveyed methods, ranked comparison by AlertD using 2017 Tier-1 industry statistics.

Descr	Work	Rank ↓	MCC	AlertD ↓	A/F	%FPR	%TPR	%Miss	%Acc	%PG	%PF	F-score	#Records	RGF
Expert	Correia et al. (2015),	1	0.596	2,060	2	0.020	80.00	20.00	99.98	100.00	44.48	0.572	5,600,000,000	n/a
Neural	Ghosh and Reilly (1994).	2	0.180	5,614	5	0.090	40.00	12.24	99.90	99.99	8.16	0.136	2,000,000	666
Neural	Ryman-Tubb (2016)	3	0.332	5,927	7	0.001	75.56	24.44	99.91	100.00	0.24	0.930	59,344,649,000	17,206
Neural	Richardson (1997)	4	0.230	8,140	7	0.130	61.41	38.59	99.86	99.99	8.64	0.152	5,000,000	n/a
Eclectic	Carminati et al. (2014)	5	0.303	11,991	11	0.190	98.00	2.00	99.81	100.00	9.36	0.171	460,264	n/a
Eclectic	Salazar et al. (2012)	6	0.184	12,128	18	0.200	60.00	40.00	99.79	99.99	5.67	0.104	10,002,005	4,988
DT	Dal Pozzolo et al. (2017)	7	0.289	12,222	11	0.195	94.43	5.57	99.80	n/a	n/a	0.162	76,594,714	525
DT	Brause et al. (1999)	8	0.239	16,486	14	0.270	90.91	0.63	99.73	100.00	6.32	0.118	548,708	93
Neural	Sahin and Duman (2011a)	9	0.134	53,684	51	0.920	92.29	7.71	99.08	100.00	1.97	0.039	22,000,978	22,495
Eclectic	Van Vlasselaer et al. (2015)	10	0.122	58,205	51	1.000	87.40	12.60	99.00	n/a	n/a	0.034	3,300,000	69
AIS	Halvaiee and Akbari (2014)	11	0.053	104,043	91	1.808	51.84	48.16	98.18	n/a	n/a	0.011	42,000	26
Neural	Zakaryazad and Duman (2016)	12	0.064	114,542	100	1.989	65.06	34.94	98.00	n/a	n/a	0.013	9,388	9
Bayes	Bahnsen et al. (2013)	13	0.079	115,323	101	2.000	80.00	20.00	98.00	100.00	0.79	0.016	80,000,000	n/a
SVM	Dheepa and Dhanapal (2012)	14	0.079	144,039	126	2.500	90.00	10.00	97.50	100.00	0.72	0.014	591	38
Eclectic	Ramaki et al. (2012)	15	0.071	172,634	169	3.000	89.40	10.60	97.00	100.00	0.59	0.012	5,721,486	n/a
Unsupervised	Zaslavsky and Strizhak (2006)	16	0.048	198,105	173	3.450	65.75	34.25	96.54	99.99	0.38	0.008	100	9
AIS	Hormozi et al. (2013)	17	0.056	198,335	230	3.452	75.28	24.72	96.54	99.99	0.43	0.009	n/a	n/a
Genetic	Wong et al. (2012)	18	0.047	212,421	185	3.700	67.10	32.90	96.29	99.99	0.36	0.007	640,000	n/a
Bayes	Panigrahi et al. (2009)	19	0.068	229,936	205	4.000	98.00	2.00	96.00	n/a	n/a	0.010	n/a	n/a
Genetic	Brabazon et al. (2010)	20	0.066	233,352	204	4.060	96.55	3.45	95.94	100.00	0.47	0.009	50,000	238
Eclectic	Cabral et al. (2006)	21	0.018	234,878	683	4.100	30.00	70.00	95.89	99.99	0.15	0.003	40,495	20
Eclectic	Jha et al. (2012)	22	0.053	259,510	273	4.520	82.98	17.02	95.48	100.00	0.37	0.007	49,858,600	n/a
Neural	Brause et al. (1999)	23	0.038	275,291	385	37.600	95.20	37.60	95.19	99.99	0.26	0.005	548,708	93
Eclectic	Zanin et al. (2017)	24	0.023	286,475	250	5.000	40.00	60.00	94.99	n/a	n/a	0.003	15,000,000	n/a
Eclectic	Kundu et al. (2009)	25	0.042	286,819	358	5.000	70.00	30.00	94.99	99.99	0.28	0.006	n/a	n/a
Genetic	Bentley et al. (2000)	26	0.057	332,353	290	5.790	100.00	0.00	94.21	100.00	0.34	0.007	2,671	3
AIS	Duman and Elikucuk (2013)	27	0.049	345,383	339	6.020	88.91	11.09	93.98	100.00	0.29	0.006	22,000,000	22,494
HMM	Chetcuti and Dingli (2008)	28	0.027	458,303	400	8.000	59.00	41.00	91.99	99.99	0.15	0.003	n/a	n/a
HMM	Bhusari and Patil (2011a)	29	0.042	458,635	455	8.000	88.00	12.00	92.00	100.00	0.22	0.004	n/a	n/a
Neural	Guo and Li (2008)	30	0.045	458,715	422	8.000	95.00	5.00	92.00	100.00	0.24	0.005	n/a	n/a
Neural	Mahmoudi and Duman (2015)	31	0.009	476,306	1,654	8.321	25.13	74.87	91.67	99.98	0.06	0.001	9,300	9
Expert	Leonard (1993)	32	0.031	495,620	611	8.650	70.76	29.24	91.35	99.99	0.16	0.003	12,709	21
AIS	Soltani et al. (2012)	33	0.043	566,887	495	9.890	100.00	0.00	90.11	99.99	0.20	0.004	n/a	n/a
Bayes	Maes et al. (2002)	34	0.027	572,813	500	10.000	68.00	32.00	90.00	99.99	0.14	0.003	n/a	n/a
Unsupervised	Sherly and Nedunchezhian (2010)	35	0.035	573,008	589	10.000	85.00	15.00	90.00	n/a	n/a	0.003	n/a	n/a
Unsupervised	Krivko (2010)	36	0.007	652,435	2,064	11.400	27.60	72.40	88.59	99.98	0.05	0.001	11,555	6
Neural	Lee (2013)	37	0.014	698,112	1,371	12.195	44.44	55.56	87.80	99.99	0.07	0.001	200	5
DT - BAYES	Stolfo et al. (1997)	38	0.028	744,561	650	13.000	80.00	20.00	87.00	100.00	0.12	0.002	500,000	n/a
Neural	Aleskerov et al. (1997)	39	0.030	771,765	674	13.475	85.00	15.00	86.53	100.00	0.13	0.003	112	1
Neural	Dorronsoro et al. (1997)	40	0.024	801,684	959	14.000	73.00	27.00	86.00	99.99	0.10	0.002	n/a	n/a
Eclectic	Kundu et al. (2006)	41	0.023	1,030,578	1,125	18.000	80.00	20.00	82.00	100.00	0.09	0.002	n/a	n/a
Unsupervised	Kültür and Çağlayan (2017)	42	0.029	1,043,358	938	18.220	97.10	97.10	0.82	n/a	n/a	0.002	150,957	21
Neural	Charleonnan (2016)	43	0.012	1,087,449	949	19.000	51.00	49.00	80.99	n/a	n/a	0.001	25,000	4
Eclectic	Seeja and Zareapoor (2014)	44	0.021	1,144,985	1,249	20.000	80.00	20.00	80.00	99.99	0.08	0.002	100,000	n/a
Unsupervised	Olszewski et al. (2013)	45	0.025	1,145,099	1,000	20.000	90.00	10.00	80.00	100.00	0.09	0.002	100	9
CBR	Wheeler and Aitken (2000)	46	0.010	1,259,048	1,099	22.000	50.00	50.00	77.99	99.99	0.05	0.001	700	6
Expert	Vatsa et al. (2009)	47	0.013	1,716,904	1,499	30.000	70.00	25.00	75.00	99.99	0.05	0.000	n/a	n/a
DT	Minegishi and Niimi (2011)	48	0.015	2,376,971	2,186	41.534	94.93	5.07	58.47	100.00	0.05	0.001	47,091	9
Example	Coin Flip	49	0.000	2,860,743	4,995	50.000	50.00	50.00	50.00	50.00	50.00	0.500	n/a	n/a
AIS	Taklikar and Kulkarni (2015)	50	0.004	2,860,924	2,498	50.000	65.79	34.21	50.00	99.99	0.03	0.001	n/a	n/a
Expert	HaratiNik et al. (2012)	51	−0.002	4,434,313	3,871	77.500	91.60	8.40	22.51	99.99	0.02	0.000	n/a	n/a

concerns. MasterCard announced a programme "*to address issues of social benefit and social good*" through "*data philanthropy*" (Forbes, 2014). With the substantial datasets available to one of the biggest card schemes this is hoped to be that start of payment participators: "*combining data and expertise to deliver positive social impact* [in fraud]"

Using just the fields in the transactional, account or cardholder datasets may not contain sufficient information to improve classification further. This leads to the suggested use of more complex data, including unstructured data that is outside that of the current datasets. Can social media be used to learn behavioural patterns to identify potential fraudsters? Data is a critical asset in the detection of fraud but is often held in siloes within an organisation. Bringing this data together and adding new data sources such as social media and information that is uploaded to the Internet every day. Profiling the cyber criminals and applying a game theoretic approach to detect the OCG and their MO may add a new approach to disrupt the growth of cyber-fraud.

5.2. Industry understanding of wider societal impact of fraud

It is suggested that improved fraud management based on research outputs for fraud detection may not be seen as conferring a sufficiently competitive advantage within the payment industry — including the incumbent FMS vendors. It will only become a mainstream accepted approach when there is the realisation of a significant risk event or crisis to stimulate change and innovation (see 1.4.8). Those working to reduce crime and its societal impact must influence those in the payments industry, including governments and regulators into supporting meaningful research to bring about improved prevention and detection methods.

Cyber-fraud is highly lucrative to criminals and the risk of being caught remains low and the punishment weak. As an example, in the USA, over four years from 2006, payment card fraud was the most common cybercrime prosecuted (80%), so that 942 payment card fraud criminals were successfully convicted and sentenced and of these only 490 (52%) received custodial sentences. Around 163 (33%) of these criminals received a sentence of less than one year (Marcum et al., 2011). Research is needed to understand the cyber criminal's cognitive model drawing upon knowledge from law enforcement organisations, think tanks and academia. Understanding this model and the dependencies that cyber criminals have on legitimate infrastructure and service providers will allow an approach for countering cyber-fraud to be developed, seeking to disrupt their model and making it harder to perpetrate and more likely to be caught.

5.3. Improving classifier performance

The surveyed studies concentrate on fraud classifiers and how these can be improved over other approaches. This is a non-trivial problem given the complexities of the real-world data. However, it is suggested that the fraud detection classifier has reached a point where there is little practical insight to be gained by concentrating on its further improvement alone. Deep Learning with neural networks has recently received much research attention, especially in applications such as image recognition and natural language processing. However, it is not clear if this method has any advantages in the fraud detection domain, e.g. Salakhutdinov and Hinton (2009). These approaches may not yield improved results over less complex methods but it is a challenging area of future research. The survey indicates that the temporal and sequential nature of transactions is important as humans develop habitual behaviours, where patterns of expenditure on certain goods, shops, brands, amounts can be observed over a period. As the FMS typically operates in real-time on a stream of data this is a key area of improvement and it appears that researchers are turning their attention to the issue of recognising sequences.

5.4. Implementation within industry

Research methods need to be implemented in hardware servers within the payment participators. Many of the approaches are known to be NP-hard problems and so pose the problem of implementation. The increasing availability of low-cost multi-core processors allows realistic concurrent processing on commodity hardware. Approaches that use now commonly available GPU hardware (Graphic Processing Units with many parallel cores) propose functional programming techniques with immutable data records will enable multiple cores to be fully exploited without any concurrency control overhead, e.g. Dubach et al. (2012). Research could therefore investigate the problem of efficient implementations of the new fraud detection methods. The aim of any new method should be to encourage adoption and so underpin rather than replace existing FMS investment and to improve the productivity of fraud prevention and investigation. Collaborating with industry partners and vendors should lead to deploying the research outputs in the development of novel products/services.

5.5. Cognitive continuous learning systems

The FMS should not be a siloed system but must exist in a wider ecosystem, including the reviewers and fraud experts. What is now termed "Good Old-Fashioned Artificial Intelligence" (Haugeland, 1989), popular in the 1980s and explored in 3.1 represents human knowledge symbolically. It can use logical reasoning from of Inductive Logic Programming (ILP) to help explain decisions/relationships and to generate new facts on the data. The ability of humans to provide knowledge where it is available is important. However, this approach alone was found to be ineffective when there are a lack of experts and when there are rapid outside changes causing the facts to become out dated. This is where adaptive machine learning approaches excel. From this survey, no methods combine rule inference and rule extraction from models built using machine learning. Traditional induction methods tend to overfit the data and the output can be counterintuitive. Neural networks are shown in the survey to be able to learn from experience, generalise from this experience and to abstract important information from abundant real-world datasets, investigated in 3.2 and 3.3. Until recently neural network approaches have been considered a black box approach that cannot explain decisions. It is suggested that rule extraction from such a system with a high level of abstraction and linguistic simplicity is a promising method.

Each of these components can be combined to form part of a cognitive approach (Haikonen, 2003; Bishop, 2015). The FMS must work in collaboration with reviewers and experts using natural language processing, generating natural language questions for humans to answer, grounding learnt information, encoding existing knowledge on wider payments, fraud and crime, able to explain reasoning and decisions, learning and adapting in real-time to streams of data, to combine neural network approaches and knowledge-based methods. More recent ILP research (Muggleton et al., 2015) demonstrate that higher concepts as meta-rules can be learnt and that this will help to integrate human and machine learning for tasks which involve collaboration between the two, so as to learn symmetrically from each other. Most surveyed methods are difficult to update with new data, as more complex fraud vectors are undertaken with a rapidly not previously experienced. As discussed, the rate of change in the financial industry coupled with a similar change to the patterns of fraud vectors, a static model that is only updated in batch will quickly become out dated. One method might be to use Multiple-Instance Learning so that the FMS can adapt and does not need to wait for reviewer feedback as it can learn from both unlabelled and labelled data while the transactions are presented. Algorithms need to be created to add new data from this human/ML feedback loop, while ensuring that current models are not adversely affected.

By joining the academic "connectionists" and "symbolists" disciplines it is suggested that improved and high impact methods will be

Table 15

Direct fraud losses ($\$Fraud$)	For 1971 and 1982 $\$Fraud$ is stated in Nilson-Report (1993). The period 1993–2010 are summarised in Nilson-Report (2013a). 2011 is detailed in Nilson-Report (2013b). 2013 is given in Heggestuen (2014). 2014 is detailed in Nilson-Report (2015a). This data is fitted to give a forecast in 2017 to be $24 bn, a CAGR of 14.6%.
Economic cost of fraud	This includes the cost of the operations necessary to prevent and detect it, loss of fees, interest, charge-backs, goods and services write-off, etc. which differs for each payment participant reported as an average multiple of at least $17.\$fraud in 2017$, giving $416 bn
Operations cost of fraud	This includes the review team, FMS hardware and software and data processing and has been estimated to be at least 30% of $\$fraud$ in 2017§.
Basis point (BP)	This is a standard industry measure of fraud per value of a transaction. It is calculated as $BP = \$Fraud/\$CEV.100$, given in US cents. 2017 is calculated to be 9.29¢.
Cardholder Expenditure Volume ($\$CEV$)	For the period up to 1993 this is stated in Stearns (2011). 1993 is calculated using country level data provided in Mann (2006a). 2013 is stated in Nilson-Report (2015c). This data is fitted to give a forecast $\$CEV$ in 2017 to be $26.3 tn.
Average Transaction Value ($\$ATV$)	For the period 1971–2012 this is stated in Mann (2006a), for 2013 (Hirsch, 2014b) reported to be $84. The $\$ATV$ in 2017 has been forecast as $\$CEV/\#T$ to be an $\$ATV$ of $75 — a reduction from the earlier period likely due to more micro-payments (low value) using contactless cards.
Average Fraud Transaction Value ($FTV)	In 2013 $\$FTV$ was reported to be an average $350 (Graves et al., 2014). It is recognised that this is likely to have a wide variance.
A fraud multiple (fm)	This is the multiple over the average value of a transaction. In 2013 $\$FTV$ and $ATV were reported and so $fm = \$350/\$84 = 4$. It is assumed that this remains true for 2017.
Number of payment card transactions ($\#T$)	For the period up to 2012, this is estimated using $CEV/$ATV. In 1993 this is given in Mann (2006a). For 2013 there is discrepancy between the reported figures between Hirsch (2014a) and Nilson-Report (2015b) and so the mean of these figures has been used. This data is fitted to give a forecast in 2017 of 349 bn payment card transactions.
Number of fraud transactions (#P)	This figure is not directly reported. It has been calculated here as $\$Fraud/\$ATV.fm$. 2017 is approximated as 70 m transactions.
Number of genuine transactions (#N)	In 2017, this has been calculated as $\#P = \#T - \#P$, 349 bn-70 m, so approximately 349 bn transactions.
Ratio of Fraud to Genuine transactions (RGF):	In 2017, this is 349 bn/70 m so approximately 5000. It is recognised that this is likely to have a wide variance between payment participators.
Review Team	An estimation of the number of reviewers can be made if an assumption is made that a reviewer spends 5 min processing each $AlertD$ with an effective 480 min a working day. This gives an average of 96 reviews per day per person. A team of around 10 is required to review 1,000 $AlertD$ and 150 when $AlertD$ is 15,000.
Number of Issuers	There are around 40,000 issuers in 2017 estimated from (Stearns, 2011) that range from small local to large banks. It is likely that all of these will make use of some form of fraud detection, either using a service or a deployed FMS.
Computing costs/MIP	In 1994, a typical server might have been 3xIBM RS/6000 930 with a cost of c. $190 k ($4 m adjusted value) and 63 MIPS (Longbottom, 2015) and so $63 k/MIP. In 2013, this might be 8xBlade (Intel i7) with a cost of c. $150 k and 1 m MIPS and so $0.15/MIP. The performance cost multiple is estimated, $63/$0.15 = 420,000.

www.fits.institute

discovered that will substantially reduce the exponential of payment crime.

6. Conclusions

Fraud losses have grown every year since 1971 despite the preventative and detection methods put in place. These methods have not been sufficiently successful either in the body of work surveyed or in deployed solutions. There are two explanations for the failure of these methods, (1) that there is little industry incentive to improve them while fraud levels are judged as a cost of business and are seen as normative. This industry benchmark and survey indicates that despite the academic validity of the research surveyed, its impact on the payment card industry has been minimal; (2) academic work in this area is difficult and marginalised in terms of funding.

As discussed in Ryman-Tubb (1994), it remains true that research methods must translate into a real-world application to have impact and to integrate with varying existing industry solutions with as little imposition as possible. There has been little incentive for industry to adopt new methods devised by research where these provide limited improvement over the earlier works and their deployment into existing IT systems has associated risks and requires support, time and further funding.

At least nine innovations are disrupting the payments industry based on innovative technology and are having a substantial impact on fraud levels, fraud vectors and the payment card fraud lifecycle. Together this forms a pivotal event that is challenging the effectiveness of current payment fraud detection. As crime migrates to these new technologies it will do so more rapidly than before as criminals use the same technology to share information. This is significant as it is established that there is a timely need for fundamental research into the effective prevention and detection of payment fraud. These new research methods must translate into a real-world deployed application to have a demonstrable impact and be able to integrate with the varying existing industry solutions.

It is concluded there is a gap in research to help reduce payment card fraud in industry. The core goal of this paper is to identify guidance on how the research community can better transition their research into industry and a list of future directions has been proposed for scholars in this area.

Table 16

Smartphone	Ownership has grown from 0.5 bn to a forecast 4.6 bn in 2017 Ericsson (2014).
e-commerce	Transaction value has grown from $545 bn in 2012 to $1.2 tn in 2017 Malik (2014).
m-commerce	Transaction value has grown from $61 bn in 2012 to $520 bn in 2017 Malik (2014).
Contactless payments	The number of cards was 580 bn in 2008 and grew to 1.5 bn within 6 years Payments-Cards-and Mobile (2015)
e-wallet	In 2015 spend was $1 bn, in 2017 it is forecast at $18 bn and by 2020 $5 tn, accounting for 15% of all payments by transaction value Allied-Market-Research (2013).
Micro payments	Payments under $5, in 2015, $39 bn and by 2020, $89 bn Burelli et al. (2011). It is reported that micropayments will largely replace the widespread use of physical coins/tender by 2030.
Virtual currencies	If Bitcoin is used as a general trend, in 2017, 37 m transactions and by 2020, 100 m Blockchain (2015).
Data breaches	From 2008 to 2013 payment processors disclosed 1,489 data breaches exposing at least 262 m payment card records Information-is beautiful (2015) with the impact put at $2.3 bn. If this trend continues, then by 2030 nearly 50% of all CHD might have been compromised.

Acknowledgements

Acknowledged contributions: James Saint, Financial Innovations & Transaction Security (FITS). Richard Everson, University of Exeter.

Appendix A

This Appendix details terms and abbreviations and importantly provides a summary on the various sources and computation of industry data used. The basis for the industry benchmark Tier-1 statistics are given. This is not intended to provide a robust econometric model for the payment card industry — figures are provided to allow an overview and benchmarking. (See Table 15.)

The industry benchmark Tier-1 statistics are estimated to provide a guide benchmark for a typical FMS operating environment and have been calculated as follows. Example below is for 2017:

- In 2015, $\$CEV$ in the USA accounted for 28% of worldwide $\$CEV$ (Nilson-Report, 2015c). This is assumed to remain similar for 2017.
- The top 15 issuers in the USA were reported to have a $\$CEV$ of $2.52 tn (Banks-around-the world, 2010).
- Top 15 issuers as proportion of worldwide $\$CEV$ is $\$CEV/\$2.52$ tn giving 9.6%.
- The average market share per top issuer is an average 9.6%/15 = 0.6%.
- $\$CEV_{tier1}$ is calculated as $\$CEV$*0.6% giving $157.9 bn.
- The number of transactions per day for a Tier-1 is calculated $\#T_{tier1} = \#T.0.6\%/365$ giving 5.7 m.
- The number of fraud transactions per day for a Tier-1 is calculated $\#P_{tier1} = \#P.0.6\%/365$ giving 1150.
- The number of genuine transactions per day $\#N_{tier1} = \#T_{tier1} - \#P_{tier1}$ giving 5.7 m
- The fraud write-off per day for each Tier-1 issuer $\$FraudD_{tier1} = BP.\$CEV_{tier1}/365$ giving $400 k.

Appendix B

It is beyond the scope of this paper to provide full details, but Table 16 indicates the growth in technologies that are argued to be disrupting payments in 2017.

References

Abdallah, A., Maarof, M.A., Zainal, A., 2016. Fraud detection system: A survey. J. Netw. Comput. Appl. 68, 90–113.

ACI-Worldwide, 2017. Enterprise Fraud Detection and Prevention. https://www.aciworldwide.com/capabilities/fraud-detection-and-aml.

Adewumi, A.O., Akinyelu, A.A., 2016. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. Int. J. Syst. Assur. Eng. Manag. 1–17.

Ahmed, M., Mahmood, A.N., Islam, M.R., 2016. A survey of anomaly detection techniques in financial domain. Future Gener. Comput. Syst. 55, 278–288.

Al-Khatib, A., 2012. Electronic payment fraud detection techniques. World Comput. Sci. Inf. Technol. J. 2, 137–141.

Aleskerov, E., Freisleben, B., Rao, B., 1997. CARDWATCH: A neural network based database mining system for credit card fraud detection. In: Computational Intelligence for Financial Engineering, CIFEr. IEEE Press, pp. 220–226.

Allied-Market-Research, 2013. Mobile Wallet Market (Applications, Mode of Payment, Stakeholders and Geography) - Global Share, Size, Industry Analysis, Trends, Opportunities, Growth and Forecast. pp. 2012–2020.

Bahnsen, A.C., Stojanovic, A., Aouada, D., Ottersten, B., 2013. Cost sensitive credit card fraud detection using Bayes minimum risk. In: Machine Learning and Applications, CMLA, 2013 12th International Conference on. Vol. 1. IEEE, pp. 333–338.

Banks-around-the world, 2010. Top credit card issuers. http://www.relbanks.com/rankings/top-credit-card-issuers.

Bar-Hillel, M., 1980. The base-rate fallacy in probability judgments. Acta Psychol (Amst) 44, 211–233.

Bentley, P.J., Kim, Jungwon, Jung, Gil-Ho, Choi, Jong-Uk, 2000. Fuzzy Darwinian detection of credit card fraud. In: 14th Annual Fall Symposium of the Korean Information Processing Society.

Bhusari, V., Patil, S., 2011a. Application of hidden Markov model in credit card fraud detection. Int. J. Distributed and Parallel Syst. 2, 203–211.

Bhusari, V., Patil, S., 2011b. Study of hidden Markov model in credit card fraudulent detection. Int. J. Comput. Appl. (0975–8887) volume.

Bishop, C.M., 1995. Neural Networks for Pattern Recognition. Oxford university press.

Bishop, C.M., 2006a. Graphical Models. In: Pattern Recognition and Machine Learning, Springer, pp. 359–418. (Ch. 8).

Bishop, C.M., 2006b. Pattern Recognition and Machine Learning. Springer.

Bishop, J.M., 2015. The Singularity, or How I Learned to Stop Worrying and Love AI.

Blockchain, 2015. Number of Bitcoin Transactions a Day. https://blockchain.info/charts/n-transactions?timespan=alldaysAverageString=1scale=0address.

Bose, R., 2006. Intelligent technologies for managing fraud and identity theft. In: Information Technology: New Generations, 2006. ITNG 2006. Third International Conference on, pp. 446–451.

Boyan, J.A., Moore, A.W., 1998. Learning evaluation functions for global optimization and boolean satisfiability. AAAI/IAAI, pp. 3–10.

Brabazon, A., Cahill, J., Keenan, P., Walsh, D., 2010. Identifying online credit card fraud using Artificial Immune Systems. In: Evolutionary Computation, CEC, 2010 IEEE Congress on, pp. 1–7.

Brause, R., Langsdorf, T., Hepp, M., 1999. Neural data mining for credit card fraud detection. In: 11th International Conference on Tools with Artificial Intelligence. IEEE Press, p. 103.

Breiman, L., 1996. Bagging predictors. Mach. Learn. 24, 123–140.

Burelli, F., Chua, K., Alty, H., Morris, C., Rothkopf, M., Rofagha, M., Tess, D.T., Healy, Rachel, Schneeberg, D., 2011. Capturing the Micropayments Opportunity. Value Partners, London UK.

Cabral, J.E., Pinto, J.O.P., Linares, K.S.C., Pinto, A.M.A.C., 2006. Methodology for fraud detection using rough sets. In: Granular Computing, 2006 IEEE International Conference on, pp. 244–249.

Campolo, A., Sanfilippo, M., Whittaker, M., Crawford, K., 2017. AI Now 2017 Report. AI Now Institute at New York University.

Carminati, M., Caron, R., Maggi, F., Epifani, I., Zanero, S., 2014. Banksealer: An online banking fraud analysis and decision support system. In: ICT Systems Security and Privacy Protection. Springer, pp. 380–394.

Castle, A., 2008. Drawing Conclusions About Financial Fraud: Crime, Development, and International Co-Operative Strategies in China and the West. Transnational Financial Crime Program, The International Centre for Criminal Law Reform & Criminal Justice Policy, Vancouver, Canada.

Chan, P.K., Fan, W., Prodromidis, A.L., 1999. Distributed data mining in credit card fraud detection. Intell. Syst. Appl. 14, 67–74.

Charleonnan, A., 2016. Credit card fraud detection using RUS and MRN algorithms. In: Management and Innovation Technology International Conference, MITicon, 2016. IEEE, pp. MIT–73–MIT–76.

Chen, R.-C., Chiu, M.-L., Huang, Y.-L., Chen, L.-T., 2004. Detecting credit card fraud by using questionnaire-responded transaction model based on support vector machines. In: Yang, Z., Yin, H., Everson, R. (Eds.), Intelligent Data Engineering and Automated Learning – IDEAL 2004. In: Lecture Notes in Computer Science, vol. 3177, Springer Berlin Heidelberg, pp. 800–806. (Ch. 119).

Chen, R.-C., Shu-Ting, L., Xun, L., 2005. Personalized approach based on SVM and ANN for detecting credit card fraud. In: International Conference on Neural Networks and Brain, vol. 2, IEEE Press, pp. 810–815.

Chetcuti, T., Dingli, A., 2008. Using hidden Markov models in credit card transaction fraud detection. In: Proceedings of the 1st Workshop in ICT, WICT 2008, Valletta, Malta.

Chiu, C.-C., Tsai, C.-Y., 2004. A web services-based collaborative scheme for credit card fraud detection. e-Technology, e-Commerce and e-Service, pp. 177–181.

Choo, K.-K.R., Smith, R.G., McCusker, R., Criminology, A.I.o, 2007. Future directions in technology-enabled crime: 2007–09. Australian Institute of Criminology, Canberra, Australia.

Chui, C.K., 1992. An Introduction to Wavelets. Academic Press.

Cohen, W.W., 1995. Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 115–123.

Correia, I., Fournier, F., Skarbovsky, I., 2015. Industry Paper: The Uncertain Case of Credit Card Fraud Detection.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20, 273–297.

Cortez, N., 2014. Regulating disruptive innovation. Berkeley Technol. Law J. 29.

Crow, E., Davis, F.A., Maxfield, M.W., 1960. In: Davis, F.A., Maxfield, M.W. (Eds.), Statistics Manual. Dover Publications, Inc, New York.

Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., Bontempi, G., 2017. Credit card fraud detection: A realistic modeling and a novel learning strategy. IEEE Trans. Neural Netw. Learn. Syst..

Danenas, P., 2015. Intelligent financial fraud detection and analysis: A survey of recent patents. Recent Patents Comput. Sci. 8, 13–23.

Dazeley, R.P., 2006. To The Knowledge Frontier and Beyond. University of Tasmania.

de Castro, L.N., Timmis, J., 2002. Artificial immune systems: A novel paradigm to pattern recognition. Artif. Neural Netw. pattern Recognit. 1, 67–84.

Dempster, A.P., 2008. Upper and lower probabilities induced by a multivalued mapping. In: Classic Works of the Dempster-Shafer Theory of Belief Functions. Springer Berlin, pp. 57–72.

Dheepa, V., Dhanapal, R., 2012. Behavior based credit card fraud detection using support vector machines. ICTACT J. Soft Comput. 4, 391–397.

Dhok, S.S., 2012. Credit card fraud detection using hidden Markov model. Int. J. Soft Comput. Eng. 2.

Domingos, P., Hulten, G., 2000. Mining high-speed data streams. In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp. pp. 71–80.

Dorronsoro, J.R., Ginel, F., Sgnchez, C., 1997. Neural fraud detection in credit card operations. IEEE Trans. Neural Netw. 8, 827–834.

Dubach, C., Cheng, P., Rabbah, R., Bacon, D.F., Fink, S.J., 2012. Compiling a high-level language for GPUs: (via language support for architectures and compilers). In: ACM SIGPLAN Notices, vol. 47, ACM, pp. 1–12.

Duman, E., Elikucuk, I., 2013. Solving credit card fraud detection problem by the new metaheuristics migrating birds optimization. In: Advances in Computational Intelligence. Springer, pp. 62–71.

Duman, E., Ozcelik, M.H., 2011. Detecting credit card fraud by genetic algorithm and scatter search. Expert Syst. Appl. 38, 13057–13063.

Dvorsky, G., 2017. Hackers Have Already Started to Weaponize Artificial Intelligence. https://gizmodo.com/hackers-have-already-started-to-weaponize-artificial-in-1797688425.

Elías, A., Ochoa-Zezzatti, A., Padilla, A., Ponce, J., 2011. Outlier analysis for plastic card fraud detection a hybridized and multi-objective approach. In: Hybrid Artificial Intelligent Systems. Springer, pp. 1–9.

Ericsson, 2014. Ericsson Mobility Report: 90 percent will have a mobile phone by 2020. http://www.ericsson.com/news/1872291.

European-Union, 2016. EU General Data Protection Regulation, GDPR. http://www.eugdpr.org.

Evans, D.S., Schmalensee, R., 2005. More than money. In: Paying with Plastic. The MIT Press, pp. 72–73. (Ch. 3).

Everett, C., 2003. Credit card fraud funds terrorism. Comput. Fraud Secur..

Fadaei Noghani, F., Moattar, M., 2017. Ensemble classification and extended feature selection for credit card fraud detection. J. AI Data Min. 5, 235–243.

Fan, W., Stolfo, S.J., Zhang, J., Chan, P.K., 1999. AdaCost: Misclassification cost-sensitive boosting, ICML, pp. 97–105.

Fang, L., Cai, M., Fu, H., Dong, J., 2007. Ontology-Based fraud detection. In: Shi, Y., van Albada, G., Dongarra, J., Sloot, P.A. (Eds.), Computational Science – ICCS 2007. In: Lecture Notes in Computer Science, vol. 4489, Springer Berlin Heidelberg, pp. 1048–1055. (Ch. 168).

Feigenbaum, E.A., 1977. The art of artificial intelligence. In: Themes and case studies of knowledge engineering, vol. 1, Stanford Univ CA Dept of Computer Science.

Feynman, R.P., Leighton, R., Hutchings, E., 1992. Surely You're Joking, Mr. Feynman!: Adventures of a Curious Character. Random House.

Financial-Fraud-Action-UK, 2014. Fraud the Facts 2014. The UK Cards Association, London, UK.

Fisher, D.H., McKusick, K.B., 1989. An empirical comparison of ID3 and back-propagation. In: International Joint Conference on Artificial Intelligence, IJCAI, pp. 788–793.

Fix, E., Hodges Jr., J.L., 1951. Discriminatory Analysis-Nonparametric Discrimination: Consistency Properties. California Univ Berkeley.

Forbes, 2014. The Worlds's Biggest Public Companies. http://www.forbes.com/global2000/.

Fu, K., Cheng, D., Tu, Y., Zhang, L., 2016. Credit card fraud detection using convolutional neural networks. In: International Conference on Neural Information Processing. Springer, pp. 483–490.

Gadi, M.F.A., Wang, X., do Lago, A.P., 2008. Credit card fraud detection with artificial immune system. In: Artificial Immune Systems. Springer, pp. 119–131.

Gates, T., Jacob, K., 2008. Payments Fraud: Perception Versus Reality Payments Conference. Federal Reserve Bank of Chicago, pp. 7–13.

Geurts, P., Ernst, D., Wehenkel, L., 2006. Extremely randomized trees. Mach. Learn. 63, 3–42.

Ghosh, S., Reilly, D.L., 1994. Credit card fraud detection with a neural network. In: International Conference on System Sciences. IEEE Press, Hawaii, pp. 621–630.

Graves, J.T., Acquisti, A., Christin, N., 2014. Should payment card issuers reissue cards in response to a data breach? In: Workshop on the Economics of Information Security, WEIS. The Pennsylvania State University, State College, Pennsylvania, USA.

Guo, T., Li, G.-Y., 2008. Neural data mining for credit card fraud detection. In: Machine Learning and Cybernetics, 2008 International Conference on, vol. 7. IEEE, pp. 3630–3634.

Haikonen, P.O., 2003. The Cognitive Approach to Conscious Machines. Imprint Academic.

Halvaiee, N.S., Akbari, M.K., 2014. A novel model for credit card fraud detection using artificial immune systems. Appl. Soft Comput. 24, 40–49.

Han, J., Pei, J., Yin, Y., 2000. Mining frequent patterns without candidate generation. In: ACM SIGMOD Record, vol. 29, ACM, pp. 1–12.

Hanagandi, V., Dhar, A., Buescher, K., 1996. Density-based clustering and radial basis function modeling to generate credit card fraud scores. In: Computational Intelligence for Financial Engineering. IEEE, New York City, NY, USA, pp. 247–251.

Hand, D., Whitrow, C., Adams, N., Juszczak, P., Weston, D., 2008. Performance criteria for plastic card fraud detection tools. J. Oper. Res. Soc. 59, 956–962.

HaratiNik, M., Akrami, M., Khadivi, S., Shajari, M., 2012. FUZZGY: A hybrid model for credit card fraud detection. In: Telecommunications, IST, 2012 Sixth International Symposium on. IEEE, pp. 1088–1093.

Hartigan, J.A., 1975. Clustering Algorithms. Wiley.

Haugeland, J., 1989. Artificial Intelligence: The Very Idea. MIT press.

Heggestuen, J., 2014. The US Sees More Money Lost To Credit Card Fraud Than The Rest Of The World Combined Business Insider. Business Insider Inc, USA.

Hirsch, D., 2014a. Banking Automation Bulletin. RBR, London.

Hirsch, D., 2014b. Global Payment Cards. Banking Automation Bulletin, London.

Hofmeyr, S.A., Forrest, S., 1999. Architecture for an artificial immune system. Evol. Comput. 7, 45–68.

Holland, J.H., 1973. Genetic algorithms and the optimal allocation of trials. SIAM J. Comput. 2, 88–105.

Hormozi, E., Akbari, M.K., Javan, M.S., Hormozi, H., 2013. Performance evaluation of a fraud detection system based artificial immune system on the cloud. In: Computer Science & Education, ICCSE, 2013 8th International Conference on. IEEE, pp. 819–823.

Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L., 2016. Densely connected convolutional networks. arXiv preprint arXiv:1608.06993.

IBM, 2015. IBM Proactive Technology Online. https://www.research.ibm.com/haifa/projects/services/proactive/index.shtml.

Information-is beautiful, 2015. World's biggest data breaches. http://www.informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks/.

Ise, M., Niimi, A., Konishi, O., 新美礼彦, 小西修, 2009. Feature selection in large scale data stream for credit card fraud detection. In: 5th International Workshop on Computational Intelligence and Applications 2009, IWCIA 2009. IEEE Systems, Man & Cybernetics Society, pp. 202–207. IWCIA2009_B1004.

Jacobson, M., 2010. Terrorist financing and the internet. Stud. Confl. Terror. 33, 353–363.

Japkowicz, N., Shah, M., 2011. Error estimation. In: Evaluating Learning Algorithms: A Classification Perspective. Cambridge University Press, pp. 172–177. (Ch. 5).

Jha, S., Guillen, M., Westland, J.C., 2012. Employing transaction aggregation strategy to detect credit card fraud. Expert Syst. Appl. 39, 12650–12657.

Jianyun, X., Sung, A.H., Qingzhong, L., 2006. Tree Based Behavior Monitoring for Adaptive Fraud Detection. Vol. 1, Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, pp. 1208–1211.

Juszczak, P., Adams, N.M., Hand, D.J., Whitrow, C., Weston, D.J., 2008. Off-the-peg and bespoke classifiers for fraud detection. Comput. Statist. Data Anal. 52, 4521–4532.

Khan, M.Z., Pathan, J.D., Ahmed, A.H.E., 2014. Credit card fraud detection system using hidden Markov model and K-Clustering. Int. J. Adv. Res. Comput. Commun. Eng. 3, 5458–5461.

Kohonen, T., 1984. Self-organizing feature maps. In: Self Organisation and Associative Memory.

Kokkinaki, A.I., (1997) 1997. On atypical database transactions: Identification of probable frauds using machine learning for user profiling. In: Knowledge and Data Engineering Exchange Workshop, pp. 107–113.

Krivko, M., 2010. A hybrid model for plastic card fraud detection systems. Expert Syst. Appl. 37, 6070–6076.

Kültür, Y., Çağlayan, M.U., 2017. A novel cardholder behavior model for detecting credit card fraud. Intell. Autom. Soft Comput. 1–11.

Kundu, A., Bagchi, A., Atluri, V., Sural, S., Majumdar, A., 2006. Two-stage credit card fraud detection using sequence alignment. In: Information Systems Security. In: Lecture Notes in Computer Science, vol. 4332, Springer Berlin/Heidelberg, pp. 260–275.

Kundu, A., Panigrahi, S., Sura, S., Majumdar, A.K., 2009. Blast-ssaha hybridization for credit card fraud detection. Dependable and Secure Computing, IEEE Transactions on. pp. 309–315.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324.

Lee, C.C., 2013. A data mining approach using transaction patterns for card fraud detection. arXiv preprint arXiv:1306.5547.

Leonard, K., 1993. Detecting credit card fraud using expert systems. Comput. Ind. Eng. 25, 103–106.

Lesot, M.-J., d'Allonnes, A.R., 2012. Credit-card fraud profiling using a hybrid incremental clustering methodology. In: Scalable Uncertainty Management. Springer, pp. 325–336.

Liu, P., Li, L., 2002. A Game Theoretic Approach to Attack Prediction. Technical Report, PSU-S-001, Penn State Cyber Security Group, pp. 2–2002.

Longbottom, R., 2015. Computer Speed Claims 1980 to 1996. http://www.roylongbottom.org.uk/mips.htm#anchorIBM7.

Lopez-Rojas, E.A., Axelsson, S., 2014. Using financial synthetic data sets for fraud detection research. In: Research in Attacks, Intrusions and Defenses: 17th International Symposium, RAID 2014, Gothenburg, Sweden, September 17–19, 2014, Proceedings, vol. 8688, Springer, pp. 17–19.

Maes, S., Tuyls, K., Vanschoenwinkel, B., Manderick, B., 2002. Credit card fraud detection using Bayesian and neural networks. In: First international congress on neuro fuzzy technologies.

Mahmoudi, N., Duman, E., 2015. Detecting credit card fraud by modified fisher discriminant analysis. Expert Syst. Appl. 42, 2510–2516.

Malik, O., 2014. There Will Be as Much Mobile Commerce in 2018 as E-Commerce in 2013. http://www.theatlantic.com/technology/archive/2014/03/goldman-there-will-be-as-much-mobile-commerce-in-2018-as-br-e-commerce-in-2013/284270/.

Mann, R.J., 2006a. Country-level data. In: Charging ahead: The growth and regulation of payment card markets. Cambridge University Press, pp. 209–240.

Mann, R.J., 2006b. The introduction of the payment card. In: Charging ahead: The growth and regulation of payment card markets. Cambridge University Press. (Ch. 7).

Marcum, C.D., Higgins, G.E., Tewksbury, R., 2011. Doing time for cyber crime: An examination of the correlates of sentence length in the united states. Int. J. Cyber Criminol. 5, 825.

Matthews, B.W., 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochimica et Biophys. Acta Protein Struct. 405, 442–451.

Minegishi, T., Niimi, A., (2011). Detection of fraud use of credit card by extended VFDT. Internet Security, WorldCIS, 2011 World Congress on, pp. 152–159.

Mishra, J.S., Panda, S., Mishra, A.K., 2013. A novel approach for credit card fraud detection targeting the indian market. Int. J. Comput. Sci. Issues 10, 172–179.

Mishra, M.K., Dash, R., 2014. A comparative study of chebyshev functional link artificial neural network, multi-layer perceptron and decision tree for credit card fraud detection. In: Information Technology, ICIT, 2014 International Conference on. IEEE, pp. 228–233.

Morgan, J.N., Sonquist, J.A., 1963. Problems in the analysis of survey data, and a proposal. J. Amer. Statist. Assoc. 58, 415–434.

Muggleton, S.H., Lin, D., Tamaddoni-Nezhad, A., 2015. Meta-interpretive learning of higher-order dyadic datalog: Predicate invention revisited. Mach. Learn. 100, 49–73.

Mule, K., Kulkarni, M., 2014. Credit Card Fraud Detection Using Hidden Markov Model (HMM).

Nilson-Report, 1993. Credit Card Fraud. Carpinteria, California, USA.

Nilson-Report, 2013a. Global Card Fraud.

Nilson-Report, 2013b. Global Credit, Debit, and Prepaid Card Fraud Losses Up 146% in 2012. http://www.paymentsnews.com/2013/08/global-credit-debit-and-prepaid-card-fraud-losses-up-146-in-2012html.

Nilson-Report, 2015a. Global Card Fraud Damages Reach $16B. http://www.pymnts.com/news/2015/global-card-fraud-damages-reach-16b/.

Nilson-Report, 2015b. Global Cards — 2013, The Nilson Report, USA.

Nilson-Report, 2015c. Purchase Volume Worlwide.

Ning, Z., Cox, A.J., Mullikin, J.C., 2001. SSAHA: A fast search method for large DNA databases. Genome Res 11, 1725–1729.

Ogwueleka, F.N., 2011. Data mining application in credit card fraud detection system. J. Eng. Sci. Technol. 6, 311–322.

Olszewski, D., 2014. Fraud detection using self-organizing map visualizing the user profiles. Knowl.-Based Syst. 70, 324–334.

Olszewski, D., Kacprzyk, J., Zadrozny, S., 2013. Employing Self-Organizing Map for Fraud Detection. In: Artificial Intelligence and Soft Computing, Springer, pp. 150–161.

Ozcelik, M.H., Duman, E., Duman, E., Cevik, T., 2010. Improving a credit card fraud detection system using genetic algorithm. In: Networking and Information Technology, ICNIT, 2010 International Conference on. IEEE, pp. 436–440.

Panigrahi, S., Kundu, A., Sural, S., Majumdar, A.K., 2009. Credit card fraud detection: A fusion approach using Dempster–Shafer theory and Bayesian learning. Inf. Fusion 10, 354–363.

Parker, D.B., 1976. Computer abuse perpetrators and vulnerabilities of computer systems. In: Proceedings of the June 7–10, 1976, national computer conference and exposition. ACM, pp. pp. 65–73.

Pasquale, F., 2015. The need to know. In: The Black Box Society: The Secret Algorithms That Control Money and Information. Havard University Press, pp. 2–3. (Ch. 1).

Patel, T., Kale, M.O., 2012. A Secured Approach to Credit Card Fraud Detection Using Hidden Markov Model.

Pawlak, Z., 1991. Rough Sets: Theoretical Aspects of Reasoning About Data. Kluwer Academic Publishing.

Payments-Cards-and Mobile, 2015. Contactless payment and US Chip and PIN adoption drives smart card growth. http://www.paymentscardsandmobile.com/contactless-payment-us-chip-pin-adoption-drives-smart-card-growth/.

Phua, C., Lee, V., Smith, K., Gayler, R.A., 2010. comprehensive survey of data mining-based fraud detection research. Cornell University.

Prasad, V.K., 2013. Method and system for detecting fraud in credit card transaction. Int. J. Innov. Res. Comput. Commun. Eng. 1.

Provost, F.J., Fawcett, T., Kohavi, R., 1998a. The case against accuracy estimation for comparing induction algorithms. In: ICML, vol. 98, pp. 445–453.

Provost, F.J., Fawcett, T., Kohavi, R., 1998b. The case against accuracy estimation for comparing induction algorithms. In: Proceedings of the Fifteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc.

Quah, J.T.S., Sriganesh, M., 2007. Real time credit card fraud detection using computational intelligence. Int. Jt Conf. Neural Netw. 863–868.

Quinlan, J.R., 1986. Induction of decision trees. Mach. Learn. 1, 81–106.

Quinlan, J.R., 2007. C5.0. http://www.rulequest.com/see5-info.html.

Ramaki, A.A., Asgari, R., Atani, R.E., 2012. Credit card fraud detection based on ontology graph. Int. J. Secur. Priv. Trust Manag. 1, 1–12.

Richardson, R., 1997. Neural networks compared to statistical techniques. In: Computational Intelligence for Financial Engineering, CIFEr. Proceedings of the IEEE/IAFE 1997. p. 89–95.

Rosenthal, R.W., 1973. A class of games possessing pure-strategy Nash equilibria. Internat. J. Game Theory 2, 65–67.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533–536.

Ryman-Tubb, N., 1994. Implementation — the only sensible route to wealth creating success: A range of applications. EPSRC: Information Technology Awareness in Engineering, London.

Ryman-Tubb, N., 2011. Computational neuroscience for advancing artificial intelligence: Models, methods and applications. In: Alonso, E., Mondragn, E. (Eds.), Neural-Symbolic Processing in Business Applications: Credit Card Fraud Detection Medical Information Science Reference. IGI Global, pp. 270–314. (Ch. 12).

Ryman-Tubb, N., 2016. Understanding Payment Card Fraud through Knowledge Extraction from Neural Networks using Large-Scale Datasets (Doctor of Philosophy thesis), University of Surrey.

Ryman-Tubb, N., d' Avila Garcez, A.S., 2010. SOAR - Sparse oracle-based adaptive rule extraction: Knowledge extraction from large-scale datasets to detect credit card fraud. In: World Congress on Computational Intelligence. IEEE Press, Barcelona, Spain, pp. 1–9.

Ryman-Tubb, N., Krause, P., 2011. Neural network rule extraction to detect credit card fraud. In: Palmer-Brown, D. Draganova, C. Pimenidis, E. Mouratidis, H. (Eds.), 12th International Conference on Engineering Applications of Neural Networks, EANN, Corfu, Greece.

Sahin, S., Tolun, M.R., Hassanpour, R., 2012. Hybrid expert systems: A survey of current approaches and applications. Expert Syst. Appl. 39, 4609–4617.

Sahin, Y., Bulkan, S., Duman, E., 2013. A cost-sensitive decision tree approach for fraud detection. Expert Syst. Appl. 40, 5916–5923.

Sahin, Y., Duman, E., 2011a. Detecting credit card fraud by ANN and logistic regression. In: Innovations in Intelligent Systems and Applications, INISTA, 2011 International Symposium on, pp. 315–319.

Sahin, Y., Duman, E., 2011b. Detecting Credit Card Fraud by Decision Trees and Support Vector Machines. In: International MultiConference of Engineers and Computer Scientists, vol. 1.

Saia, R., 2017. A discrete wavelet transform approach to fraud detection. In: International Conference on Network and System Security. Springer, pp. 464–474.

Salakhutdinov, R.R., Hinton, G.E., 2009. Deep Boltzmann machines International Conference on Artificial Intelligence and Statistics, AISTATS, Florida, USA.

Salazar, A., Safont, G., Soriano, A., Vergara, L., 2012. Automatic credit card fraud detection based on non-linear signal processing. In: Security Technology, ICCST, 2012 IEEE International Carnahan Conference on. IEEE, pp. 207–212.

Seeja, K., Zareapoor, M., 2014. FraudMiner: A novel credit card fraud detection model based on frequent itemset mining. Sci. World J. 2014.

Sethi, N., Gera, A., 2014. A revived survey of various credit card fraud detection techniques. Int. J. Comput. Sci. Mob. Comput. 3, 780–791.

Shafer, G., 1976. A Mathematical Theory of Evidence, vol. 1. Princeton university press Princeton.

Shannon, C.E., 1948. A mathematical theory of communication. Bell Syst. Tech. J. 27, 379–423, 623–656.
Shao, Y.P., Wilson, A., Oppenheim, C., 1995. Expert systems in UK banking. In: Artificial Intelligence for Applications, 1995. Proceedings. 11th Conference on, pp. 18–23.
Shen, A., Tong, R., Deng, Y., 2007. Application of classification models on credit card fraud detection. In: International Conference on Service Systems and Service Management, pp. 1–4.
Sherly, K.K., Nedunchezhian, R., 2010. BOAT adaptive credit card fraud detection system. Computational Intelligence and Computing Research, ICCIC, 2010 IEEE International Conference on, pp. 1–7.
Shokri, R., 2015. Privacy games: Optimal user-centric data obfuscation. Proc. Priv. Enhanc. Technol. 2015, 1–17.
Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Inf. Process. Manage. 45, 427–437.
Soltani, N., Akbari, M.K., Javan, M.S., 2012. A new user-based model for credit card fraud detection based on artificial immune system. In: Artificial Intelligence and Signal Processing, AISP, 2012 16th CSI International Symposium on. IEEE, pp. 029–033.
Srivastava, A., Kundu, A., Sural, S., 2008. Credit card fraud detection using hidden Markov model. Dependable Secur. Comput. 5, 37–48.
Stanfill, C., Waltz, D., 1986. Toward memory-based reasoning. Commun. ACM 29, 1213–1228.
Stanford-Research-Institute, 2008. Timeline of SRI International Innovations: 1940s - 1950s. http://www.sri.com/about/timeline.
Stearns, D.L., 2011. Core System Statistics. In: Electronic Value Exchange, vol. XXVIII, Springer, p. 219.
Stolfo, S., Fan, W., Lee, W., Prodromidis, A., Chan, P., 1997. Credit card fraud detection using meta-learning. Working notes of AAAI Workshop on AI Approaches to Fraud Detection and Risk Management.
Svigals, J., 2012. The long life and imminent death of the mag-stripe card. IEEE Spectr. 49, 72–76.
Tafti, M.H.A., 1990. Neural networks: A new dimension in expert systems applications. In: Proceedings of the 1990 ACM SIGBDP Conference on Trends and Directions in Expert Systems. ACM, Orlando, Florida, USA, pp. 423–433.
Taklikar, S.H., Kulkarni, R., 2015. Credit card fraud detection system based on user based model with ga and artificial immune system. J. Multidiscip. Eng. Sci. Technol. 2.
Tasoulis, D., Adams, N., Weston, D., Hand, D., 2008. Mining information from plastic card transaction streams. In: Proceedings in Computational Statistics: 18th Symposium, COMPSTAT 2008, vol. 2, pp. 315–322.
Thosani, J.C., Bhadane, C., Avlani, H.M., Parekh, Z.H., 2014. Credit card fraud detection using hidden Markov model. Int. J. Sci. Eng. Res. 5, 1348–1351.
Tsung-Nan, C., 2007. A novel prediction model for credit card risk management. In: Second International Conference on Innovative Computing, Information and Control, pp. 211–215.
Turvey, B.E., 2011. Case linkage. In: Criminal Profiling: An Introduction to Behavioral Evidence Analysis. Academic Press, pp. 310–311. (Ch. 11).
UK-Government, 2017. Industrial Strategy: Building a Britain Fit for the Future, London.
Vaidya, A.H., Mohod, S., 2012. Internet banking fraud detection using HMM and BLAST-SSAHA hybridization. Int. J. Sci. Res..
Value-Penguin, 2017. Largest U.S. Credit Card Issuers: 2017 Market Share Report, https://www.valuepenguin.com/largest-credit-card-issuers.
Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., Baesens, B., 2015. APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. Decis. Support Syst. 75, 38–48.
Vatsa, V., Sural, S., Majumdar, A.,.
Vuk, M., Curk, T., 2006. ROC curve, lift chart and calibration plot. Metodoloski zvezki 3, 89–108.
Waikato, U.o., 2010. Data Mining Software in Java. http://www.cs.waikato.AC.nz/ml/weka/.
Watkins, A., Timmis, J., 2002. Artificial immune recognition system (AIRS): Revisions and refinements. In: 1st International Conference on Artificial Immune Systems, ICARIS2002, vol. 5, University of Kent at Canterbury Printing Unit, pp. 173–181.
Wen-Fang, Y., Na, W., 2009. Research on credit card fraud detection model based on distance sum. In: Artificial Intelligence, 2009. JCAI '09. International Joint Conference on, pp. 353–356.
Weston, D.J., Hand, D.J., Adams, N.M., Whitrow, C., Juszczak, P., 2008. Plastic card fraud detection using peer group analysis. Adv. Data Anal. Classif. 2, 45–62.
Wheeler, R., Aitken, S., 2000. Multiple algorithms for fraud detection. Knowl.-Based Syst. 13, 93–99.
Whitrow, C., Hand, D.J., Juszczak, P., Weston, D., Adams, N.M., 2009. Transaction aggregation as a strategy for credit card fraud detection. Data Min. Knowl. Discov. 18, 30–55.
Wong, N., Ray, P., Stephens, G., Lewis, L., 2012. Artificial immune systems for the detection of credit card fraud: An architecture, prototype and preliminary results. Inf. Syst. J. 22, 53–76.
Yuen, S., 2008. Exporting trust with data: Audited self-regulation as a solution to cross-border data transfer protection concerns in the offshore outsourcing industry. Colum. Sci. Tech. L. Rev. 9, 41.
Yufeng, K., Chang-Tien, L., Sirwongwattana, S., Yo-Ping, H., 2004. Survey of fraud detection techniques. In: Networking, Sensing and Control, 2004 IEEE International Conference on., vol. 2, pp. 749–754. Vol.742.
Zakaryazad, A., Duman, E., 2016. A profit-driven artificial neural network (ANN) with applications to fraud detection and direct marketing. Neurocomputing 175, 121–131.
Zanin, M., Romance, M., Moral, S., Criado, R., 2017. Credit card fraud detection through parenclitic network analysis. arXiv preprint arXiv:1706.01953.
Zareapoor, M., Shamsolmoali, P., 2015. Application of credit card fraud detection: Based on bagging ensemble classifier. Procedia Comput. Sci. 48, 679–686.
Zaslavsky, V., Strizhak, A., 2006. Credit card fraud detection using self-organizing maps. Cybercrime Cybersecur. 4, 8–63.
Zhaohao, S., Finnie, G., 2004. Experience based reasoning for recognising fraud and deception. In: Hybrid Intelligent Systems, 2004. HIS '04. Fourth International Conference on, pp. 80–85.