Detection of Anomalies in Large Scale Accounting Data using Deep Autoencoder Networks

Marco Schreyer
University of St. Gallen
8 PUBLICATIONS **74** CITATIONS
SEE PROFILE

Timur Sattarov
University of Bonn
5 PUBLICATIONS **26** CITATIONS
SEE PROFILE

Damian Borth
Deutsches Forschungszentrum für Künstliche Intelligenz
70 PUBLICATIONS **1,639** CITATIONS
SEE PROFILE

Andreas Dengel
Deutsches Forschungszentrum für Künstliche Intelligenz
659 PUBLICATIONS **5,182** CITATIONS
SEE PROFILE

Detection of Anomalies in Large-Scale Accounting Data using Deep Autoencoder Networks

Marco Schreyer[1], Timur Sattarov[2], Damian Borth[1],
Andreas Dengel[1], and Bernd Reimer[2]

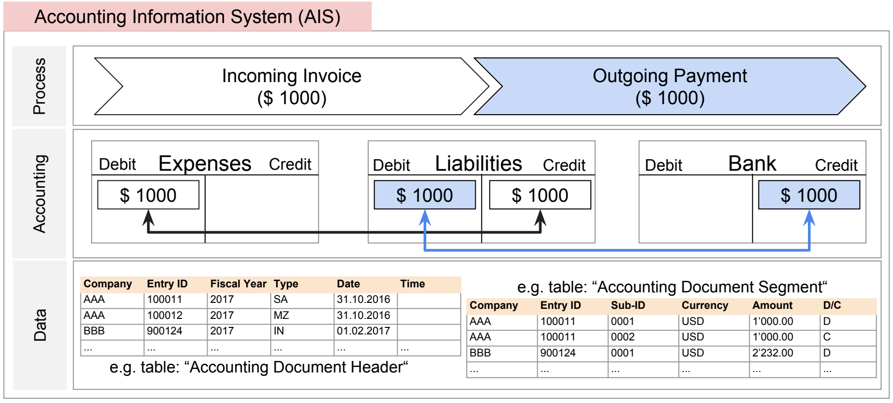
[1] German Research Center for Artificial Intelligence (DFKI) GmbH
Trippstadter Straße 112, Kaiserslautern, Germany
firstname.lastname@dfki.de
[2] PricewaterhouseCoopers (PwC) GmbH,
Friedrichstraße 14, Stuttgart, Germany
lastname.firstname@pwc.com

Abstract. Learning to detect fraud in large-scale accounting data is one of the long-standing challenges in financial statement audits or fraud investigations. Nowadays, the majority of applied techniques refer to handcrafted rules derived from known fraud scenarios. While fairly successful, these rules exhibit the drawback that they often fail to generalize beyond known fraud scenarios and fraudsters gradually find ways to circumvent them. To overcome this disadvantage and inspired by the recent success of deep learning we propose the application of deep autoencoder neural networks to detect anomalous journal entries. We demonstrate that the trained network's reconstruction error obtainable for a journal entry and regularized by the entry's individual attribute probabilities can be interpreted as a highly adaptive anomaly assessment. Experiments on two real-world datasets of journal entries, show the effectiveness of the approach resulting in high f_1-scores of 32.93 (dataset A) and 16.95 (dataset B) and less false positive alerts compared to state of the art baseline methods. Initial feedback received by chartered accountants and fraud examiners underpinned the quality of the approach in capturing highly relevant accounting anomalies.

Keywords: Accounting Information Systems · Computer Assisted Audit Techniques (CAATs) · Journal Entry Testing · Forensic Accounting · Fraud Detection · Forensic Data Analytics · Deep Learning

1 Motivation

The Association of Certified Fraud Examiners estimates in its Global Fraud Study 2016 [1] that the typical organization lost 5% of its annual revenues due to fraud. The term ”fraud” refers to ”the abuse of one's occupation for personal enrichment through the deliberate misuse of an organization's resources or assets” [47]. A similar study, conducted by PwC, revealed that nearly a quarter



Company	Entry ID	Fiscal Year	Type	Date	Time
AAA	100011	2017	SA	31.10.2016	
AAA	100012	2017	MZ	31.10.2016	
BBB	900124	2017	IN	01.02.2017	
...	...	...	...	...	...

e.g. table: “Accounting Document Header“

e.g. table: “Accounting Document Segment“

Company	Entry ID	Sub-ID	Currency	Amount	D/C
AAA	100011	0001	USD	1’000.00	D
AAA	100011	0002	USD	1’000.00	C
BBB	900124	0001	USD	2’232.00	D
...	...	...	...	...	...

Fig. 1. Hierarchical view of an Accounting Information System (AIS) that records distinct layer of abstractions, namely (1) the business process, (2) the accounting and (3) technical journal entry information in designated database tables.

(22%) of respondents experienced losses between $100’000 and $1 million due to fraud [39]. The study also showed that financial statement fraud caused by far the highest median loss of the surveyed fraud schemes[3].

At the same time, organizations accelerate the digitization and reconfiguration of business processes [32] affecting in particular Accounting Information Systems (AIS) or more generally Enterprise Resource Planning (ERP) systems. Steadily, these systems collect vast quantities of electronic evidence at an almost ”atomic” level. This holds in particular for the journal entries of an organization recorded in its general ledger and sub-ledger accounts. SAP, one of the most prominent enterprise software providers, estimates that approx. 76% of the world’s transaction revenue touches one of their ERP systems [41]. Figure 1 depicts a hierarchical view of an AIS recording process of journal entry information in designated database tables.

To detect potentially fraudulent activities international audit standards require the direct assessment of journal entries [2],[22]. Nowadays, the majority of applied techniques to examine journal entries refer to rules defined by experienced chartered accountants or fraud examiners that are handcrafted and often executed manually. The techniques, usually based on known fraud scenarios, are often referred to as ”red-flag” tests (e.g. postings late at night, multiple vendor bank account changes, backdated expense account adjustments) or statistical analyses (e.g. Benford’s Law [9], time series evaluation). Unfortunately, they often don’t generalize beyond historical fraud cases already known and therefore

[3] The ACFE study encompasses an analysis of 2’410 cases of occupational fraud investigated between January 2014 and October 2015 that occurred in 114 countries. The PwC study encompasses over 6’000 correspondents that experienced economic crime in the last 24 months.

fail to detect novel schemes of fraud. In addition, such rules become rapidly outdated while fraudsters adaptively find ways to circumvent them.

Recent advances in deep learning [31] enabled scientists to extract complex nonlinear features from raw sensory data leading to breakthroughs across many domains e.g. computer vision [29] and speech recognition [34]. Inspired by those developments we propose the application of deep autoencoder neural networks to detect anomalous journal entries in large volumes of accounting data. We envision this automated and deep learning based examination of journal entries as an important supplement to the accountants and forensic examiners toolbox [38].

In order to conduct fraud, perpetrators need to deviate from regular system usage or posting pattern. Such deviations are recorded by a very limited number of "anomalous" journal entries and their respective attribute values. Based on this observation we propose a novel scoring methodology to detect anomalous journal entries in large scale accounting data. The scoring considers (1) the magnitude of a journal entry's reconstruction error obtained by a trained deep autoencoder network and (2) regularizes it by the entry's individual attribute probabilities. This anomaly assessment is highly adaptive to the often varying attribute value probability distributions of journal entries. Furthermore, it allows to flag entries as "anomalous" if they exceed a predefined scoring threshold. We evaluate the proposed method based on two anonymized real-world datasets of journal entries extracted from large-scale SAP ERP systems. The effectiveness of the proposed method is underpinned by a comparative evaluation against state of the art anomaly detection algorithms.

In section 2 we provide an overview of the related work. Section 3 follows with a description of the autoencoder network architecture and presents the proposed methodology to detect accounting anomalies. The experimental setup and results are outlined in section 4 and section 5. In section 6 the paper concludes with a summary of the current work and future directions of research.

2 Related work

The task of detecting fraud and accounting anomalies has been studied both by practitioners [47] and academia [3]. Several references describe different fraud schemes and ways to detect unusual and "creative" accounting practices [44]. The literature survey presented hereafter focuses on (1) the detection of fraudulent activities in Enterprise Resource Planning (ERP) data and (2) the detection of anomalies using autoencoder networks.

2.1 Fraud Detection in Enterprise Resource Planning (ERP) Data

The forensic analysis of journal entries emerged with the advent of Enterprise Resource Planning (ERP) systems and the increased volume of data recorded by such systems. Bay et al. in [8] used Naive Bayes methods to identify suspicious general ledger accounts, by evaluating attributes derived from journal entries measuring any unusual general ledger account activity. Their approach was enhanced by McGlohon et al. applying link analysis to identify (sub-) groups of high-risk general ledger accounts [33].

Kahn et al. in [27] and [26] created transaction profiles of SAP ERP users. The profiles are derived from journal entry based user activity pattern recorded in two SAP R/3 ERP system in order to detect suspicious user behavior and segregation of duties violations. Similarly, Islam et al. used SAP R/3 system audit logs to detect known fraud scenarios and collusion fraud via a ”red-flag” based matching of fraud scenarios [23].

Debreceny and Gray in [17] analyzed dollar amounts of journal entries obtained from 29 US organizations. In their work, they searched for violations of Benford’s Law [9], anomalous digit combinations as well as unusual temporal pattern such as end-of-year postings. More recently, Poh-Sun et al. in [43] demonstrated the generalization of the approach by applying it to journal entries obtained from 12 non-US organizations.

Jans et al. in [24] used latent class clustering to conduct an uni- and multivariate clustering of SAP ERP purchase order transactions. Transactions significantly deviating from the cluster centroids are flagged as anomalous and are proposed for a detailed review by auditors. The approach was enhanced in [25] by a means of process mining to detect deviating process flows in an organization procure to pay process.

Argyrou et al. in [6] evaluated self-organizing maps to identify ”suspicious” journal entries of a shipping company. In their work, they calculated the Euclidean distance of a journal entry and the code-vector of a self-organizing maps best matching unit. In subsequent work, they estimated optimal sampling thresholds of journal entry attributes derived from extreme value theory [7].

Concluding from the reviewed literature, the majority of references draw either (1) on historical accounting and forensic knowledge about various ”red-flags” and fraud schemes or (2) on traditional non-deep learning techniques. As a result and in agreement with [46], we see a demand for unsupervised and novel approaches capable to detect so far unknown scenarios of fraudulent journal entries.

2.2 Anomaly Detection using Autoencoder Neural Networks

Nowadays, autoencoder networks have been widely used in image classification [21], machine translation [30] and speech processing [45] for their unsupervised data compression capabilities. To the best of our knowledge Hawkins et al. and Williams et al. were the first who proposed autoencoder networks for anomaly detection [20], [48].

Since then the ability of autoencoder networks to detect anomalous records was demonstrated in different domains such as X-ray images of freight containers [5], the KDD99, MNIST, CIFAR-10 as well as several other datasets obtained from the UCI Machine Learning Repository[4] [15], [4], [50]. In [51] Zhou and

[4] https://archive.ics.uci.edu/ml/datasets.html

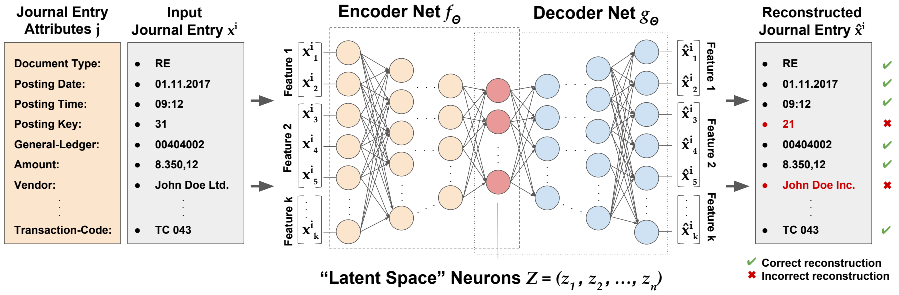


Fig. 2. Schematic view of an autoencoder network comprised of two non-linear mappings (fully connected feed forward neural networks) referred to as encoder $f_\theta : \mathbb{R}^{d_x} \mapsto \mathbb{R}^{d_z}$ and decoder $g_\theta : \mathbb{R}^{d_z} \mapsto \mathbb{R}^{d_y}$.

Paffenroth enhanced the standard autoencoder architecture by an additional filter layer and regularization penalty to detect anomalies.

More recently, autoencoder networks have been also applied in the domain of forensic data analysis. Cozzolino and Verdoliva used the autoencoder reconstruction error to detect pixel manipulations of images [13]. In [16] the method was enhanced by recurrent neural networks to detect forged video sequences. Lately, Paula et al. in [35] used autoencoder networks in export controls to detect traces of money laundry and fraud by analyzing volumes of exported goods.

To the best of our knowledge, this work presents the first deep learning inspired approach to detect anomalous journal entries in real-world and large-scale accounting data.

3 Detection of Accounting Anomalies

In this section we introduce the main elements of autoencoder neural networks. We furthermore describe how the reconstruction error of such networks can be used to detect anomalous journal entries in large-scale accounting data.

3.1 Deep Autoencoder Neural Networks

We consider the task of training an autoencoder neural network using a set of N journal entries $X = \{x^1, x^2, ..., x^n\}$ where each journal entry x^i consists of a tuple of K attributes $x^i = (x^i_1, x^i_2, ..., x^i_j, ..., x^i_k)$. Thereby, x^i_j denotes the j^{th} attribute of the i^{th} journal entry. The individual attributes x_j encompass a journal entry's accounting specific details e.g. posting type, posting date, amount, general-ledger. Furthermore, n^i_j counts the occurrence of a particular attribute value of attribute x_j e.g. a specific document type or account.

An autoencoder or replicator neural network defines a special type of feed-forward multilayer neural network that can be trained to reconstruct its input. The difference between the original input and its reconstruction is referred to as reconstruction error. Figure 2 illustrates a schematic view of an autoencoder neural network. In general, autoencoder networks are comprised of two nonlinear mappings referred to as encoder f_θ and decoder g_θ network [40]. Most commonly the encoder and the decoder are of symmetrical architecture consisting of several layers of neurons each followed by a nonlinear function and shared parameters θ. The encoder mapping $f_\theta(\cdot)$ maps an input vector x^i to a compressed representation z^i in the latent space Z. This latent representation z^i is then mapped back by the decoder $g_\theta(\cdot)$ to a reconstructed vector $\hat{x}^i$ of the original input space. Formally, the non-linear encoder and decoder mapping of an autoencoder encompassing several layers of neurons can be defined by:

$$f^l_\theta(\cdot) = \sigma^l(W^l(f^{l-1}_\theta(\cdot)) + b^l), \text{ and } g^l_\theta(\cdot) = \sigma'^l(W'^l(g^{l-1}_\theta(\cdot)) + d^l), \tag{1}$$

where σ and σ' denote non-linear activations e.g. the sigmoid function, θ denote the model parameters $\{W, b, W', d\}$, $W \in \mathbb{R}^{d_x \times d_z}, W' \in \mathbb{R}^{d_z \times d_y}$ are weight matrices, $b \in \mathbb{R}^{d_z}, d \in \mathbb{R}^{d_y}$ are offset bias vectors and l denotes the number of hidden layers.

In an attempt to achieve $x^i \approx \hat{x}^i$ the autoencoder is trained to learn a set of optimal encoder-decoder model parameters θ^* that minimize the dissimilarity of a given journal entry x^i and its reconstruction $\hat{x}^i = g_\theta(f_\theta(x^i))$ as faithfully as possible. Thereby, the autoencoder training objective is to learn a model that optimizes:

$$\arg\min_\theta \|X - g_\theta(f_\theta(X))\|, \tag{2}$$

for all journal entries X. As part of the network training one typically minimizes a loss function $\mathcal{L}_\theta$ defined by the squared reconstruction loss or, as used in our experiments, the cross-entropy loss given by:

$$\mathcal{L}_\theta(x^i; \hat{x}^i) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{k} x^i_j ln(\hat{x}^i_j) + (1 - x^i_j) ln(1 - \hat{x}^i_j), \tag{3}$$

for a set of n-journal entries x^i, $i = 1, ..., n$ and their respective reconstructions $\hat{x}^i$ over all journal entry attributes $j = 1, ..., k$. For binary encoded attribute values, as used in this work, the $\mathcal{L}_\theta(x^i; \hat{x}^i)$ measures the deviation between two independent multivariate Bernoulli distributions, with mean x and mean $\hat{x}$ respectively [10].

To prevent the autoencoder from learning the identity function the number of neurons of the networks hidden layers are reduced indicating $\mathbb{R}^{d_x} > \mathbb{R}^{d_z}$ (usually referred to as "bottleneck" architecture). Imposing such a constraint onto the network's hidden layer forces the autoencoder to learn an optimal set of parameters θ^* that result in a "compressed" model of the most prevalent journal entry attribute value distributions and their dependencies.

3.2 Classification of Accounting Anomalies

To detect anomalous journal entries we first have to define "normality" with respect to accounting data. We assume that the majority of journal entries recorded within an organizations' ERP system relate to regular day-to-day business activities. In order to conduct fraud, perpetrators need to deviate from the "normal". Such deviating behavior will be recorded by a very limited number of journal entries and their respective attribute values. We refer to journal entries exhibiting such deviating attribute values as *accounting anomalies*.

When conducting a detailed examination of real-world journal entries, recorded in large-scaled ERP systems, two prevalent characteristics can be observed: First, journal entry attributes exhibit a high variety of distinct attribute values and second, journal entries exhibit strong dependencies between certain attribute values e.g. a document type that is usually posted in combination with a certain general ledger account. Derived from this observation and similarly to Breunig et al. in [11] we distinguish two classes of anomalous journal entries, namely *global* and *local anomalies*:

Global accounting anomalies, are journal entries that exhibit unusual or rare individual attribute values. Such anomalies usually relate to skewed attributes e.g. rarely used ledgers, or unusual posting times. Traditionally, "red-flag" tests performed by auditors during an annual audit, are designed to capture this type of anomaly. However, such tests often result in a high volume of false positive alerts due to events such as reverse postings, provisions and year-end adjustments usually associated with a low fraud risk. Furthermore, when consulting with auditors and forensic accountants, "global" anomalies often refer to "error" rather than "fraud".

Local accounting anomalies, are journal entries that exhibit an unusual or rare combination of attribute values while their individual attribute values occur quite frequently e.g. unusual accounting records, irregular combinations of general ledger accounts, user accounts used by several accounting departments. This type of anomaly is significantly more difficult to detect since perpetrators intend to disguise their activities by imitating a regular activity pattern. As a result, such anomalies usually pose a high fraud risk since they correspond to processes and activities that might not be conducted in compliance with organizational standards.

In regular audits, accountants and forensic examiners desire to detect journal entries corresponding to both anomaly classes that are "suspicious" enough for a detailed examination. In this work, we interpret this concept as the detection of (1) any unusual individual attribute value or (2) any unusual combination of attribute values observed. This interpretation is also inspired by earlier work of Das and Schneider [14] on the detection of anomalous records in categorical datasets.

3.3 Scoring of Accounting Anomalies

Based on this interpretation we propose a novel anomaly score to detect global and local anomalies in real-world accounting datasets. Our score accounts for both of the observed characteristics, namely (1) any "unusual" *attribute value occurrence* (global anomaly) and (2) any "unusual" *attribute value co-occurrence* (local anomaly):

Attribute value occurrence: To account for the observation of unusual or rare attribute values we determine for each value x_j its probability of occurrence in the population of journal entries. This can be formally defined by $\frac{n^i_j}{N}$ were N counts the total number of journal entries. For example, the probability of observing a specific general ledger or posting key in X. In addition, we obtain the sum of individual attribute value log-probabilities $P(x^i) = \sum_{j=1}^{k} \ln(1 + \frac{n^i_j}{N})$ for each journal entry x^i over all its j attributes. Finally, we obtain a min-max normalized attribute value probability score AP denoted by:

$$AP(x^i) = \frac{P(x^i) - P_{\min}}{P_{\max} - P_{min}}, \tag{4}$$

for a given journal entry x^i and its individual attributes x^i_j, where P_{max} and P_{min} denotes min- and max-values of the summed individual attribute value log-probabilities given by P.

Attribute value co-occurrence: To account for the observation of irregular attribute value co-occurrences and to target local anomalies, we determine a journal entry's reconstruction error derived by training a deep autoencoder neural network. For example, the probability of observing a certain general ledger account in combination with a specific posting type within the population of all journal entries X. Anomalous co-occurrences are hardly learned by the network and can therefore not be effectively reconstructed from their low-dimensional latent representation. Therefore, such journal entries will result in a high reconstruction error. Formally, we derive the trained autoencoder network's reconstruction error E as the squared- or $L2$-difference $E_{\theta^*}(x^i; \hat{x}^i) = \frac{1}{k} \sum_{j=1}^{k} (x^i_j - \hat{x}^i_j)^2$ for a journal entry x^i and its reconstruction $\hat{x}^i$ under optimal model parameters θ^*. Finally, we calculate the normalized reconstruction error RE denoted by:

$$RE_{\theta^*}(x^i; \hat{x}^i) = \frac{E_{\theta^*}(x^i; \hat{x}^i) - E_{\theta^*,min}}{E_{\theta^*,max} - E_{\theta^*,min}}, \tag{5}$$

where E_{min} and E_{max} denotes the min- and max-values of the obtained reconstruction errors given by E_{θ^*}.

Accounting anomaly scoring: Observing both characteristics for a single journal entry, we can reasonably conclude (1) if an entry is anomalous and (2) if it was created by a "regular" business activity. It also implies that we have seen enough evidence to support our judgment. To detect global and local accounting anomalies in real-world audit scenarios we propose to score each journal entry x^i by its reconstruction error RE regularized by its normalized attribute probabilities AP given by:

$$AS(x^i; \hat{x}^i) = \alpha \times RE_{\theta^*}(x^i; \hat{x}^i) + (1 - \alpha) \times AP(x^i), \tag{6}$$

for each individual journal entry x^i and optimal model parameters θ^*. We introduce α as a factor to balance both characteristics. In addition, we flag a journal

AE	Fully Connected Layers and Neurons
AE 1	[401; 576]-3-[401; 576]
AE 2	[401; 576]-4-3-4-[401; 576]
AE 3	[401; 576]-8-4-3-4-8-[401; 576]
AE 4	[401; 576]-16-8-4-3-4-8-16-[401; 576]
AE 5	[401; 576]-32-16-8-4-3-4-8-16-32-[401; 576]
AE 6	[401; 576]-64-32-16-8-4-3-4-8-16-32-64-[401; 576]
AE 7	[401; 576]-128-64-32-16-8-4-3-4-8-16-32-64-128-[401; 576]
AE 8	[401; 576]-256-128-64-32-16-8-4-3-4-8-16-32-64-128-256-[401; 576]
AE 9	[401; 576]-512-256-128-64-32-16-8-4-3-4-8-16-32-64-128-256-512-[401; 576]

Table 1. Evaluated architecture ranging from shallow architectures (AE 1) encompassing a single fully connected hidden layer to deep architectures (AE 9) encompassing several hidden layers.

entry as anomalous if its anomaly score AS exceeds a threshold parameter β, as defined by:

$$AS(x^i; \hat{x}^i) = \begin{cases} AS(x^i; \hat{x}^i), & AS(x^i; \hat{x}^i) \geq \beta \\ 0, & \text{otherwise} \end{cases}, \tag{7}$$

for each individual journal entry x^i under optimal model parameters θ^*.

4 Experimental Setup and Network Training

In this section we describe the experimental setup and model training. We evaluated the anomaly detection performance of nine distinct autoencoder architectures based on two real-world datasets of journal entries.

4.1 Datasets and Data Preparation

Both datasets have been extracted from SAP ERP instances, denoted SAP ERP dataset A and dataset B in the following, encompassing the entire population of journal entries of a single fiscal year. In compliance with strict data privacy regulations, all journal entry attributes have been anonymized using an irreversible one-way hash function during the data extraction process. To ensure data completeness, the journal entry based general ledger balances were reconciled against the standard SAP trial balance reports e.g. the SAP ”RFBILA00” report.

In general, SAP ERP systems record a variety of journal entry attributes predominantly in two tables technically denoted by ”BKPF” and ”BSEG”. The table ”BKPF” - ”Accounting Document Header” contains the meta information of a journal entry e.g., document id, type, date, time, currency. The table ”BSEG” - ”Accounting Document Segment”, also referred to journal entry line-items, contains the entry details e.g., posting key, general ledger account, debit and credit information, amount. We extracted a subset of 6 (dataset A) and 10 (dataset B) most discriminative attributes of the ”BKPF” and ”BSEG” tables.

The majority of attributes recorded in ERP systems correspond to categorical (discrete) variables, e.g. posting date, account, posting type, currency. In order

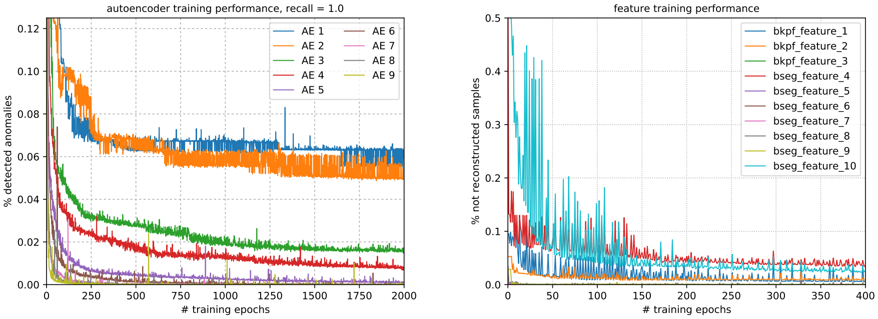


Fig. 3. Training performance using dataset A of the evaluated autoencoder architectures AE 1 - AE 9 (left). Training performance using dataset B of the individual journal entry attributes (right).

to train autoencoder neural networks, we preprocessed the categorical journal entry attributes to obtain a binary (”one-hot” encoded) representation of each journal entry. This preprocessing resulted in a total of 401 encoded dimensions for dataset A and 576 encoded dimensions for dataset B.

To allow for a detailed analysis and quantitative evaluation of the experiments we injected a small fraction of synthetic *global* and *local anomalies* into both datasets. Similar to real audit scenarios this resulted in highly unbalanced class distribution of ”anomalous” vs. ”regular” day-to-day entries. The injected global anomalies are comprised of attribute values not evident in the original data while the local anomalies exhibit combinations of attribute value subsets not occurring in the original data. The true labels (”ground truth”) are available for both datasets. Each journal entry is labeled as either synthetic *global anomaly*, synthetic *local anomaly* or non-synthetic *regular entry*. The following descriptive statistics summarize both datasets:

- **Dataset A** contains a total of 307’457 journal entry line items comprised of 6 categorical attributes. In total 95 (0.03%) synthetic anomalous journal entries have been injected into dataset. These entries encompass 55 (0.016%) global anomalies and 40 (0.015%) local anomalies.
- **Dataset B** contains a total of 172’990 journal entry line items comprised of 10 categorical attributes. In total 100 (0.06%) synthetic anomalous journal entries have been injected into the dataset. These entries encompass 50 (0.03%) global anomalies and 50 (0.03%) local anomalies.

4.2 Autoencoder Neural Network Training

In annual audits auditors aim to limit the number of journal entries subject to substantive testing to not miss any error or fraud related entry. Derived from this desire we formulated three objectives guiding our training procedure: (1) minimize the overall autoencoder reconstruction error, (2) focus on models that

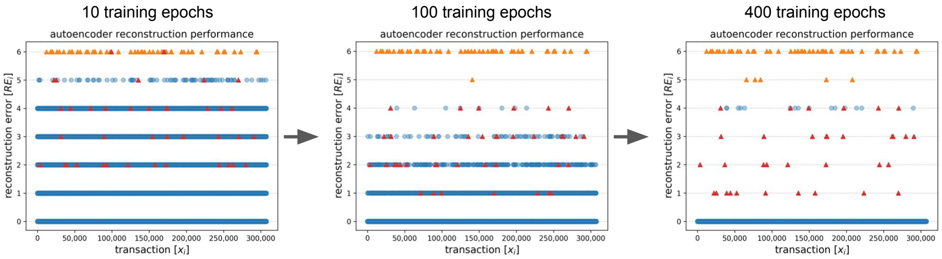


Fig. 4. Journal entry reconstruction error RE obtained for each of the 307.457 journal entries x_i contained in dataset A after 10 (left), 100 (middle) and 400 (right) training epochs. The deep autoencoder (AE 8) learns to distinguish global anomalies (orange) and local anomalies (red) from original journal entries (blue) with progressing training epochs.

exhibit a recall of 100% of the synthetic journal entries, and (3) maximize the autoencoder detection precision to reduce the number of false-positive alerts.

We trained nine distinct architectures ranging from shallow (AE 1) to deep (AE 9) autoencoder networks. Table 1 shows an overview of the evaluated architectures [5] The depth of the evaluated architectures was increased by continuously adding fully-connected hidden layers of size 2^k neurons, where $k = 2, 3, ..., 9$. To prevent saturation of the non-linearities we choose leaky rectified linear units (LReLU) [49] and set their scaling factor to $a = 0.4$.

Each autoencoder architecture was trained by applying an equal learning rate of $\eta = 10^{-4}$ to all layers and using a mini-batch size of 128 journal entries. Furthermore, we used adaptive moment estimation [28] and initialized the weights of each network layer as proposed in [19]. The training was conducted via standard back-propagation until convergence (max. 2'000 training epochs). For each architecture, we run the experiments five times using distinct parameter initialization seeds to guarantee a deterministic range of result.

Figure 3 (left) illustrates the performance of the distinct network topologies evaluated for dataset A over progressing training epochs. Increasing the number of hidden units results in faster error convergence and decreases the number of detected anomalies. Figure 3 (right) shows the individual attribute training performance of dataset B with progressing training. We noticed that the training performance of individual attributes correlates with the number of distinct attributes values e.g. the attribute *bseg_attrubute_10* exhibits a total of 19 distinct values whereas *bseg_attribute_2* exhibits only 2 values and is therefore learned faster.

[5] The notation: $[401; 576] - 3 - [401; 576]$, denotes a network architecture consisting of three fully connected layers. An input-layer consisting of 401 or 576 neurons (depending on the encoded dimensionality of the dataset respectively), a hidden layer consisting of 3 neurons, as well as, an output layer consisting of 401 or 576 neurons.

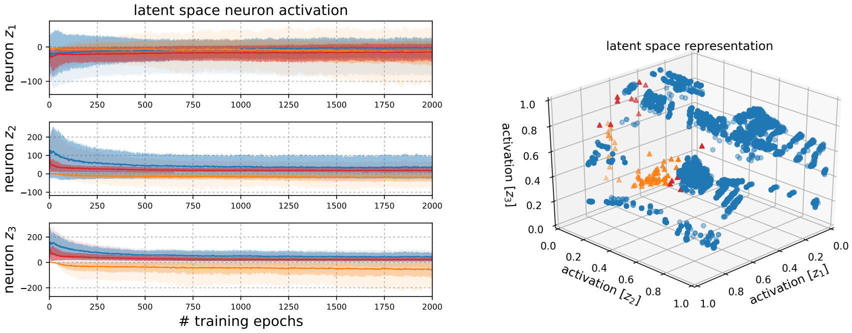


Fig. 5. Learned latent space neuron activations z_1, z_2 and z_3 of autoencoder architecture AE 9 and dataset A with progressing training epochs (left). Latent space representation of dataset A learned by autoencoder architecture AE 9 after 2'000 training epochs (left).

Once the training converged the trained models are used to obtain the reconstruction errors RE of each journal entry. Figure 4 illustrates the reconstruction errors obtained for the 307'457 journal entries contained in dataset A after 10 (left), 100 (middle) and 400 (right) training epochs. The used autoencoder (AE 8) learns to reconstruct the majority of original journal entries (blue) with progressing training epochs and fails to do so for the *global anomalies* (orange) and *local anomalies* (red).

We set the anomaly threshold $\beta = 0.01$ implying that a journal entry is labeled "anomalous" if one of its attributes was not reconstructed correctly or occurs very rarely. This was done in compliance with real-world audit scenarios in which auditors tend to handle fraudulent journal entries in a conservative manner to mitigate risks and not miss a potential true positive.

5 Experimental Results

This section describes the results of our evaluation. Upon successful training we evaluated the proposed scoring according to two criteria: (1) Are the trained autoencoder architectures capable of learning a model of the regular journal entries and thereby detect the injected anomalies ("quantitative" evaluation)? (2) Are the detected and non-injected anomalies "suspicious" enough to be followed up by accountants or forensic examiners ("qualitative" evaluation)?

5.1 Quantitative Evaluation

To quantitatively evaluate the effectiveness of the proposed approach, a range of evaluation metrics including precision, standard f_1-Score, top-k precision, absolute and relative number of detected anomalies are reported. The choice of f_1-Score is to account for the highly unbalanced anomalous vs. non-anomalous

Model	Dataset	Precision	F_1-Score	Top-k	Anomalies	Anomalies
					[%]	[#]
AE 1	A	0.0049	0.0098	0.0049	6.26	19’233
AE 2	A	0.0063	0.0126	0.0063	4.87	14’966
AE 3	A	0.0098	0.0194	0.6632	3.16	9’719
AE 4	A	0.0290	0.0564	0.7684	1.07	3’275
AE 5	A	0.0641	0.1204	0.6632	0.48	1’483
AE 6	A	0.0752	0.1398	0.5263	0.41	1’264
AE 7	A	0.0796	0.1474	0.7895	0.39	1’194
AE 8	A	0.1201	0.2144	0.5684	0.26	791
AE 9	**A**	**0.1971**	**0.3293**	**0.6947**	**0.16**	**482**
AE 1	B	0.0020	0.0040	0.0020	28.84	49’897
AE 2	B	0.0030	0.0059	0.0030	19.52	33’762
AE 3	B	0.0052	0.0104	0.6200	11.04	19’102
AE 4	B	0.0076	0.0150	0.7300	7.65	13’238
AE 5	B	0.0087	0.0173	0.7400	6.62	11’444
AE 6	B	0.0251	0.0489	0.6100	2.30	3’986
AE 7	B	0.0268	0.0522	0.6400	2.15	3’735
AE 8	B	0.0197	0.0387	0.6700	2.93	5’070
AE 9	**B**	**0.0926**	**0.1695**	**0.4200**	**0.62**	**1’080**

Table 2. Evaluation of anomaly detection performance for both datasets A and B using autoencoder architectures AE 1 - AE 9. The training was constrained to models exhibiting a recall of 100% of the synthetic journal entries using standard back propagation. Models were trained until reconstruction error convergence or max. 2’000 epochs. The best detection performances was obtained by architecture AE 9 consisting of 17 hidden layers and LReLU activations.

class distribution of the datasets. To calculate the top-k precision we set $k = 95$ (dataset A) and $k = 100$ (dataset B) corresponding to the number of synthetic anomalies in both benchmark datasets.

Table 2 shows the obtained results of both benchmark datasets using distinct network architectures. Increasing the number of hidden layers reduces the number of detected anomalies. While preserving recall of 100%, the deepest trained autoencoder architecture (AE 9) results in a low fraction of 0.16% detected anomalies in dataset A and 0.62% detected anomalies in dataset B. The observed results show that the autoencoder depth substantially affects its ability to model the inherent manifold structure within each dataset. Figure 6 (left) illustrates the anomaly score AS_i distributions of the distinct journal entry classes using a trained deep autoencoder (AE 9) and $\alpha = 0.3$.

To understand the observed difference in detection performance for both datasets we also investigated the learned latent space representations. Figure 5 (left) shows the mean neuron activation $[z_1, z_2, z_3]$ of the deep autoencoder (AE 9) three ”bottleneck” neurons z_1, z_2 and z_3. With progressing training epochs the network learns a distinctive activation pattern for each journal entry class. Upon 2’000 training epochs a mean activation of $[-12.99, -20.57, -50.55]$ can be observed for textitglobal anomalies (orange), $[-15.09, 18.99, 23.40]$ for *local anomalies* (red) and $[-1.67, 37.28, 44.76]$ for regular journal entries (blue). Figure 5 (right) shows the learned manifolds structure after the training completion.

Method	Data	Precision	F_1-Score	ROC AUC	Anomalies [%]	Anomalies [#]
PCA (c=30)	A	0.0055	0.0110	0.9608	1.63	17’162
HDBSCAN (mcl=1’100)	A	0.0256	0.0499	0.9787	1.29	3’714
LOF (k-NN=50)	A	0.0499	0.0952	0.9979	0.62	1’901
OC-SVM (nu=0.005, γ=0.95)	A	0.2769	0.4338	0.9999	0.11	343
AE (m=AE 9)	A	**0.5688**	**0.7251**	**0.9999**	**0.05**	**167**
PCA (c=36)	B	0.0331	0.0640	0.9978	1.75	3’025
HDBSCAN (mcl=7’445)	B	0.0383	0.0720	0.9744	1.65	2’680
LOF (k-NN=10)	B	0.0518	0.0986	0.9984	1.12	1’929
OC-SVM (nu=0.01, γ=0.95)	B	0.1397	0.2451	0.9997	0.41	716
AE (m=AE 9)	B	**0.1616**	**0.2782**	**0.9997**	**0.35**	**619**

Table 3. Comparative evaluation of the autoencoder based approach against several unsupervised and non-parameteric anomaly detection techniques (best performing parameters of each technique in brackets). For each method the best detection performance is reported that results (1) in a recall of 100% of the synthetic anomalies and (2) the best obtainable ROC-AUC.

5.2 Qualitative Evaluation

To qualitatively evaluate the character of the detected anomalies contained in both datasets we reviewed all non-synthetic journal entries detected by the AE 9 architecture. To distinguish local from global anomalies we empirically choose to set $\alpha = 0.3$ and flagged the journal entries exhibiting an $AS \geq 0.4$ as local anomalies and $AS < 0.4$ as global anomalies.

As anticipated the review of the *global anomalies* revealed that the majority of anomalies correspond to journal entries that exhibit one or two rare attribute values e.g. journal entries that correspond to seldom vendors or seldom currencies. In addition, we also detected journal entries referring to: (1) posting errors due to wrongly used general ledger accounts; (2) journal entries of unusual document types containing extremely infrequent tax codes; and (3) incomplete journal entries exhibiting missing currency information. Especially, the latter observations indicated a weak control environment around certain business processes of the investigated organization.

The review of the *local anomaly* journal entry population showed that these anomalies correspond to journal entries exhibiting attributes that are frequently observable but rarely occur in combination e.g. changes of business process or rarely applied accounting practices. A more detailed investigation of the detected instances uncovered: (1) shipments to customers that are invoiced in different than the usual currency; (2) products send to a regular client but were surprisingly booked to another company code; (3) postings that exhibit an unusual large time lag between document date and posting date; and, (4) irregular rental payments that slightly deviate from ordinary payments. Our initial feedback received by auditors underpinned not only their relevance from an audit but also a forensic perspective.

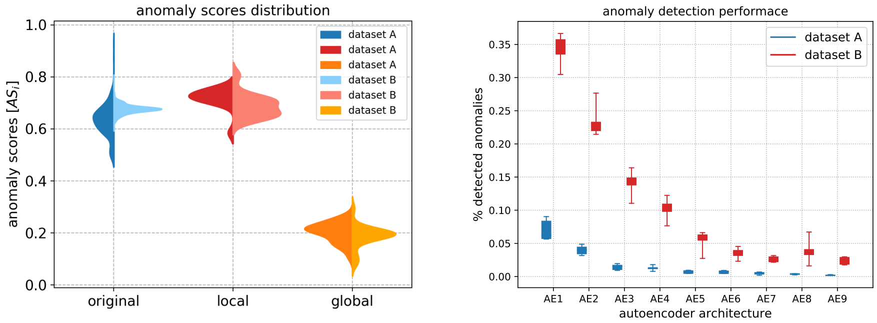


Fig. 6. Distribution of anomaly scores AS_i of the distinct journal entry classes in both datasets using a trained deep autoencoder (AE 9) and $\alpha = 0.3$ (left). Fraction of detected anomalies in both datasets for the evaluated autoencoder architectures (right). Result variation originates from initializing the weights of each model with five distinct seed values.

5.3 Baseline Evaluation:

We evaluated the autoencoder network based approach against unsupervised and non-parametric anomaly detection techniques, namely (1) reconstruction error-based: Principal Component Analysis (PCA) [36], (2) kernel-based: One Class Support Vector Machine (OC-SVM) [42], (3) density based: Local-Outlier Factor (LOF) [11], and (4) hierarchical nearest-neighbor based: Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [18] anomaly detection. For all methods, besides DBSCAN, the performance was assessed using their implementations of the sci-kit machine learning library [37]. For DBSCAN we used the optimized HDBSCAN implementation developed by Campello et al.[12].

To conduct a fair comparison an exhaustive grid search over each techniques parameter space was conducted to determine their best performing parameters. We report the anomaly detection performance of each technique in table 3. The best performing results are selected based on parameterizations that (1) result in a recall of 100% of the synthetic anomalies and correspond to (2) the highest area under the ROC curve (ROC-AUC).

For both evaluation datasets, best ROC-AUC results are obtained for the OC-SVM and the AE 9 autoencoder architecture. However, the autoencoder based approach outperforms the other benchmark techniques in terms of its anomaly detection precision. Comparing both the OC-SVM and the AE 9 for dataset A, the autoencoder results in 176 less detected false positive anomalies; while for dataset B, the autoencoder results in 97 less detected false positive anomalies. The lower but highly accurate number of false positive alerts is of great relevance in the context of real financial audit scenarios where substantiative evaluation of a single detected anomaly can results in considerable effort.

6 Conclusion and Future Work

In this work we presented the first deep learning based approach for the detection of anomalous journal entries in large scaled accounting data. Our empirical evaluation using two real-world accounting datasets demonstrates that the reconstruction error of deep autoencoder networks can be used as a highly adaptive anomaly assessment of journal entries. In our experiments we achieved a superior f_1-score of 32.93 in dataset A and 16.95 in dataset B compared to state of the art baseline methods. Qualitative feedback, received by auditors and forensic accountants, on the detected anomalies underpinned that our method captures journal entries of high relevance for a detailed follow-up audit.

We are excited about the future of deep learning based audit approaches and plan to conduct a more detailed investigation of the journal entries' latent space representations learned by deep autoencoders. We believe that investigating the latent manifolds will provide additional insights into the fundamental structures of accounting data and underlying business processes. Furthermore, we aim to evaluate the anomaly detection ability of more recently proposed autoencoder architectures e.g. adversarial autoencoder neural networks.

Given the tremendous amount of journal entries recorded by organizations annually, an automated and high precisions detection of accounting anomalies can save auditors considerable time and decrease the risk of fraudulent financial statements.

The code we used to train and evaluate our models is available at:
https://github.com/GitiHubi/deepAI.

Acknowledgments This work was supported by the NVIDIA AI Lab (NVAIL) program. We thank Adrian Ulges, all members of the Deep Learning Competence Center at the DFKI, as well as, the PwC Europe's Forensic Services team for their comments and support.

References

1. ACFE: Report to the Nations on Occupational Fraud and Abuse, The 2016 Global Fraud Study. Association of Certified Fraud Examiners (ACFE) (2016), https://s3-us-west-2.amazonaws.com/acfepublic/2016-report-to-the-nations.pdf
2. AICPA: Consideration of Fraud in a Financial Statement Audit. American Institute of Certified Public Accountants (AICPA) (2002), https://www.aicpa.org/Research/Standards/AuditAttest/DownloadableDocuments/AU-00316.pdf
3. Amani, F.A., Fadlalla, A.M.: Data mining applications in accounting: A review of the literature and organizing framework. International Journal of Accounting Information Systems **24**, 32–58 (2017)
4. An, J.: Variational Autoencoder based Anomaly Detection using Reconstruction Probability. Tech. rep. (2015)
5. Andrews, J.T.A., Morton, E.J., Griffin, L.D.: Detecting Anomalous Data Using Auto-Encoders. International Journal of Machine Learning and Computing **6**(1), 21–26 (2016)

6. Argyrou, A.: Auditing Journal Entries Using Self-Organizing Map. In: Proceedings of the Eighteenth Americas Conference on Information Systems (AMCIS). pp. 1–10. No. 16, Seattle, Washington (2012)
7. Argyrou, A.: Auditing Journal Entries Using Extreme Vale Theory. Proceedings of the 21st European Conference on Information Systems **1**(2013) (2013)
8. Bay, S., Kumaraswamy, K., Anderle, M.G., Kumar, R., Steier, D.M., Blvd, A., Jose, S.: Large Scale Detection of Irregularities in Accounting Data. In: Data Mining, 2006. ICDM'06. Sixth International Conference on. pp. 75–86. IEEE (2006)
9. Benford, F.: The Law of Anomalous Numbers. Proceedings of the American Philosophical Society **78**(4), 551–572 (1938)
10. Bengio, Y., Yao, L., Alain, G., Vincent, P.: Generalized denoising auto-encoders as generative models. In: Advances in Neural Information Processing Systems, pp. 899–907 (2013)
11. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: Identifying Density-Based Local Outliers. In: Proceedings of the 2000 Acm Sigmod International Conference on Management of Data. pp. 1–12 (2000)
12. Campello, R.J.G.B., Moulavi, D., Sander, J.: Density-Based Clustering Based on Hierarchical Density Estimates. Tech. rep. (2013)
13. Cozzolino, D., Verdoliva, L.: Single-image splicing localization through autoencoder-based anomaly detection. In: 8th IEEE International Workshop on Information Forensics and Security, WIFS 2016. pp. 1–6 (2017)
14. Das, K., Schneider, J.: Detecting anomalous records in categorical datasets. ACM SIGKDD International conference on Knowledge discovery and data mining pp. 220–229 (2007)
15. Dau, H.A., Ciesielski, V., Song, A.: Anomaly Detection Using Replicator Neural Networks Trained on Examples of One Class. In: Asia-Pacific Conference on Simulated Evolution and Learning. pp. 311–322 (2014)
16. D'Avino, D., Cozzolino, D., Poggi, G., Verdoliva, L.: Autoencoder with recurrent neural networks for video forgery detection. arXiv preprint (March), 1–8 (2017), https://arxiv.org/abs/1708.08754
17. Debreceny, R.S., Gray, G.L.: Data mining journal entries for fraud detection: An exploratory study. International Journal of Accounting Information Systems **11**(3), 157–181 (2010)
18. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining. pp. 226–231 (1996)
19. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS) **9**, 249–256 (2010)
20. Hawkins, S., He, H., Williams, G., Baxter, R.: Outlier Detection Using Replicator Neural Networks. In: International Conference on Data Warehousing and Knowledge Discovery. pp. 170–180. No. September, Springer Berlin Heidelberg (2002)
21. Hinton, G.E., Salakhutdinov, R.R.: Reducing the Dimensionality of Data with Neural Networks. Science **313**(5786), 504–507 (2006)
22. IFAC: International Standards on Auditing 240, The Auditor's Responsibilities Relating to Fraud in an Audit of Financial Statements. International Federation of Accountants (IFAC) (2009), http://www.ifac.org/system/files/downloads/a012-2010-iaasb-handbook-isa-240.pdf

23. Islam, A.K., Corney, M., Mohay, G., Clark, A., Bracher, S., Raub, T., Flegel, U.: Fraud detection in ERP systems using Scenario matching. IFIP Advances in Information and Communication Technology **330**, 112–123 (2010)
24. Jans, M., Lybaert, N., Vanhoof, K.: Internal fraud risk reduction: Results of a data mining case study. International Journal of Accounting Information Systems **11**(1), 17–41 (2010)
25. Jans, M., Van Der Werf, J.M., Lybaert, N., Vanhoof, K.: A business process mining application for internal transaction fraud mitigation. Expert Systems with Applications **38**(10), 13351–13359 (2011)
26. Khan, R., Corney, M., Clark, A., Mohay, G.: Transaction Mining for Fraud Detection in ERP Systems. Industrial Engineering and Management Systems **9**(2), 141–156 (2010)
27. Khan, R., Corney, M.: A role mining inspired approach to representing user behaviour in ERP systems. In: Proceedings of The 10th Asia Pacific Industrial Engineering and Management Systems Conference. pp. 2541–2552. No. December (2009)
28. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. arXiv preprint pp. 1–15 (2014), http://arxiv.org/abs/1412.6980
29. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. Advances In Neural Information Processing Systems pp. 1–9 (2012)
30. Lauly, S., Larochelle, H., Khapra, M.: An Autoencoder Approach to Learning Bilingual Word Representations. In: Advances in Neural Information Processing Systems. pp. 1853–1861 (2014)
31. LeCun, Y., Bengio, Y., Hinton, G., Y., L., Y., B., G., H.: Deep learning. Nature **521**(7553), 436–444 (2015)
32. Markovitch, S., Willmott, P.: Accelerating the digitization of business processes. McKinsey & Company pp. 1–5 (2014)
33. McGlohon, M., Bay, S., Anderle, M.G.M., Steier, D.M., Faloutsos, C.: SNARE: A Link Analytic System for Graph Labeling and Risk Detection. Kdd-09: 15Th Acm Sigkdd Conference on Knowledge Discovery and Data Mining pp. 1265–1273 (2009)
34. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 pp. 1–12 (2013)
35. Paula, E.L., Ladeira, M., Carvalho, R.N., Marzagão, T.: Deep learning anomaly detection as suppor fraud investigation in Brazilian exports and anti-money laundering. In: Proceedings - 2016 15th IEEE International Conference on Machine Learning and Applications, ICMLA 2016. pp. 954–960 (2017)
36. Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space. Philosophical Magazine Series 6 **2**(11), 559–572 (1901). https://doi.org/10.1080/14786440109462720
37. Pedregosa, F., Varoquaux, G.: Scikit-learn: Machine learning in Python, vol. 12 (2011). https://doi.org/10.1007/s13398-014-0173-7.2, http://dl.acm.org/citation.cfm?id=2078195
38. Pedrosa, I., Costa, C.J.: New trends on CAATTs: what are the Chartered Accountants' new challenges? ISDOC '14 Proceedings of the International Conference on Information Systems and Design of Communication, May 1617, 2014, Lisbon, Portugal pp. 138–142 (2014)
39. PwC: Adjusting the Lens on Economic Crime, The Global Economic Crime Survey 2016. PricewaterhouseCoopers LLP (2016), https://www.pwc.com/gx/en/economic-crime-survey/pdf/GlobalEconomicCrimeSurvey2016.pdf

40. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representation by error propagation. In: Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations, pp. 318–362 (1986)
41. SAP: SAP Global Corporate Affairs, Corporate Factsheet 2017 (2017), `https://www.sap.com/corporate/de/documents/2017/04/4666ecdd-b67c-0010-82c7-eda71af511fa.html`
42. Scholkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J., Platt, J.: Support Vector Method for Novelty Detection. In: Advances in neural information processing systems. vol. 12, pp. 582–588 (2000)
43. Seow, Poh-Sun; Sun, Gary; Themin, S.: Data Mining Journal Entries for Fraud Detection : a Replication of Debreceny and Gray ' S (2010). Journal of Forensic Investigative Accounting **3**(8), 501–514 (2016)
44. Singleton, T., Singleton, A.J.: Fraud auditing and forensic accounting. John Wiley & Sons (2006)
45. Tóth, L., Gosztolya, G.: Replicator Neural Networks for Outlier Modeling in Segmental Speech Recognition. Advances in Neural NetworksISNN 2004 pp. 996–1001 (2004)
46. Wang, S.: A comprehensive survey of data mining-based accounting-fraud detection research. In: 2010 International Conference on Intelligent Computation Technology and Automation, ICICTA 2010. vol. 1, pp. 50–53 (2010)
47. Wells, J.T.: Corporate Fraud Handbook: Prevention and Detection. John Wiley & Sons (2017)
48. Williams, G., Baxter, R.: A comparative study of RNN for outlier detection in data mining. IEEE International Conference on Data Mining (December 2002), 1–16 (2002)
49. Xu, B., Wang, N., Chen, T., Li, M.: Empirical Evaluation of Rectified Activations in Convolution Network. ICML Deep Learning Workshop pp. 1–5 (2015)
50. Zhai, S., Cheng, Y., Lu, W., Zhang, Z.: Deep Structured Energy Based Models for Anomaly Detection. In: International Conference on Machine Learning. vol. 48, pp. 1100–1109 (2016)
51. Zhou, C.: Anomaly Detection with Robust Deep Auto-encoders. In: KDD '17 Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 665–674 (2017)