Occupational Fraud Detection through Agent-based Data Generation

Julian Tritscher, Alexander Roos, Daniel Schlör, Andreas Hotho, and Anna Krause

University of Würzburg, Am Hubland, 97074 Würzburg, Germany
{tritscher, roos, schloer, hotho,
anna.krause}@informatik.uni-wuerzburg.de

Abstract. Occupational fraud is an increasing concern for enterprises that is estimated to cause losses of around 5% of company revenue each year. With the increasing data tracked by companies through enterprise resource planning systems, recent research has taken interest in the automated detection of occupational fraud. Automated detection is however hindered by the unavailability of labeled fraud cases which require known occupational frauds within company data and costly expert annotation. Even despite the existence of anomaly detection methods that can be trained on unsupervised data, selecting the ideal preprocessing techniques, the most suitable model, and the optimal hyperparameters necessitates the availability of labeled data for evaluation purposes. To alleviate this issue, we propose to use simulation through multi-agent systems for generating business processes according to best practices from economics and creating labeled synthetic data that closely matches a given unlabeled real-world dataset. We extend an existing simulation by incorporating functionality for including, tracking and automatic labeling of occupational fraud cases. Using this simulation, we propose a framework that decides on important design choices for fraud detection models in enterprise resource planning data and does not require labeled real-world data. We demonstrate in multiple experiments that the framework can aid automated occupational fraud detection through data generation.

Keywords: Data generation · Fraud detection · Simulation.

1 Introduction

Occupational fraud describes the deliberate misuse of a company's assets by an employee for their personal enrichment and includes cases such as theft or bribery. This type of fraud is an ongoing issue for companies that causes estimated losses of around 5% of company revenue each year [1]. As many companies track large amounts of data regarding their business in Enterprise Resource Planning (ERP) systems, researchers have turned towards the automated detection of occupational fraud using data-driven machine learning approaches on ERP

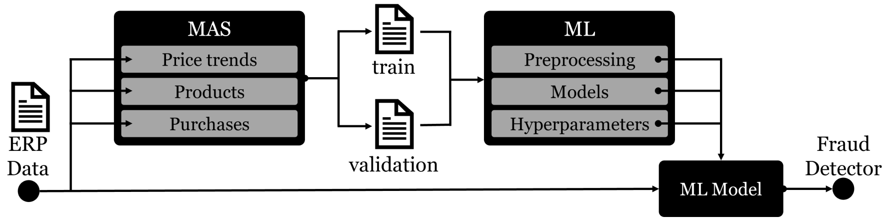


Fig. 1: Framework for automated fraud detection on ERP system data using the MAS to determine design choices for anomaly detectors using market information from the dataset.

data [23,24,33]. As with all machine learning methods, the design of an automated fraud detection system includes multiple data dependent design choices regarding preprocessing, choice of machine learning model, and hyperparameter configuration that are typically made through labeled validation data with known fraud cases and normal behavior. One key issue with automated fraud detection in this domain is that labeled ERP data from companies for validation is not readily available to the public, especially when fraud is known to be included in the data [31]. This does not just limit the progress of open research through limiting reproducibility. Even for research with private data, it poses challenges for employing the introduced methods in practice, since the cleaning and labeling of data requires costly domain experts without any guarantee that suitable examples of fraudulent activities required for validation are present and can be discovered. Therefore the requirement for labeled data is currently an obstacle for companies employing automated fraud detection systems in practice.

In this paper, we explore the use of multi-agent systems (MAS) to generate labeled ERP data with occupational fraud scenarios included. The generation of normal business processes is based on an existing MAS for make-to-stock production and generates data according to best practices from economics [8,9]. We further extend this approach by integrating multiple occupational fraud cases into the simulation, resulting in an MAS that appropriately reacts to the consequences of fraud, in contrast to other data augmentation-based techniques where anomalies are simply added into historic data [14]. Our simulation is adaptable to different market scenarios through provided information on market behavior, either directly or from available unlabeled ERP data. We further propose a framework for the design of automated fraud detection models that operates on ERP data where no real-world fraud labels are available. Our framework, depicted in Figure 1, extracts market trend information from the original unlabeled ERP data and uses our adapted MAS to generate corresponding synthetic data with multiple labeled fraud scenarios. Through the synthetic labeled data, our framework can identify suitable design choices for anomaly detection approaches such as data preprocessing, choice of machine learning model, and hyperparameter configuration that would otherwise have to be set to default values due to the lack of label information. The found design choices can then be used when detecting fraud in the provided ERP data. Multiple experiments show that the proposed framework successfully provides fraud detection models that significantly improve over the use of default models, highlighting the suitability of our method to detect occupational fraud within unlabeled ERP data.[1]

In summary, our contributions are as follows. We (1) extend an existing MAS framework for make-to-stock production to include occupational fraud scenarios for labeled data generation. Using the simulation, we (2) provide a framework for the automated detection of occupational fraud in ERP data. The framework uses the simulation for labeled data generation, which in turn allows the rigorous choice of data preprocessing schemes, detection models, and hyperparameter configurations even when no labels for the real data are available.

The remainder of this paper is structured as follows. Section 2 discusses literature related to our study. Section 3 introduces the MAS simulation for normal and fraudulent data in detail, and gives an overview of our proposed anomaly detection framework. Section 4 showcases the data generated through the simulation. Section 5 includes our experiments for automated detection of occupational fraud in ERP data using our proposed framework. Section 6 concludes the paper.

2 Related Work

Due to the limited availability of ERP system data, multiple research works in the past have focused on data generation for occupational fraud detection in ERP systems. Islam et al. [14] generate fraudulent ERP data for validating their ERP fraud detection approach by randomly adding noise to normal real-world ERP data which results in noisy transactions that are afterwards detected as anomalies. While this enables quick generation of anomalous data, the anomalies do not correspond in any way to fraud. Yannikos et al. [34] propose 3LSPG, a synthetic data generator for occupational fraud detection based on Markov chains. The framework takes input probabilities of possible actions within the system, and subsequently constructs transition probabilities between actions to generate data using a Markov chain for individual processes. Generated action sequences are first converted to an application-specific data format that results from the actions and then shifted in time to prevent easily detectable time patterns that result from regularities in the simulation. In contrast to our work, this framework is based entirely on Markov chains and leverages no established economical processes, making the quality of obtained data highly dependent on transition probabilities that have to be provided as inputs by the user. To generate more realistic data for occupational fraud detection, Baader et al. [3] propose to generate data of the standardized purchase-to-pay business process directly within a real ERP system through carrying out rule-based transactions. Manually crafted rules are created that trigger transactions within the ERP system, and may be used to generate normal and fraudulent data with respect to an initial simulation input that provides the probabilities of each normal and

[1] Code and data are available at https://professor-x.de/erp-fraud-mas

fraudulent activity. While the resulting data more closely mimics real ERP data due to it's direct integration into the ERP system back-end, generation is limited to the purchase-to-pay scenario and requires hand-crafted transaction rules. Additionally, none of the discussed methods provide any code or generated data to the public, hindering the use of these approaches in practice.

Beyond the use of purely automated data generation approaches, some works have explored user interaction for generating ERP fraud detection data. Baader et al. [2] generates fraud cases through user interaction by conducting a serious game where participants attempt to construct and hide fraudulent activities within provided normal ERP data. Tritscher et al. [31] build on the automated data generation approach of Baader et al. [3], and use the serious game ERPSim [15] to generate ERP data with both normal and fraudulent activities with multiple research participants. The authors also propose a gamification approach through a serious game [32], that avoids the need for participants to operate an entire ERP system and aims to discover novel fraud cases through emergent gameplay. In contrast to these works, our approach does not require continuous user input during the simulation, enabling a more cost-efficient data generation process without the need for research participants.

Given the high losses associated with occupational fraud, multiple works have in the past turned to ERP data for automated occupational fraud detection. Approaches may broadly be divided into three categories. Early works focus on statistical and clustering based methods [19,21,27,28,29] for detecting occupational fraud. Subsequently, process mining emerged as a way of detecting fraud by summarizing event data as process graphs and highlighting uncommon transitions within the graphs [2,10,17,22]. More recently, the increasing popularity of artificial intelligence has paved the way for multiple works that use machine learning approaches from the domain of anomaly detection to directly identify fraud cases in transactional data [18,23,24,25,33,35]. While all of these works demonstrate potential for detecting occupational fraud, choosing which model to use in practice, alongside finding suitable data preprocessing schemes and hyperparameter settings is dependent on the available data and requires labeled data for quantitative evaluation.

To this end, our MAS-based framework for occupational fraud detection provides labeled data to select suitable occupational fraud detection models and set important design choices regarding their parameters when no labeled data is available. The simulation itself uses best practices from established economics research to jointly simulate fraud and normal business processes and does not require continuous user input. In contrast to many previous works on data generation, we also provide the simulation code base, the generated data, and our framework implementation to the public.

3 Methodology

In this section, we first describe our methodology for modeling normal business processes of a make-to-stock production company using multi-agent sys-

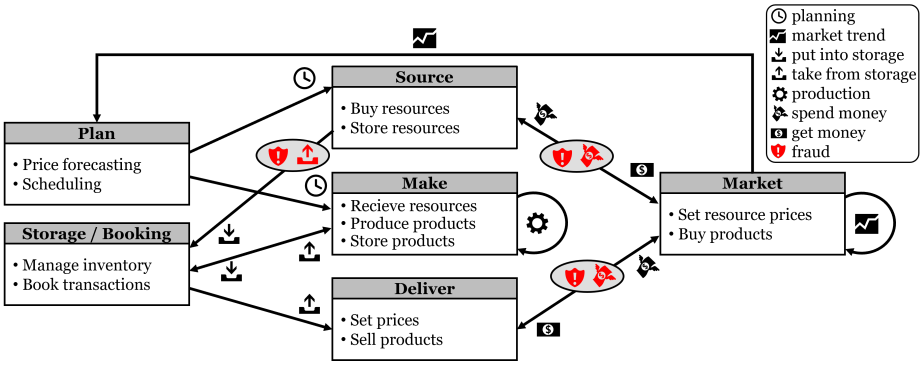


Fig. 2: Visualization of ERP simulation as MAS, following [9] and additionally integrating fraudulent behavior and adding booking functionality for labeled ERP data generation.

tems (MAS), then our approach to integrate occupational fraud into the MAS, and finally how we incorporate the MAS in our framework for simulation-based design choices.

3.1 Modeling normal business processes

In economics, supply chain modeling is the task of modeling the interactions of suppliers, customers, and several internal entities of a company in an economic environment [8]. One technique for modeling supply chains are MAS which use single agents to model the aforementioned entities.

Adapted Simulation using Multi-Agent Systems In this work, we adapt the economical MAS presented by Dominguez et al. [9] and obtain the MAS shown in Figure 2. In the following, we introduce the components of the MAS and discuss and motivate our adaptations. Our MAS has six agents: Market, Plan, Storage/Booking, Source, Make, and Deliver.

The Market agent represents all market participants outside the simulated company, that is suppliers and customers. We merge these into one single agent as this is less computationally expensive overall and the benefits from modeling suppliers and customers separately mainly applies to full supply chain simulations instead of our fraud detection task [8]. The Market agent communicates with the Plan, Source, and Deliver agents. It provides the Plan agent with historical market data for planning. It further accepts purchase offers and sends goods and bills to the Source agent, and sends payments to the Delivery agent from which it receives sales offers.

The Plan agent combines all planning tasks in a company, such as planning for resource acquisition and production, and scheduling the Source and Make agents. While [9] uses individual agents for each planning task, we aggregate their functionality into a single planning agent here to lower computational complexity while still providing the functionality of [9] regarding all planning and scheduling operations.

The Storage/Booking agent constitutes an additional tracking agent that we introduce over the original agents presented in the economical MAS. This tracking agent manages the inventory of the simulated company and books transactions into a simulated ERP system. We specifically make this addition to both allow fraud cases that involve accessing the company storage, and to directly obtain ERP system data from the simulation that is internally collected by the Storage/Booking agent.

The Source agent procures resources for production based on a schedule provided by the Plan agent. To this end, it queries the Market agent for offers and receives resources and bills. We assume, that bills are always received after delivery. The receipt is booked by communicating with the Storage/Booking agent and the purchased goods are added into storage.

The Make agent is in charge of production. It receives its schedule from the Plan agent with information on production quantities for different products. Then resources are retrieved from storage, and their retrieval is booked by the Storage/Booking agent. After production, finished products are stored and booked into the ERP system by communicating with the Storage/Booking agent.

The Deliver agent is in charge of sales and is the only internal agent that is not scheduled by the Plan agent as it is configured to sell daily. It queries the Storage/Booking agent about stored products and costs. It then calculates the sell price from the variable and fixed production costs plus markup. Goods are offered to the Market agent at the calculated price. If the Market agent decides to buy, the Deliver agent receives the order and queries the Storage/Booking agent to realize shipment and booking.

Market simulation To simulate the full business process of a make-to-stock production company, both a supplying market for purchasing materials and a sales market for offering produced products are needed in addition to the agents that represent the company. As the choice and price of products depends on the simulated company, we provide an interface for defining material and product price trends, as well as usual sales amounts for the simulation.

In case price trends are known and readily available, e.g. when continuous real-world historical economic data is available or extensive market research has already been conducted, price information for each business day of a year may be directly supplied to the simulation. If such data is not available, we provide functionality to generate a price trend from prices observed at discrete points in time in a provided unlabeled dataset through interpolation using a Logistic Regression model. This enables companies to generate synthetic data when prices are only known at the specific times of purchases. The price trends resulting from the Logistic Regression model are set as maximum prices. This might, however, introduce a suvivorship bias as only the prices of successful sales are known. To counteract this, we additionally incorporate an artificial markup percentage to allow prices to rise slightly above the observed values. Although materials and products both use these price trends, they are handled differently within the simulation. Materials are assumed as functionally infinite and are implemented as spot market, i.e. the company can buy as many resources as needed at the given price. For products, we employ a simplified trade simulation. The Market agent interprets the provided price trend data as the maximum price at which the market will buy the product. To determine how many products are purchased by the market, the Logistic Regression model may again be fitted to the observed historical sales points, providing an amount of potentially purchaseable products. This amount is again treated as maximum purchasing amount for the market.

Planning and Forecasting For the activities of scheduling production and determining sales prices, agents require functionality for predicting near future price and cost forecasts. For historical market data where all previous prices are available, we use Holt-Winters method [6] to forecast prices, setting a window length of 5 days. The Holt-Winters method applies exponential smoothing separately to the de-trended component, the trend, and the seasonality of a given timeseries. Our implementation of Holt-Winters follows [11], and can therefore only be applied to fully known historical market data without missing values. For all other forecasts, moving average is used, taking all values in a time window and calculating the average.

Our production planning simulation follows the methods introduced in an established economics textbook [5] to maximize company revenue and is applied daily. In economics, this production planning is known as Capacitated Dynamic Lot Sizing Problem (CLSP), a constrained optimization problem [30]. The constraints for the planning problem included in our simulation are maximum storage capacities, simultaneous production capacity, and setup time for switching production to a different product. These storage and production capacities and setup times can be provided by the user prior to simulation start, as this information is commonly available in companies and allows to tailor the simulation to a specific business scenario. Solving the CLSP determines what and how much to produce while taking time and capacity constraints into account. As the number of parameters in our MAS make full CLSP implementations infeasible due to its NP-hard nature and the fact that not all business cases in our simulation can be covered by a single efficient solver [30], we use a non-optimal general heuristic implementation. The heuristic makes the following assumptions: forecasting errors are small, product prices are independent from the produced amount, production time and storage space are constant, and setup time for switching products is irrelevant. These assumptions enable us to reduce the NP-hard problem of general CLSP to a direct selection of the most profitable product to produce, while respecting the remaining constraints of maximum storage capacities and maximum production capacity. While the resulting heuristic provides a simple approach to planning based on common practices from economics, it can also be replaced by alternative strategies that best fit the given company, if such expert knowledge is available.

3.2 Modeling Occupational Fraud

We implement four basic fraud schemes that can occur between different agents: a type of Invoice Kickback, Selling Kickback, Non-Cash Larceny, and Corporate Injury. Kickback describes a type of bribery between an employee and an outside partner. The employee manipulates a payment or process such that the outside partner profits from the manipulation. The employee receives kickback money from the outside partner in return. In our simulation, we include two cases of kickback fraud: an Invoice Kickback and a Selling Kickback. Our Invoice Kickback fraud may occur when the Source agent buys materials from the market. Here, the agent creates a manual purchase request and increases the price of the raw materials, resulting in purchases that are above the currently requested market price which provides a direct benefit to the offering vendor at the expense of company funds. Our Selling Kickback, on the other hand, may occur when the Deliver agent sets selling prices for end products for the market. The Deliver agent changes sales order documents to provide an external vendor with prices that are below the current ask-price of the sold products, damaging the company's revenue for the benefit of the external vendor. In our Non-Cash Larceny case, an employee alters an already payed purchase request to book a smaller amount of raw materials into storage than have been purchased. This allows the fraudster to steal the remaining difference in materials upon delivery. Non-Cash Larceny can occur every time materials are purchased and subsequently booked into storage, making this a fraud case conducted by the Source agent. Finally, Corporate Injury denotes any action that is taken to cause damages to the corporation, without requiring a monetary benefit for the fraudster. We model a case of Corporate Injury, where regular fix costs are arbitrarily increased to cause increased payments by the company.

Each type of fraud is adjustable using two probabilities. An occurrence probability allows for adjusting the frequency of the conducted fraud cases, while an amount probability determines the percentage of stolen goods or reduced profits for the respective fraud cases. The probabilities for each fraud case may be set prior to simulation start.

This simulation of normal and fraudulent behavior and its manifestation in generated synthetic data is then incorporated into our framework allowing design choices of automated fraud detectors to be evaluated even though the real-world dataset at hand is unlabeled.

3.3 Framework for Simulation-based Design Choices

Although there are unsupervised anomaly and fraud detection approaches that do not require labels for training, the evaluation and therefore preprocessing, model and hyperparameter choices require a notion of ground-truth in the form of labels to build the selection process on quantitative metrics. This is especially important with current anomaly detection models that provide large numbers of possible parameter combinations, preventing a feasible manual inspection of possible models. In this scenario, our framework for simulation-based design choices, as depicted in Figure 1, offers a solution when no ground-truth is available. The key for this is that our simulation as described in Section 3.1 closely resembles the normal behavior within the unlabeled data to be examined by following price trends, product selection and purchase behavior. As detailed in Section 3.2, fraudulent activities are modeled within this simulation as well, such that the actual label, i.e. benign or fraudulent activities, are known for each data point produced by the simulation. Based on these labeled simulated datasets, different preprocessing, model selection and hyperparameter choices are evaluated, revealing the optimal model and parameter choices. This optimal parameter set determines the final model that is most promising for the real data and can be employed for unsupervised fraud detection in practice.

By grounding our experiment on ERP data for which ground truth labels are already available, we can experimentally evaluate the proposed framework for simulation-based design choices quantitatively.

4 Data Generation

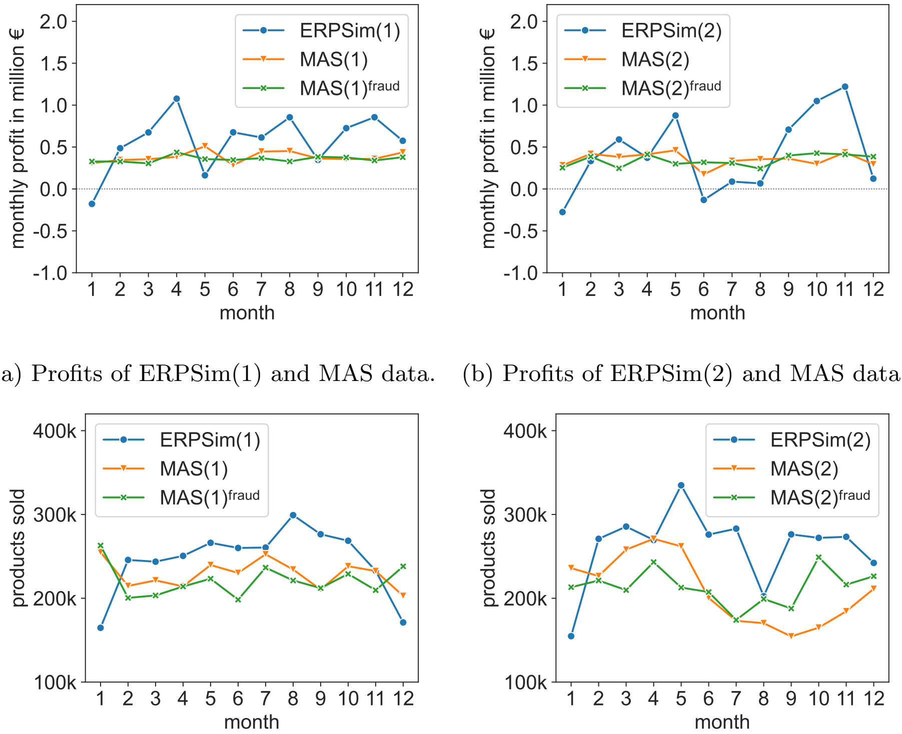
In this section we first outline the data used as real-world dataset for our experiments, describe our simulation setup and briefly analyze the synthetic data generated by our MAS.

4.1 Comparison Data

To conduct reproducible experiments evaluating our MAS, we use the data provided by [31] for direct comparison. While real ERP data is currently not openly available, the provided data is generated through a serious game [4] using a real ERP system and currently constitutes the only openly available ERP data for occupational fraud detection. In this serious game, research participants make production and sales decisions for a cereal production company in a standardized make-to-stock production scenario within a real ERP system instance using a simulated market scenario. The use of a real ERP system allows participants to additionally conduct different fraud cases directly within the ERP system interface, resulting in realistic responses of the ERP system. While multiple years of operation are provided that contain varying amounts and types of fraud cases, we focus on comparing their two more complex fraud datasets fraud2 and fraud3, which we refer to as ERPSim(1) and ERPSim(2) hereafter.

4.2 Simulation Setup

In our evaluation scenario, we assume that only unlabeled company data is available, and that this data might include previously undiscovered fraud cases (we use ERPSim(1) and ERPSim(2) without their label information in this study). We therefore set our MAS parameters according to the business scenario of the supplied ERPSim(1)/ERPSim(2) data. Production and storage capacities, as



(a) Profits of ERPSim(1) and MAS data.

(b) Profits of ERPSim(2) and MAS data.

(c) Sales of ERPSim(1) and MAS data.

(d) Sales of ERPSim(2) and MAS data.

Fig. 3: Achieved profits and product sales in the two ERPSim datasets compared to the respective generated MAS data, showing that MAS data maintains comparable value ranges continuously making profits during simulated production.

well as setup times for products are set according to the ERPSim business scenario [4], while market prices and sales figures are obtained directly through the recorded purchases within the datasets. To achieve continuous price and sales trends for the simulation, we use the interpolation techniques introduced in Section 3.1 to closely mimic the market behavior based on the available data. Using this setup, we generate two synthetic datasets for each provided ERPSim dataset. One generated dataset includes no fraud cases and is used to train fraud detection models on clean data, while one dataset is generated with fraudulent behavior alongside normal business processes to obtain a labeled dataset for choosing a well performing model configuration. To showcase the simulation's ability to generate realistic data, we first compare the generated datasets to the underlying ERPSim data.

4.3 Data Showcase

For each of the two comparison datasets ERPSim(1) and ERPSim(2), we generate one year of company operation through our simulation by first extracting the price trends and purchase behavior from the ERPSim datasets individually and then simulating a run with each of the extracted trends which we label MAS(1) and MAS(2) respectively. We repeat this data generation process for each dataset while enabling the four fraud scenarios described in Section 3.2 within the simulation to obtain the partially fraudulent labeled datasets MAS(1)^{fraud} and MAS(2)^{fraud}. Fraud probabilities were set to 2% for Invoice Kickback, 0.2% for Selling Kickback, 2% for Larceny, and 1% for Corporate Injury, while fraud amounts were set to damages of 75% of the normal transaction. This results in fraud cases that cause notable damage to the company, while providing a fairly even distribution of fraud cases aside from Corporate Injury that is applied only to generally rare fix cost transactions that occur on a weekly basis.

When visualizing the monthly profit obtained throughout the year in Figures 3a and 3b for ERPSim(1) and ERPSim(2) respectively, we observe that the MAS is capable of closely mimicking the underlying ERPSim data for both datasets, achieving consistent profits throughout all months that are in a comparable range to the ERPSim datasets. While the ERPSim data shows stronger fluctuation in profits that result from the company buying resources in large bulks instead of carrying out continuous procurement planning, causing large expenses in some months, the chosen planning and sales strategies within the MAS are capable of achieving similar profits throughout the business year. The used production planning in the MAS also manages to produce similar numbers of products compared to the underlying data, which is highlighted through the products sold per month in Figures 3c and 3d.

An overview of the resulting datasets with the overall number of ERP data transactions and included fraud cases in comparison to the ERPSim datasets is given in Table 1. While the fraud cases contained in the ERPSim data are more diverse and contain multiple different types of Kickback and Larceny frauds

Table 1: Number of transactions and fraud cases of ERPSim and generated MAS data. Note that ERPSim fraud labels are not known during MAS data generation and anomaly detector training and only used for final evaluation.

Dataset	Transactions	Frauds	Invoice Kickback	Selling Kickback	Larceny	Corporate Injury
ERPSim(1)	36778	50	24	0	22	4
MAS(1)	92985	0	0	0	0	0
MAS(1)^{fraud}	93356	223	51	104	66	2
ERPSim(2)	37407	86	30	0	48	8
MAS(2)	59378	0	0	0	0	0
MAS(2)^{fraud}	64858	187	51	102	34	0

compared to the cases included in our MAS, we group these frauds together into the broad categories of Invoice Kickbacks, Larceny and Corporate Injury to provide an overview. Also note that while we can freely access the data labels of the MAS^{fraud} datasets, knowledge of the number and type of fraud cases in the ERPSim data is not used for data generation and detector training, and is only used to verify that fully trained and optimized fraud detectors successfully detect true occupational fraud cases in the data.

5 Fraud Detection through MAS Data

Based on this simulated labeled data, we conduct experiments to asses the benefit of our proposed framework for fraud detection introduced in Section 3.3. We therefore first outline the experimental setup and then present the results in this section.

5.1 Experimental Setup

Our experimental setup follows a scenario where a company only has a single year of real data with unknown labels available. This makes automated fraud detection difficult, as important choices like preprocessing strategies, used models, and hyperparameters can not be explored and evaluated, in spite of their strong impact on detection performance. In our evaluation, we use the ERPSim(1) and ERPSim(2) data described in Section 4.1 separately as substitute for unlabeled real-world company data. We additionally use the MAS datasets generated through our simulation in Section 4.1 by mimicking the general business process contained in the ERPSim data. MAS(1) and MAS(2), which only contain benign data of normal business processes, are used for model training and the partially fraudulent datasets MAS(1)^{fraud} and MAS(2)^{fraud} are used for validation. This synthetic labeled data allows us to validate and select best performing preprocessing steps, detection models, and hyperparameter configurations. We then use these best performing choices to re-train a model on the original ERPSim(1)/ERPSim(2) data in an unsupervised fashion. We finally use the ERPSim(1)/ERPSim(2) labels in this study to show that our synthetic hyperparameter tuning process indeed improves performance in comparison to the default parameter settings of different unsupervised anomaly detection algorithms. Note that the default parameters constitute the only alternative option when no labeled data for judging performance is available. Commonly used default parameter settings are supplied in a well known python library [36] and are used in evaluation scenarios where labeled validation data is unavailable [13].

Anomaly detection design choices In our experimental study, we vary multiple design choices that may impact a machine learning based fraud detection system.

Preprocessing encompasses the preparation of data prior to providing it to the anomaly detection model. In our study, we use one-hot encoding for categorical features and two options for numerical features: *zscore* scaling [20] is

Table 2: Hyperparameter grid with tested parameter sets for each model.

model	hyperparameters
OCSVM	kernel ∈ [rbf], γ ∈ [1e3, 1e2, 1e1, 1e0, 1e-1, 1e-2, 1e-3], ν ∈ [0.2, 0.4, 0.6, 0.8]
AE	neurons ∈ {[32, 16, 8, 16, 32], [64, 32, 16, 32, 64], [128, 64, 32, 64, 128], [64, 32, 16, 8, 16, 32, 64], [128, 64, 32, 16, 32, 64, 128], [256, 128, 64, 32, 64, 128, 256], [128, 64, 32, 16, 8, 16, 32, 64, 128], [256, 128, 64, 32, 16, 32, 64, 128, 256], [512, 256, 128, 64, 32, 64, 128, 256, 512]}, learning rate ∈ [1e-2, 1e-3, 1e-4], batch size ∈ [2048]
IF	trees ∈ [16, 32, 64, 128], max samples ∈ [0.4, 0.6, 0.8, 1.0], max features ∈ [0.4, 0.6, 0.8]

a well-known technique that normalizes feature columns to a mean of zero and a standard deviation of one. *Quantizated* preprocessing [33] converts numerical features into categorical buckets using two outlier buckets that contain the 1% highest and lowest values and then equally distributing the remaining data.

Model choice describes the decision which unsupervised anomaly detection model is used for detecting the occupational fraud cases. Our study includes three well-established machine learning models from the domain of anomaly detection that are capable of training entirely on unlabeled data. *Autoencoder neural networks* (AE) [12] are reconstruction-based anomaly detection methods that encode the input to a lower-dimensional representation and afterwards attempt to reconstruct the original input. *One-class support vector machines* (OCSVM) [26] adapt maximum-margin-based regressors to the one-class setting by learning a separation of observed training data from origin in hyperspace. *Isolation Forests* (IF) [16] are ensemble-based methods that learn a representation of normal data within multiple decision trees that are subsequently used to isolate anomalous data points. All three selected models are popular methods for unsupervised anomaly detection and have in the past been successfully used in occupational fraud detection on ERP system data [23,24,33].

Hyperparameters denote important parameters that are set prior to model training and directly influence the respective model's architecture or training procedure. As the labeled data obtained through our simulation enables optimization of hyperparameters, we conduct a hyperparameter search using the parameter grid reported in Table 2 to identify suitable hyperparameter combinations that allow the model to best detect the synthetic fraud cases contained in the simulated data.

Evaluation metrics To judge the performance of the investigated anomaly detection approaches, we follow the evaluation setting of [33]. We report results using area under the precision recall curve (PR) and area under the receiver operator characteristic (ROC) scores that provide performance metrics for anomaly detection methods without a predefined threshold. While all choices are made regarding the PR score in our heavily unbalanced anomaly detection setting, as PR scores are known to be less sensitive to class imbalance [7], we additionally report ROC scores for completeness. We also repeat all experiments that contain the non-deterministic AE and IF models 5 times with different random seeds and report mean and standard deviations of scores to mitigate statistical fluctuation.

5.2 Results

Results on the ERPSim(1) dataset in Table 3 show that anomaly detection models using default hyperparameters and different data preprocessing schemes achieve strongly varying results. Especially AE achieves very low detection performance using default parameters, which may be attributed to the high sensitivity to hyperparameters that is typical of neural network based models. Using the synthetic labeled data in our proposed framework to set hyperparameters results in large improvements on PR score for all individual models, allowing all models to achieve more than three times the PR score of their default parameter settings. Additionally, the overall best model on the synthetic data, the OCSVM, corresponds to the best model on the real data, managing to find the overall best combination of investigated design choices for the ERPSim(1) dataset.

On the ERPSim(2) dataset in Table 4 default parameters manage to retain higher PR scores compared to ERPSim(1), especially for AE and OCSVM using quantized preprocessing. However, other default parameter configurations especially using zscore preprocessing for AE and OCSVM produce considerably lower scores in comparison. This is especially noteworthy as without labeled data the choice of used model and preprocessing scheme is arbitrary and may result in poor performance. Even still, our proposed optimization scheme using synthetic data manages to considerably improve on the PR scores achieved by all models that use default hyperparameters. While the best observed performance for ERPSim(2) is achieved by the AE optimized through our framework, the best performance on the synthetic data is given by the OCSVM with the AE remaining a close second. Nevertheless, choosing the model according to the synthetic

Table 3: Mean and standard deviation of performance when setting hyperparameters (HP) through labeled synthetic data vs using default HP. Showing PR score on synthetic data that was used for parameter selection, and PR and ROC scores on ERPSim(1), ordered by PR score on ERPSim. Higher is better.

HP choice	model	preprocessing	PR$_{synth}$	PR$_{ERPSim(1)}$	ROC$_{ERPSim(1)}$
synthetic	AE	synthetic	17.5 ± 2.6	26.5 ± 4.3	**99.1 ± 0.2**
	IF	synthetic	3.8 ± 0.1	12.3 ± 0.8	97.5 ± 0.3
	OCSVM	synthetic	**21.5 ± 0.0**	**28.1 ± 0.0**	96.0 ± 0.0
default	AE	quantized	N/A	1.7 ± 0.3	73.0 ± 4.2
	AE	zscore	N/A	1.4 ± 0.2	72.6 ± 7.8
	IF	quantized	N/A	5.3 ± 2.3	98.2 ± 0.3
	IF	zscore	N/A	3.4 ± 0.6	97.4 ± 0.6
	OCSVM	quantized	N/A	8.6 ± 0.0	98.8 ± 0.0
	OCSVM	zscore	N/A	5.2 ± 0.0	97.2 ± 0.0

Table 4: Mean and standard deviation of performance when setting hyperparameters (HP) through labeled synthetic data vs using default HP. Showing PR score on synthetic data that was used for parameter selection, and PR and ROC scores on ERPSim(2), ordered by PR score on ERPSim. Higher is better.

HP choice	model	preprocessing	PR_{synth}	$\mathrm{PR}_{ERPSim(2)}$	$\mathrm{ROC}_{ERPSim(2)}$
synthetic	AE	synthetic	16.6 ± 1.0	**53.7 ± 5.7**	**99.4 ± 0.3**
	IF	synthetic	10.9 ± 0.7	21.3 ± 2.6	98.2 ± 0.3
	OCSVM	synthetic	**17.4 ± 0.0**	38.3 ± 0.0	92.3 ± 0.0
default	AE	quantized	N/A	24.4 ± 13.6	99.2 ± 0.2
	AE	zscore	N/A	8.3 ± 2.3	98.3 ± 0.3
	IF	quantized	N/A	10.5 ± 2.0	98.9 ± 0.2
	IF	zscore	N/A	11.3 ± 4.0	98.7 ± 0.2
	OCSVM	quantized	N/A	23.7 ± 0.0	99.3 ± 0.0
	OCSVM	zscore	N/A	11.0 ± 0.0	98.2 ± 0.0

PR score just like hyperparameters and preprocessing obtains an OCSVM model that strongly outperforms all models that rely on default hyperparameters.

Comparing the results across both ERPSim datasets also shows that default models achieve strongly varying results with no clear ordering of well performing models and preprocessing schemes, suggesting that there is no common well performing choice for the combination of preprocessing, detection model, and default hyperparameters. This again highlights the benefit of our proposed MAS-based framework enabling design choice selection in scenarios where other labeled validation data is unavailable.

6 Conclusion

In this paper, we proposed a framework for detecting occupational fraud in a single unlabeled ERP dataset using multi-agent systems to provide necessary labeled data. We adapted an established MAS from economics research to generate labeled ERP data including fraud cases, and introduced a framework that leverages the simulation to select preprocessing schemes, fraud detection models and hyperparameters for fraud detection on a given unlabeled dataset. We showed the suitability of our MAS by comparing the given unlabeled data to data generated through the simulation, and showcased the potential of our proposed framework for detecting occupational fraud on two ERP datasets when no real-world label information is available.

Despite the promising performance of our proposed framework, the presented work currently only contains a general proof of concept with basic economic procedures and limited number of fraud cases included in the simulation. These may be extended in future work to provide more realistic data for setting common design choices in machine learning models and obtain a more realistic data generator for other tasks based on ERP data.

References

1. ACFE: Occupational Fraud 2022: A Report to the nations. Report To the Nations (2022), https://legacy.acfe.com/report-to-the-nations/2022/, [Online; accessed 18. Jul. 2023]
2. Baader, G., Krcmar, H.: Reducing false positives in fraud detection: Combining the red flag approach with process mining. Int. Journal of Accounting Information Systems **31**, 1–16 (2018)
3. Baader, G., Meyer, R., Wagner, C., Krcmar, H.: Specification and Implementation of a Data Generator to Simulate Fraudulent User Behavior. In: Business Information Systems. pp. 67–78. Lecture Notes in Business Information Processing, Springer International Publishing, Cham (2016)
4. Babin, G., Léger, P.M., Robert, J., Bourdeau, S.: ERPsim BI: A Problem-based Learning approach in Teaching Business Analytics. The Proceedings of DYNAA **2** (2011)
5. Bloech, J., Bogaschewsky, R., Buscher, U., Daub, A., Götze, U., Roland, F.: Einführung in die Produktion. Springer-Lehrbuch, Springer, Berlin, Heidelberg (2014)
6. Chatfield, C.: The Holt-Winters Forecasting Procedure. Journal of the Royal Statistical Society. Series C (Applied Statistics) **27**(3), 264–279 (1978)
7. Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings of the 23rd int. conf. on Machine learning. pp. 233–240 (2006)
8. Dominguez, R., Cannella, S.: Insights on Multi-Agent Systems Applications for Supply Chain Management. Sustainability **12**(5), 1935 (Jan 2020)
9. Domínguez, R., Cannella, S., Framinan, J.M.: SCOPE: A Multi-Agent system tool for supply chain network analysis. In: IEEE EUROCON 2015 - International Conference on Computer as a Tool (EUROCON). pp. 1–5 (Sep 2015)
10. Febriyanti, K.D., Sarno, R., Effendi, Y.A.: Fraud detection on event logs using fuzzy association rule learning. In: 2017 11th International Conference on Information & Communication Technology and System (ICTS), pp. 149–154. IEEE (10 2017). https://doi.org/10.1109/ICTS.2017.8265661
11. Frank, E., Hall, M.A., Witten, I.H.: The weka workbench. online appendix for ”data mining: Practical machine learning tools and techniques” (2016)
12. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org
13. Han, S., Hu, X., Huang, H., Jiang, M., Zhao, Y.: ADBench: Anomaly Detection Benchmark. In: Thirty-Sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Sep 2022)
14. Islam, A.K., Corney, M.W., Mohay, G.M., Clark, A.J., Bracher, S., Tobias, R., Flegel, U.: Fraud detection in ERP systems using scenario matching. In: International Information Security Conference (SEC 2010) : Security and Privacy : Silver Linings in the Cloud. pp. 112–123. Springer, Brisbane Convention & Exhibition Centre, Brisbane, Queensland (Sep 2010)
15. Léger, P.M.: Using a Simulation Game Approach to Teach ERP Concepts. Journal of Information Systems Education **17**, 441–447 (Jul 2006)
16. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: 2008 Eighth IEEE Int. Conf. on Data Mining. pp. 413–422. IEEE (2008)
17. Naufal, M.F.: Fraud detection using process mining and analytical hierarchy process with verification rules on erp business process. In: International Conference on Informatics, Technology, and Engineering (InCITE)-2nd. (2019)

18. Nonnenmacher, J., Kruse, F., Schumann, G., Gómez, J.M.: Using autoencoders for data-driven analysis in internal auditing. In: Proceedings of the 54th Hawaii Int. Conf. on System Sciences (2021)
19. Oliverio, W.F.M., Silva, A.B., Rigo, S.J., da Costa, R.L.B.: A Hybrid Model for Fraud Detection on Purchase Orders. In: Intelligent Data Engineering and Automated Learning – IDEAL 2019, pp. 110–120. Springer, Cham, Switzerland (10 2019). https://doi.org/10.1007/978-3-030-33607-3_13
20. Patro, S., Sahu, K.K.: Normalization: A preprocessing stage. arXiv preprint arXiv:1503.06462 (2015)
21. Sabau, A.S.: Survey of clustering based financial fraud detection research. Informatica Economica **16**(1), 110 (2012)
22. Sarno, R., Dewandono, R.D., Ahmad, T., Naufal, M.F., Sinaga, F.: Hybrid association rule learning and process mining for fraud detection. IAENG International Journal of Computer Science **42**(2) (2015)
23. Schreyer, M., Sattarov, T., Borth, D., Dengel, A., Reimer, B.: Detection of anomalies in large scale accounting data using deep autoencoder networks. arXiv preprint arXiv:1709.05254 (2017)
24. Schreyer, M., Sattarov, T., Schulze, C., Reimer, B., Borth, D.: Detection of accounting anomalies in the latent space using adversarial autoencoder neural networks. In: 2nd KDD Workshop on Anomaly Detection in Finance. ACM (2019)
25. Schultz, M., Tropmann-Frick, M.: Autoencoder neural networks versus external auditors: Detecting unusual journal entries in financial statement audits. In: Proceedings of the 53rd Hawaii Int. Conf. on System Sciences (2020)
26. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Computation **13**(7), 1443–1471 (Jul 2001)
27. Singh, K., Best, P.: Interactive visual analysis of anomalous accounts payable transactions in sap enterprise systems. Managerial Auditing Journal (2016)
28. Singh, K., Best, P., Mula, J.: Automating vendor fraud detection in enterprise systems. Journal of Digital Forensics, Security and Law **8**(2), 1 (2013)
29. Singh, K., Best, P., Mula, J.M.: Proactive fraud detection in enterprise systems. In: Proceedings of the 2nd International Conference on Business and Information: Steering Excellence of Business Knowledge (ICBI 2011). University of Kelaniya, Faculty of Commerce and Management Studies (2011)
30. Suwondo, E., Yuliando, H.: DYNAMIC LOT-SIZING PROBLEMS: A Review on Model and Efficient Algorithm. Agroindustrial Journal **1**(1), 36 (May 2017)
31. Tritscher, J., Gwinner, F., Schlör, D., Krause, A., Hotho, A.: Open ERP System Data For Occupational Fraud Detection (Jul 2022)
32. Tritscher, J., Krause, A., Schlör, D., Gwinner, F., Von Mammen, S., Hotho, A.: A financial game with opportunities for fraud. In: 2021 IEEE Conference on Games (CoG). pp. 1–5 (Aug 2021)
33. Tritscher, J., Schlör, D., Gwinner, F., Krause, A., Hotho, A.: Towards Explainable Occupational Fraud Detection. In: Machine Learning and Principles and Practice of Knowledge Discovery in Databases. pp. 79–96. Communications in Computer and Information Science, Springer Nature Switzerland, Cham (2023)
34. Yannikos, Y., Franke, F., Winter, C., Schneider, M.: 3LSPG: Forensic Tool Evaluation by Three Layer Stochastic Process-Based Generation of Data. In: Computational Forensics. pp. 200–211. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2011)

35. Yu, J., Oh, H., Kim, M., Jung, S.: Unusual insider behaviour detection framework on enterprise resource planning systems using adversarial recurrent auto-encoder. IEEE Transactions on Industrial Informatics (2021)
36. Zhao, Y., Nasrullah, Z., Li, Z.: Pyod: A python toolbox for scalable outlier detection. Journal of Machine Learning Research **20**(96), 1–7 (2019), http://jmlr.org/papers/v20/19-011.html