# Auto-encoder Based Medicare Fraud Detection

Mansour Zoubeirou a Mayaki, Michel Riveill

*4rd International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2022), 19-21 October 2022, Corfu, Greece*

**(7010)**

# Auto-encoder Based Medicare Fraud Detection

**Mansour Zoubeirou A Mayaki and Michel Riveill**
University Côte d'Azur, CNRS, INRIA, 2004 Rte des Lucioles, 06902 Valbonne, France
E-mail: mansour.zoubeirou-a-mayaki@inria.fr

**Summary:** In this study, we used deep learning based multiple inputs classifier with a Long-short Term Memory (LSTM) autoencoder component to detect medicare fraud. The proposed model is made of two separate blocks: MLP block and auto encoder feature extraction block. The MLP block extracts high level feature from the invoice data and the auto encoder block extracts high level features from that describes the provider behavior over time. This architecture makes it possible to take into account many sources of data without mixing them. The latent features extracted from the LSTM autoencoder have a strong discriminating power and separate the providers into homogeneous clusters. We use the data sets from the Centers for Medicaid and medicare Services (CMS) of the US federal government. Our results show that baseline artificial neural network give good performances compared to classical machine learning models but they are outperformed by our model.

**Keywords:** Medicare fraud, Anomaly detection, Deep learning, Auto encoder, Machine learning.

## 1. Introduction

Insurance fraud results in considerable losses for governments and insurance companies and results in higher premiums from clients. In the European Union, fraud would cost 13 billion per year to European customers and insurance companies [1]. In France for example, over 260 million of medicare fraud transactions are detected each year, mainly due to medicare providers and institutions. In the United States, medicare fraud represents 5-10 % of medicare claims and costs insurance companies between 21 billion and 71 billion per year.

Traditionally, business rule-based systems are used for fraud detection. These methods, although effective, are often very difficult to set up and maintain. Indeed, a rule-based fraud detection system constantly requires the presence of experts in the field and constant updates of the rules. Models based on machine learning make it possible to automatically build patterns and thus detect fraudulent behavior effectively. The main challenge when using the methods in fraud detection is that there not enough data labeled as fraudulent, this leads to an imbalance situation. In the field of medicare, fraudulent transactions represent less than 5 % of all transactions. This high imbalance ratio makes it very difficult for machine learning algorithms to learn as they will tend to favor the majority class.

To detect medicare fraud, we propose a multiple inputs deep neural networks model with a Long-short Term Memory (LSTM) autoencoder component. We call this architecture AE-MFD for Auto Encoder based medicare fraud detection method. This architecture makes it possible to take into account many sources of data without mixing them and makes the classification task easier for the final model. The LSTM autoencoder component plays a dimension reduction role for the provider data and its latent vector describes the provider behavior over time.

We use the publicly available medicare data sets from the Centers for medicare and Medicaid Services (CMS) of the United States federal government for period 2017-2019 [2]. The CMS data sets contain the hospitalization requests, the outpatient care requests etc.

The rest of the paper is outline as follows. The **Related Works** section discusses the other studies related to imbalance data handling, deep learning for anomaly and medicare fraud detection. In the third section, we describe our methodology, the model's architecture and the choice of hyperparameters. Section **Experimental Data Sets** we describe the experimental data sets and pre-processing steps. The results are presented and discussed in the last section.

## 2. Related Works

To deal with class imbalance issue, there are two main approaches with varying performance depending on the field of application and the complexity of the problem: the resampling approach (or data level) which consists in balancing the classes by adding or removing data from classes and the approach which consists in modifying the learning algorithms so that they take into account the class imbalance (algorithm level).

The Centers for medicare and Medicaid Services (CMS) data has been used in numerous studies to detect medicare fraud. Most of these studies use resampling techniques to overcome the imbalance class issue (Bauder et al. [3] Liu et al. [4], Herland et al. [5]; Johnson et al. [6], Van et al. [7]). In [5], the authors combined also the three parts of CMS data set and showed that it leads to better performance. They used logistic regression, random forest and gradient boosting classifiers to detect fraudster providers in the CMS. Their results show that the performance of all classifiers improves significantly when they used all

parts of the CMS data. In [6], Johnson et al. the CMS data over the period 2012-2016. They used neural network models with algorithm level and data level techniques to predict medicare fraud medicare. They tested random undersampling (RUS), random oversampling (ROS), mean square error (MSE) and Focal Loss techniques among others. Their results suggest that multi layers perceptrons (MLP) classifiers combined with ROS outperforms all other models. They also noted that RUS can improve model's performance up to certain level of imbalance ratio. Thus, RUS improves the performance if the majority class is above 99 %. Lin et al. [10] rewrite the classical entropy loss function by integrating two new parameters: $\alpha$ and $\gamma$ (gamma). They called the new loss function Focal Loss. The focal lost consist of multiplying the classical cross entropy (CE) by a modulation factor $\alpha(1 - p)\gamma$. Hyper parameter $\gamma \geq 0$ adjusts the rate at which easy examples are down weighted and $\alpha \geq 0$ is a class-wise weight used to give more importance to the minority class [11]. Wang et al. [9] proposed to use a cost matrix with an artificial neural network-based model to predict readmission of patients to a hospital. The cost matrix is defined such that the cost of misclassified readmission is greater than that of misclassified non-readmission. During back propagation, the model penalizes more or pay more attention to the readmission class which is the class of interest. Jason et al. [7], compared different resampling techniques using 11 types of classifiers and 35 different data sets. The imbalance ratio of the experimental data sets varies between 1.33 % and 35 %. The resampling techniques used in this article are: random undersampling (RUS), random oversampling (ROS), one-sided selection (OSS), cluster-based oversampling (CBOS), Wilson's editing (WE), SMOTE (SM) and borderline-SMOTE (BSM). Their results show that RUS tends to give better performance (32 % of the time).

Most of data level experiments studies come to the conclusion that undersampling gives belter results than over-sampling. This goes against what one might have expected as undersampling often leads to a loss of information. One possible explanation is that in cases, adding new artificial data brings more noise than useful information to the model. Algorithm level methods often give better results than resampling methods as they don't alter the training data and don't lead to a loss of information. However, in some cases, when labeled data is limited, oversampling techniques are good way to extend the data set. Moreover, when the majority class distribution is stationary (the samples are very close to each other) undersampling may work very well as we don't lose lot of information by deleting some samples.

## 3. Methodology

In this section, we present the AE-MFD model architecture and the other classifiers we tested. We refer to the baseline neural network as MLP}.

### 3.1. Baseline Classifiers

We compared AE-MFD to baseline Multi-Layer Perceptrons (MLP) networks and state-of-the-art classifiers such as logistic regression (LR), random forest (RF), gradient boosting (GB) and XGBoost. These classifiers are good baseline models for classification tasks. They take an invoice as input and predicts if it's fraud or not. The Multi-Layer Perceptrons (MLP) model consists of a single input layer, multiple hidden layers, and an output layer. This model takes an invoice as input and predicts if it's fraud or not. The number of layers and the number of neurons in each layer of the MLP model are variables (hyper-parameters) that must be chosen carefully for neural network models to give good results. These variables remain constant throughout the training process and have a direct impact on the performance of the models. The choice of hyper-parameters is described in Subsection 3.4.

### 3.2. AE-MFD Model's Architecture

AE-MFD is made up of two different inputs layers. The MLP part input layer receives the claims details and the auto encoder component input layer receives the data relating to the healthcare provider. The model is thus composed of two blocks which meet at the end. Each block consists of an input layer, hidden layers and an output layer. The outputs of the two blocks are then concatenated to form a single vector. Such an architecture makes it possible to simultaneously take into account the details of claims and the healthcare provider behavior separately.

In our version of the multi-input model, the second block is a Long-short Term Memory (LSTM) autoencoder. We first trained the LSTM autoencoder on the provider level data. This autoencoder learns to reconstitute a healthcare provider behavior over time. Then we used the latent vector from the LSTM autoencoder as an input vector for our final model. The final model is thus composed of an input layer which takes as input the claims details and another input layer which makes it possible to inject the latent vectors coming from the autoencoder. In this architecture, the autoencoder plays a dimension reduction role for the provider data and its latent vector describes the healthcare provider behavior. Note that the autoencoder parameters remain constant when learning the final model.

### 3.3. Performance Metrics

The classifiers are evaluated using the precision-recall curve (PRC). This plot shows precision values for corresponding recall values. It provides a model-wide evaluation like the receiver operating characteristic (ROC) plot or the cost curve (CC). The area under the curve (AUC) score of precision-recall curve, denoted as AUC (PRC), is

likewise effective in multiple-classifier comparisons [11]. The AUC (PRC) measures the entire two-dimensional area under the entire precision-recall curve (by integral calculations) from (0,0) to (1, 1). We don't use AUC (ROC) as most studies do because as Saito et al. [11] show in their study, the AUC (ROC) is not well suited in case of imbalance class. They proved that AUC (ROC) could be misleading when applied in imbalanced classification scenarios instead AUC (PRC) should be used. Their study showed via multiple simulations that AUC (ROC) fails to capture the variation in class distribution contrary to the AUC (PRC).

As the AUC (ROC) is used as performance metric in most studies in the literature, we will give its value for each of our classifiers just as an indication. In order to have more detail on classifiers performance, we also compute the precision and the G-means score. The precision gives the performance of the classifier on the positive class and the G-mean metric makes a compromise between the true positive rate TPR or recall and the true negative rate (TNR).

### 3.4. Hyperparameters Optimization

We used the mini-batch stochastic gradient descent (SGD) with a batch size of 200. We used the version called SGD Adam which allows to adapt the step of the gradient during the training of the model. This optimizer is known for its better performance compared to other versions of SGD. We kept the default values of the other hyper parameters: $lr = 0.001$, $\beta_1 = 0.9$ and $\beta_1 = 0.999$. The rectified linear unit (ReLU) activation function is used in neurons of the hidden layers, and the sigmoid activation function is used for output layer to estimate posterior probabilities. We choose the best hyper parameters using the KerasTuner library and on a small holdout set (20 %) of the validation dataset. KerasTuner is an easy-to-use, scalable hyperparameter optimization framework that solves the pain points of hyperparameter search [12]. We also added between the hidden layers a Batch Normalization layer followed by a dropout layer. Batch Normalization makes artificial neural network faster and more stable by normalizing and rescaling layers inputs. Dropout consists in "deactivating" randomly some neurons during training [13]. Each neuron being possibly inactive during a learning iteration, this forces each unit to "learn well" independently of the others and thus avoid overfitting.

## 4. Experimental Data Sets

In order to evaluate our method's capabilities, we compared its performance to those of four state-of-the-art classifiers on four other publicly available bench mark data sets. The data sets are highly imbalanced and present some big data properties. Note that we put more emphasis on the CMS data as our primary focus is medicare fraud detection.

The Centers for medicare and Medicaid Services (CMS) publishes each year a series of publicly available data containing information on the use and payments of medical procedures, services and prescription drugs provided to beneficiaries as well as data on physicians and other actors in the healthcare system. These CMS data combined with the Office of Inspector General's list of excluded individuals and entities (LEIE) [14] containing the list of healthcare providers excluded from the healthcare system for illegal activity allows researchers in the field of statistics and artificial intelligence to propose new methods to fight against fraud in medicare.

The CMS data sets contains mainly two types of information: hospitalization requests (Inpatient Data) and outpatient care requests (Outpatient Data). The Inpatient Data contains information on patients admitted to hospitals. The Outpatient Data gathers information on patients who have visited the hospital without being hospitalized there. We thus have information such as the unique identifier of the healthcare provider (NPI), the refunded amount (AmtReimbursed) etc. Records within the data set also contain various provider-level attributes, e.g., National Provider Identifier (NPI), first and last name, gender, credentials, address etc.

Note that for a fraudster provider, we do not know which of its claims are fraudulent and which are legitimate. To overcome this label issue, we considered that if a provider is fraudster, all its claims are also fraudulent [6]. This assumption makes sense because if a provider has been declared as a fraudster, the decision certainly comes from a deep analysis of his recent activity and his claims reflect illegal activities. We created additional features for the providers by aggregating the variables at the invoice level. For each provider we created new variables by taking the mean, the variance, the sum, the skewness coefficient of the numerical variables per trimester. In order to capture the provider behavior over time, we use a slicing technique called bucketing [9]. The behavior of each provider with respect to each variable can be considered as a time series. Indeed, over the years the provider makes several claims for different patients. For example, we can calculate the total amount paid by the insurance company to the provider each month. The bucketing technique consists of separating the claims from each provider into groups according to time and aggregation indicators are calculated in each group. There are two possible levels of aggregation: first order (first order features) and second order (second order features). The first order indicators are mean, standard deviation, variance, sum, maximum, minimum, skewness and kurtosis. The second order indicators are: Energy (E1), Entropy (E2), Correlation ($\rho_{x,y}$), Inertia and Local Homogeneity (LH). In this study, we only calculate the first order indicators for each numeric variable per trimester. The data of each provider over each year is separated into 4 blocks. In each block, the aggregation

variables are calculated: mean, standard deviation, skewness, maximum, minimum, sum etc. After cleaning and preprocessing, the final data set has a fraud rate of 0.5 %. Table 1 shows a subset of the provider level data and Table 2 a subset of the LEIE data.

**Table 1.** Subset of the aggregated data set on provider level.

| NPI | BeneID count | Deductible AmtPaid mean | InscClaimAmt Reimbursed sum | Fraud |
|---|---|---|---|---|
| 1000051001 | 24 | 213.60 | 104640 | No |
| 1000051101 | 117 | 502.16 | 605670 | Yes |
| 1000051005 | 138 | 2.08 | 52170 | No |
| 1000051007 | 58 | 45.33 | 33710 | No |
| 1000051009 | 36 | 53.86 | 35630 | No |

## 5. Results and Discussions

The classifiers performances are listed in Table 3. Recall that we refer to the baseline neural network as MLP and our Auto encoder model as AE-MFD. MLP weighted stands for MLP with weighted loss, MLP focal with focal loss, MLP mfe with the mean false error loss and MLP rus the best MLP obtained by random under sampling. Despite the class imbalance in the training data, AE-MFD outperform all other classifiers in terms of AUC (PRC). Our model's AUC (PRC) is 0.765 and that of the second-best classifier is 0.741. Note that the no skill} (random) classifier has an AUC (PCR) of 0.03. Using the ROC (AUC) as performance metric, the baseline neural network with mean false error function (MLP mfe) has the best performance (0.864) but it has a very low precision (0.445) compared to our model (0.77). This is due to the fact that state-of-the-art classifiers (logistic regression, random forest, Gradient boosting and XGBoost) they fail to capture complex structures in sequence datasets and large-scale data [15]. As context matters in fraud detection, the advantage of AE-MFD is that the autoencoder separates the providers into homogeneous groups and creates contextual features. The main disadvantage of our method is that it requires lot of historical data to train the LSTM auto encoder. Thus, the final model's performance depends on the auto encoder performance.

## 5. Conclusion

Frauds or anomalies are very rare events but results in considerable losses for governments, insurance companies and taxpayers. In this study, we proposed a deep neural network with auto encoder to detect medicare fraud. We also tested some classical classifiers (random forest, logistic regression, gradient boosting) and simple MLP models. We use the publicly available medicare data sets from the Centers for medicare and Medicaid Services (CMS) of the United States federal government. The results of our experiments show that this kind of architecture outperforms a classical machine learning models and multi-layer perceptron models using a single input layer. The Long-short Term Memory (LSTM) autoencoder component learns high level contextual features from the input data. In addition, the capability of the LSTM auto encoder to extract strong discriminating latent features makes the model robust toward the imbalance class issue. Future work will include employing the multiple inputs models with data sampling techniques or algorithm level techniques to combat the imbalanced nature of the data.

**Table 2.** Subset of LEIE data set.

| NPI | CITY | STATE | EXCLTYPE | EXCLDATE |
|---|---|---|---|---|
| 1306111111 | GARDEN CITY | NY | 1128a1 | 20180220 |
| 1306111111 | WARREN | OH | 1128b5 | 20190220 |
| 1306111111 | PHILADELPHIA | PA | 1128b7 | 20191231 |
| 1306111111 | FLUSHING | NY | 1128a1 | 20190220 |
| 1306111111 | SPRINGFIELD | MO | 1128b4 | 20200220 |

**Table 3.** Experimental results of the proposed method and some state-of-the-art methods. Mean time refers to the execution time expressed in minutes.

| Models | Precision | AUC(ROC) | Gmean | AUC (PRC) | Mean Time |
|---|---|---|---|---|---|
| Random | - | 0 | 0.5 | 0.03 | - |
| LR | 0.438 | 0.827 | 0.826 | 0.629 | 0.51 |
| RF | 0.599 | 0.807 | 0.796 | 0.658 | 2.35 |
| GB | 0.713 | 0.715 | 0.666 | 0.617 | 15.43 |
| XGBoost | 0.707 | 0.713 | 0.663 | 0.613 | 0.53 |
| MLP | 0.436 | 0.863 | 0.862 | 0.739 | 2.54 |
| MLP weighted | 0.429 | 0.860 | 0.859 | 0.733 | 2.96 |
| MLP focal | 0.432 | 0.861 | 0.861 | 0.737 | 2.50 |
| MLP mfe | 0.445 | 0.864 | 0.863 | 0.741 | 6.03 |
| MLP RUS | 0.534 | 0.720 | 0.668 | 0.512 | 1.20 |
| **AE-MFD (Ours)** | **0.770** | **0.794** | 0.762 | **0.765** | 10 |

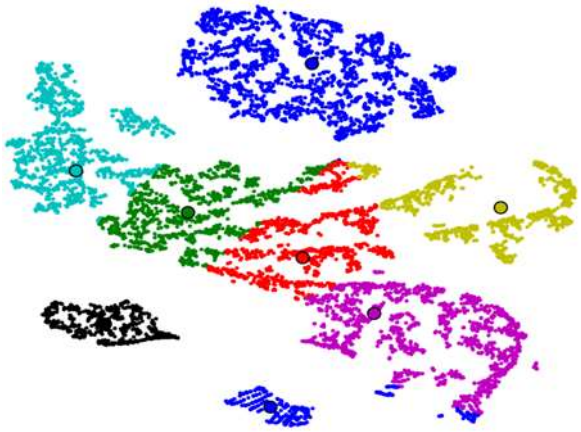

**Fig. 1.** Mean-shift clustering on the autoencoder latent vector. This output of the autoencoder separates the providers into homogeneous groups.

## References


[1]. Fraud Prevention, https://www.insuranceeurope.eu/priorities/23/fraud-prevention

[2]. US Government, Centers for Medicare, Medicaid Services. Medicare Fee-for-service Provider Utilization, Payment Data Physician and Other Supplier Public Use File: A Methodological Overview, https://www.cms.gov/research-statistics-data-and-systems/statistics-trends-and-reports/medicare-provider-charge-data/physician-and-other-supplier

[3]. R. A. Bauder, T. M. Khoshgoftaar, The detection of medicare fraud using machine learning methods with excluded provider labels, in *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA'18)*, 2018, pp. 858-865.

[4]. Q. Liu, M. Vasarhelyi, Healthcare fraud detection: A survey and a clustering model incorporating geo-location information, in *Proceedings of the 29th World Continuous Auditing and Reporting Symposium (WCARS'13)*, Brisbane, Australia, 2013.

[5]. M. Herland, T. M. Khoshgoftaar, R. A. Bauder, Big data fraud detection using multiple medicare data sources, *Journal of Big Data*, Vol. 5, 2018, 29.

[6]. J. M. Johnson, T. M. Khoshgoftaar, Medicare fraud detection using neural networks, *Journal of Big Data*, Vol. 6, 2019, 63.

[7]. J. Van Hulse, T. M. Khoshgoftaar, A. Napolitano, Experimental perspectives on learning from imbalanced data, in *Proceedings of the 24th International Conference on Machine learning (ICML'07)*, 2007, pp. 935-942.

[8]. S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, P. J. Kennedy, Training deep neural networks on imbalanced data sets, in *Proceedings of the International Joint Conference on Neural Networks (IJCNN'16)*, 2016, pp. 4368-4374.

[9]. H. Wang, Z. Cui, Y. Chen, M. Avidan, A. B. Abdallah, A. Kronzer, Predicting hospital readmission via cost-sensitive deep learning, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2018, Vol. 15, pp. 1968-1978.

[10]. T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV'17)*, 2017.

[11]. T. Saito, M. Rehmsmeier, The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets, *PloS One*, Vol. 10, 2015, e0118432.

[12]. T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al., Keras tuner, *Retrieved May*, Vol. 21, 2019.

[13]. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, Vol. 15, 2014, pp. 1929-1958.

[14]. US Government, Office of Inspector General. List of Excluded Individuals and Entities, https://oig.hhs.gov/exclusions/authorities.asp

[15]. R. Chalapathy, S. Chawla, Deep learning for anomaly detection: A survey, *arXiv Preprint*, 2019, arXiv:1901.03407.

[16]. J. M. Johnson, T. M. Khoshgoftaar, Survey on deep learning with class imbalance, *Journal of Big Data*, Vol. 6, 2019, 27.

[17]. Bilan 2018 des actions de lutte contre la fraude et actions de contrôles, https://www.ameli.fr/sites/default/files/2019-10-01-dp-controles-fraudes.pdf