International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 07 | July - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Autoencoders for Anomaly Detection Based on Isolation Forest Algorithm

Vijayalakshmi N[1], Thanuja J C[2]

[1]*Student, Department of MCA, Bangalore Institute of Technology, Karnataka, India*

[2]*Professor, Department of MCA, Bangalore Institute of Technology, Karnataka, India*

ABSTRACT: With the increased usage of credit cards in this, credit card fraud and the loss to credit card users have increased. Supervised learning has is widely been used to discover anomalies in records of credit card transactions under the presumption that the pattern of fraud would depend on past transactions. Unsupervised learning does not undermine the fact that fraudsters change their approaches from time to time according to customers' behaviors and patterns. Anomaly detection has become the most critical aspects within several applications, involving the security of networks, fraud detection, and quality control. Several conventional methods failed to find an anomaly in nature accurately within these complex and high-dimensional data. The present study is aimed at finding a better approach in such anomaly detection by integrating the autoencoder neural network with the Isolation Forest algorithm.

Key Words: Anomaly Detection, Isolation Forest Algorithm, AutoEncoders, Reconstruction error.

1. INTRODUCTION

One of the many variants of neural networks, autoencoders are specialized in learning representations of input data in an unsupervised way. They work by compressing an incoming input onto a latent space from which it can reconstruct with minimal loss. As such, this architecture is good at capturing the underlying patterns and structures within the data. When trained on normal data, the reconstruction error of an autoencoder is higher for the anomalous data, thus giving a robust anomaly detection method. Unlike methods that isolate anomalies based on profiling normal points, isolation forest is a tree-based ensemble method. The algorithm builds trees by randomly selecting a feature and subsequently a split value in between that feature's minimum and maximum values. Anomalies differ and are sparse; therefore, they will be efficiently isolated and are usually characterized with a path of a shorter length in the tree structure. In the present paper, we introduce another way in which Autoencoders and Isolation Forest benefit from each other.

First, the autoencoder is trained to reconstruct data. Then, the reconstruction error features are utilized by the isolation forest algorithm. This will make it very effective in identifying anomalies, for it will capture the linear and nonlinear relationship within this data.

2. LITERATURE SURVEY

Seminal work by Aggarwal about challenges of anomaly detection in high-dimensional data. He underlines here the actuality that distance measures become meaningless with high dimensionality; hence meaningful processes are hard to take place[1]. Chalapathy and Chawla survey many deep learning techniques applied to anomaly detection. Their procedures are divided into supervised, semi-supervised, and unsupervised learning frameworks[2]. The paper by Liu, Ting, and Zhou presents a new algorithm for anomaly detection: Isolation Forest. Traditional approaches typically profile normal data points, while Isolation Forest isolates anomalies by partitioning data using random splits[3]. Sakurada and Yairi use autoencoders for unsupervised anomaly detection. The authors describe the architecture of an auto-encoder and its application in the detection of anomalies in time series data[4]. The paper by Zhang and Zulkernine surveys the machine learning application techniques in network traffic anomaly detection. In the process, several methodologies are compared, ranging from k-Nearest Neighbors and Support Vector Machines to Neural Networks[5]. Campos et al. provide a comprehensive survey of various outlier detection algorithms. The paper provides a systematic comparison of the presentation of techniques such as Local Outlier Factor, One-Class SVM, and Isolation Forest

on many benchmark datasets[6]. The paper by Rousseeuw and Driessen is devoted to the proposal of robust statistical techniques to detect anomalies. The authors propose an estimator of the Minimum Covariance Determinant, which provides a robust estimate of data covariance and thus possibly leads to better outlier detection[7]. Zong and colleagues proposed a hybrid of deep auto-encoders with Gaussian Mixture Models for unsupervised anomaly detection. Conceptually, this deep autoencoder learns to map the input data onto a compact representation, so that later, GMM can model the probability distribution in this latent space[8]. Blázquez-García et al. provide a survey of methods for time-series data anomaly detection. The paper organizes these general approaches into statistical, distance-based, and machine learning techniques[9]. The survey by Hodge and Austin presents a comprehensive review of outlier detection methodologies. Their paper categorizes those techniques into statistical, distance-based, density-based, and clustering methods[10]. Akçay et al. presented adversarial autoencoders as a more robust algorithm for abnormality detection in images. In the paper, the authors combine reconstruction criteria of auto-encoders with adversarial training to come up with a model resilient to noisy and adversarial attacks[11]. Schlegl et al. propose an anomaly detection method using GANs, which means training GANs to generate normal data samples and then detecting anomalies based on their deviation from generated samples[12]. A model constructed by Umaporn Yokkampon et al. for the accurate detection of anomalies in the multivariate time series data has captivated more awareness because of its importance in a broad range of applications[13]. Chandola, Banerjee, and Kumar provide a comprehensive survey of anomaly detection techniques. The paper distinguishes the techniques developed under statistical, proximity-based, clustering-based, and machine learning approaches[14].Ping Jiang et al. developed a model to balance the samples between the majority and the minority classes. An oversampling algorithm is used for synthesizing new minority class samples, and it might introduce noise in it[15].

3. EXISTING SYSTEM

The existing system is based on training the classification model using transaction details from only the legitimate data class, further working out the reconstruction error for a given dataset, and rendering a decision based on a predefined threshold.

The methodology used by the current model is to first use oversampling, which will help in transforming an imbalanced dataset into a balanced dataset. Next is denoising, which helps get a noised-free data set. Following this, it uses a deep fully connected neural network model for final classification.

4. PROBLEM STATEMENT

In particular, anomaly detection is very important in systems that need measures of resistance against malicious activities to guarantee operational reliability in industrial processes and detect fraud in financial transactions. Conventional methods for detecting anomalies are very fast rendered ineffective by the high-dimensional characteristics and complexity of modern datasets. As such, this basically creates a growing need for developing a sophisticated framework of anomaly detection that can overcome such challenges efficiently.

To develop software for anomaly detection using Autoencoders which will detect the abnormalities in the high-dimensional input datasets, i.e., credit card transaction details, using a neural network and unsupervised deep learning algorithms.

Input: Credit card transaction details' dataset.

Output: Detection of anomalies in the given dataset.

4. PRELIMINARIES

AUTOENCODERS

Autoencoders are one class of neural networks which can be trained without teacher supervision. The purpose of Autoencoders is to find a compressed form of the input data in a bottleneck layer. This encoding can be used to reconstruct the initial input. To check the anomaly score, reconstruction error can be used, where a high reconstruction error indicates anomalous behavior.

ISOLATION FOREST

The Isolation Forest is constructed on the "separate-away" mechanism, whereby anomalies are isolated. Isolation Forest

involves building a collection of decision trees, and the main intuition is to discard the errors by way of division. The Isolation Forest undoubtedly provides an ensemble of decision trees built on the data set. The least centralized anomaly results of the binary tree are collected as the most anomalies result. Isolation Forest represents the ensemble of binary decision trees. Each tree in an Isolation Forest is called an Isolation Tree (I Tree). This gives the algorithm an advantage in always learning from data and creating Isolation Trees. The other advantage in the working of the Isolation Forest is that it works faster as compared to other anomaly detection algorithms and it also uses less memory.

5. PROPOSED SYSTEM

The system for Anomaly Detection in use makes use of oversampling and Autoencoders. Autoencoder neural networks can sometimes misclassify genuine transactions as fraudulent. Proposed system implements Isolation Forest algorithm to the outputs obtained from the AE. This aids to get enhanced accuracy than the accuracy obtained from AE alone.

The dataset involves only numerical input after doing PCA transformation. First, the uneven dataset is converted to a balanced dataset by oversampling. Then this is passed through a denoised autoencoder to receive a denoised dataset.

It involves two steps of anomaly detection whereby the output from the first stage becomes the input to the second stage. At Layer1, running the test dataset through the autoencoder performs.

There is segregation of abnormal and normal transaction data-sets into two sets. However, the resulted sets contain data points that ideally do not belong to them. Layer2 then uses Isolation Forest in an attempt to identify these misfit outlier data points to enhance the accuracy overall.

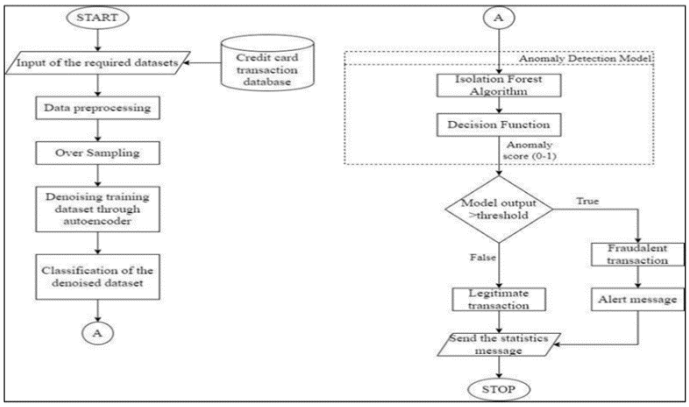

Fig -1: SYSTEM ARCHITECTURE

6. METHODOLOGY

Dataset Description:

Credit Card Fraud Transaction Dataset utilized in this work is accessed from Kaggle, updated as of 2021. The current dataset contains transactions for two days where we have 492 fraud cases out of 284,807 transactions. It contains numerical input after doing the PCA transformation. Features V1, V2 … V28 are the main components; the only features that remain untransformed by PCA are 'Time' and 'Amount'. It takes in a target variable, 'Class,' which is set to 1 in cases of fraudulent transactions and otherwise 0.

Data Preprocessing:

Because the datasets have formal and numerical values, the training and testing datasets are normalized. Normalizing the values is done in order to assign equal weight to each feature. This approach examines every aspect of this dataset. Thus, every feature has the same significance. For preprocessing dataset, drop the "TIME" data, and normalized the "AMOUNT" part. Other features are obtained by PCA and does not need to do normalization. Then select the test sample, which is 20% of the total sample.

Oversampling:

The train dataset is over-sampled. Before over-sampling, the amount of normal and abnormal classes is uneven. After over-sampling, both normal and abnormal classes contain an equal number of samples in the training dataset.

Train Autoencoder:

The autoencoder is instructed on normal datasets only. There are various advantages of this approach.

1. Training the AE on only normal datasets overcomes class uneveness of the credit card transaction datasets.

2. Data congestion of normal type will be captured and the attacks will be discarded.

3. More visual for real-time applications like fraud detection where decisions over normal and abnormal data sets have to be built in real time.

Denoising Autoencoder:

The autoencoder contains 7 layers in the procedure of denoising the dataset as shown in Fig. After getting a balanced training dataset from oversampling, Gaussian noise is attached to the training dataset and then fed into the denoised autoencoder. After instructing the autoencoder can denoise the testing dataset in the prediction process

Isolation Forest algorithm:

It will provide an equal number of outputs as inputs, but with reconstruction loss. The reconstruction loss for the abnormal data becomes astronomically huge in comparison to the normal data as AE has only been trained on "normal" data. By varying the reconstruction loss value, one can determine a preferred threshold. Data points which have reconstruction loss values greater than the threshold value can be categorized as "normal sets" (set1) or "abnormal sets" (set2). The first and second sets of data contain aberrant and normal data, respectively, hence the AE result is not entirely correct.

These two sets are subsequently sent as inputs to two Isolation Forest modules in order to increase accuracy, which entails detecting more incursions. The "abnormal" output of the AE is fed into the first module, Isolation Forest1, which searches for anomalies—normal data points. Comparably, the AE's "normal" output is used by the second module, Isolation Forest2, to search for anomalies, or unusual data points. To put it simply, outliers and anomalies are what the abnormal data in the "normal" set and the normal data in the "abnormal" set are.

7. OUTPUT AND RESULTS

In machine learning, you still see differences in data grouping even when you add more info. To address this, researchers turned to AutoEncoder and oversampling. The proposed system combines a stacked denoising AutoEncoder neural network with the oversampling approach for better clustering of things. It performs optimal at 83.56% under an optimal setting of 0.2. The new system utilizes AutoEncoders in detecting pesky patterns and Isolation Forest for weird data points. The combination achieves an accuracy of 95.4 per cent at a setting of 0.0022. Basically, the Isolation Forest works well on false alarm detection, uniquely working in more of its function compared to AutoEncoder. However, mixing them complicates things because it is time-consuming and requires more computer power when handled with large datasets.

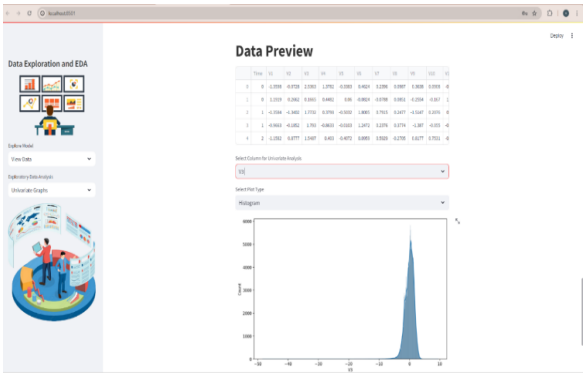

Fig -2: Data Preview and Univariant Graph

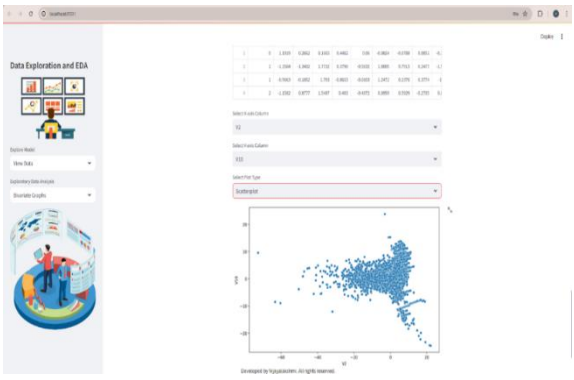

Fig -3: Bivariant Graph

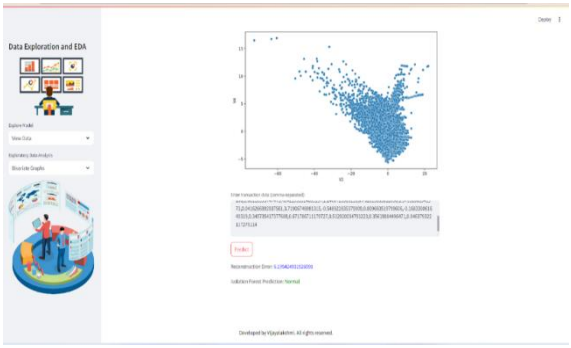

Fig -4: Normal Transaction

Fig -5: Fraud Transaction

7. CONCLUSION

In this study, we checked out how to mix Autoencoders and the Isolation Forest algorithm to spot weird stuff better in big messy datasets. We wanted to use the good parts of both to make a strong system that can find outliers in things like network security catching frauds, and making sure stuff is good quality. Autoencoders are impressive because they can learn how normal data looks without anyone instructing them what to do. They did a good job of figuring out what's normal and what's not by looking at how well they could rebuild the data. The Isolation Forest, which uses a bunch of decision trees, was able to unrelate the weird stuff by splitting data randomly. This made it fast and able to handle lots of data.

We came up with a method to combine these two ideas. We took the errors from rebuilding data in Autoencoders and mixed them with how the Isolation Forest scores weird stuff. This combo lets the system see both simple and tricky patterns in the data making it better at finding odd things than the old ways of doing it. We tried out our new system on a bunch of different test datasets. It did well to get more right answers and missing fewer weird things than other methods. The results show that our mixed-up model can find strange stuff more , which makes it super useful for real-world problems.

REFERENCES

[1] Aggarwal, C. C. (2013). *Anomaly Detection in High-Dimensional Spaces*. In: Outlier Analysis. Springer, New York, NY.

[2] Chalapathy, R., & Chawla, S. (2019). *Deep Learning for Anomaly Detection: A Survey*. arXiv preprint arXiv:1901.03407.

[3] Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). *Isolation Forest*. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 413-422.

[4] Sakurada, M., & Yairi, T. (2014). *Anomaly Detection with Autoencoders*. In: Proceedings of the 2nd Workshop on Machine Learning for Sensory Data Analysis (MLSDA 2014), pp.

[5] Zhang, Y., & Zulkernine, M. (2006). *Anomaly Detection in Network Traffic Using Machine Learning Techniques*. In: Proceedings of the 2006 IEEE International Conference on Communications, pp. 1-5.

[6] Campos, G. O., et al. (2016). *A Comparative Evaluation of Outlier Detection Algorithms: Experiments and Analysis*. Pattern Recognition, 74, 406-421.

[7] Rousseeuw, P. J., & Driessen, K. V. (1999). *Anomaly Detection Using Robust Covariance Estimation*. Technometrics, 41(3), 212-223.

[8] Zong, B., et al. (2018). *Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection*. In: Proceedings of the International Conference on Learning Representations (ICLR).

[9] Blázquez-García, A., et al. (2020). *Anomaly Detection in Time-Series Data*. ACM Computing Surveys (CSUR), 53(4), 1-28.

[10] Hodge, V. J., & Austin, J. (2004). *A Survey of Outlier Detection Methodologies*. Artificial Intelligence Review, 22(2), 85-126.

[11]Akçay, S., Atapour-Abarghouei, A., & Breckon, T. P. (2019). *Robust Anomaly Detection in Images Using Adversarial Autoencoders*. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 169-177.

[12] Schlegl, T., et al. (2017). *Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery*. In: Proceedings of the International Conference on Information Processing in Medical Imaging (IPMI), pp. 146-157.

[13] Chandola, V., Banerjee, A., & Kumar, V. (2009). *Anomaly Detection: A Survey*. ACM Computing Surveys (CSUR), 41(3), 1-58.

[14] Golan, I., & El-Yaniv, R. (2018). *Deep Anomaly Detection Using Geometric Transformations*. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS), pp. 9758-9769.

[15] Mohammad Mahdi Rezapour Mashhadi, "Anomaly Detection using Unsupervised Methods: Credit Card Fraud Case Study", International Journal of Advanced Computer Science and Applications, 2019.

[16] Tom Sweers, "Autoencoding Credit Card Fraud," Radboud University, 2018.

[15] Kubilay Muameleci, "Anomaly Detection in Credit Card Transactions using Multivariate Generalized Pareto Distribution Comparison in Performance for Supervised and Unsupervised Machine Learning Algorithm", Chalmers University of Technology, 2022.

Web References

[16] Credit Card Fraud Detection using Autoencoders in Keras — TensorFlow for Hackers, https://venelinvalkov.medium.com/credit-card-fraud-detection-using-autoencoders-in-keras tensorflow-for-hackers-part-vii-20e0c85301bd

[17] How to perform anomaly detection with the Isolation Forest algorithm, https://towardsdatascience.com/how-to-perform-anomaly-detection-with-the-isolation-forest algorithm-e8c8372520bc

[14]J.P.Wilkinson,–Nonlinear resonantcircuitdevices,‖ U.S.Patent3624 12,July16, 1990.**(Standards style)**

[15] *LetterSymbolsforQuantities*,ANSI StandardY10.5-1968. **(Handbookstyle)**

[16]*TransmissionSystemsforCommunications*,3rded.,WesternElectricCo.,Winston-Salem,NC,1985, pp. 44-60.

[17] *Motorola Semiconductor Data Manual*, Motorola SemiconductorProducts Inc.,Phoenix,AZ, 1989.