Analyzing the Classification Accuracy of Deep Learning and Machine Learning for Credit Card Fraud Detection

Mohammad Naveed Hossain
Computer Science and Engineering
BRAC University
Dhaka,Bangladesh
naveedhossain99@gmail.com

Md. Mahedi Hassan
Computer Science and Engineering
BRAC University
Dhaka,Bangladesh
mahedi.hassan11001@gmail.com

Raiyan Janik Monir
Computer Science and Engineering
BRAC University
Dhaka,Bangladesh
raiyan.janik.monir@g.bracu.ac.bd

***Abstract*—The purpose of this study is to classify a dataset of credit card security problems by employing six different machine learning (ML) approaches. The Support Vector Machine (SVM), Random Forest (RF), Bagged Tree, K-Nearest Neighbor (KNN), Naive Biased Classifier, and Extreme Gradient Boosting were selected as the classifiers to use (XGBoost). The classification accuracy of the machine learning algorithms was compared with that of a technique for categorization that is based on deep learning called Long Short-Term Memory (LSTM). The KNN machine learning approach had a maximum accuracy of 97.50 percent, while the LSTM machine learning method had an accuracy of more than 96 percent and promised to give biologically appropriate control of upper-limb movement. In addition to enhancing accuracy, the research has investigated how the effects of removing the channel with the most noise from the algorithms can have on accuracy. This was done in an effort to handle data in a more effective manner.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

The use of a payment card, such as a credit card or debit card, to commit fraud is known as credit card fraud. The goal might be to obtain goods or services or to move money from a particular account to another one controlled by the criminals. In the credit card sector, credit card fraud is an increasing issue. Losses from all forms of credit card theft are anticipated to exceed $850 million in the US alone, a 10 % rise from 1991. [13] Despite being negligible in comparison to credit card losses brought on by charge-offs of severely past-due accounts ($8.5 billion in losses in 1992), Fraud's share of total charge volume is rising, showing that it is expanding faster than the credit card industry as a whole. The scale of the fraud problem increased from 8 basis points to almost 20 between 1988 and 1991. The identification of credit card fraud has received a lot of attention recently, and various research papers have been published in this area. As a result, we make an effort to carefully read a large number of research papers from related fields and critically assess them from various perspectives.

II. Related Works

Unauthorized credit card fraud occurs when the account holder does not give permission for the payment to proceed and a third party completes the transaction. On the other hand Authorized credit card fraud occurs when the legitimate customer themselves processes payment to another account that is controlled by a criminal. In the UK, losses from unauthorized financial fraud involving credit cards and remote banking were £844.8 million in 2018. While in 2018, banks and card issuers stopped 1.66 billion in unauthorized fraud. Unauthorized individuals may use someone else's credit card information to make purchases, conduct other transactions, or open new accounts, which is referred to as credit card fraud. Account takeover fraud, new account fraud, cloned cards, and cards-not-present schemes are a few instances of credit card fraud. Phishing, information skimming, and information sharing by a user—often without their knowledge lead to this unlawful access.

According to paper [13], a neural network-based credit card fraud detection model was trained on an extensive sample of pre-labeled credit card activities and evaluated on a holdout data set that included all transaction history over the course of the next two months, using information from a card provider. The neural network has been trained using real-world cases of fraud involving missing or stolen cards, applications, forgeries, mail-order fraud, and NRI fraud. In comparison to rule-based threat detection processes, the network discovered noticeably more fraudulent accounts with noticeably fewer false - positives. In terms of detection speed and precision of detecting fraud, they talked about how the network performed on this collection of data.

The primary focus of the paper [14] is machine learning algorithms. The Random Forest and the Adaboost Algorithms are applied. The two algorithms' outputs are based on F1-score, accuracy, precision, recall, and other metrics. On the basis of the confusion matrix, the ROC curve is plotted. When the algorithms from Random Forest and Adaboost are compared, the method with the highest accuracy, precision,

recall, and F1-score is regarded as the best one for identifying fraudulent activities. It is evident from the data above that a variety of machine learning techniques are applied to identify fraud, however, we can see that the outcomes are not up to the mark. Therefore, in order to detect credit card fraud more accurately, the authors suggested applying deep learning techniques.

Cyberspace has grown due to the widespread use of the Internet and mobile devices. Cyberattacks that are automated and sustained are now more likely to occur in cyberspace. This paper [15] aims to present literature on ML techniques for cyber security, including intrusion detection, spam detection, and malware detection on computer networks and mobile networks in the last decade, in order to provide a comprehensive overview of the challenges that ML techniques face in defending cyberspace against attacks. Cybersecurity approaches improve security procedures for spotting and responding to threats. Due to hackers' increased intelligence and ability to circumvent traditional security measures, the previously employed protection mechanisms are no longer enough. Ineffective at spotting previously unknown and polymorphic security assaults, conventional security systems. In many cyber security applications, machine learning (ML) techniques are essential. Nevertheless, despite ongoing achievements, there remain considerable difficulties in guaranteeing the reliability of ML systems. In cyberspace, there are motivated opponents that are eager to game and take advantage of such ML weaknesses.

Despite the fact that deep learning has been effectively used to solve a variety of data mining issues, relatively little research has been done on deep learning for anomaly identification. Existing deep anomaly detection techniques use indirect optimization of anomaly scores, which results in data-inefficient learning and inadequate anomaly detection. These techniques concentrate on learning novel feature representations to support downstream anomaly detection techniques. Because there aren't many large-scale datasets with labels for anomalies, they are frequently constructed as unsupervised learning. In response, when such knowledge is accessible, such as in numerous real-world anomaly detection applications, it is challenging to use previous knowledge. In order to solve these issues, this study offers a unique anomaly detection framework and its instantiation. This research [16] satisfies an end-to-end requirement without using representation learning. In order to solve these issues, this study offers a unique anomaly detection framework and its instantiation. Authors use a few labeled anomalies and a prior probability to impose statistically substantial deviations of the anomaly scores of anomalies from those of normal data objects in the upper tail, replacing representation learning with neural deviation learning in our technique. Numerous findings demonstrate that this method outperforms state-of-the-art techniques in terms of anomaly classification and the ability to train the data far more efficiently.

Machine learning techniques are being used more often than have ever been in cyber security. The use of machine learning is one of the potential solutions that can be successful against zero-day attacks, starting with the categorization of IP traffic and filtering harmful traffic for intrusion detection. Utilizing statistical traffic features and ML approaches, a new study is being conducted. This research [17] conducts a concentrated literature review on machine learning and its usage in cyber analytics for email filtering, traffic categorization, and intrusion detection. Each approach was recognized and summarized in accordance with its importance and the number of citations. Some well-known datasets are also discussed because they are a crucial component of ML techniques. Concerning when to utilize a certain algorithm is also offered advice. On MODBUS data gathered from a gas pipeline, four ML algorithms have been evaluated. Applying ML algorithms, different assaults have been categorized, and each algorithm's performance has then been evaluated.

This research [18] suggested a stacked auto-encoder (SAE) featured deep learning model to create machine-learned characteristics for transmission SCADA threats in order to enhance more elevated characteristics for ML-based threat surveillance. The proposed approach utilizes the automaticity of unlabeled feature learning in comparison to the state-of-the-art ML detectors to lessen the dependency on framework models and human experience in complicated security contexts. The effectiveness of the machine-learned characteristics in enabling more precise discriminating against SCADA attacks in power transmission systems was proved in simulations using data from a high-fidelity smart grid test-bed.

III. Data Collection and dataset explanation

The data set was acquired using Kaggle. [12] In the data collection, five frequent credit card problems were identified. Examples include card-not-present fraud, counterfeiting and skimming fraud, and lost and stolen cards. This category includes card fraud, non-arrival of card fraud, and bogus application fraud. CNP fraud is the fraudulent use of a card while its owner is not present. Skimming is a white-collar crime in which money is taken from a firm before it is recorded in the records. Since the funds are taken before being recorded in the books, skimming is considered "off-book" fraud. Consequently, it is not recorded in the company's records. Under the pretense of "card not received," fraud is conducted when a customer orders a card but never gets it. In false application fraud, the identity or information of another person is used to establish an account. Before the possessing process could begin, a comprehensive data purification was performed. Must be present to fill in blanks, delete tuples, and remove missing data. In addition to regression and clustering, the binning approach was used to eliminate unnecessary data. The remaining data transformation operations comprised normalization, attribute selection, discretization, and creating a concept hierarchy. The size of the dataset was not lowered for this study. [4]

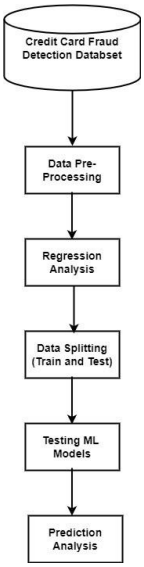

Fig. 1. Data prepossessing flow

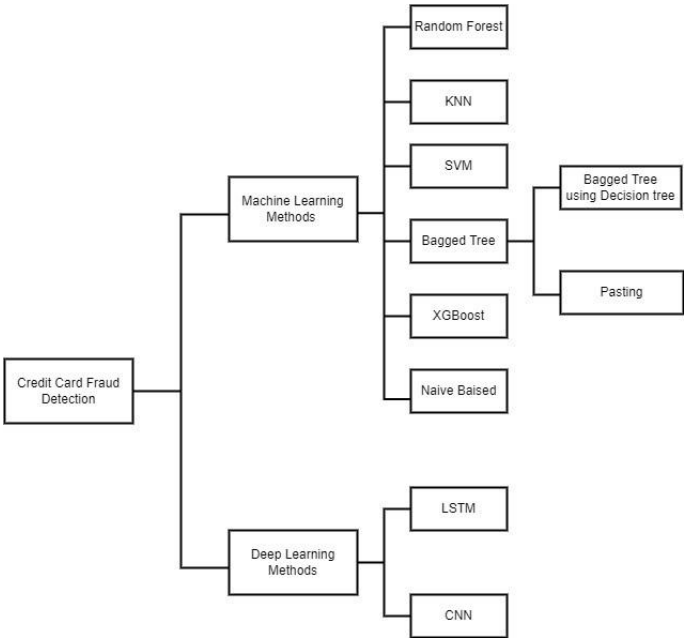

Fig. 2. Applied Models

IV. Applied Methods

A. Random Forest

Random forest uses decision trees. A bootstrap sample is a randomized portion of a training set used to generate trees. Half of the training sample becomes test data. Out-of-bag (oob) samples will be discussed again. Feature bagging adds a second random event to the dataset, [2] making decision trees more different. The problem dictates how to predict. In classification, the predicted class is the one with the most votes or the most frequent categorical variable. In regression, the individual decision trees are summed. Cross-validation is used to verify the prediction's correctness.

B. Support Vector Machine

SVM is basically a supervised learning model that tries to tell the classes apart as much as possible. This is done by converting the data coming into different feature spaces. The feature space affects how hard the algorithm is to understand. The SVM makes use of three core functions. Here are the nuts and bolts: 1.) Linear kernel 2.) Polynomial kernel for the Radial Basis Function (RBF). In this study, the RBF kernel gave more accuracy than was planned. Attributes of data must be split in a way that is not linear. [3] The following equation is a way to describe the RBF function:

$$f(X1, X2) = (a + X1^T * X2)^b$$

This is a pretty simple kernel formula for a polynomial. The polynomial decision limit for this set of data is f. (X1, X2). The data are shown in two formats: X1 and X2. Most questions about how to classify text fall into this category, and they are often brought up. The following function tells us what the linear kernel is:

$$f(X) = w^T * X + b$$

Given the data to be grouped (X) and the expected linear coefficient (b), find the smallest weight vector (w) that fits the data (obtained from the training data). This equation shows the SVM's decision threshold. [3]

C. XGBoost

Extreme gradient-boosting (XGBoost) XGBoost is a versatile gradient-boosting library. Gradient Boosting is used. Parallel tree boosting handles data science problems rapidly and accurately. XGBoost is an open-source gradient boosted trees implementation. Gradient boosting combines the estimates of simpler, weaker models to forecast a target variable. [1] When employing gradient boosting for regression, the weak learners are regression trees, and each transfers an input data point to a leaf with a continuous score. XGBoost minimizes a regularized (L1 and L2) objective function that combines a convex loss function with model complexity penalty (in other words, the regression tree functions). Iterative adding new trees that anticipate the residuals or errors of preceding trees is used to make the final prediction. Gradient boosting minimizes model loss by using a gradient descent approach.

D. Bagged Tree

Bagging is a meta-algorithm used to increase the stability and accuracy of machine learning algorithms used in statistical classification and regression. It also helps to minimize variance and prevent overfitting by reducing the number of observations. Bagging reduces the size of the data, and the classification is attained with the help of other machine learning algorithms. [7] Fig. 3 portrays the classification process. Although it is most often associated with decision tree techniques, other techniques can also opt as well.

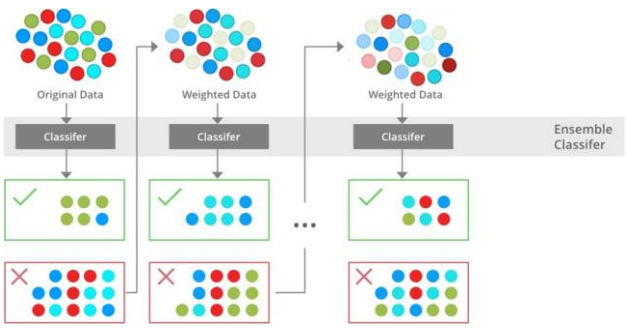

Fig. 3. Bagged Tree
[11]

E. KNN

KNN is a supervised ML algorithm used for classification and regression predicting issues. It's mostly utilized for industrial classification challenges. Two characteristics describe are KNN is a slow learning method because it uses all the data for training and classification. KNN is a non-parametric algorithm since it takes no assumptions about the data. KNN algorithm predicts the values of fresh data sets depending on how closely they resemble the training set. Any algorithm needs data. First, we load training and test data for KNN. Next, choose K, the nearest data point. K is an integer. Use Euclidean, Manhattan, or Hamming distance to calculate the distance between test and training data. [9] [5] Euclidean distance is the most prevalent. Sort them by distance, then. The top K rows of the sorted array are then chosen. Now, the test point's class will be based on the most frequent class of these rows.

F. Naive Bayes

The work of Nave Bayes exemplifies the adage that the simplest answers are frequently the best ones. The current developments in Machine Learning have not changed the fact that it is simple, fast, accurate, and reliable. It has proven useful in many settings, but its true calling is in the realm of natural language processing (NLP) problems. Naive Bayes, a machine learning method based on the Bayes Theorem, is widely used for various classification tasks. This article will leave no stone unturned as we explore the Nave Bayes algorithm and its core concepts.

G. LSTM

LSTM employs deep learning to learn long-term dependencies. Their adaptability makes them popular for solving many difficulties. LSTMs prevent habit development. An LSTM RNN has four linked layers. Change or delete the LSTM's gated cell state. Lines carry outputs and inputs from nodes. Orange boxes represent neural network layers, whereas green circles represent component actions. Forking copies text, whereas concatenation links lines. The sigmoid layer creates 0-1 weights for each input element to decide how much of it to send to the next layer. [9] Three gates govern LSTM cell state. LSTM uses a sigmoid function at each iteration to disregard cell input. Sigmoid and tanh functions select whether values pass (0 or 1) and apply weights (-1 to 1) to them. In the last step, the output percentage is determined using a sigmoid function and a tanh function. Each RNN stage will choose its own data from a large store.

V. Result Analysis and Discussion

Method name	Accuracy
Random Forest	91%
Bagges Tree	91.13%
KNN	%
SVM	92%
Naive Baised	83.33%
XGBoost	94.22%
LSTM	98%

TABLE I
Accuracy table

This research analyzes our outcomes using the performance criteria accuracy, precision, recall, and F1-Score. The data utilized for both training and testing is divided into 80-20 unique sets using all existing methodologies. All classification model hyper parameters were fine-tuned to get the most accurate metrics possible. Table (1) displays the accuracy the six classification models assessed in this research. The label "Counterfeiting and skimming" had the greatest recall and f1-score values (0.96 and 0.93, respectively) in Random Forest. With an accuracy of 0.93, "Counterfeiting and skimming" is the gesture that achieves the greatest exact capture. Using SVM, the "Card-not-present" class obtains the maximum recall (0.93) and f1-score (0.94), while the "False application" class achieves the best precision (0.92). For the Lost and Stolen Card class, the highest accuracy that XGBoost can achieve is 0.93. The subcategory "Counterfeiting and skimming" had the highest recall and f1-score of 0.97 and 0.95, respectively. In Bagged Tree, the "Last and stolen card" class has the greatest precision (0.92), while the "False application" class has the highest recall (0.95), as well as the highest f1-score (0.94). With a rating of 0.96, "counterfeiting and skimming" is the category with the highest KNN precision. The category with the greatest recall values and f1-score is "Card-never-received" (0.97 and 0.95, respectively). Naive Biased earned the greatest recall (0.91) and F1-score (0.85) for the category "Counterfeiting and skimming." LSTM's efficacy was shown by the outcomes. The highest accuracy, recall, and F1 score attained by any classifier in the scenario of counterfeiting and skimming were 0.99. Fig. (5) displays the accuracy of the various classifiers examined for the proposed study. The

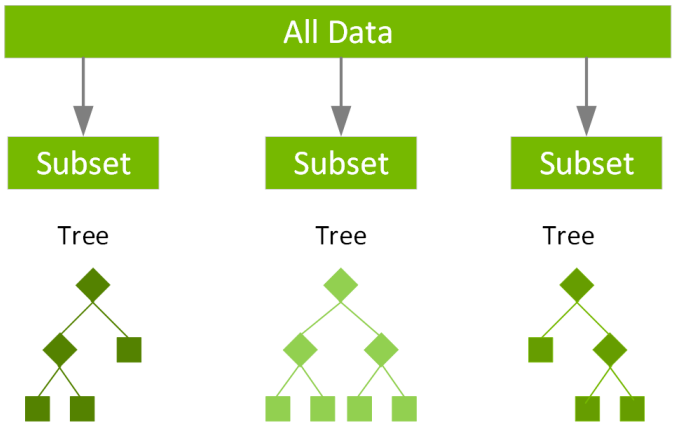

Fig. 4. XGBoost data processing flow
[10]

Bagged tree classifier (with Random Sub spaces) achieves 91.31 percent accuracy, but the SVM classifier achieves 92 percent accuracy. Both the RF and XGBoost classifiers have a 91 percent prediction accuracy rate. For classifiers based on machine learning, KNN has the highest accuracy, while Naive biased has the lowest at 83.33 percent. Compared to other techniques, LSTM's success record of 98 percent makes it the obvious winner. This level of precision calls for forty time periods. Figure (4) illustrates the correlation between the number of epochs and accuracy on both the training and validation data sets. XGBoost has the potential to outperform other machine learning (ML) algorithms due to its ability to create classifiers incrementally. This is the key to developing an effective classifier since any approach may be used to get the optimal weight values that minimize prediction error. To do this, deep neural networks use the backward propagation method to establish a bidirectional feedback network, which eventually offers ANNs an advantage over ML approaches. To

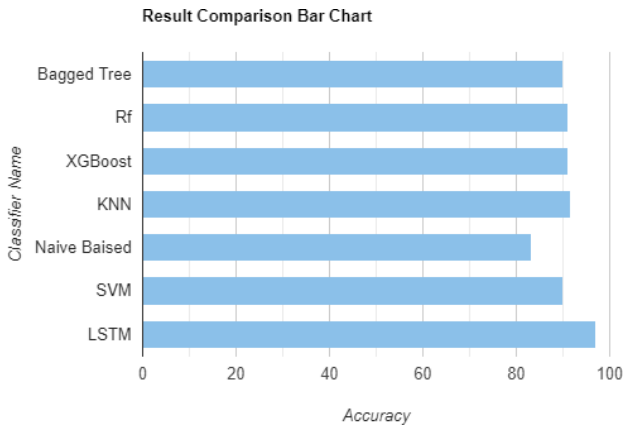

Fig. 5. Comparison Bar chart

obtain the highest level of accuracy, LSTM is preferable since they can recall patterns selectively over extended time periods. Categorization is facilitated by long short-term memory (LSTM) cells, which enable the acquisition of extra elements when the data has a longer-term pattern.

VI. Conclusion and Future Work

This study's major purpose was to assess the level of accuracy attained using machine learning and deep learning approaches (LSTM). In addition, the study aimed for a very high degree of precision, analyzed the impacts of numerous algorithms on the precision of the Bagged tree, and assessed the influence of the noisiest channel on the precision of the overall analysis. By using the ML technique, KNN was able to achieve 98% accuracy. Despite this, around 97% accuracy has been attained with the application of deep learning. The study reported in this article was effective in proving that the DL algorithm is capable of delivering high precision, which has identifies that card not present is the mostly occurred fraud among other fraud methods.

Card-not-present fraud cost $477,920,701. [8] Retailers should study secure payment methods to prevent this mistake. CVV number use must alter to avoid card-not-present fraud. Confirming and verifying financial activities protects customers. This research aims to strengthen its basis by introducing and validating additional fraud detection techniques and by giving countermeasures to new and evolving fraud tactics.

References

[1] Ghosh, S., & Reilly, D. L. (1994, January). Credit card fraud detection with a neural-network. In System Sciences, 1994. Proceedings of the Twenty-Seventh Hawaii International Conference on (Vol. 3, pp. 621-630). IEEE.

[2] Raj, S. B. E., & Portia, A. A. (2011, March). Analysis on credit card fraud detection methods. In 2011 International Conference on Computer, Communication and Electrical Technology (ICCCET) (pp. 152-156). IEEE.

[3] Srivastava, A., Kundu, A., Sural, S., & Majumdar, A. (2008). Credit card fraud detection using hidden Markov model. IEEE Transactions on dependable and secure computing, 5(1), 37-48.

[4] Chan, P. K., Fan, W., Prodromidis, A. L., & Stolfo, S. J. (1999). Distributed data mining in credit card fraud detection. IEEE Intelligent Systems and Their Applications, 14(6), 67-74.

[5] Patidar, R., & Sharma, L. (2011). Credit card fraud detection using neural network. International Journal of Soft Computing and Engineering (IJSCE), 1(32-38).

[6] Thennakoon, A., Bhagyani, C., Premadasa, S., Mihiranga, S., & Kuruwitaarachchi, N. (2019, January). Real-time credit card fraud detection using machine learning. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 488-493). IEEE.

[7] Abakarim, Y., Lahby, M., & Attioui, A. (2018, October). An efficient real time model for credit card fraud detection based on deep learning. In Proceedings of the 12th international conference on intelligent systems: theories and applications (pp. 1-7).

[8] Rajeshwari, U., & Babu, B. S. (2016, July). Real-time credit card fraud detection using streaming analytics. In 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) (pp. 439-444). IEEE.

[9] Ghosh, S., & Reilly, D. L. (1994, January). Credit card fraud detection with a neural-network. In System Sciences, 1994. Proceedings of the Twenty-Seventh Hawaii International Conference on (Vol. 3, pp. 621-630). IEEE.

[10] NDVIA, What is XGBoost?, https://www.nvidia.com/en-us/glossary/data-science/xgboost/

[11] Geeksfor geeks, Bagging vs Boosting in Machine Learning, https://www.geeksforgeeks.org/bagging-vs-boosting-in-machine-learning/
[12] Kaggle dataset, Credit Card Fraud Detection, https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud
[13] Ghosh, S. & Reilly, D. Credit card fraud detection with a neural-network. *System Sciences, 1994. Proceedings Of The Twenty-Seventh Hawaii International Conference On*. **3** pp. 621-630 (1994)
[14] Sailusha, R., Gnaneswar, V., Ramesh, R. & Rao, G. Credit card fraud detection using machine learning. *2020 4th International Conference On Intelligent Computing And Control Systems (ICICCS)*. pp. 1264-1270 (2020)
[15] Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. & Xu, M. A survey on machine learning techniques for cyber security in the last decade. *IEEE Access*. **8** pp. 222310-222354 (2020)
[16] Pang, G., Shen, C. & Hengel, A. Deep anomaly detection with deviation networks. *Proceedings Of The 25th ACM SIGKDD International Conference On Knowledge Discovery & Data Mining*. pp. 353-362 (2019)
[17] Das, R. & Morris, T. Machine learning and cyber security. *2017 International Conference On Computer, Electrical & Communication Engineering (ICCECE)*. pp. 1-7 (2017)
[18] Wilson, D., Tang, Y., Yan, J. & Lu, Z. Deep learning-aided cyber-attack detection in power transmission systems. *2018 IEEE Power & Energy Society General Meeting (PESGM)*. pp. 1-5 (2018)