National College of Ireland

Project Submission Sheet

Student Name: ...Emin Cem Koyluoglu...

Student ID: ....x23192542 ....

Programme: ...MSCAI...................................... **Year:** ...2024-2025... ....

Module: ...Foundations of Artificial Intelligence **(H9FAI)**..

Submission Due Date: ......20/11/2024...

Project Title: ... Credit Card Fraud Detection with Machine Learning.........

Word Count: ...3091...

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
<u>ALL</u> internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature: ....

Date: ....19/11/2024..

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.

3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

[Foundations of Artificial Intelligence (H9FAI)]

[Credit Card Fraud Detection with Machine Learning]

Your Name/Student Number	Course	Date
Emin Cem Koyluoglu/ x23192542	Foundations of Artificial Intelligence **(H9FAI)**	19/11/2024

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool
ChatGPT	Provided assistance with finding references and language translation	https://chat.com/

Quillbot	Helped with grammar checking and improving sentence clarity	https://quillbot.com/

Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

ChatGPT	
Assist in brainstorming and structuring the assignment's content.	
Translate text from Turkish to English accurately.	Guide on how to format IEEE-style references.

Quillbot
Help was taken from Quillbot Grammer Tools to fix grammar errors.

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence:

ChatGPT o1-preview >

What is IEEE format and how can I use it in my research paper?

Thought for a few seconds

The IEEE format refers to a set of guidelines for formatting academic papers, specifically tailored for engineering, computer science, and related fields. Developed by the Institute of Electrical and Electronics Engineers (IEEE), this format standardizes the structure and presentation of research papers to ensure consistency and professionalism across publications.

Key Components of the IEEE Format:

1. **Paper Structure:**
   - **Title Page:** Includes the paper title, author names, affiliations, and contact information.
   - **Abstract:** A brief summary of the paper, usually around 150-250 words.
   - **Keywords:** A list of relevant keywords following the abstract.
   - **Main Body:** Divided into sections such as Introduction, Methodology, Results, Discussion, and Conclusion.
   - **References:** A numbered list of all sources cited in the paper.
   - **Appendices (if any):** Additional material that supports the main text.
2. **Formatting Guidelines:**
   - **Page Layout:** Typically double-column format with single spacing.
   - **Margins:** Usually set to 1 inch on all sides.
   - **Font:** Times New Roman is commonly used; font size varies (e.g., 24 pt for the title, 10 pt for the body text).
   - **Headings:** Section headings are numbered and formatted in a consistent style.
   - **Figures and Tables:** Centered and labeled with captions; figures are numbered consecutively.
3. **Citation Style:**
   - **In-text Citations:** Uses numerical citations in square brackets, e.g., [1], [2], [3].
   - **Reference List:** Entries are listed numerically in the order they appear in the text.

How to Use IEEE Format in Your Research Paper:

1. **Obtain the Official Template:**
   - **Microsoft Word Template:** IEEE provides a Word template that ...udes predefined styles for headings, text, and references.

Message ChatGPT

o1-preview

ChatGPT o1-preview >

Sınırlamalar

Bu güçlü yönlere rağmen, önerilen yöntemin bazı zayıflıkları şunları içerir:

1. Veri Kalitesine Bağımlılık: Performansı etkileyen diğer önemli faktör, veri setinin kalitesi, çeşitliliği ve temsil edilebilirliğidir. Gerçek hayattaki sahtekarlık kalıpları, modelleme için kullanılan açık veri setlerindeki sahtekarlık biçimlerinden oldukça farklıdır.
2. Dinamik Sahtekarlık Kalıpları: Sahtekarlık kalıpları sık sık değişme eğilimindedir. Bu nedenle, statik modeller zamanla kötü performans gösterebilir. Ansambl ve derin öğrenme yöntemleri uyarlanabilir olsa da, sürekli güncellenmiş verilerle yeniden eğitilmeleri gerekecektir.
3. Hesaplama Maliyeti: Derin öğrenme ve hibrit modellerin hesaplama maliyeti, düşük maliyetli uygulamalar için henüz uygun değildir.
4. Açıklanabilirlik: Sinir ağı modelleri ve ansambl yöntemleri, genellikle düzenlemeye tabi sektörlerin paydaşları tarafından istenmeyen kara kutu modeller olarak kabul edilir.

PROMPT:
I don't want you to change the meaning of the sentence or add an extra layer of information, I want you to directly translate my own sentences into English.

Thought for 40 seconds

Translating the document
Translating a lengthy Turkish text into English, ensuring the core meaning and critical details are preserved.

Translating a need
I'm pulling together the Turkish text to deliver a clear and accurate English translation, ensuring the essence is preserved without adding extraneous details.

Securing transactions
I'm exploring a fraud detection framework using machine learning, focusing on data anonymization and algorithms like logistic regression and gradient boosting to ensure high detection accuracy and minimal false alarms.

Understanding credit card fraud
Credit card fraud, a significant menace, generates losses over $28 billion annually. Despite digital payment systems' growth, sophisticated fraud attempts persist, necessitating adaptable detection methods to maintain consumer trust and control operational costs.

Shifting focus
Hmm, I'm thinking about how fraud detection techniques need an upgrade to keep up with modern threats. Machine learning seems ideal for sifting through vast datasets to spot fraud, especially with preprocessing techniques that boost both accuracy and adaptability.

Balancing and adapting
To spot fraud effectively, models must balance class imbalances. Techniques like oversampling, undersampling, and SMOTE

Message ChatGPT

o1-preview

Credit Card Fraud Detection with Machine Learning

Abstract

Fraud detection in credit cards is something quite urgent that the financial sector needs to attempt. Losses need to be limited and the transaction be made more secure. This identifies a fraud detection framework using machine learning with a number of sophisticated data treatment and classification techniques. As a result, such an experiment would be based on the dataset of credit card transactional information, so each of the feature-values is anonymized: transaction amount, location, and time. Thus, the analysis involves imbalance data processing of the dataset, feature engineering, and several machine learning algorithms including logistic regression, random forests, gradient boosting, and methods of deep learning. This would confirm the result in terms of high accuracy of detection and least number of false alarms based on metrics such as precision, recall, and F1-score. The insight brought from this project will highly contribute to fraud prevention systems and build up more trust in financial systems.

Keywords:

Credit card fraud detection, machine learning, financial transaction, imbalanced data, classification model.

I. Introduction

Applications: Credit card fraud is one of the most frequent and costly types of fraud with which any financial sector has to deal. Global losses due to credit card fraud may exceed 28 billion dollars in 2022 alone [2]. In this scenario, though digital support for modern payment systems is increasing, the attempts to fraud are also on the rise. Such frauds have an A&M effect: it jolts consumer confidence, cripples financial institutions, and raises operation costs.

While fraudsters' techniques employ increasingly sophisticated means, efficiency in fraud detection systems is dependent on the precision of tunings and adaptability. Traditional fraud detection systems usually use a set of rules for fraud detection. In general, fraud detection systems are rule-based; normally, fraud is detected by static thresholds set up from predefined rules.

Whereas this was quite all right some years ago, these methods hardly work against new fraud schemes. This also means there are extremely high levels of false positives and false negatives in systems. All this has favored the movement toward more data-driven machine learning approaches that find their intrinsic ability to mine large datasets for complex, hidden patterns indicative of fraud [3].

Therefore, modern fraud detection systems have achieved state-of-the-art performance by incorporating machine learning with strong preprocessing techniques both with great accuracy and high adaptiveness. Given the fraud detection, motivations of the project can be drawn from the following key challenges:

Skewed Data Distribution: Usually consists of less than 1% of total transactions and generally introduce skew in data distribution. Unless these imbalances are balanced, any machine learning model is biased to predict the majority class and hence will not do any good for fraud detection [4]. Various techniques available for handling these issues include oversampling, under-sampling, and synthetic data generation like SMOTE [5].

Evolution of Fraud Techniques: The techniques of fraud also tend to evolve very fast. Hence, fraud detection should be dynamic in nature with less re-training required.

Real-Time Constraints: The fraud detection system needs to give predictions near real-time in order to keep transactions frictionless. This immediately creates a tussle between computational efficiency and model complexity.

That is something challenging, and to such a challenge, this present study would surely help in the initiation of systems for fraud detection, which must be apt and efficient.

The proposed project tries to answer the following general research question: How is this particular machine learning technique to be manifested in order to detect credit card frauds faced with problems like imbalanced data, fraud patterns that change, and applicability in real time? To achieve this, the following have been set to be the objectives of the project:

1. Data Preprocessing: Design a preprocessing pipeline to clean up, normalize, and balance the dataset, employing missing data handling, feature scaling, and class imbalance with more advanced techniques such as SMOTE [5].

2. Model Explorations: The comparison of performances from different ML algorithms such as Logistic Regression, Decision Trees, Ensemble Methods—Random Forest and Gradient Boosting, and Deep Learning Architectures is to be presented.

3. Feature Engineering: Feature engineering is done presently by domain knowledge of transaction time intervals, location-based patterns, which will result in greater performance and interpretability [4].

4. Evaluation: It needs to be evaluated on various metrics like precision, recall, F1-score, AUC-ROC, etc. More emphasis shall be given toward reducing the number of false negatives, which means the cases of fraud that have not been detected [7].

II. Related Work

There are three fundamental strengths we observe:

1. Adaptability: It may be achieved through machine learning models, especially ensemble and deep learning methods, which make them adaptive to changes in fraudulent patterns. They can also be re-trained on updated data with good results.

2. Accuracy: The results from these models are not only good but also state-of-the-art, especially when embedded with good preprocessing techniques.

3. Efficiency: Techniques such as SMOTE and cost-sensitive learning enable dealing with so-called “imbalanced” datasets—a common case when faced with fraud detection scenarios.

Weaknesses:

1. Quality of Data: The models constructed are highly dependent on the quality and diversity of datasets used at the time of training. The form that actual fraud takes can be quite different from the forms taken by fraud as seen in the open-source datasets used for modeling. It is so stated in [8].

2. Dynamic Fraud Patterns: Much like any other static model, there could be gradual degradation in the fraud detection models that we come up with, simply because fraud as an industry innovates much like any legitimate business. The detection patterns could get outdated rather quickly once their retraining isn’t done on a continuous basis.

3. Computational Cost: When we have numerous patterns to detect and a variety of data to work with, our deep learning hybrid models can get prohibitively expensive. They consume a great deal of CPU time and memory; due to the parallel nature of algorithms presently used for them—despite the trends in modern computing—it is not obvious how effectively they could be run on a single, low-cost machine. Are they worth all the hassle if they’re going to use so much energy? [9], [11].

4. Explainability: Luring a number of interesting patterns from the complex set of data is possibly the reason for which we might use methods like neural network. But when we do do it, what’s it doing? It is all well and good to say that it is ideal for finding patterns as Edwardes did but no one knows for certain why it is so good at it—if you do not know how something works it can be dangerous to use it in a regulated industry [10].

Reuse of Methods

If you plan to reuse methods that have already been applied to that particular dataset, then you are in for a number of advantages:

- Highly Accurate: Most of the classical techniques, including SMOTE—Synthetic Minority Over-sampling Technique—and cost-sensitive learning have already been performed successfully to detect fraud.

- Class Imbalance Handling Efficiency: These methods turn out to be effective while dealing with yet another commonality of fraud detection, that is class imbalance.

- Adaptability: Fraud is something that is not static; it keeps evolving, and again, the best way to handle its evolution is continuous retraining of the model with present data.

Deep Dive into Machine Learning Algorithms

Because such algorithms can find in high-volume data complex patterns and anomalies, nowadays they are an essential piece of most modern fraud detection systems. Let's now look at some of the most common algorithms: how they work, when to use them, and why they succeed or fail at the task at hand.

A. Classification

1. Logistic Regression

Logistic regression is a type of statistical model used in predicting binary or no outcome probabilities. That would be whether or not a transaction is fraudulent. It is relatively simple and straightforward; hence, any other advanced model, if derived based on this, would have to outperform this one in order to be legitimate.

2. Decision Trees

Being more specific, decision trees are a popular means whereby fraud detection was done; thus, it decomposes complicated problems into parts that belong to subsets and are easier to deal with. The possible outcomes come with branches from a tree and they can be prodded and poked to get us the pattern most useful in identifying instances of fraud. But trees tend to overfit, or learn patterns that are just peculiar to the training data, and not to the kinds of data we should expect to see in the wild. Random forests can help us avoid this pitfall.

3. Ensemble Learning Methods

Ensemble learning methods take a large number of models and combine them to do better. Random forests employ many decision trees in order to expand the number of data types and conditions tested. RFs can also pick up non-linear relationships in data, which especially suits them to find "there's fraud happening" type patterns. In addition, random forests are very accurate, robust, compared with many other models, in a kind of "if it works, use it" way.

4. Gradient Boosting

The next section shall present the use of gradient boosting models. Gradient boosting frameworks including XGBoost and LightGBM are particularly well-apt for processing a dataset with high class imbalance. The model learns from the previous iterations of mistakes and in fraud detection applications, this attribute of the model makes it particularly well-suited for minimizing error types that matter most [13].

B. Anomaly Detection

1. K-Nearest Neighbors (KNN)

While K-Nearest Neighbors is an unsupervised machine learning algorithm, it does provide quite decent classification. In K-Nearest Neighbors, the algorithm commits to memory the training data and thereafter does all its computation with respect to the stored data. So, while K-Nearest Neighbors enjoys a straight path to accuracy, it may hit a wall when dealing with large-scale computation.

2. Autoencoders

These might be employed in anomaly detection, ordinal. However, autoencoders and their evil twin, variational auto-encoders, probably do the best job of learning the unsupervised patterns born from truly massive—and high-dimensional—datasets.

C. Deep Learning

Neural Networks

Overall, neural networks and deep learning neural networks are excellent at finding complex patterns within big data datasets. With minimal explicit programming, they reach a model of the pattern representing fraudulent transactions. They can, therefore, be an exceptional resource in fraud detection and prevention.

D. Hybrid Models

1. Stacking

In stacking, the generation of one model is controlled by the actual combination of disparate models through a meta-model. For example, the usage of a meta-model based on Logistic Regression for an ensemble consisting of model types Random Forests, ANNs, and other types can strengthen a fraud detection system. Finally, the attractive feature of some stacking ensemble methods is their potential to leverage the strengths of very different kinds of models in developing better system performance.

2. Ensemble Learning

Fraud detection models demonstrated some potential in adopting an ensemble learning technique, taking an already good model accurate and robust with combinations of algorithms in choosing Random Forests and XGBoost. Some of the variance reduction methods are as follows. They're performing much better versus what any one model can do by itself. It really demonstrates well the power of the algorithms as state-of-the-art regression and classification tools.

Still worth noting: it's only enthusiasm over the ensemble methods in and of themselves that ranks second to a thing we know we can do—build a robust, adaptive model able to detect the many forms fraud takes these days.

In other words, the methodology and related work sections are very interrelated. The methodology section outlines the concrete steps of building and deploying a fraud detection system; the related work lists almost exhaustively strengths and weaknesses of various machine learning algorithms which can serve as the foundation for such a system. If you will ever be working with any of those algorithms, you must know their limitations and relative advantages.

III. Methodology

This section outlines how this methodology was adapted for the following research question: "How will machine learning methods go about in efficiently detecting credit card fraud, keeping perspective on problems of data imbalance, evolving fraud patterns, and applicability in real time?" It consists of three phases in the methodology, namely, dataset preparation, model development, and its evaluation.

A. Data Preprocessing

It basically prepares the data, dealing with an imbalance problem in a class, its quality, and efficiency of model training.

1. Data Source and Characteristics

The major dataset to be utilized is the Kaggle Credit Card Fraud Detection Dataset [1]. Key characteristics include:

- **Size:** 284,807 transactions of which 492 are labelled fraudulent (~0.17%).
- **Features:** 30 anonymized numerical features produced by PCA are combined with Time and Amount.
- **Class Labels:**
- **0:** Valid transaction.
- **1:** Fraudulent transaction.

2. Data Cleaning

- **Handling Missing Values:** The given dataset is complete, but in case of entries, an elongated dataset received. Missing at places, numerical features will consider median imputation, and for categorical features mode imputation will be considered.

- **Outlier Analysis:**

- **Z-score analysis** (threshold >3) will be used to identify outliers in transaction amounts.

- Those would otherwise be outlier data points that flag up for analysis for a pattern indicative of fraud.

3. Feature Scaling and Transformation

First, numeric features will be standardized using StandardScaler; it regularizes their contributions. Then, Time will be converted to time-interval-like transactions within 1-hour window sizes for the possible time-based analysis.

4. Handling Class Imbalance

- **Synthetic Oversampling:**

- **SMOTE:** Synthetic Minority Over-sampling Technique, this will create synthetic examples for the minority class while preserving the relationship [5].

- **Random Undersampling:**

- Oversampling should rather under-sample it by choosing randomly the majority class to avoid the overfitting from the result of oversampling [11].

- **Cost-Sensitive Learning:**

- Class weights will be included in the loss functions of the machine-learning models in order to penalize misclassification of minority-class samples [12].

5. Feature Engineering

Feature engineering allows improving model interpretability and performance with new features:

- **Simplification:**
  - Frequency of user transaction in a certain time period.
  - Average transaction value over 24 hours.
- **Geolocation and Distance Metrics:**
  - Output would include, when possible, the geographical distance between successive transactions.
- **Clustering-Based Anomalies:**
  - K-means clustering groups transactions, and those falling out in the region of low density will mark potential anomalies.

B. Model Development and Implementation

Traditional and advanced models balance between accurate interpretability and computational efficiency, each populating machine learning pipelines.

1. Reference Model

- **Logistic Regression:**
  - Offers comprehensible outputs and serves as a benchmark for comparison across performances.
  - It will involve L1 or L2 to regularize and avoid overfitting.

2. Tree-based Models

- **Decision Trees:**
  - Geometrical boundaries can be visualized and, although simple, it’s very powerful for binary classification; suffers from the problem of overfitting.
- **Random Forests:**
  - Ensembling of Decision Trees reduces variance, hence making it more robust.

    - **Preprocessing:** Hyperparameters that can be tuned are the number of estimators and maximum depth for both.

- **Gradient Boosting:**

    - **XGBoost and LightGBM:** Both of these techniques would be used as they can efficiently learn complex feature interactions in high imbalance datasets [13].

3. Deep Learning Models

- **Artificial Nutral Networks:**

    - Multi-layer perceptrons will use ReLU activation and dropout as its regularizer.

- **Autoencoders:**

    - Autoencoder could be applied in training anomaly detection, which would unmask these fraudulent transactions based on reconstruction error [14].

- **LSTM-Long Short-Term Memory:**

    - LSTMs could model temporal dependencies between sequences of transactions, hence be suitable for the time series analysis itself [15].

4. Hybrid Models

- **Stacking:**

    - It integrates many models with the help of Random Forests, ANN, and many more using a meta-model employing Logistic Regression to perform well [12].

- **Ensemble Learning:**

    - Ensemble algorithm-based techniques that use the Random Forest and XGBoost algorithms.

IV. Evaluation

This makes the performance for the models measurable, interpretable, and suitable for a real-world deployment.

A. Indicators

- **Precision:** Evaluates how many flagged transactions are truly fraudulent, minimizing false positives.
- **Recall (Sensitivity):** Measures the ability to detect all fraudulent transactions, minimizing false negatives.
- **F1-Score:** This is a better balance between precision and recall for an overall measure.
- **AUC-ROC Curve:** A plot that shows the relation between the true positive rate and the false positive rate for different thresholds.

B. Testing Approach

- **Stratified k-Fold Cross-Validation:**
  - Ensures consistent class distributions across the training and validation folds [16].
- **Out-of-Sample Test Set:**
  - It cross-validates to hold out 15% of the dataset for the final testing.

C. Real-Time Applicability

- **Latency vs. Throughput:**
  - Inference times of the models have to be measured and optimized for real-time systems [7].
- **Scalability Analysis:**
  - Batch processing and online learning would be reviewed for scalability of big datasets.

V. Conclusions and Future Work

This project presents a general framework for credit card fraud detection against highly imbalanced datasets using machine learning. In addition to this, this project consists of more robust data preparation methodology, class imbalance through SMOTE, cost-sensitive learning, and feature engineering with high predictive accuracy.

Those models ranged from the simplest logistic regressions, tree-based methods, to autoencoders and LSTMs. Different metrics measure both the accuracy and scalability of the solutions, including but not limited to precision, recall, F1-score, and AUC-ROC.

These techniques will enable the following direct research questions to be answered:

1. **How might machine learning methods detect fraudulent transactions?**
   - These techniques listed hereby identify subtle patterns that exist in transactional data, which, in the end, identify fraudulent and nonfraudulent transactions [3].
2. **How can one handle data imbalance and/or a shift in fraud patterns?**
   - Class imbalance is handled using SMOTE and cost-sensitive models, whereas evolving fraud patterns are handled using the introduction of adaptable machine learning techniques combined with real-time validation strategies [5], [6].

Critical implications include:

- Better fraud catching rates by reducing false negatives (missed fraud) and false positives (legitimate transactions marked as fraud).
- These are now scalable and efficient methods for real-world financial applications.

Limitations

Despite these strengths, some of the weaknesses of the proposed method include:

1. **Dependence on Data Quality:** The other major factor governing performance is the quality, variety and representativeness of the data set. Fraud patterns in real life are very different from those occurring in publicly available data sets as well [8].
2. **Dynamic Fraud Patterns:** The fraudulent patterns tend to vary quite frequently. In that respect, the static models may thus poorly perform after some time. The ensemble and deep learning methods, though adaptable, would require retraining in such a case continuously with updated data [6].

3. **Computational Cost:** The computational cost of deep learning and hybrid models is yet unaffordable in the context of low-cost implementations [9].

4. **Explainability:** Neural network models, just like the ensemble methods, represent black boxes that normally are unwanted by the stakeholders of regulated industries [10].

References

[1] Kaggle, "Credit Card Fraud Detection Dataset," 2016. [Online]. Available: https://www.kaggle.com/mlg-ulb/creditcardfraud

[2] The Nilson Report, "Global Card Fraud Losses Reach $28.65 Billion," Issue 1197, August 2021.

[3] R. J. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," *Statistical Science*, vol. 17, no. 3, pp. 235–255, 2002.

[4] N. Japkowicz and S. Stephen, "The Class Imbalance Problem: A Systematic Study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[5] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[6] J. T. Quah and M. Sriganesh, "Real-time Credit Card Fraud Detection Using Computational Intelligence," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1721–1732, 2008.

[7] D. A. Zuech, T. M. Khoshgoftaar, and R. Wald, "Intrusion Detection and Big Heterogeneous Data: A Survey," *Journal of Big Data*, vol. 2, no. 1, p. 3, 2015.

[8] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," in *2015 IEEE Symposium Series on Computational Intelligence*, 2015, pp. 159–166.

[9] A. K. Jain, J. Mao, and K. M. Mohiuddin, "Artificial Neural Networks: A Tutorial," *Computer*, vol. 29, no. 3, pp. 31–44, 1996.

[10] Z. C. Lipton, "The Mythos of Model Interpretability," *Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[11] G. E. Batista, R. C. Prati, and M. C. Monard, "A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data," *SIGKDD Explorations*, vol. 6, no. 1, pp. 20–29, 2004.

[12] C. Elkan, "The Foundations of Cost-Sensitive Learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, 2001, pp. 973–978.

[13] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[14] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[15] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] R. Kohavi, “A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection,” in *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, vol. 2, 1995, pp. 1137–1143.