Feature engineering strategies based on a One-point Crossover for fraud detection on Big Data Analytics

M Soleh*, E R Djuwitaningrum1, M Ramli1 and M Indriasari1

1 Prodi Informatika, Institut Teknologi Indonesia, Institut Teknologi Indonesia, Jl. Raya Puspiptek, Setu, Kec. Serpong, Kota Tangerang Selatan, Banten 15314, Indonesia

*Email: muhamad.soleh@iti.ac.id

Abstract. A wide range of new opportunities for fraudulent online activities has arisen with the growing popularity of online shopping and big data issue. E-payment fraud schemes are collecting billions of dollars from customers, distributors and service providers every year. A lot of machine learning methods for fraud detection problems have been proposed which can be categorized into supervised, unsupervised and semi-supervised methods. In this paper, we proposed biologically inspired technic in the feature engineering phase for handling imbalanced data by oversampling. One-point crossover used to generate the new data of minority classes. The best algorithm performance obtained to predict the fraud transaction from various machine learning models is Classification and Regression Tree with the corresponding accuracy, precision, recall, and F-1 Score are 96%.

Keyword: Fraud detection, One-point-crossover, Machine learning, Big data, Oversampling

1. Introduction
The identification of fraud is a hot topic among the societies of data mining. Credit card fraud identification gets the most academic research in the field of financial transactions [1]. A wide range of new opportunities for fraudulent online activities has arisen with the growing popularity of online shopping. E-payment fraud schemes are collecting billions of dollars from customers, distributors and service providers every year. Damage caused by electronic payment fraud throughout the world has risen from $7.6 billion in 2010 to $21.81 billion in 2015, reflecting an increase of 300 million over five years. In 2020, fraudulent damage in the banking sector is expected to reach up to $31.67 billion [2]. As the digital fraud study Cyber Source confirmed in 2013, the total loss of online sales due to fraud in the United States and Canada reached a remarkable figure of $3.5 billion in 2012, a rise of $100 million compared to 2011 [3]. In another place, there are over 70 million credit cards in mainland China by the end of 2007. But with the increasing credit card industry, in recent years, the crime of credit card fraud has increased [4]. Many strategies for detecting fraud in financial transactions have been developed recently.

Several methods of fraud detection have been discussed in the literature to reduce as much as possible such financial losses. One of the most common strategies for fraud detection is anomaly-based approaches in which each cardholder's evaluated spending behavior makes up the cardholder profile. Any incoming payment that contradicts the profile of the cardholder would be deemed suspicious [5]. Traditional methods of identification rely primarily on the database system and...
consumer training, which are typically slow, unreliable and not in-time. Methods are then commonly used based on differential analysis and regression analysis which can detect interest rate fraud for cardholders and credit card transactions, however, with a large data gap. In recent years, the dominant data mining concerns credit card fraud detection systems based on data mining [6].

A lot of machine learning methods for fraud detection problems have been proposed which can be categorized into supervised, unsupervised and semi-supervised methods. We know both fraudulent and legitimate transactions in past data in supervised learning, we believe that the future is close to the past. But fraudsters are constantly changing their tactics. Supervised methods are impotent in detecting fraud of a new type. We do not know in the past data about fraudulent or legitimate transactions in unsupervised education, yet we know that fraud is distinct from most transactions. Unmonitored approaches can detect new kinds of fraud. Yet their false alarm rate is usually lower than the methods being monitored. Semi-supervised approaches are attempting to benefit from both supervised and unsupervised learning [7].

This area became a fascinating field for researchers in addition to the interest of the financial institutions in the development of a fraud detection program. The explanation for this concern is the information imbalance, time dependency between samples, idea drift and real-time detection. Besides that, the difficulty of this issue is to detect fraud in a large dataset where legitimate transactions are more common and fraudulent transactions are small or near negligible [8]. Different methods have been proposed to solve imbalanced issues. More use is made of over-sampling, under-sampling and SMOTE, other sampling methods such as GANs used to increase the number of samples of minority classes [9].

In this paper, we proposed biologically inspired technic in the feature engineering phase for handling imbalanced data to increase the total data of a small number of classes. One-point crossover used to generate the new data of minority classes. Crossover plays a major role in searching based on the Genetic Algorithm (GA). In the literature, there were many statistical correlations of different crossover operators [10]. In this paper, we used a one-point crossover to oversampling the dataset. GA is a technique of optimization, motivated by the natural evolution theory. The GA process begins by establishing a pool of candidate solutions (called a population) followed by assigning each of them a fitness score [11].

2. Research methodology

This research was motivated by one of the completed Kaggle competitions. Kaggle is a known machine learning competition organizer. Researchers from the IEEE Computational Intelligence Society (IEEE-CIS) want to improve this figure, while also improving the customer experience. With higher accuracy fraud detection without the hassle.

2.1. Dataset

The data set for this paper was provided by Vesta Corporation. Vesta Corporation is the predecessor to assured payment solutions for e-commerce. Founded in 1995, Vesta pioneered the system of the telecommunications industry's fully guaranteed card-not-present (CNP) payment transactions. The data comes from real-world e-commerce purchases and provides a wide range of features ranging from device type to product features. In this paper, we used a demanding large-scale dataset to test machine learning models by oversampling datasets to boost our performance [12].

In this dataset, there are two tables, the transaction and identity table. The two tables can be mixed according to the transaction id. Transaction Table consist of some features:

- TransactionDT: timedelta from the reference date (not the actual timestamp)
- TransactionAMT: transaction payment sum in USD
- ProductCD: product code, product for each transaction
- card1 — card6: payment card details such as card size, card class, issuance account, nation, etc.
- Addr: address
• Distance:
• P and R email; purchaser and recipient email domain
• C1 – C14: counting, such as how many addresses the payment card contains
• D1-D15: timedelta, such as previous payment days, etc.
• M1-M9: match, like card and address names, etc.
• Vxxx: Vesta has developed a wealth of features, including ranking, counting, and other relationships. Where xxx is the number starts from 001 and so on.

These features are divided into categorical and numerical features. The following are categorical features form those feature ProductCD, card1 - card6, addr1, addr2, Pemaildomain Remaildomain, and M1 - M9. The rest are numerical features. Variables in identity table are transaction-related identity data (IP, ISP, Proxy, etc.) and digital signature (UA / browser / os / version, etc.). Categorical Features from identity tables consist of DeviceType, DeviceInfo, and id12 - id38 [12].

2.2. Pre-processing
Data is often tidy and homogeneous in the real world. Usually, they appear to be incomplete, noisy, and inconsistent, so filling missing values is an important task for a data scientist to prepossess the data. The real-world dataset was too complex. There are some missing data we have to carry out. It is important to manage them because they could lead to incorrect predictions or classification [13].

Basically, there are some other steps for handling this problem. First, we can ignore the data raw as well as provided. This is a quick solution and is favored in situations where the amount of missing values (< 5 percent) is relatively low. As you lose records, it's a dirty approach. We can also do backfilling or forward filling, in order to spread the next or previous values. Besides those scenarios, filling with the constant value would be a good idea to handle the missing value. In this research, we used the median and mean from each feature to fill the missing data [14].

2.3. Feature engineering
A solution to mitigate the imbalanced class problem is to reduce the difference among classes by under-sampling the majority class in a training set randomly eliminating certain instances of the majority class on the premise that information consistency renders such exclusion meaningless for classification purposes. Even trying to over-sample the minority class by replicating minority class instances. Such a strategy's downside is that it does not incorporate insightful content, thereby restricting a classifier's ability to generalize change [9,15]. In this research, we do the oversampling for replicating the instance of minority class using One-Point-Crossover.

One-Point-Crossover is a basic technique for crossover. This chooses a breakpoint randomly in the chromosomes of children. The first offspring is produced by the first parent concatenating the first part and the second parent's last part. Likewise, another offspring comes from the first part of the second and the last part of the first part as sown in figure 1. Each child comes from the reproduction process assume to have identical biologic gen with their parent [10-11].

![Parent and Offspring](image)

**Figure 1.** One-Point-Crossover Process to generate new data.
2.4. Modeling and prediction

Machine learning is a subset of artificial intelligence. In machine learning, there are three tasks to solve related to the problems. Those are Regression, Clustering, and Classification. Regression is a technique to predict a continuous value given some data observation. Otherwise, classification is used to predict a discrete value given data observation. Both are known as supervised learning. In Machine learning generally, the process of learning algorithm consist of supervised and unsupervised learning. Supervised learning needs a label or class as a supervisor while learning observed data training to build the model. Otherwise unsupervised no need any label or class. The unsupervised technique is well known as the clustering process.

Fraud detection was developed to predict while the transaction was fraud or not. Those in this research or paper will tend to use the classification model of machine learning instead of clustering. But, in Kaggle competition we have to predict the probability of transaction is being a fraud or not. So we have to use some modifications to overcome this case. Nowadays, machine learning or artificial intelligence become very popular. Most of the research in computer science talking about this topic. As long as big data become a more interesting topic, machine learning would take part in this hip.

Neural network, support vector machine, k nearest neighbors, decision tree, logistic regression, naïve Bayes, and other methods are some familiar algorithms in supervised machine learning. Evaluating our feature engineering strategies based on one-point-crossover would be done by using those algorithms. Compare to the original one, we use some evaluation matrix to show how significant our proposed method can increase robustness in our model.

Accuracy, Precision, Recall, and F1-Score would be appropriate matrix related to the imbalanced dataset for the original one. A confusion matrix is a table often used to define a classification model's performance on a collection of test datasets for the true values as shown in Table 1. It enables the show of an algorithm's performance by finding each Accuracy, Precision, Recall, and F1-Score from this matrix.

| True Label | TP = 45 | FN = 5 |
| False Label | FP = 5 | TN = 45 |
| T = 100 | True Prediction | False Prediction |

\[
\begin{align*}
\text{Precision} & = \frac{TP}{TP+FP} \\
\text{Recall} & = \frac{TP}{TP+FN} \\
\text{Accuracy} & = \frac{TP+TN}{TP+TN+FP+FN} \\
F1 \text{ Score} & = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}
\end{align*}
\]

Definition of the Terms:

- True Positive (TP): label is a fraud, and is predicted to be a fraud.
- False Negative (FN): label is a fraud, but is predicted not a fraud.
- True Negative (TN): label is not fraud, and is predicted to be not a fraud.
- False Positive (FP): label is not fraud, but is predicted fraud.
- Total (T): Total number of training datasets.

Accuracy is defined by finding the ratio of the total data correctly predicted both positive and negative with all along with the testing dataset (T). F1 Score is a harmonic average of precision and recall. Precision is inferred from all the expected positive cases as the indicator of the correctly predicted positive cases. The recall is the indicator of positive cases from all the actual positive cases that are correctly identified. Accuracy can be used when the distribution of classes is similar, while
F1-score is a better metric when classes are imbalanced as in the fraud case to handle false positive and negative.

3. Result and analysis
In this section, we will explain the result and our analysis consists of data exploration, data statistics, handling missing value, feature engineering, and classification performance result.

3.1. Data exploration
First of all, we have to consider the total data used in this research. The dataset consists of 590540 instances with 432 related features and 1 class. Class is labeled with 0 and 1, 0 is corresponding to not fraud and 1 is fraud transaction. There are 569877 instances labeled with 0 otherwise only 20663 belong to 1. Figure 2 shows the unique samples consist of nan and categorical instance, which means in our dataset there some data are missing. Each data missing can be seen in table null from figure 2. Since most of the machine learning algorithms not allowed any missing data, so we have to handle it as well as possible. The categorical instance must be transformed into a numerical feature, so a machine learning algorithm can process them as well.

| dataFeatures | dataType | null | nullPct | unique                     | uniqueSample                  |
|--------------|----------|------|---------|----------------------------|------------------------------|
| 0            | card1    | 0    | 0.00    | 13553                      | [1444, 8113, 16533, 9500, 6394, 18132, 14642, ...]|
| 1            | card2    | 8933 | 1.51    | 500                        | [nan, 321.0, 170.0, 562.0, 111.0, 490.0, 408.0,...]|
| 2            | card3    | 1565 | 0.27    | 114                        | [150.0, 185.0, nan, 143.0, 117.0, 144.0, 203.0,...]|
| 3            | card4    | 1577 | 0.27    | 4                         | [visa, mastercard, discover, american express,...]|
| 4            | card5    | 4259 | 0.72    | 119                        | [226.0, 147.0, 224.0, 166.0, 202.0, 195.0, 162,...]|
| 5            | card6    | 1571 | 0.27    | 4                         | [debit, credit]               |

![Figure 2](image1.png)

Figure 2. Samples data exploration.

### 3.1.1. Handling missing value
Missing data would be a problem when we running a machine-learning algorithm to classify fraud detection. In this research, we used the mean and median value for each data missed. Firstly, we have to consider a large number of missing values by using a threshold to remove the data feature from our dataset. Those features are worry to affect the classification model. We used 50% of the missing value threshold. Figure 3 shows us there is no more missing value in our dataset. Figure 3 also shows us no more categorical features in unique sample tables.

| dataFeatures | dataType | null | nullPct | unique                     | uniqueSample                  |
|--------------|----------|------|---------|----------------------------|------------------------------|
| 0            | card1    | 0    | 0.0     | 13553                      | [8755, 6489, 6832, 1030, 10856, 4272, 6650, 81,...]|
| 1            | card2    | 0    | 0.0     | 501                        | [111.0, 321.0, 455.0, 298.0, 348.0, 490.0, 418,...]|
| 2            | card3    | 0    | 0.0     | 115                        | [150.0, 185.0, 153.2, 203.0, 146.0, 147.0, 143,...]|
| 3            | card4    | 0    | 0.0     | 5                         | [4, 3, 2, 1, 0]               |
| 4            | card5    | 0    | 0.0     | 120                        | [117.0, 102.0, 226.0, 195.0, 224.0, 166.0, 126,...]|
| 5            | card6    | 0    | 0.0     | 5                         | [3, 2, 0]                     |

Figure 3. Samples handing missing value.
3.1.2. Data statistics from a numerical feature. The statistical report of the dataset would be necessary since we have to replace the missing data by any value related to the characteristic of our data. The statistical report hopefully will help us to adjust the best value. It would make sense if we have the minimal value 0 and the maximal value is 1, so we are not allowed to replace the missing value with 0 or 1. This can lead to bias in performing our model. Figure 4 shows the sample of statistical data distribution from our dataset.

| isFraud | TransactionDT | TransactionAmt | card1 | card2 | card3 | card4 | card5 | addr1 | addr2 | P_emaildomain |
|---------|---------------|----------------|-------|-------|-------|-------|-------|-------|-------|---------------|
| count   | 590540.000000 | 5.905400e+05   | 590540.000000 | 590540.000000 | 581607.000000 | 588975.000000 | 586281.000000 |
| mean    | 0.034000      | 7.372311e+06   | 135.027176    | 9898.734658    | 362.555488     | 153.104025     | 199.278897    |
| std     | 0.183755      | 4.617224e+06   | 239.162522    | 4901.170153    | 157.793246     | 11.336444     | 41.244453     |
| min     | 0.000000      | 8.640000e+04   | 0.000000      | 0.000000       | 0.000000       | 0.000000       | 0.000000      |
| 25%     | 0.000000      | 3.027052e+06   | 43.321000     | 6219.000000     | 214.000000     | 150.000000     | 166.000000    |
| 50%     | 0.000000      | 7.306523e+06   | 68.769000     | 9678.000000     | 361.000000     | 150.000000     | 226.000000    |
| 75%     | 0.000000      | 1.124662e+07   | 125.000000    | 14184.000000    | 512.000000     | 150.000000     | 226.000000    |
| max     | 1.000000      | 1.581113e+07   | 31937.391000  | 18396.000000    | 600.000000     | 231.000000     | 237.000000    |

8 rows × 402 columns

**Figure 4.** Samples data statistic.

3.1.3. Feature engineering based on one-point-crossover. Feature engineering used to generate new data for oversampling the minority class from our dataset. This research use one-point-crossover to reproduce the data. Figure 5 describes the process of oversampling the minority class by using the one-point-crossover operator. Parent from this process was to get from the original dataset, and offspring is the new data gotten from one-point-crossover. We are assuming that offspring would be inherited characteristics from their parent from the same class.

Parent:

```
| TransactionDT | TransactionAmt | ProductCD | card1 | card2 | card3 | card4 | card5 | addr1 | addr2 | P_emaildomain |
|---------------|----------------|-----------|-------|-------|-------|-------|-------|-------|-------|---------------|
| 0             | 8889157        | 400.0     | 1     | 12807 | 172.0 | 150.0 | 4     | 226.0 | 3     | 191.0         | 87.0 | 36 |
| 1             | 6007144        | 200.0     | 2     | 15497 | 490.0 | 150.0 | 4     | 226.0 | 3     | 299.0         | 87.0 | 17 |
| 2             | 6865613        | 196.0     | 4     | 11167 | 211.0 | 150.0 | 3     | 327.0 | 2     | 254.0         | 87.0 | 54 |
| 3             | 7164923        | 78.5      | 4     | 2456  | 321.0 | 150.0 | 4     | 226.0 | 2     | 244.0         | 87.0 | 54 |
| 4             | 3215669        | 687.0     | 4     | 10516 | 555.0 | 150.0 | 4     | 226.0 | 2     | 441.0         | 87.0 | 17 |
```

5 rows × 218 columns

Offspring:

```
| TransactionDT | TransactionAmt | ProductCD | card1 | card2 | card3 | card4 | card5 | addr1 | addr2 | P_emaildomain |
|---------------|----------------|-----------|-------|-------|-------|-------|-------|-------|-------|---------------|
| 0             | 5968682        | 77.000000 | 4     | 17373 | 490.0 | 150.0 | 4     | 226.0 | 3     | 310.0         | 87.0 | 20 |
| 1             | 2216099        | 12.71938  | 0     | 13832 | 375.0 | 185.0 | 3     | 224.0 | 3     | 291.0         | 86.625| 20 |
| 2             | 10100827       | 58.375000 | 9     | 9917  | 142.0 | 185.0 | 4     | 138.0 | 3     | 291.0         | 86.625| 17 |
| 3             | 10100783       | 56.375000 | 9     | 9917  | 142.0 | 185.0 | 4     | 138.0 | 3     | 291.0         | 86.625| 17 |
| 4             | 4224213        | 97.000000 | 4     | 17186 | 321.0 | 150.0 | 4     | 226.0 | 3     | 122.0         | 87.0 | 0  |
```

5 rows × 219 columns
3.1.4. Classification performance on various machine learning algorithms

Table 2. Classification performance result.

|                | Oversampling (One Point Crossover) | Without Oversampling |
|----------------|-----------------------------------|----------------------|
|                | LDA  | CART | LR  | NB  | KNN | LDA  | CART | LR  | NB  | KNN |
| Precision      | 76   | 96   | 67  | 39  | 90  | 73   | 75   | 56  | 47  | 68  |
| Recall         | 71   | 96   | 51  | 49  | 94  | 62   | 77   | 50  | 48  | 53  |
| F1 Score       | 73   | 96   | 43  | 25  | 91  | 65   | 76   | 49  | 5   | 55  |
| Accuracy       | 78   | 96   | 68  | 32  | 92  | 96   | 97   | 96  | 5   | 96  |

Definition of the Terms:
LDA: Linear Discriminant Analysis, CART: Classification and Regression Tree, LR: Logistic Regression, NB: Naïve Bayes, and KNN: K-Nearest Neighbors.

Table 2 describes the comparison from various machine learning classification performance between oversampling based on one-point-crossover and without oversampling. Most of the proposed methods used in this research show better performance for handling imbalanced datasets with corresponding to the higher F1 score. Classification and Regression Tree would be of a great and suitable algorithm for fraud detection by using oversampling based on one-point-crossover. This research conduct using google research collab.

4. Conclusion
This paper aims to recognize the fraud using various machine learning algorithms combined with feature engineering based on the one-point-crossover operator in a genetic algorithm. To achieve this objective, the dataset was taken from Kaggle competition with a huge dimension of instances and features. One-point cross over operator is used to oversampling the minority class of fraud to create a more robust model. The best algorithm performance obtained from various machine learning models is CART with the corresponding accuracy, precision, recall, and F-1 Score are 96%.
Acknowledgment
This research is supported and funded by LLDIKTI III with contract number: 225/SP2H/LT/DRPM/2019, 4/AKM/PNT/2019, and 048/KP/LPKT-ITI/IV/2019. The title of the research grant is “Rancang Bangun Purwarupa Laboratorium Big Data”.

References
[1] Yuh-Jen Chen and Chun-Han Wu, “On Big Data-based Fraud Detection Method for Financial Statements of Business Groups”, 6th IIAI International Congress on Advanced Applied Informatics, 2017
[2] Mohamad Zamini and Gholamali Montazer, “Credit Card Fraud Detection using autoencoder based clustering”, 9th International Symposium on Telecommunications (IST), 2018
[3] Kunlin Yang, “A Memory-Enhanced Framework for Financial Fraud Detection”, 17th IEEE International Conference on Machine Learning and Applications, 2018
[4] Chun-Hua JU and Na Wang, “Research on Credit Card Fraud Detection Model Based on Similar Coefficient Sum”, 1st International Workshop on Database Technology and Applications, 2009
[5] N.Malini and Dr.M.Pushpa, “Analysis on Credit Card Fraud Identification Techniques based on KNN and Outlier Detection”, 3rd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEEICB) 2017
[6] Dr. Radhakrishna Rambola, Prateek Varshney, and Prashant Vishwakarma, “Data Mining Techniques for Fraud Detection in Banking Sector” 4th International Conference on Computing Communication and Automation (ICCCA), 2018
[7] Xing Min and Rongheng Lin, “K-Means Algorithm: Fraud Detection Based on Signaling Data”, IEEE World Congress on Services, 2018
[8] Donia Malekian, Mahmoud Reza Hashemi, “An adaptive profile-based fraud detection framework for handling concept drift”, 10th International ISC Conference on Information Security and Cryptology (ISCISC), 2013
[9] Ugo Fiore, Alfredo De Santis, Francesca Perla, Paolo Zanetti, Francesco Palmieri, “Using Generative Adversarial Networks for Improving Classification Effectiveness in Credit Card Fraud Detection”, Information Sciences (2017), doi: 10.1016/j.ins.2017.12.030
[10] Dilok Pumsuwan, Sunisa Rimcharoen and Nutthanon Leelathakul, “Front-Rear Crossover: A New Crossover Technique for Solving a Trap Problem”, 14th International Joint Conference on Computer Science and Software Engineering (JCSSE), 2017
[11] Michal Przewozniczek, “Towards finding an effective uniform and single-point crossover balance for optimization of Elastic Optical Networks”, Second European Network Intelligence Conference, 2015
[12] www.kaggle.com/c/ieee--fraud-detection
[13] Soheil Jamshidi and Mahmoud Reza Hashemi, “An Efficient Data Enrichment Scheme for Fraud Detection Using Social Network Analysis”, 6th International Symposium on Telecommunications, 2012
[14] www.towardsdatascience.com/how-to-handle-missing-data
[15] Simon Delecourt and Li Guo, “Building a robust mobile payment fraud detection system with adversarial examples”, IEEE 2nd International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), 2019