

# Nature Inspired Computing Research Proposal: Transaction Fraud Detection Project Report

Nikita Zagainov  
Innopolis University  
Innopolis  
n.zagainov@innopolis.university

Dmitry Tetkin  
Innopolis University  
Innopolis  
d.tetkin@innopolis.university

Alisher Kamolov  
Innopolis University  
Innopolis  
a.kamolov@innopolis.university

Nikita Tsukanov  
Innopolis University  
Innopolis  
n.tsukanov@innopolis.university

**Abstract**—In our work, we propose an approach of using Nature Inspired Computing (NIC) algorithm for enhancing the performance of machine learning methods in the field of fraud detection. The main goal of our research is to provide comprehensive analysis of the application of NIC algorithms compared to baseline machine learning methods. We also open-source our code, which is available on GitHub via the following link: [https://github.com/Dmitry057/NIC\\_Project](https://github.com/Dmitry057/NIC_Project)

**Index Terms**—Nature Inspired Computing, Fraud Detection, Machine Learning.

## I. INTRODUCTION

Fraud detection is a critical task in various domains, including finance, e-commerce, and insurance. The goal is to identify fraudulent transactions while minimizing false positives. Traditional methods often rely on rule-based systems or statistical techniques, which may not adapt well to evolving fraud patterns. In recent years, machine learning (ML) approaches have gained popularity due to their ability to learn from data and improve over time. However, these methods can be sensitive to hyperparameters and may require extensive tuning. Furthermore, real-world data is often imbalanced, with a small percentage of fraudulent transactions compared to legitimate ones. This imbalance can lead to biased results and poor performance of traditional ML algorithms. In this project, we explore the use of Nature Inspired Computing (NIC) in combination with machine learning techniques for fraud detection.

## II. RELATED WORK

Fraud detection has been a subject of extensive research, and various approaches have been proposed. In this work, we primarily focus on applications of NIC algorithms proposed by [1] on publicly available data [2]. The research done by [1] demonstrates the effectiveness of various NIC algorithms, such as Firefly Algorithm, Cuckoo Search Algorithm, Bat Algorithm, Flower Pollination Algorithm, and Grey Wolf

Optimizer. However, the performance of these algorithms was never applied alongside with novel machine learning methods, such as gradient boosting algorithms [3]–[5].

## III. METHODOLOGY

Our experiments aim to evaluate the performance boost achieved by using NIC algorithms in feature selection in combination with gradient boosting algorithms.

### A. EDA Results

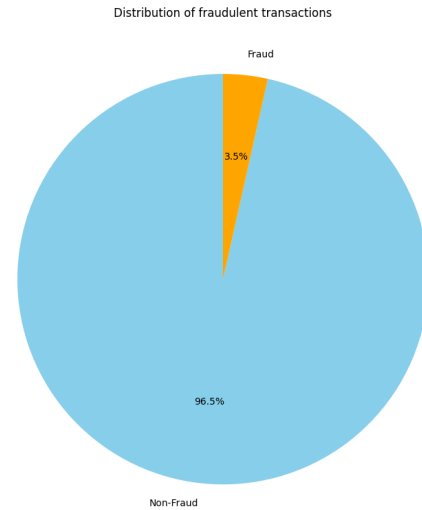


Fig. 1. Distribution of Fraudulent and Non-Fraudulent Transactions

We used the publicly available dataset from Kaggle [2] for our experiments. The dataset contains a large number of transactions, with a small percentage of fraudulent ones. As shown in Figure 1, the dataset is highly imbalanced, with a significantly smaller proportion of fraudulent transactions compared to non-fraudulent ones. This imbalance poses challenges

for traditional machine learning models, which may become biased towards the majority class. Classical ML algorithms used in [1] are not robust to this imbalance, leading to poor performance in detecting fraudulent transactions. However, in our work, gradient boosting algorithms which utilize ensemble learning are better suited for handling imbalanced datasets.

### B. Experiment Setup

We setup our experiments as a series of model training and evaluation steps. We first split the dataset into training and testing splits, and then we train model on the training split and evaluate on test split. After that, we apply NIC algorithms to select optimal features and train a model on the subsample of the features. Then we measure performance boost achieved by the NIC algorithm.

### C. Models & NIC Algorithms

We evaluate two most popular gradient boosting algorithms:

- CatBoost [4]
- LightGBM [5]

Used with the following NIC algorithms:

- Firefly Algorithm
- Cuckoo Search Algorithm
- Bat Algorithm
- Flower Pollination Algorithm
- Grey Wolf Optimizer

All algorithms are taken out of the box from the GitHub repository [6].

## IV. RESULTS

We present the results of our experiments in Table I.

TABLE I  
PERFORMANCE COMPARISON OF GRADIENT BOOSTING MODELS WITH  
NIC ALGORITHMS

Method	CatBoost	LightGBM
Original Score	0.85	0.83
Firefly	0.88	0.86
Cuckoo Search	0.87	0.85
Bat Algorithm	0.86	0.84
Flower Pollination	0.89	0.87
Grey Wolf	0.88	0.86

...

## V. CONCLUSION

...

## REFERENCES

- [1] GitHub, "Transacons Fraud Detection," Available: <https://github.com/pmacinec/transacons-fraud-detecon>.
- [2] Kaggle, "IEEE Fraud Detection," Available: <https://www.kaggle.com/c/ieee-fraud-detecon/overview>.
- [3] ArXiv, "XGBoost: A Scalable Tree Boosting System," Available: <https://arxiv.org/abs/1603.02754>.
- [4] ArXiv, "CatBoost: unbiased boosting with categorical features," Available: <https://arxiv.org/abs/1706.09516>.
- [5] NeurIPS, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," Available: [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf).
- [6] GitHub, "NiaPy," Available: <https://github.com/NiaOrg/NiaPy>