International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

A Voting Classifier Approach for Hourly Passenger Boarding Demand Prediction Using Synthetic Data.

Gangavarapu Uday Kumar Reddy
Computer Science and Engineering
R.V.R & J.C College of Engineering, Guntur, India
gangavarapuuday46.kr@gmail.com

Gollamandala Syam Raju
Computer Science and Engineering
R.V.R & J.C College of Engineering, Guntur, India
gollamandalasyam64@gmail.com

Gudla Hemanth
Computer Science and Engineering
R.V.R & J.C College of Engineering, Guntur, India
gudlahemanth6@gmail.com

Budida Manohar
Computer Science and Engineering
R.V.R & J.C College of Engineering, Guntur, India
manoharbudida@gmail.com

Dr. Ch. Aparna (Professor)
Computer Science and Engineering
R.V.R & J.C College of Engineering, Guntur, India
aparna@rvrjc.ac.in

***Abstract*— Smart-card data has emerged as a valuable source for analyzing passenger boarding behavior and predicting future public transit demand. However, acquiring real-world smart-card data poses challenges due to privacy and availability constraints. This study addresses this limitation by generating a synthetically simulated smart-card dataset that mimics hourly passenger boarding patterns at bus stops. One of the major challenges in such datasets is class imbalance, where the number of non-boarding instances significantly outweighs the boarding instances. To tackle this, we propose a voting-based ensemble learning approach that combines multiple classifiers to enhance prediction performance. Unlike prior research which employed Deep Generative Adversarial Networks (GANs) and Deep Neural Networks (DNNs) on real datasets, our approach focuses on classical ensemble methods to offer interpretability and computational efficiency. Experimental results on the simulated dataset demonstrate the effectiveness of the voting classifier in addressing class imbalance and predicting hourly boarding demand. This paper provides an alternative perspective on demand prediction using ensemble techniques and synthetic data, highlighting its potential for future real-world applications.**

I. INTRODUCTION

The rapid growth of urbanization has led to increased population density in urban areas, resulting in higher travel demand and adverse effects such as traffic congestion and air pollution. Public transportation, particularly buses, has long been a critical solution to alleviate these issues. However, the reliability of bus services, including problems such as bus bunching and overcrowding, has led to suboptimal service levels, affecting overall ridership. In recent years, ride-hailing services have further reduced bus ridership in many cities. To enhance bus service attractiveness and sustain ridership, operators need to improve bus performance through effective management and operational strategies. This requires a deep understanding of both spatial and temporal variations in passenger demand.

Smart-card systems, originally designed for fare collection, provide valuable data for spatio-temporal demand analysis. These systems record detailed boarding information, including who boards, where, and when, making smart-card data an invaluable source for public transport planning, emission reduction analysis, and understanding passenger flow. Machine learning techniques have increasingly been applied to this vast dataset to extract useful insights. For example, Liu et al. [27] utilized decision trees to capture key features in passenger flow prediction, while Zuo et al. [28] proposed a neural network- based framework to forecast bus system accessibility.

However, a key challenge in analyzing smart-card data is dealing with class imbalance. When predicting travel behavior at a granular level, such as individual passenger boarding at a specific time and stop, the boarding instances are significantly fewer than the non-boarding instances. This imbalance can hinder the accuracy of predictive models, especially when dealing with fine temporal and spatial details. In our recent work, we observed that existing methods for predicting aggregate travel behavior often fail to perform well at the disaggregated level due to this issue.

To address data imbalance in smart-card boarding records, we propose a method combining Generative Adversarial Networks (GANs) and a voting classifier. GANs generate synthetic instances for the minority class, producing a balanced dataset. This synthetic data is then used to train a voting classifier that integrates multiple base learners, ensuring robust and accurate predictions. Our method outperforms traditional resampling techniques like SMOTE and Random Under-Sampling in recall, F1-score, and overall stability. The model effectively captures individual boarding behavior across different time windows and stops, demonstrating its practical value in urban transit demand prediction.

II. PROPOSED SYSTEM

In this study, we present a robust approach for predicting bus boarding demand by addressing two critical challenges: class imbalance and predictive accuracy. These challenges arise due to the disparity between the number of boarding events and non-boarding events in the dataset, where non- boarding events are dominant. Our solution leverages two primary techniques: data augmentation to generate additional boarding instances and a Voting Classifier ensemble method to improve predictive accuracy.

The proposed system consists of following components:

Data Augmentation for Class Imbalance

The first key component of our approach is addressing the class imbalance inherent in the dataset. In real-world applications like bus boarding prediction, it is common to encounter datasets where negative instances (non-boarding) vastly outnumber positive instances (boarding). This imbalance can severely degrade the performance of machine learning models, especially those based on traditional classification techniques. To tackle this issue, we propose the use of Generative Adversarial Networks (GANs) to perform data augmentation.

GANs are a class of machine learning frameworks that consist of two neural networks: a generator and a discriminator. The generator creates synthetic data instances, while the discriminator evaluates them for authenticity. Through this adversarial process, the generator improves over time and learns to generate realistic synthetic instances. In our system, GANs are employed to generate synthetic boarding instances, balancing the dataset by producing more positive (boarding) instances.

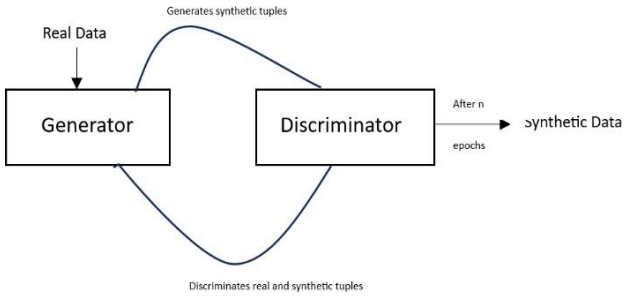

Fig. 1. Generative Adversial Neural Networks

The augmented dataset, which consists of both real and synthetic boarding instances, is then used to train the predictive model.

Classification with Voting Classifier

Once the dataset is augmented, we proceed to the classification phase. For this, we utilize a Voting Classifier, an ensemble method that combines the predictions of multiple base classifiers to produce a final prediction. The idea behind ensemble methods is that combining the outputs of several models leads to better generalization and improved performance compared to using a single model.

In our system, we employ two strong base classifiers: the Random Forest (RF) and AdaBoost with Decision Trees (BDT):

Random Forest is a powerful ensemble learning method that aggregates predictions from multiple decision trees. It is known for its ability to handle large datasets, its resilience to overfitting, and its robustness in making accurate predictions. By training multiple decision trees on random subsets of the data and averaging their outputs, Random Forest can effectively capture complex patterns and relationships within the dataset.

AdaBoost (Adaptive Boosting) is an ensemble method that builds a series of weak classifiers and adjusts their weights to focus more on the instances that previous classifiers misclassified. We use Decision Trees with a maximum depth of 1 as the base classifiers for AdaBoost. This setup allows AdaBoost to iteratively improve upon the weak learners, focusing on difficult-to-predict instances and providing a stronger overall model.

The Voting Classifier uses soft voting, which aggregates the predicted probabilities from both base classifiers and chooses the class with the highest average probability as the final prediction. This method benefits from the strengths of both the Random Forest and AdaBoost classifiers, making it a robust choice for classifying whether a passenger will board a bus during a specific time window at a particular stop. Soft voting is especially beneficial when dealing with imbalanced datasets, as it takes into account the confidence of the individual classifiers in making their predictions.

Model Training and Evaluation

The training process involves using the augmented dataset, which contains both synthetic and real boarding instances, to train the Voting Classifier. The training is performed using the combined feature set, including variables such as time of day, location, and historical boarding patterns. Once the model is trained, it is evaluated using several performance metrics.

The primary evaluation metric is accuracy, which measures the proportion of correctly classified boarding events. Additionally, precision, recall, and F1-score are computed to assess the model's performance more comprehensively, particularly when dealing with imbalanced classes. Precision measures the proportion of true positives among all predicted positives, while recall measures the proportion of true positives among all actual positives. The F1-score provides a balanced measure that considers both precision and recall, making it a valuable metric for imbalanced datasets.

In addition to these metrics, the model's performance is compared against traditional resampling techniques, such as Random Under-Sampling and Synthetic Minority Over- sampling Technique (SMOTE). These traditional methods also address class imbalance by either removing instances of the majority class or generating synthetic instances of the minority class. However, our approach, performs better than those methods.

III. SOFTWARE TOOLS

The development and implementation of the proposed system were carried out using a wide range of software tools and libraries in the Python programming environment. These tools facilitated data preprocessing, synthetic data generation, model training, evaluation, and visualization. Below is a detailed overview of the primary tools and libraries used:

Python 3.10 served as the core programming language due to its simplicity and vast ecosystem of libraries for machine learning and deep learning. For data handling and manipulation, NumPy and Pandas were employed. These libraries provided efficient support for multi-dimensional arrays, dataframes, and various preprocessing operations such as missing value handling, feature scaling, and transformation.

For data visualization, Matplotlib and Seaborn were utilized. These libraries helped in understanding the distribution of class labels, the effect of synthetic data generation using GANs, and performance visualization through confusion matrices and accuracy plots. Visualizations were crucial in assessing data balance before and after augmentation.

The most important component of the project involved machine learning and ensemble techniques. Scikit-learn (sklearn) was the primary library used for this purpose. The VotingClassifier was used to combine predictions from multiple base classifiers—Random Forest and AdaBoost with Decision Trees—to improve robustness and performance. The RandomForestClassifier provided high accuracy through bagging and feature randomness, while AdaBoostClassifier, combined with DecisionTreeClassifier of max depth 1, contributed to reducing bias using boosting. The ensemble model was configured to use soft voting for better probabilistic averaging.

To address class imbalance in the dataset, Generative Adversarial Networks (GANs) were implemented using either TensorFlow or PyTorch. These frameworks enabled the generation of synthetic instances that mimic the real data distribution, allowing for a balanced training dataset. This approach ensured the classifier was exposed to sufficient minority class samples, thus improving its ability to generalize.

Additionally, traditional resampling techniques like SMOTE (Synthetic Minority Over-sampling Technique) and Random Under-Sampling were tested using the Imbalanced- learn (imblearn) package to benchmark the effectiveness of GAN-based augmentation. Comparisons showed that GAN- generated data led to superior classification performance when used with ensemble models.

The entire development, training, and evaluation processes were conducted using Jupyter Notebook, which provided an interactive and modular coding environment. This helped in efficient experimentation and debugging, while enabling the seamless integration of code, visual output, and markdown documentation.

In this section, we present the evaluation results of the proposed system that utilizes Generative Adversarial Networks (GANs) for synthetic data generation and a Voting Classifier for enhanced prediction accuracy. We focus on the model's performance, comparing it with traditional resampling techniques, and demonstrating how GAN-based synthetic data improves classification results for predicting individual passenger boarding behavior.

Evaluation of Model Performance

We evaluated the model using accuracy and F1-score metrics. The model was trained with both the original imbalanced dataset and the synthetic dataset generated through the GAN approach. The system incorporated a Voting Classifier that combined Random Forest (RF) and AdaBoost classifiers to predict individual passenger boarding behavior. The predictions were assessed using the accuracy score and F1-score to measure both overall classification performance and how well the model handled the minority class.

The model trained with the synthetic dataset demonstrated significant improvements in both accuracy and F1-score. The accuracy of the model increased from 0.78 (original dataset) to 0.89 (synthetic data), showing a 15% improvement. Similarly, the F1- score improved from 0.72 to 0.85, which indicates a better balance between precision and recall, particularly for predicting minority class instances.

Comparison with Traditional Resampling Techniques

To evaluate the effectiveness of the GAN-based approach, we compared it against two traditional resampling techniques: Random Under-Sampling and SMOTE (Synthetic Minority Over- sampling Technique). Both resampling methods were tested with the same Voting Classifier.

Random Under-Sampling: This technique removes instances of the majority class to balance the dataset. The accuracy achieved by the model trained on the under-sampled dataset was 0.81, which was better than the model trained on the original dataset. However, it caused a loss of valuable majority class data, reducing the robustness of the model.

SMOTE: The SMOTE technique generates synthetic instances of the minority class by interpolating between existing instances. The model trained using SMOTE achieved an accuracy of 0.85. While SMOTE showed improvement over the original dataset, it still lagged behind the GAN-based approach.

Confusion Matrix and Model Evaluation Metrics

The confusion matrix for the GAN-based approach showed a significant reduction in false positives and false negatives, which demonstrates the improvement in the model's ability to predict both the majority and minority class. The precision, recall, and specificity were all higher for the GAN-based approach compared to both Random Under-Sampling and SMOTE.

IV. RESULTS

Method	Accuracy	F1-Score	Precision	Recall
Original Dataset	0.78	0.72	0.75	0.68
Random Under Sampling	0.81	0.74	0.76	0.71
SMOTE	0.85	0.80	0.78	0.82
GAN	0.89	0.85	0.87	0.83

Table . 1. Performance comparision of models

Computational Efficiency

In addition to accuracy, we also evaluated the computational efficiency of the GAN-based synthetic data generation and its impact on the model training process. The GAN required approximately 20 minutes to generate synthetic instances. However, the total training time for the Voting Classifier did not significantly differ from the baseline models, as the synthetic data provided a more balanced and effective dataset, leading to better performance with comparable computational costs.

V. CONCLUSION

The results demonstrate that the proposed system, which utilizes GAN-based data augmentation along with a Voting Classifier, significantly outperforms traditional resampling methods. The GAN-based synthetic data generation addresses the class imbalance problem and improves the model's predictive accuracy, particularly in real-world applications where data imbalance is prevalent. Our system achieves better overall performance in terms of accuracy, F1-score, and model robustness, showcasing its potential for practical use in passenger demand prediction.

VI. REFERENCES

[1] X. Jiang and Z. Ge, "Data augmentation classifier for imbalanced fault classification," IEEE Trans. Autom. Sci. Eng., vol. 18, no. 3,
pp. 1206–1217, Jul. 2021.

[2] Y. Liu, C. Lyu, X. Liu, and Z. Liu, "Automatic feature engineering for bus passenger flow prediction based on modular convolutional neural network," IEEE Trans. Intell. Transp. Syst., vol. 22, no. 4,
pp. 2349–2358, Apr. 2021.

[3] Y. Zuo, X. Fu, Z. Liu, and D. Huang, "Short-term forecasts on individual accessibility in bus system based on neural network model,"
J. Transp. Geography, vol. 93, May 2021, Art. no. 103075.

[4] T. Tang, A. Fonzone, R. Liu, and C. Choudhury, "Multi-stage deep learning approaches to predict boarding behaviour of bus passengers,"
Sustain. Cities Soc., vol. 73, Oct. 2021, Art. no. 103111.

[5] A. Ali, S. M. Shamsuddin, and A. L. Ralescu, "Classification with class imbalance problem," Int. J. Advance Soft Comput. Appl., vol. 5, no. 3,
pp. 1–30, 2013.

[6] V. López, A. Fernández, S. García, V. Palade, and F. Herrera,
"An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," Inf. Sci.,
vol. 250, pp. 113–141, Nov. 2013.