Mitigating Risk in P2P Lending Network: Enhancing Predictions with GenAI and SMOTE

Lina Devakumar Louis[1], Andrew Dunton[1], Sourab Rajendra Saklecha[1], Swetha Neha Kutty Sivakumar[1], Abdul Sohail Ahmed[1], Smeet Sheth[1], and Shih Yu Chang[2]
[1]Department of Applied Data Science, San Jose State University, Washington Sq, San Jose, CA 95192, United States
[2]Assistant Professor, Department of Applied Data Science, San Jose State University, Washington Sq, San Jose, CA 95192, United States

Peer-to-peer (P2P) lending is a major revolution in the field of finance where it transformed the market by eliminating the need for middlemen or conventional intermediaries such as banks, connecting borrowers directly with investors. This transformation offers several advantages, including potentially lower interest rates for borrowers and higher returns for investors. However, it also introduces risks, particularly the possibility of borrowers defaulting on their loans which could lead to significant losses. Research indicates that classification models can be leveraged to address this risk. However, the real-world datasets available are heavily skewed which could lead to bias in the prediction and model over-fitting. Existing research utilize conventional approaches such as Synthetic Minority Over-sampling Technique (SMOTE) for balancing data and ensemble models. This study addresses these challenges by implementing a comparative study between SMOTE and generative AI for data synthesis to rationalize the effects of modern approaches. Further it also explores the inclusion of additional features as compared to existing research. Ensemble modeling approaches were adopted for the purpose of this study. Logistic Regression, Support Vector Machine (SVM), KNN, and Random Forest were selected to determine the best base model to be used for stacking. XGBoost, LightGBM, and AdaBoost were the three selected models for stacking. XGBoost outperformed all other models, achieving an average accuracy of 99.4% and average F1-score of 97.4% using SMOTE synthesis. GenAI synthesis obtained similar performance.

***Index Terms*—Peer-to-Peer lending, loan default, imbalanced dataset, SMOTE, GenAI, Logistic Regression, KNN, Random Forest, Support Vector Machine, XGBoost, LightGBM, AdaBoost, ensemble**

I. Introduction

PEER-TO-PEER (P2P) lending has become significantly popular in the financial technology landscape by establishing a direct connection between lenders and borrowers eliminating the traditional financial middlemen. The investment opportunities for individuals and financiers are limitless but come with the disadvantage of determining the borrower's credit worthiness. The lending system requires a recommendation to determine the risk in the lending decision for borrowers who can become potential defaulters. This project aims to provide a solution to this problem by developing machine learning models to enhance prediction accuracy. To address the challenges posed by highly imbalanced datasets in domains such as peer-to-peer (P2P) lending, researchers have commonly adopted conventional techniques for data balancing. Among these, SMOTE stands out as a widely utilized approach for oversampling method. SMOTE functions by generating synthetic samples of the minority class, hence increasing its representation in the dataset. This technique is particularly effective in scenarios where the minority class is inadequately represented, as it helps mitigate the bias towards the majority class and enables the machine learning task to better perceive patterns and relationships inherent in the data[1]. This project explores the Lending Club dataset [11].

In reference to P2P lending, where defaulting loans are the minority class, SMOTE can act as an agent to balance the dataset to ensure that predictive models are not biased towards the majority class which is non-defaults. By augmenting the dataset with synthetic instances of defaulting loans, SMOTE helps classifiers better capture the underlying patterns indicative of loan default risk, thereby enhancing the accuracy and reliability of predictions.

In addition to SMOTE, research also shows ensemble modeling techniques have been widely employed in conjunction. Ensemble models combine the predictions of multiple base classifiers to produce a more robust and accurate predictions. By leveraging the uniqueness of individual classifiers and aggregating their outputs, ensemble models can handle the limitations of a single classifier and thereby improve overall prediction performance [3].

Exploring available modern open-source generative AI (GenAI) tools like Mostly AI for generating synthetic data presents an opportunity to address this issue of imbalance in P2P data-driven networks using newer technology. Mostly AI is an open-source tool which is a leading provider of synthetic data generation solutions, leveraging advanced learning algorithms to create high quality data that closely mimics actual data while also ensuring compliance with data privacy regulations [2].

The significance of each feature to differentiate defaulters from the total list of borrowers is identified by implementing feature engineering. To precisely classify the defaulters, the discrepancies in the data are resolved and important features are further considered for model development. The research will classify the borrowers as defaulters and non-defaulters by

Manuscript received January 31, 2024; revised June 13, 2024. Corresponding author: Lina Devakumar Louis (email: lina2396@gmail.com).

developing baseline machine-learning models with ensemble techniques like boosting. The borrowers who may eventually default are finally recognized.

II. Key Contributions

Our research contributes to the field of P2P lending risk assessment by providing insights into effective techniques and feature engineering. The combination of MostlyAI, XGBoost, and novel features yields promising results. Future work could explore additional ensemble methods and further enhance model robustness. The following adoptions explain the key contributions of this research.

A. Comparison of MostlyAI and SMOTE

We evaluate the performance of MostlyAI and SMOTE in handling imbalanced datasets. MostlyAI generates synthetic data, while SMOTE over-samples the minority class. Results indicate that MostlyAI achieves impressive accuracy, contributing to better predictions.

B. Predictive Modeling with XGBoost

We employ XGBoost, an ensemble learning algorithm, for predicting defaulters. Our model achieves an outstanding accuracy of 99.4%. The synthetically generated data plays a crucial role in achieving this high accuracy.

C. Feature Engineering Beyond the Basics

Beyond existing features, we explore new variables impacting loan default risk. These features enhance the model's ability to identify potential defaulters. Our approach improved overall prediction performance.

D. Ensemble Modeling Approaches

We compare different ensemble models for defaulter prediction. By combining base models, we optimize accuracy. The best combination of base and meta models is determined through rigorous evaluation.

III. Related Work

The Lending Club dataset is highly imbalanced towards the non-default loans [11]. Data balancing between the non-default and default loan classes should be considered before model training. A study by Mukherjee & Khushi (2021) involved an in-depth exploration of strategies to address class imbalance in credit scoring [4]. Specifically, it discussed the importance of the Synthetic Minority Over-sampling Technique (SMOTE) in such contexts. SMOTE generates new samples of the minority class, balancing the class distribution to help improve model performance. Furthermore, Mukherjee & Khushi (2021) emphasized the necessity of encoding categorical features before applying SMOTE, as it ensures the synthetic data generation is grounded in the numerical space, essential for the algorithm's processing. These insights are particularly relevant to our study, where feature encoding and SMOTE may play a pivotal role in loan default model performance.

Muslim and his team in 2023 discuss their study on improving the accuracy of default risk prediction by balancing data and using a stacking model [5]. The problem at hand was that the prediction was inaccurate with imbalanced data and low-performing algorithms. They used the data from Lending club and chose KNN, SVM, and Random Forest as their base model on top of which they used to build their ensemble mode. To evaluate their models, they compared their accuracy scores. The evaluation's findings indicate that the optimal ensemble model for the dataset is LGBFS-StackingXGBoost.It obtained a 99.82% accuracy rate.In subsequent research, they intend to experiment with larger datasets or datasets from various nations, trying to optimize new models for improved performance.

The study by Shen and his collaborators proposes a novel ensemble classification model for imbalance credit risk evaluation [6]. Evaluation metrics of the traditional classification models were compared with the proposed ensemble model to which it was observed that the ensemble model outperforms other models. The Synthetic Minority Over-Sampling Technique was performed to balance the training data. The Particle Swarm Optimization (PSO) algorithm was used to assign proper weights. AdaBoost was combined with base classifiers as an ensemble approach. The model was then tested on German and Australian real-world imbalance datasets and its evaluation metrics were noted down. The model got an accuracy of 70% for the German Dataset and 95% for the Australian Dataset. The literature survey considers the common challenge of the imbalance dataset in the credit scoring system and highlights the gap in existing research by proposing an effective and efficient ensemble classification model.

Another study emphasizes the qualitative features of the applicants on top of the quantitative indicators[7]. A new cluster analysis to handle such mixed data was proposed in this study. The newly developed model proved to be more effective compared to the traditional methods that were proposed in previous studies, indicating that the qualitative features are informative and improve credit risk evaluation. The study was performed on a credit risk evaluation dataset from Germany which was downloaded from the UCI Machine Learning Repository. The study also mentions other approaches where clusters were formed of different objects using the k-types algorithm and a cost function was defined where the relationship between numerical and categorical variables was defined. Banks can utilize the proposed method for the correct identification of good and bad customers and improve their loan grant ratio along with the efficient allocation of funds.

Chen and Zhang proposed K-means SMOTE algorithm and BP neural networks to predict the defaulter in allocating credit card risk. The research is conducted on credit card usage data published on the Kaggle platform [8]. The first step in the research is to balance the disparities that exist both between and within credit card categories. The minority cluster is found using the K-means cluster, and it is then subjected to smooth-oversampling. Decision Trees are used to determine the features' importance, which is then used to replace the original weights in the BP Neural Network. The model's accuracy increased from 0.765 to 0.929 when comparing the

before and after implementation of the K-means SMOTE. In addition to the proposal, the author suggests the research requires a lot of features and implementation of Delphi expert method that can be used to identify the individual's credit which can significantly impact the evaluation system for credit card approvals.

Considering small businesses are the main employers in the United States, Wang and Cheng's research from 2021 focused on assessing the loan risk associated with them [9]. They employed the U.S. Small Business Administration (SBA) dataset, which has historical data with 899,164 observations from 1987 through 2014, to do this. Different machine learning algorithms and popular boosting algorithms such as Linear Regression, Support Vector Machine (SVM), Random Forest (RF), Multi-layer Perceptron (MLP), Gradient Boosting Machine and XGBoost , Light Gradient Boosting Machine (LightGBM) and Categorical Boosting (CatBoost) were experimented. Furthermore, in order to produce additional features, random arithmetic operations were applied to the top two features in a process known as synthetic feature synthesis. In order to increase model correctness, this method assists in determining the strongest feature. CatBoost outperforms the current boosting algorithms, achieving the greatest accuracy of 95.84% while using synthetic features. Apart from the current feature, the bank or investors have the option to take into account elements like the loan duration, the borrower's status, and the credit line's condition. They may then set higher standards and implement more stringent guidelines.

Synthetic consumer credit data was generated using Generative Adversarial Networks (GANs) for educational and research purposes [10]. The research uses the Korea Credit Information Services dataset which holds multiple credit history information of many car owners for the period of 2015 to 2019. By using the extension of GAN i.e conditionalGAN produces consumer credit data. Using conditionalGAN, the random noise and conditional variable are concatenated as input to the Generator to generate fake data. Further Discriminator determines if the data is real or fake. Finally the data is executed in the performing model to present the outputs. The results show the uni-variate and multivariate distribution is higher in the synthetic data with the exposure risk of 0.05%. To increase the prediction potential, more research can be done by grouping the consumer segments.

Considering the similarities and differences in the various studies discussed, it seems there is some leeway to explore available genAI tools to generate synthetic data for this domain. This could lead to enhancing the robustness and generalizability of predictive models across diverse applications. For the purpose of this research it would be substantial to assess genAI against SMOTE that has gathered positive results in the past for the same dataset. This research also stands out with respect to the fact that it considers additional features that could be effective for this domain.

IV. Methodology

In this research the lending club dataset[11] consisting of loan information, borrower's personal and financial information with the loan status is used. The significance of each feature to differentiate defaulters from the total list of borrowers is identified by implementing feature engineering. To precisely classify the defaulters, the discrepancies in the data are resolved and important features are further considered for model development. The research will classify the borrowers as defaulters and non-defaulters by developing baseline machine-learning models with ensemble techniques like boosting. The borrowers who may eventually default are finally recognized.

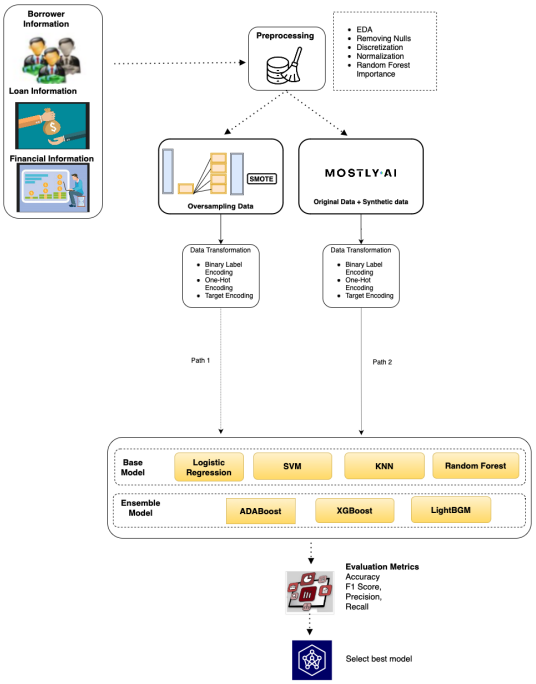


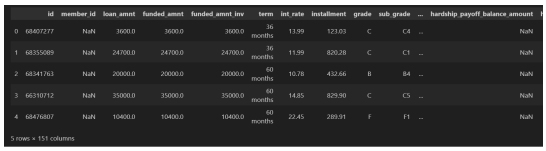
Fig. 1. Process flow diagram of project lifecycle

Figure 1 depicts the process flow diagram of the project life-cycle. The dataset consisting of the borrower's personal and financial information, and loan information requires to be pre-processed. Firstly Exploratory Data Analysis (EDA) is performed to understand the characteristics and patterns, identify anomalies, and understand the relationship between features. The data-cleaning phase further requires null value removal and handling of inconsistent and noisy data. Then feature importance using Random Forest Importance (RFI) is performed to filter features that have high impact on the target. During the EDA phase, inconsistency in the target feature was observed. To balance the dataset SMOTE technique and generation of synthetic data using mostly.ai were proposed indicated by the two distinguished paths. Post data synthesis the data is transformed for model training. The categorical data were converted into numerical data by performing the Label encoding, target encoding, and one-hot encoding. The pre-processed data is then split into training and testing data for model development. The base models such as Logistic regression, Support Vector Machine (SVM), KNN, and Random Forest were implemented to first identify the model that results with the highest accuracy. At the end of this step, the best performing model is selected as the base model to be

further used for stacking alongside AdaBoost, XGBoost, and LightBGM to improve the accuracy of prediction. The final evaluation metrics such as Accuracy, F1 Score, Recall, and Precision are used to compare the performance of the models to select the best performing model post hyper-tuning. Further the results for the two paths will be compared to gauge the results of the different approaches.

A. Data Cleaning

Data cleaning is performed on the raw dataset to ensure the data quality is reliable to perform further operations. This study uses the Lending Club loan data[11] for data preparation and modeling. Raw data was initially read to determine the number of columns or features present in it and it came out to be 151 which is shown in Figure 2.

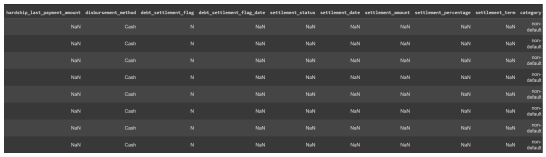


	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	...	hardship_payoff_balance_amount
0	68407277	NaN	3600.0	3600.0	3600.0	36 months	13.99	123.03	C	C4	...	NaN
1	68355089	NaN	24700.0	24700.0	24700.0	36 months	11.99	820.28	C	C1	...	NaN
2	68341763	NaN	20000.0	20000.0	20000.0	60 months	10.78	432.66	B	B4	...	NaN
3	66310712	NaN	35000.0	35000.0	35000.0	60 months	14.85	829.90	C	C5	...	NaN
4	68476807	NaN	10400.0	10400.0	10400.0	60 months	22.45	289.91	F	F1	...	NaN

5 rows × 151 columns

Fig. 2. Raw Dataset

After the initial dataset was read a new feature was created with the name ‘category’ which is the target feature that contains values as either default or non-default. It was derived from the ‘loan_status’ dataset where the value is the default if the loan_status has values such as charged off, default, late (31-120 days), and in the grace period; and if other values were there then they were considered as non-default. Figure 3 shows the newly created category feature added to the data frame.



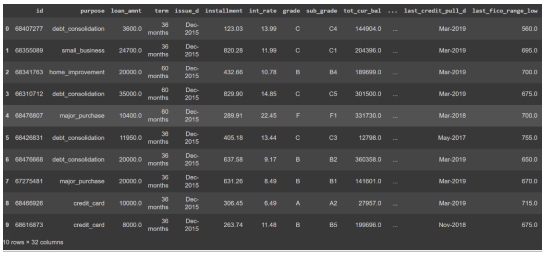
hardship_last_payment_amount	disbursement_method	debt_settlement_flag	debt_settlement_flag_date	settlement_status	settlement_date	settlement_amount	settlement_percentage	settlement_term	category
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default
NaN	Cash	N	NaN	NaN	NaN	NaN	NaN	NaN	non-default

Fig. 3. Category feature added in the Dataframe

It was determined that all 152 features were not required and a thorough investigation was done to identify the features of interest. It was identified that only 32 features are required among them to make default predictions. The operation was performed to remove the rest of the features or columns from the original Dataframe and keep only the required ones in the new Dataframe that are used to perform further operations and analysis. Figure 4 shows the top 10 rows of the Dataframe after removing unnecessary features.

The next step after selecting the necessary features and creating the target feature was to determine if there were multiple data types present in the Dataframe. It was identified that all the features were of a single datatype. Figure 5 shows the data types of each feature.

Checked for null values in different features and removed them if the count was less and the removal of these instances



	id	purpose	loan_amnt	term	issue_d	installment	int_rate	grade	sub_grade	tot_cur_bal	...	last_credit_pull_d	last_fico_range_low
0	68407277	debt_consolidation	3600.0	36 months	Dec-2015	123.03	13.99	C	C4	144904.0	...	Mar-2019	560.0
1	68355089	small_business	24700.0	36 months	Dec-2015	820.28	11.99	C	C1	204396.0	...	Mar-2019	695.0
2	68341763	home_improvement	20000.0	60 months	Dec-2015	432.66	10.78	B	B4	189699.0	...	Mar-2019	700.0
3	66310712	debt_consolidation	35000.0	60 months	Dec-2015	829.90	14.85	C	C5	301500.0	...	Mar-2019	675.0
4	68476807	major_purchase	10400.0	60 months	Dec-2015	289.91	22.45	F	F1	331730.0	...	Mar-2018	700.0
5	68426831	debt_consolidation	11950.0	36 months	Dec-2015	405.18	13.44	C	C3	12798.0	...	May-2017	755.0
6	68476668	debt_consolidation	20000.0	36 months	Dec-2015	637.58	9.17	B	B2	360358.0	...	Mar-2019	650.0
7	67275481	major_purchase	20000.0	36 months	Dec-2015	631.26	8.49	B	B1	141601.0	...	Mar-2019	670.0
8	68406926	credit_card	10000.0	36 months	Dec-2015	306.45	6.49	A	A2	27957.0	...	Mar-2019	715.0
9	68616873	credit_card	8000.0	36 months	Dec-2015	263.74	11.48	B	B5	199696.0	...	Nov-2018	675.0

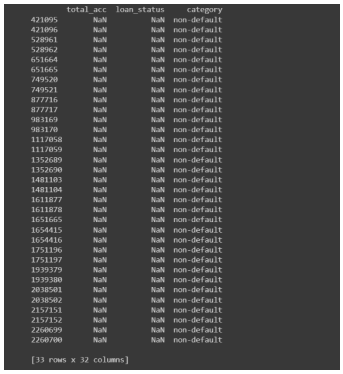
10 rows × 32 columns

Fig. 4. New Dataframe with required features

```
id                          object
purpose                     object
loan_amnt                  float64
term                        object
issue_d                     object
installment                float64
int_rate                   float64
grade                       object
sub_grade                   object
tot_cur_bal                float64
pymnt_plan                  object
addr_state                  object
total_rec_int              float64
total_rec_late_fee         float64
total_rec_prncp            float64
out_prncp                  float64
emp_title                   object
emp_length                  object
annual_inc                 float64
home_ownership              object
delinq_2yrs                float64
inq_last_12m               float64
last_credit_pull_d          object
last_fico_range_low        float64
mort_acc                   float64
mths_since_last_delinq     float64
num_actv_bc_tl             float64
open_il_12m                float64
tot_hi_cred_lim            float64
total_acc                  float64
loan_status                 object
category                    object
dtype: object
```

Fig. 5. Data types of features

does not cause issues in the future. Figure 6 shows the ‘purpose’ feature that was cleaned by identifying the number of null values and removing them.



	total_acc	loan_status	category
421095	NaN	NaN	non-default
421096	NaN	NaN	non-default
528961	NaN	NaN	non-default
528962	NaN	NaN	non-default
651664	NaN	NaN	non-default
651665	NaN	NaN	non-default
749520	NaN	NaN	non-default
749521	NaN	NaN	non-default
877716	NaN	NaN	non-default
877717	NaN	NaN	non-default
983169	NaN	NaN	non-default
983170	NaN	NaN	non-default
1117058	NaN	NaN	non-default
1117059	NaN	NaN	non-default
1352689	NaN	NaN	non-default
1352690	NaN	NaN	non-default
1481103	NaN	NaN	non-default
1481104	NaN	NaN	non-default
1611877	NaN	NaN	non-default
1611878	NaN	NaN	non-default
1651665	NaN	NaN	non-default
1654415	NaN	NaN	non-default
1654416	NaN	NaN	non-default
1751196	NaN	NaN	non-default
1751197	NaN	NaN	non-default
1939379	NaN	NaN	non-default
1939380	NaN	NaN	non-default
2038501	NaN	NaN	non-default
2038502	NaN	NaN	non-default
2157151	NaN	NaN	non-default
2157152	NaN	NaN	non-default
2260699	NaN	NaN	non-default
2260700	NaN	NaN	non-default

[33 rows x 32 columns]

Fig. 6. Null values in purpose feature

Some of the features had a lot of numeric values present in them which made it difficult to perform analysis and would have been difficult to handle during modeling. To resolve this issue the bins were created. One such feature on which this operation was performed was ‘total_acc’ which provides information about the total number of credit lines present in the borrower’s account. Bins were selected to be 0-10, 10-20,

20-50', '50-100', and '100+'. Figure 7 shows the creation of category bins based on 'total_acc' values.

	total_acc	total_acc_new
0	13.0	10-20
1	38.0	20-50
2	18.0	10-20
3	17.0	10-20
4	35.0	20-50
5	6.0	0-10
6	27.0	20-50
7	15.0	10-20
8	23.0	20-50
9	18.0	10-20

Fig. 7. Creation of category bins based on total_acc values

Categories were also created for the feature 'last_fico_range_low' that gives information about the credit score of the person. Here, the bins were created as 'No Credit', 'Poor', 'Average', 'Good', and 'Excellent' based on credit score. If the person has less than a 500 credit score then that person will be placed in the no-credit category, between 500 to 600 then poor, between 600 to 700 then average, 700 to 800 then good, and anything above it is considered excellent. Figure 8 shows the categorization of the values of last_fico_range_low.

	last_fico_range_low	last_fico_range_low_new
0	560.0	Poor
1	695.0	Average
2	700.0	Good
3	675.0	Average
4	700.0	Good
5	755.0	Good
6	650.0	Average
7	670.0	Average
8	715.0	Good
9	675.0	Average

Fig. 8. Creation of categories based on last_fico_range_low values

B. Exploratory Data Analysis

A key stage in the data preparation process is Exploratory Data Analysis (EDA), which involves identifying, analyzing, and visualizing a dataset's primary characteristics. The Exploratory Data Analysis revealed a pronounced class imbalance in the loan-status feature, leaning heavily towards the non-default loan class, as indicated by Figure 9.

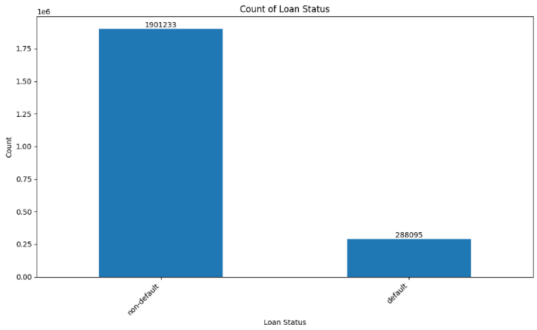


Fig. 9. Total count of the non-default and default classes

The imbalance shows the importance of considering data synthesis. A correlation matrix of the feature space was generated, highlighting the strong linkage between interest rates and sub-grades, as highlighted in Figure 10.

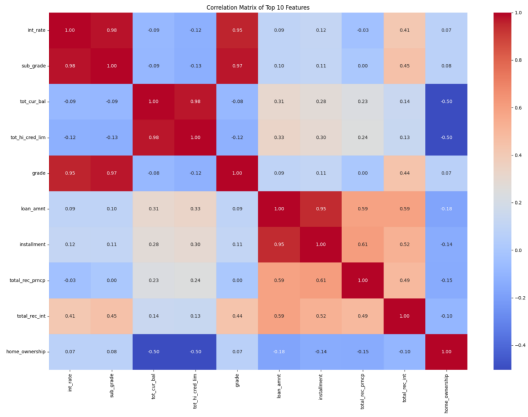


Fig. 10. Correlation matrix of the Lending Club dataset

The box plot analysis in Figure 11, depicting the interest rate variations across sub-grades, shows that there are many outliers in the dataset and should potentially be removed before training.

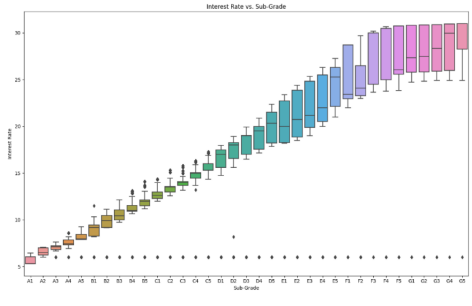


Fig. 11. Box plot of the interest rate at each sub-grade category

Delinquency data, presented in Figure 12, suggested a correlation between the number of delinquencies and recentness, informing the binning strategy in model development.

The FICO score's impact on loan default probability was evident in Figure 13, advocating for its inclusion in predictive modeling.

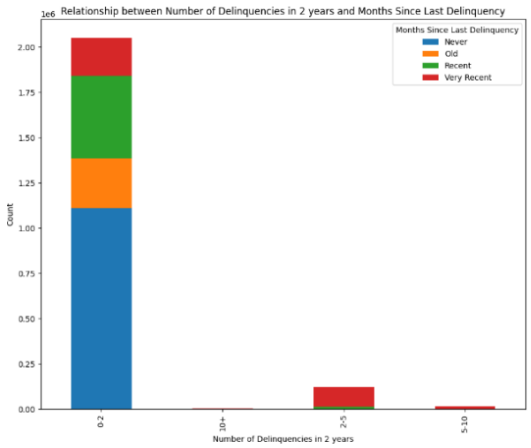


Fig. 12. Relationship between a number of delinquencies and recentness

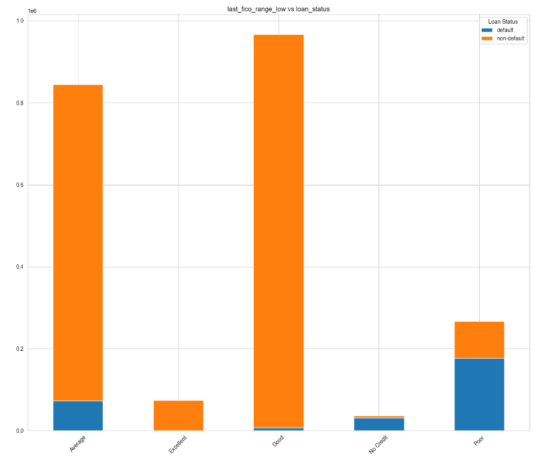


Fig. 13. Relationship between FICO score and loan status

C. Data Normalization

Data Normalization is the crucial component of data pre-processing where the numerical features are brought to the standardized level that is within a particular range like [0,1]. This operation is performed to ensure that no particular feature dominates the performance of the model. In this study, some of the features 'like loan_amnt', 'int_rate', 'installment' etc were normalized to the range of 0 to 1. Figure 14 shows some of the features after they were normalized.

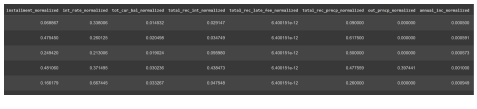


Fig. 14. Normalized features

D. Feature Engineering

Feature Engineering was performed on the selected columns to get accurate results in order to have a secure lending environment and reduce the risk to a minimum. Out of all the mixed data types available, the most relevant features that hold substantial importance in depicting the loan status were selected. This was done by observing error values of the model for different sets of features. The importance of the feature was considered based on the values of the model error while shuffling the different sets. It is calculated by below equation[15], where Error(Original) is the error of the model before permuting feature F and Error(Permuted (F)) is the error after permuting feature F.

Importance of Feature F = Error(Original) - Error(Permuted (F))

It was observed that considering the top five features based on their importance, the model gave an accuracy of 96.5%. Once the feature engineering was performed additional 10 features were added and the accuracy of the model was calculated. The accuracy vs number of features graph was plotted to get an idea about the ideal size of the features that need to be considered for model building. Based on that total of 11 features were considered for the final model building. Figure 15 shows the ranking of feature importance in the descending order.

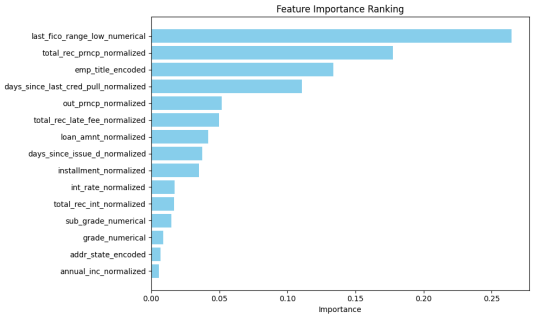


Fig. 15. Feature Importance Ranking

E. Data Synthesis: Mostly.AI and SMOTE

Oversampling is employed in machine learning models to resolve the issues of imbalance present in the dataset. Most commonly used to balance target features where one is the majority class and the other is the minority class. The imbalance present in the dataset could make the model biased and would result in poor prediction generated in the scenario of the minority class. To resolve this issue synthetic data are generated for the minority class by using Mostly.AI and SMOTE for the purpose of this project. Data Synthesis was performed in this study due to the imbalance in the division of instances having non-default and default. After performing data cleaning, 100000 rows were extracted from the original dataset to make processing faster, which was provided as an input to Mostly.ai[2]. It is a website that helps to create synthetic data using original dataset as the baseline. The newly created dataset would be unique but the values present in it would be within the expected range. This means that synthetic data will value only two categories in 'term' column that are '36 months' and '60 months' and nothing else. This website uses AI model to determine the patterns present in the data, correlations between different fields, how the data

is distributed in each category of the field and then generates the synthetic data. The majority of the instances had a target feature value as non-default which can be seen in Figure 16. Then, data synthesis was performed on the dataset and it can be observed in Figure 17 that the distribution of instances for different categories of the target feature became balanced. Similarly for the SMOTE path, the same prepped data was simultaneously over-sampled.

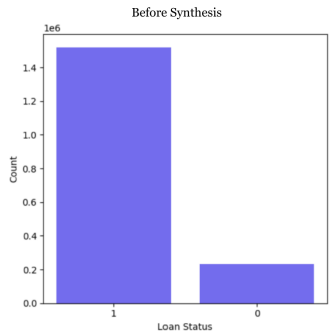


Fig. 16. Distribution of instances in different target categories before Synthesis

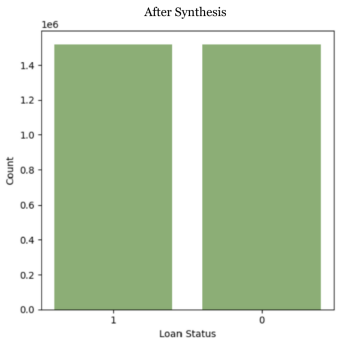


Fig. 17. Distribution of instances in different target categories after Synthesis

F. Original Data vs Mostly.AI Synthesized Data Analysis

Figure 18 illustrate the distribution of instances of the original and synthetic dataset respectively among different categories of the field 'last_fico_rang_low'. It is quite evident that the number of instances with the case result as non-default is more in the synthetic dataset than in the original one whereas the number of instances with case result as default is less in the synthetic dataset than original one. Figure 19 shows the distribution of instances of the original and synthetic dataset respectively among the different categories of the field 'grade'. It is easy to see that the number of instances with case results as non-default is more in synthetic data for the category values 'A', 'B', 'C', 'D', and 'E' and it is less for the case result as default in synthetic data for the same set of category values. The number of instances having case results as either default or non-default has decreased in synthetic data where the category values are 'F', and 'G'. Similarly, Figure 20 shows the distribution of instances among the categories of the field 'term' for original data vs synthetic data. It is quite visible that the instances with case results as non-default have decreased and with case results as default have increased in the new synthetic data. Possible reasons could be the employment of different techniques used for sampling that helped to generate data, some noise or error could have been added that resulted in the generation of data with more bias, etc.

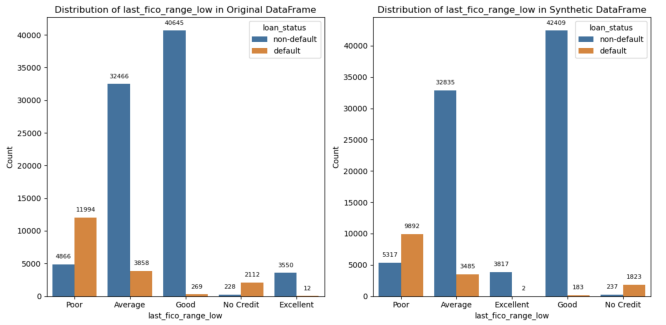


Fig. 18. Original vs synthetic data distribution for FICO range

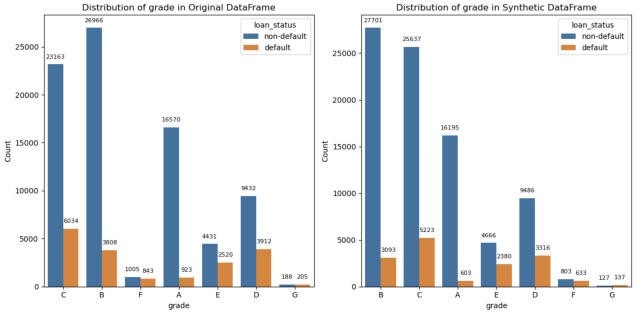


Fig. 19. Original vs synthetic data distribution for grade

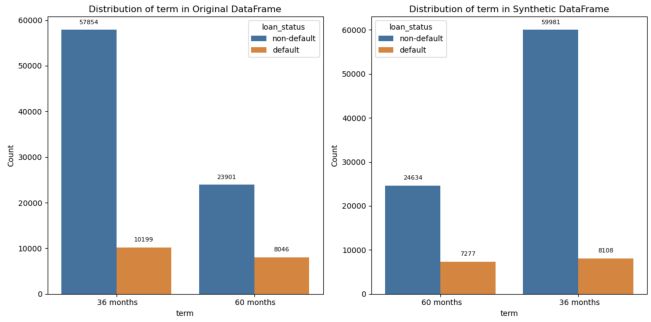


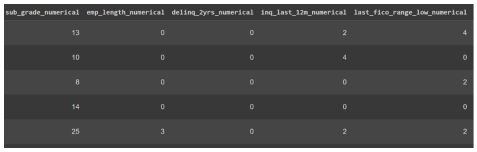
Fig. 20. Original vs synthetic data distribution for term

G. Data Transformation

Data Transformation is the process where the data is prepared to be properly loaded into the machine learning model. Some of the prominent operations are performing feature encoding.

Feature Encoding can be considered one of the most important steps that helps to prepare data that could be loaded

into the machine learning models. It involves the process of converting the categorical features into their numerical form which makes them compatible with different algorithms and ensures that the successful prediction can be made by the model. Some of the commonly used encoding techniques are label encoding, one-hot encoding, and target encoding. Few of the encoding techniques were implemented in this study. Figure 21 shows the implementation of label encoding where each category is assigned a numerical value and some of the features on which this was implemented were 'grade', 'sub_grade', 'emp_length' etc.



sub_grade_numerical	emp_length_numerical	delinq_2yrs_numerical	inq_last_12m_numerical	last_fico_range_low_numerical
13	0	0	2	4
10	0	0	4	0
8	0	0	0	2
14	0	0	0	0
25	3	0	2	2

Fig. 21. Label encoding performed on some features

One-hot encoding was also performed that converts each of the categories present in it as a separate binary value feature where the value is 1 if the original feature for that instance has that category value otherwise it is 0. Some of the features on which it was performed were 'purpose' and 'home_ownership'. Figure 22 shows each category of 'purpose' feature that was created as separate binary-valued features due to one-hot encoding.

purpose_car	purpose_credit_card	purpose_debt_consolidation	purpose_house	purpose_other
0	0	1	0	0
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	0	1	0

Fig. 22. One hot encoding performed on purpose feature

V. Modeling

The following modeling strategy was structured to set up a high-performing model for the Lending Club dataset. Initially, the dataset was partitioned into training sets derived from both the SMOTE and GenAI synthesized datasets. The models Logistic Regression, SVM, KNN, and Random Forest, were chosen as base models for their varied strengths with regards to classification tasks. The most effective base model, as determined by initial performance metrics, was then subjected to feature engineering and hyperparameter tuning to enhance its predictive accuracy. Then the refined model served as the foundation for stacked models incorporating XGBoost, AdaBoost, and LightGBM, leveraging their strengths to further improve prediction performance.

Figure 23 displays the different models that are used for predictive modeling tasks. Each of the models works in a different way on different datasets. Here, Logistic Regression, SVM, KNN, and Random Forest were considered as our base classifiers. These models were comparatively easy to implement and were able to handle and process high-dimensional as well as non-linear data. On top of that XGBoost, AdaBoost, and LightGBM were the ensemble models that were designed to improve overall accuracy in predicting the loan status. These models were able to handle the mixed data types very well even for large datasets and were resilient to overfitting. The proposed models in the study were proved to work better than the traditional methods especially when dealing with mixed data types where qualitative features are equally important as the quantitative features.

Model	Advantages	Disadvantages
Logistic Regression	-Good interpretability and Training speed	-Assumes linearity. - May underperform with non-linear relationships
SVM	-Effective in high-dimensional spaces. -Maximizes margin between classes	-Can be memory-intensive. -Requires feature scaling
KNN	-No assumptions about data. -Simple to implement	-Slow with large datasets -Sensitive to irrelevant features
Random Forest	-Handles non-linear data well -Resilient to overfitting	-Can be complex -Less interpretable than simpler models
XGBoost	-Handles various data types -Good with unbalanced data	-Prone to overfitting without proper tuning -More parameters to tune
AdaBoost	-Focus on difficult cases -Can improve weak learners	-Sensitive to noisy data -Sequential training can be slow
LightGBM	-Fast training and prediction -Efficient with large datasets	-May underperform with small datasets -Complex model tuning

Fig. 23. Advantage and Disadvantages of base and stacked models for the Lending Club dataset

A. Boosting Techniques for an Integrated Classifier

Boosting is one of the ensemble techniques which involves the process of combining multiple weak learners with each other to create a strong learner. The main motive of this process is to resolve the issues that were observed in the previous model in the newer ones.

XGBoost also known as Extreme Gradient Boosting is one of such ensemble technique which based on the concept of Boosting and uses one of its variants named Gradient boosting where new models are added with the goal of reducing the loss gradient value. It performs pruning of the trees involved in Random Forest to remove the less useful strips and to improve the performance of the model. It also incorporates the L1 and L2 regularization to control the complexity of the model and prevent the scenario of overfitting[13].

The XGBoost is used as the ensemble technique in the project to improve the performance of Random Forest model by primarily reducing the loss function value by effectively handling the loss gradient. It helps to capture complex patterns present in the data more effectively and can also handle missing values which was not the scenario with Random Forest. Figure 24 shows XGBoost prediction mechanism.

B. Hyperparameter Tuning

The modeling phase for picking the base model for stacking determined the Random Forest classifier achieved the highest accuracy as seen in Table 1. So the hyperparameter tuning is performed on the random forest model using the RandomizedSearchCV module from sci-kit-learn. The tuning process

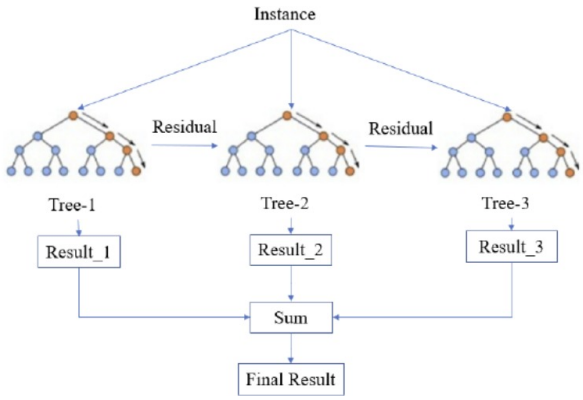


Fig. 24. XGBoost Prediction Overview — Source : Image from [14]

aimed to minimize overfitting while maximizing accuracy. The param dictionary defines the hyperparameters for the Random Forest classifier, including the number of estimators, maximum depth, and minimum samples per leaf. After that, the RandomizedSearchCV object does a randomized search across the hyperparameter space, adapting the model several times with various hyperparameter combinations and using cross-validation to assess each model's performance. The accuracy is used to identify the best-performing set of hyperparameters.

VI. Evaluation

Accuracy can be defined as the instances that are correctly predicted to all the instances that are there. It serves as a general indicator of how accurate the model is [12]. Contrarily, precision determines the proportion of accurate positive forecasts among all positive predictions [12]. The fraction of true positive predictions among all actual positive instances is called recall, or sensitivity, and it indicates how well the model finds all relevant cases [12]. By using their harmonic mean to balance recall and precision, a model's performance is assessed using the F1 score [12]. The respective equations can be seen in equations 1, 2, 3, and 4 [12].

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{Total Predictions}} \tag{1}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{2}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{3}$$

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{4}$$

Table 1 shows the evaluation metrics such as Accuracy, Precision, Recall, and F1-Score for both the base models as well as stacked models.

VII. Experimental Results and Analysis

Table 1 depicts the model performance metrics at class level. In scenarios where classes are imbalanced, average metric values may result in higher values due to the dominance of the majority class. Therefore, monitoring metric values class-wise helps to identify if the model performance is satisfactory for minority classes as well. '0' denotes 'default' class while '1' denotes 'non-default' class.

Firstly, it can be seen at an overall level that GenAI method performs similar to SMOTE method is most cases with an average difference of 1% for all metrics. When observing the performance of just the minority 'default' class, it can be seen that the performance of LR, SVM, and KNN in terms of F1-Score is quite low when compared to the 'non-default' class. This is a good example of why imbalanced datasets should be evaluated at a class level. RF outperformed the other 3 models in the phase 1 of training with an average accuracy of 98.9% and average F1-score of 95.4% for SMOTE. RF achieved an average accuracy of 97.6% and average F1-score of 95.4% for GenAI. At class level the accuracies were equal and F1-score was having an acceptable difference. It was hence selected as the base model to be stacked with XGBoost, AdaBoost, and LightBGM. The resilience of Random Forest to overfitting, due to its ensemble nature, may explain its superiority over the other base models. Additionally, it is known for handling high-dimensional data efficiently and maintaining accuracy when data has complex structures.

It is important to note that while SMOTE is often beneficial for handling class imbalance, it may not be optimal for all cases. Factors such as noise and the marginality of the minority class can adversely affect the efficacy of SMOTE, leading to a decrease in model performance.

By employing Random Forest in a stacked model, it can be seen that the performance improved. Interestingly, hyperparameter tuning of the Random Forest model did not enhance its performance, suggesting that the default settings were already well-optimized for the dataset. RF-StackingXGBoost and RF-StackingLightGBM outperformed RF-StackingAdaBoost with a similar average accuracy of 99.3% and average F1-score of 97.3% and 97.2% respectively, using SMOTE synthesis.

Interestingly, when XGBoost was trained without stacking it outperformed all other models, achieving an average accuracy of 99.4% and average F1-score of 97.4% using SMOTE synthesis. Hence, this was was selected as the best model. Existing study trained on fewer features, achieved 99.98% using LGBFS+StackingXGBoost. This study shows a competitive accuracy with additional features and just XGBoost. XGBoost's advantage over other models could be attributed to its use of regularization techniques which can reduce overfitting as well as its ability to manage sparse data.

Figure 25 shows the comparison between the accuracies obtained using SMOTE and GenAI techniques for synthesis. It can be seen that SMOTE is ahead by 1% at least for all models.

VIII. Conclusion

In conclusion, the adoption of GenAI and SMOTE techniques for synthesis presents a promising approach to mitigate

TABLE I: Model Performance Metrics

Model	Method	Class	Accuracy	Precision	Recall	F1-Score
Logistic Regression	GenAI	0	0.968	0.954	0.846	0.897
		1	0.968	0.970	0.992	0.981
Logistic Regression	SMOTE	0	0.984	0.927	0.824	0.873
		1	0.984	0.988	0.995	0.991
SVM	GenAI	0	0.968	0.959	0.843	0.897
		1	0.968	0.970	0.992	0.981
SVM	SMOTE	0	0.993	0.993	0.895	0.941
		1	0.993	0.993	0.999	0.996
KNN	GenAI	0	0.879	0.660	0.540	0.594
		1	0.879	0.913	0.945	0.929
KNN	SMOTE	0	0.940	0.581	0.372	0.454
		1	0.940	0.956	0.981	0.968
Random Forest	GenAI	0	0.976	0.980	0.868	0.921
		1	0.976	0.975	0.997	0.986
Random Forest	SMOTE	0	0.989	0.995	0.840	0.911
		1	0.989	0.989	0.999	0.994
RF-StackingXGBoost	GenAI	0	0.982	0.965	0.921	0.943
		1	0.982	0.985	0.993	0.989
RF-StackingXGBoost	SMOTE	0	0.993	0.989	0.912	0.949
		1	0.993	0.994	0.999	0.997
RF-StackingAdaBoost	GenAI	0	0.979	0.965	0.906	0.935
		1	0.979	0.982	0.994	0.988
RF-StackingAdaBoost	SMOTE	0	0.990	0.960	0.889	0.923
		1	0.990	0.992	0.997	0.995
RF-StackingLightGBM	GenAI	0	0.981	0.968	0.914	0.940
		1	0.981	0.983	0.994	0.989
RF-StackingLightGBM	SMOTE	0	0.993	0.986	0.910	0.947
		1	0.993	0.994	0.999	0.996
XGBoost	GenAI	0	0.982	0.971	0.919	0.944
		1	0.982	0.984	0.995	0.989
XGBoost	SMOTE	0	0.994	0.989	0.915	0.951
		1	0.994	0.994	0.999	0.997

risk within Peer-to-Peer (P2P) lending networks. By leveraging advanced ensemble models and synthetic data generation, finance domains can enhance their predictive models and better identify potential defaulters. To maintain the project's relevance and efficacy in the constantly changing peer-to-peer lending landscape, cooperation with industry stakeholders, ongoing model refinement, and investigation of cutting-edge technologies will be essential to help build a more informed and responsible lending ecosystem. The comparative study between GenAI and SMOTE techniques resulted in SMOTE performing better by 1% on average for most cases. This result underscores the efficacy of SMOTE technique in risk mitigation in lending networks. This however gives rise to the potential of experimenting with open source GenAI tools in other domains to see if it can act as an alternative owing to its advantages such as adaptability and dynamic optimization. XGBoost, when trained on a balanced dataset using SMOTE, achieves superior performance metric values compared to stacking approaches, thereby also being more optimized.

IX. Future Scope

By adding more features and using more advanced machine learning algorithms, the predictive model can be continuously improved. The predictive power of the model may be improved by integration with cutting-edge technologies, such as sentiment analysis of borrower narratives through natural language processing. Working together with P2P lending platforms would provide the model access to real-time data and feedback, allowing it to adjust to changing market conditions. Moreover, verifying the model's efficacy will require determining whether it can be implemented in a real-world environment and evaluating how well it works there. GenAI offers the capability to adapt and optimize models dynamically which can potentially improve model accuracy and robustness over time.

References

[1] 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, Jun. 2002, doi: 10.1613/jair.953.

[2] Levi. "Synthetic Data Generation with the Highest Accuracy for Free." MOSTLY AI, Nov. 21, 2023. Available at: https://mostly.ai/.

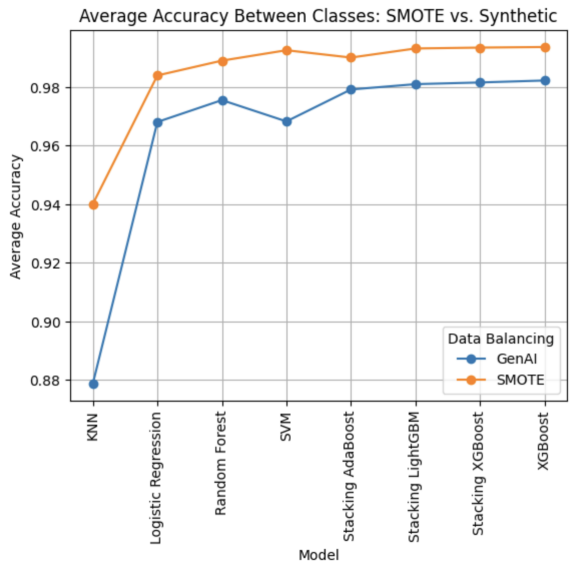


Fig. 25. Advantage and Disadvantages of base and stacked models for the Lending Club dataset

[3] Y. Pristyanto, A. F. Nugraha, I. Pratama, and A. Dahlan, "Ensemble Model Approach For Imbalanced Class Handling on Dataset," in *2020 3rd International Conference on Information and Communications Technology (ICOIACT)*, Yogyakarta, Indonesia, 2020, pp. 17-21, doi:10.1109/ICOIACT50329.2020.9331984.

[4] Mukherjee, Mimi, and Matloob Khushi. "SMOTE-ENC: A novel SMOTE-based method to generate synthetic data for nominal and continuous features." *Applied System Innovation* 4.1 (2021): 18. Publisher: MDPI.

[5] Muslim, Much Aziz, Tiara Lailatul Nikmah, Dwika Ananda Agustina Pertiwi, Yosza Dasril, and others. "New model combination meta-learner to improve accuracy prediction P2P lending with stacking ensemble learning." *Intelligent Systems with Applications* 18 (2023): 200204. Publisher: Elsevier.

[6] Shen, Feng, Xingchao Zhao, Zhiyong Li, Ke Li, and Zhiyi Meng. "A novel ensemble classification model based on neural networks and a classifier optimisation technique for imbalanced credit risk evaluation." *Physica A: Statistical Mechanics and its Applications* 526 (2019): 121073. Publisher: Elsevier.

[7] Caruso, Giulia, SA Gattone, Francesca Fortuna, and Tonio Di Battista. "Cluster Analysis for mixed data: An application to credit risk evaluation." *Socio-Economic Planning Sciences* 73 (2021): 100850. Publisher: Elsevier.

[8] Y. Chen and R. Zhang, "Research on credit card default prediction based on K-Means SMOTE and BP Neural Network," *Complexity*, vol. 2021, pp. 1–13, Mar. 2021, doi: 10.1155/2021/6618841.

[9] H. Wang and L. Cheng, "CatBoost model with synthetic features in application to loan risk assessment of small businesses," *arXiv* (Cornell University), Jun. 2021, doi: 10.48550/arxiv.2106.07954.

[10] N. Park, Y. H. Gu, and S. J. Yoo, "Synthesizing individual consumers' credit historical data using generative adversarial networks," *Applied Sciences*, vol. 11, no. 3, p. 1126, Jan. 2021, doi: 10.3390/app11031126.

[11] Wordsforthewise (n.d). *Lending Club*. Retrieved from https://www.kaggle.com/datasets/wordsforthewise/lending-club (2018).

[12] M. Hossin and M. R. Sulaiman, "A Review on Evaluation Metrics for Data Classification Evaluations," *Int. J. Data Min. Knowl. Manag. Process*, vol. 5, no. 2, pp. 01–11, 2015, doi: 10.5121/ijdkp.2015.5201.

[13] Chen, T., Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 2016 Conference on Knowledge Discovery and Data Mining, 785-794.

[14] Wang, W., Chakraborty, G., Chakraborty, B. (2020). Predicting the Risk of Chronic Kidney Disease (CKD) Using Machine Learning Algorithm. Applied Sciences, 11(2).

[15] Wang, S., You, S. D., Zhou, S. (2023). Loan prediction using machine learning methods. Advances in Economics, Management and Political Sciences, 5(1), 210–215.

Lina Devakumar Louis received her M.S in Data Analytics from San Jose State University in 2023 and is currently pursuing her career in data engineering. Her research interests include generative AI applications, recommender systems and LLMs.

Andrew Dunton is currently working as an optical engineer in the display hardware industry. He specializes in writing software for the characterization and analysis of optical performance in display products. He seeks to further his strengths in data analytics and is pursuing an M.S. in Data Analytics from San Jose State University.

Sourab Saklecha received his B.E in Computer Science from Visvesvaraya Technological University (VTU) in 2020 and is currently pursuing his M.S in Data Analytics degree at San Jose State University. His research interests include Machine Learning and Image compression.

Swetha Neha Kutty Sivakumar received her B.E in Computer Science from Anna University in 2018. Currently, the author is pursuing Master's in Data Analytics from San Jose State University. Her research area of interests include Machine Learning and Natural Language Processing.

Abdul Sohail Ahmed received his B.Tech. in Computer Science and Engineering from the Vellore Institute of Technology (VIT) in 2021 and is currently pursuing the M.S. in Data Analytics from San Jose State University. His research interests include machine learning, and artificial intelligence.

Smeet Sheth received his B.Tech. in Computer Science and Engineering from Indus University in 2022 and is currently pursuing M.S. in Data Analytics from San Jose State University. His research interests include Machine Learning and Sports Analytics.

Shih Yu Chang received a B. S. E. E. degree from National Taiwan University, Taiwan, in 1998, and Ph. D. degrees in electrical engineering and computer engineering from University of Michigan, Ann Arbor, in 2006. From August 2006 to February 2016, he was the faculty in the Department of Computer Engineering, National Tsing Hua University, Hsinchu, Taiwan. From July to August 2007, Dr. Chang had been a visiting assistant professor at Television and Networks Transmission Group, Communications Research Centre, Ottawa, Canada. From June 2018, he began to provide lectures about machine learning, data science, and AI in San Jose State University, San Jose, CA, USA. Besides academic positions, Dr. Chang also provides consulting work as an AI technical lead focusing on applying machine learning techniques to automate office work. Dr. Chang has published more than 100 peer-refereed technical journals and conference articles in electrical and computer engineering. His research interests include the areas of wireless networks, wireless communications and signal processing. He currently serves as the technical committee, symposium chair, track chair, or the reviewer in networking, signal processing, communications, and computers.