**AI HACKATHON – BITE308L**

**CREDIT CARD APPROVAL PREDICTION**

**Team Members:**

23BIT0096 - Rahul R

23BIT0273 - Tarun S

23BIT0297 - Harie Nivas B A

23BIT0361 - Johnathan John

23BIT0398 - Bharath Kumaar S K

**Project Description:**

Credit scorecards are widely used in the financial industry as a risk control measure. These cards utilize personal information and data provided by credit card applicants to assess the likelihood of potential defaults and credit card debts in the future. Based on this evaluation, the bank can make informed decisions regarding whether to approve the credit card application. Credit scores provide an objective way to measure and quantify the level of risk involved.

Credit card approval is a crucial process in the banking industry. Traditionally, banks rely on manual evaluation of creditworthiness, which can be time-consuming and prone to errors. However, with the advent of Machine Learning (ML) algorithms, the credit card approval process has been significantly streamlined. Machine Learning algorithms have the ability to analyse large volumes of data and extract patterns, making them invaluable in credit card approval. By training ML models on historical data that includes information about applicants, their financial behaviour, and credit history, banks can predict creditworthiness more accurately and efficiently.

**AI in the Prediction:**

Artificial intelligence plays a transformative role in credit scoring. Traditional credit scoring models often fail to account for the complexity and variability of individual financial behaviours. AI, on the other hand, can process vast amounts of data, identify patterns, and make predictions with a high degree of accuracy. This allows for a more personalized and fair assessment of creditworthiness. AI credit scoring also has the potential to extend credit

opportunities to underserved populations, such as those with thin credit files or those who are new to credit, by considering alternative data in the scoring process.

**Research Paper Findings:**

**1)** **An Integrated Machine Learning and Deep Learning Framework for Credit Card Approval Prediction**

Traces the evolution of credit scoring from traditional statistical methods to modern machine and deep learning. It highlights that early methods like Logistic Regression struggled with large, imbalanced datasets. The review shows how newer techniques such as **Random Forests**, **Gradient Boosting**, and **Neural Networks** were introduced to improve accuracy and handle complex data. The authors also emphasize the importance of using techniques like **SMOTE** to address the common problem of data imbalance in credit approval datasets. The paper concludes by building on this research, proposing an integrated framework that combines these advanced methods for superior performance.

**2) Smart Credit Card Approval Prediction System using Machine Learning.**

The literature review shows that credit assessment has evolved from traditional methods to advanced machine learning and deep learning. It highlights the importance of using Explainable AI (XAI) for transparency and regulatory compliance , as well as addressing

**fairness and bias** in models. The review also emphasizes the value of comparative analysis of different algorithms and the use of unconventional data sources for more inclusive and accurate assessments.

**3)** **Outlier Detection Using Gaussian Mixture Model Clustering to Optimize XGBoost for Credit Approval Prediction**.

The literature review shows that recent research in loan approval prediction focuses on hybrid models and ensemble learning. The paper identifies a research gap where there has been little exploration of combining probabilistic outlier detection methods, such as the Gaussian Mixture Model (GMM), with classification models like XGBoost.

**4) Privacy-Preserving Credit Card Approval Using Homomorphic SVM: Toward Secure Inference in FinTech Applications**

The literature review focuses on the challenge of data privacy in machine learning, especially in cloud-based financial applications. It notes that while previous work has explored using Homomorphic Encryption (HE) for SVMs, these approaches often lack support for practical features like batch processing and noise tolerance. The paper aims to fill this gap by proposing a fully encrypted inference pipeline using a hybrid-kernel SVM under the CKKS scheme

**5) Data Attribute Selection with Information Gain to Improve Credit Approval Classification Performance using K-Nearest Neighbour Algorithm**

The literature review explains that bad credit can cause financial loss for companies. To prevent this, data mining and classification algorithms are used to analyse past customer data to make objective decisions about new loan applicants. The paper points out that irrelevant data attributes can decrease the performance and accuracy of a classification algorithm. The research aims to show that using feature selection with Information Gain can improve the accuracy of the K-Nearest Neighbours (KNN) algorithm.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Paper Name** | **Algorithms** | **Data Set** | **Performance measures** | **Limitations:** |
| 1) An Integrated Machine Learning and Deep Learning Framework for Credit Card Approval Prediction | LR, SVM, KNN, DT, RF, GB), Neural Networks (NN), XGBoost, Synthetic Minority Over-sampling Technique (SMOTE) | The research used extensive datasets of user application records and credit history. These two datasets were merged into a unified dataset with 777,715 samples. | * Precision * Recall * F1-score * Area Under the Curve (AUC) * Cohen's Kappa | their struggle with large, high-dimensional datasets and data imbalance issues  note that future work will focus on further optimization and exploring real-time applications of their methodology |
| 2)Smart Credit Card Approval Prediction System using Machine Learning. | Random Forest, Gradient Boosting, XGBoost, and Logistic Regression | historical credit card application data. This data includes applicant demographics, financial history, and employment details | accuracy, precision, recall, F1-score, and ROC-AUC. |  Integrating real-time data to enable dynamic decision-making.   Improving bias detection and mitigation to ensure fairness across demographic groups. |
| 3)Outlier Detection Using Gaussian Mixture Model Clustering to Optimize XGBoost for Credit Approval Prediction | Random Forest (RF), LOF, CBLOF, DBSCAN, IF, and K-Means | The dataset used is the credit card approval dataset from the  UCI Machine Learning Repository |  Accuracy: 90.58%   Precision: 90.75%   Recall: 92.20% | It does make recommendations for future work, which often implies areas for improvement |
| 4)Privacy-Preserving Credit Card Approval Using Homomorphic SVM: Toward Secure Inference in FinTech Applications | Homomorphic Encryption (FHE) , CKKS soft-margin Support Vector Machine (SVM). hybrid Polynomial-RBF kernel. SIMD (Single Instruction, Multiple Data) | Credit Card Approval dataset from the UCI ML Repository. |  Accuracy: 97.06%   Precision: 98.33%   Recall: 95.16%   F1-Score: 96.74%   Time: 44.9 ms | Suggests areas for future work, which can be interpreted as current limitations. |
| 5) Data Attribute Selection with Information Gain to Improve Credit Approval Classification Performance using K-Nearest Neighbor Algorithm | Information Gain for feature selection and K-Nearest Neighbors (KNN) for classification | A public credit card dataset from the UCI repository, | * Without Feature Selection: The highest accuracy achieved was 74.93%. * With Feature Selection: The accuracy increased to 82.46%. * Increase: The improvement in accuracy was 7.53%. | It does highlight that not all data attributes are relevant for the classification outcome |

**Flowchart of the Project:**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**1. Data Ingestion and Preprocessing**

This is the initial phase where raw data is prepared for the model. The process starts with a CSV file containing applicant information. You perform several key steps here:

* **Load Raw Data**: The pipeline reads the raw data file, which contains both features (applicant information) and the target variable (default status).
* **Feature Engineering**: This involves creating new, more informative features from the existing ones.
* **Data Cleaning**: The data is cleaned to handle missing values and inconsistencies. You replace infinite values with NaN and then fill them with a suitable measure like the median. You also drop any features that have zero variance (i.e., contain only one value) because they offer no predictive power.
* **Categorical Encoding**: All non-numeric (categorical) features, such as 'maritalStatus' or 'occupation', are converted into a numerical format that machine learning models can process. One-hot encoding is a common method for this, creating new binary columns for each category.

**2. Ensemble Modelling and Training**

Instead of relying on a single model, this stage uses a powerful **ensemble approach** to combine the strengths of multiple algorithms.

* **Data Splitting**: The cleaned data is split into a **training set** and a **testing set**. The training set is used to teach the models, while the testing set is reserved to evaluate their performance on data they have never seen before.
* **Base Model Training**: Multiple diverse machine learning algorithms are trained independently on the training data. The models you mentioned are excellent choices for this:
  + **XGBoost**, **LightGBM, LR, RF**, and **CatBoost**: These are gradient boosting models that sequentially build decision trees, each one correcting the errors of the previous. They are known for their high performance and efficiency on structured data.
  + **Neural Networks (CNN, MLP, LSTM)**: Although more commonly used for complex data types, these deep learning models can be applied to tabular data to capture non-linear relationships that boosting models might miss.
* **ADASYN Balancing**: To address the common problem of class imbalance (where there are far fewer defaulters than non-defaulters), **ADASYN** is used. This technique generates synthetic data points for the minority class, helping the models learn a more accurate representation of both classes.

**3. Evaluation and Decisioning**

This stage is about assessing the models' performance and turning their output into a business-ready decision.

* **Model Scoring**: Each base model produces a prediction score (a probability of default). These scores are then fed into a final model.
* **Ensemble Scoring (Stacking/Blending)**: The final model combines the individual scores from the base models. This can be done through:
  + **Stacking**: A "meta-model" learns how to best combine the predictions of the base models to produce a final, more accurate prediction.
  + **Blending**: A simpler method that uses a weighted average of the base model predictions.
* **Test Metrics**: The final ensemble model is evaluated using key metrics to validate its performance on the unseen test data. The most important metrics are:
  + **Precision**: Measures the accuracy of the positive predictions (i.e., when the model predicts "default," how often is it right?).
  + **ROC-AUC**: Measures the model's ability to distinguish between the two classes across different thresholds.
* **Optimal Threshold**: The **Youden's J statistic** or a similar method is used to find the best probability threshold for making a final decision. This threshold balances the trade-off between false positives and false negatives, which is crucial for business outcomes.

**4. Deployment and Visualization**

This final stage focuses on putting the model into action and providing useful insights.

* **Final Model Training**: The full dataset is used to retrain the best-performing model (or the entire ensemble), ensuring it has learned from all available data before deployment.
* **API Endpoint**: The model is exposed via a **FastAPI endpoint**. This allows the front-end application to send a user's data to the backend and receive a prediction in real time.
* **Dynamic Visualization**: After a prediction is made, a new set of visualizations is dynamically generated and saved. These plots, such as the confusion matrix and feature importances, provide transparency and explainability for the model's decision, which is vital for a business to trust the results.
* **Results**: The final result—the approval decision, the risk probability, and the visualization URLs—is sent back to the front-end to be displayed to the user.

**Model Comparison:**

1. XGBOOST:

A screenshot of a computer

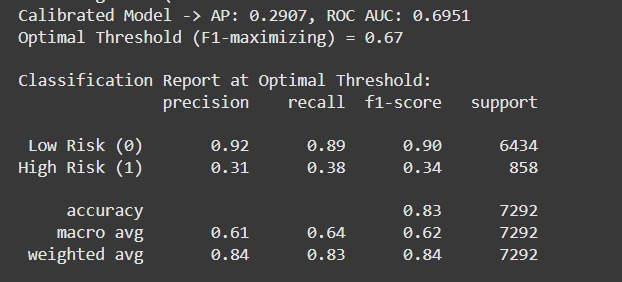
AI-generated content may be incorrect.

1. CATBOOST:

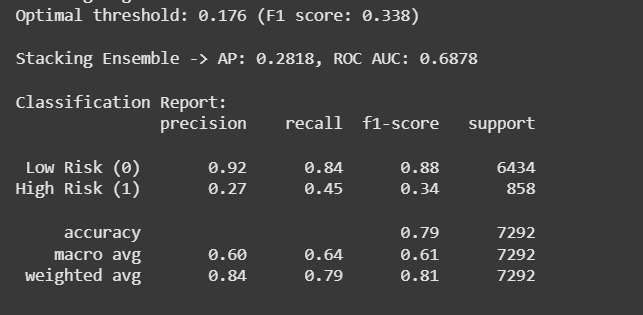
A screenshot of a computer

AI-generated content may be incorrect.

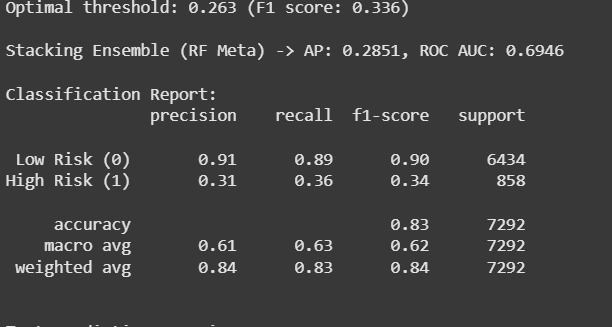
1. LightGBM:



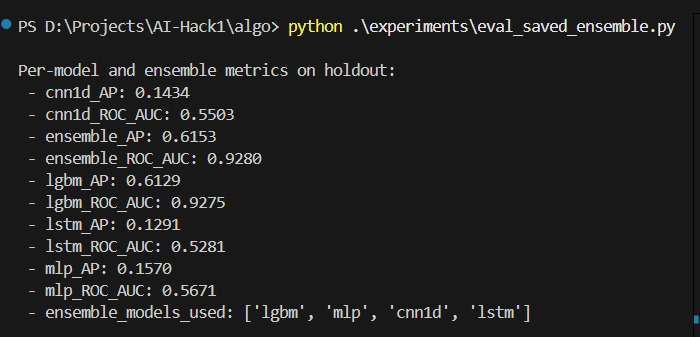
1. Logistic Regression:



1. Random Forest:



1. Deep Learning Algorithms:



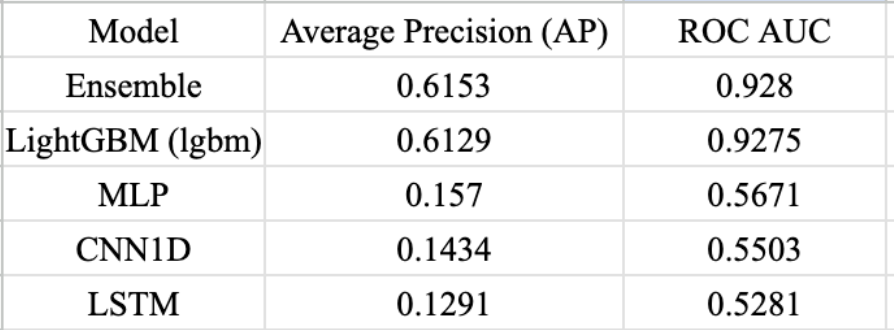
**Comparison Table:**

Classical ML Algorithms:

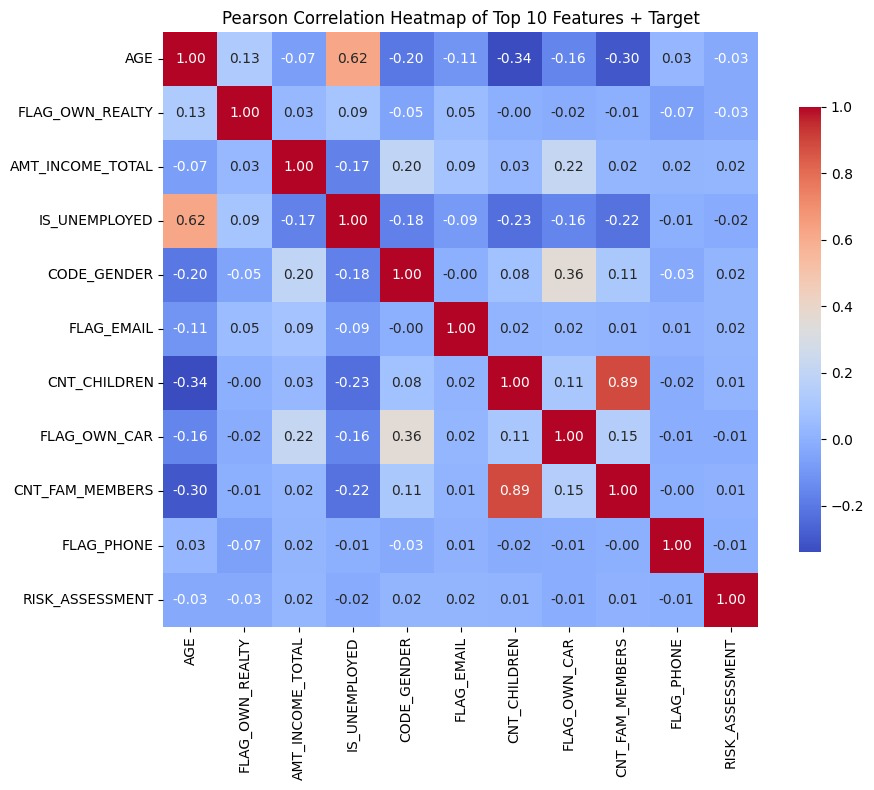
**A table with numbers and text

AI-generated content may be incorrect.**

Deep Learning Algorithms:



**Pearson Correlation Heatmap of top 10 Features:**

****

**References**

1) L. Thomas, J. Crook, and D. Edelman, "Credit scoring and its applications." SIAM, 2017.

2) Adekanmbi, Olusegun; Akinola, Caleb; Afolabi, Samson, "Credit Card Approval Prediction in Emerging Markets: A Case Study in Nigeria" Proceedings of the International Conference on Data Science, 2022.

3) Kim, Jiyoung; Lee, Minhyung, "Recent Advances in Credit Card Approval Prediction" Proceedings of the International Conference on Machine Learning, 2021.

4) S. Sahu, R. Ganeshan, and V. Muneeswaran, “Homomorphic encryption enabled svm for preserving privacy of p2p communication,” in 2024 IEEE International Students’ Conference on Electrical, Electronics and Computer Science (SCEECS). IEEE, 2024, pp. 1–6.

5) Amancio, D. R., Comin, C. H., Costa, L. D., & Fora, O. (2013). *The Performance of an Algorithm can be Affected by the Dataset and Data Type Used*.