CSE / ECE 343: Machine Learning Project Final Report
Title: Visa Acceptance Prediction

Guneet Pal Singh
guneet22190@iiitd.ac.in

Mann Nariya
mann22278@iiitd.ac.in

Sujal Soni
sujal22513@iiitd.ac.in

Dushyant Singh
dushyant22181@iiitd.ac.in

Abstract

Millions of individuals apply for visas each year for various purposes like- work, study, travel and some for the purpose of immigration. However, the visa approval process is often challenging as it is very complex, lengthy and time consuming, this leads to frequent rejections. By developing a model for predicting visa approval outcomes, applicants can gain valuable insights into the data and the chances for the success of their applications, helping them minimize uncertainty and better prepare for the future by identifying the possible gaps in their applications. The link to our github repository is given here: Link

1. Introduction

The H-1B visa is a non-immigrant work permit that enables U.S. companies to employ foreign professionals in specialized roles. These roles usually require advanced knowledge and at least a bachelor's degree or an equivalent qualification. Fields commonly associated with H-1B holders include IT, engineering, finance, healthcare, architecture, and other technical domains. Obtaining approval for this visa can be challenging, as applications may be denied due to factors like insufficient qualifications or limited visa availability. To address these challenges, we propose developing a machine learning model to predict the likelihood of visa approval based on individual applicant data.

2. Literature Review

We conducted a literature review to understand the challenges and existing approaches related to visa approval predictions.

2.1. An Analytical Study of Regression Techniques for H-1B Visa Prediction

The paper reviews the application of Decision Tree and Random Forest regression to analyze H-1B visa outcomes based on data from 2020 to 2022. Besides case status, other characteristics such as job title, the standardized occupational classification (SOC) code, employer information, wage, and full- or part-time job status were considered. After data cleaning and transformation, exploratory analysis revealed an imbalanced class distribution, with significantly more approved cases than denied ones. To address this imbalance, the SMOTE method was employed to reduce the risk of overfitting. The Decision Tree model, while easy to interpret, showed a tendency to overfit. In contrast, the Random Forest model, an ensemble method that creates multiple decision trees and averages their outputs, was more accurate and less prone to overfitting. For performance evaluation, we used metrics like Mean Squared Error (MSE), Mean Absolute Error (MAE), and Logarithmic Loss, with Random Forest slightly outperforming Decision Tree. Although both models were suitable for predictions, Random Forest demonstrated better generalization capability, while Decision Tree was easier to analyze. Further research can refine these models and explore combining them to enhance prediction accuracy.[1]

2.2. Work Visa Analysis using Machine Learning Techniques

The paper focuses on the different techniques that can be used in Machine Learning for prediction of the H1-B visa outcomes. The various features are described below:

1) Previous Works: The authors of the papers have refenced various previous works exploring the prediction of the techniques used in machine learning model for prediction of the outcomes. Some of the techniques used by the referenced authors are the Artificial Neural Networks(ANN), K-means clustering and other classification algorithms like logistic regressions to effectively predict the classes in which they would be classified to. Studies have shown that all the above methods mentioned above are effective in the classification of their respective samples in their datasets.

2) Addressing the issue of the Imbalanced Datasets An-

other research addresses the issue of highly imbalanced datasets using the different techniques using techniques like E-SMOTE, under-sampling, over-sampling and SVMs thus demonstrating the feature extraction and balanced sampling to improve the prediction accuracy. The concepts of under and over sampling to change the number of samples in the classes which are used for the training of the model to strike a balance between the number of samples.

3) Algorithms Evaluated: The paper implements their work using various machine learning models including the Random Forest, K Nearest Neighbors, Support Vector Machines and Logistic Regression for training of their model on the dataset. For the evaluation of the models various metrics like the F1 score, Recall, Accuracy were used in the implementation of the paper.

4) Outcome: The paper improves upon the current work that has been done in the field working on the improvements that can be made in the work of the current field and highlighting the importance of the balanced datasets in the training of the model.

5) Future Work: The paper also suggests the future work that can be done further on the field to improve the model by incorporating the different visa categories into the model like the student or work visas and the other related visas leading to more comprehensive understanding of the visa approval processes.[2]

3. Dataset

3.1. Dataset Description

The dataset is sourced from this link. More information about the dataset columns can be found in this file. The dataset contains 260 columns and 664,616 entries.

3.2. Dataset Analysis

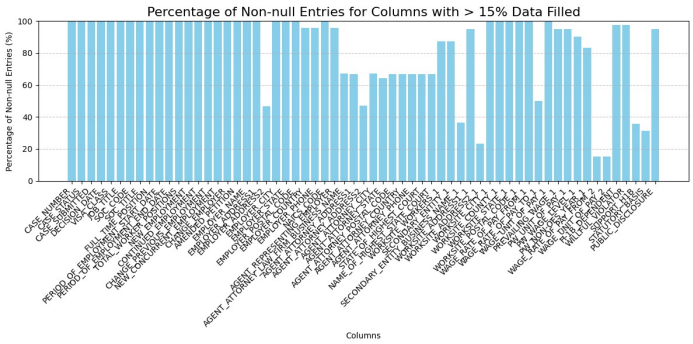

Figure 1. Percentage of Non-Null Entries

The target variable we are analyzing is CASE_STATUS, which can have the following values: CERTIFIED, WITHDRAWN, CERTIFIED-WITHDRAWN, and DENIED. There were only about 60 columns, which had more than 15% data filled. So we started analyzing these 60 columns.

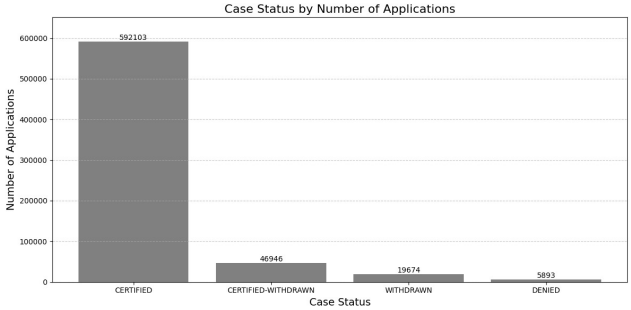

Figure 2. Bar Plot showing the number of sampled in each class.

3.3. Dataset Processing

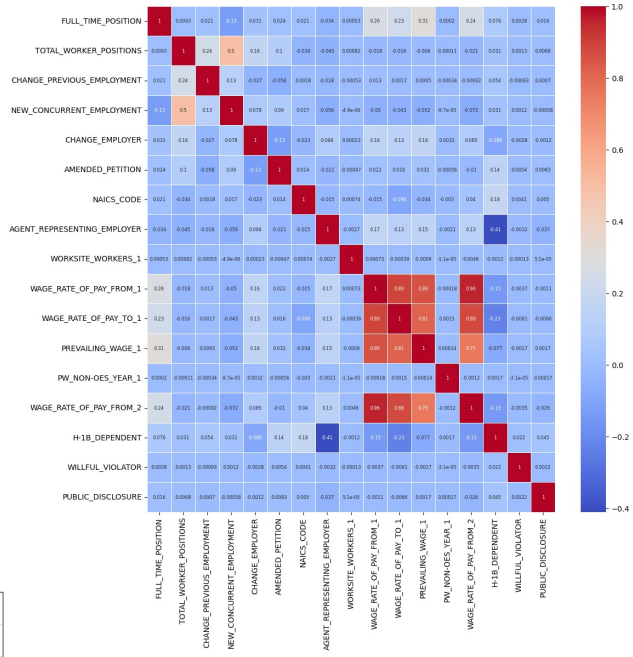

Figure 3. Heatmap of all the numeric features in the dataset.

- Columns containing entries as ”Y” and ”N” were replaced with 1 and 0, respectively, to make the data easier to interpret and plot.Also most of the columns contained 99% of entries as 0 and 1 so removed the other entries
- In the columns ”PREVAILING_WAGE_1” and ”PW_UNIT_OF_PAY_1”, we converted all entries to a yearly format to maintain consistency, ensuring the pay unit remained uniform across the dataset.
- The column ”WORKSITE_STATE_1” contained both full state names and abbreviations (e.g., ”CA” for

California) so we created a new column, "WORKSITE_STATE_FULL," where all abbreviations were replaced with full state names and then merged this column with "WORKSITE_CITY_1" to form a new column, "WORKSITE," which includes both the city and state for each case.

- We utilized a CSV file containing the latitude and longitude of U.S. states and cities. This information was stored in new columns, "lat" and "long," which were later used for prediction purposes. The CSV file can be found HERE.
- Another CSV file containing COLI (Cost of Living Index) data was used. We extracted this information and stored it in a new column, "coli," linking it with the states and cities for each case. The two CSV files used were CITY and STATE.
- Using the HeatMap that we created comparing the similarities between all the different columns that we had, we dropped the columns that had values ¿= 0.5 because the greater values represent a higher correlation among features, and using both features for generating models will not be of much value.
- We added more data points in the datasets with classes other than CERTIFIED to reduce the class imbalance and make the database more reliable. The data that we added is from the same official source but contains the entries from the data in the year 2018 and 2019.

Finally, we are left with only 22 columns. The dataset was reduced from 7 lakh data entries to 2 lakh data entries Detailed description of these columns are available HERE

4. Methodology

4.1. Correcting Imbalances

While preprocessing the data for the model we came across the problem of a very high imbalance between the number of cases in the various case statuses.For solving the problem, the following steps were implemented to handle the problem. The various methods are given below:

4.1.1 Under Sampling

We applied undersampling to the majority class by reducing its size to 10 times the count of the minority class while keeping the minority classes unchanged. We then combined these minority and majority classes of data in a random fashion in the data frame. This method helps balance the imbalanced datasets, thus preventing the model from favoring the majority class and having a more balanced view, which reduced the risks overfitting of the model while training.

4.1.2 Over Sampling

We used over sampling to address class imbalance in a dataset. We separate the features and target variable from the original DataFrame by identifying categorical columns to encode them with 'LabelEncoder' after which the dataset is split into the training and testing set. We verify the class balancing after SMOTE to ensure balanced dataset for fair model training.

4.2. Models Details

We trained the following models over the semister for finding the best possible accuracies.

4.2.1 Decision Tree

Decision trees is a tree-based classification algorithm which uses metrics like entropy and Gini index for splitting the dataset into different classes using the best possible classification of the data. Their models are prone to overfitting which can reduce it's performance when we move to the testing data.

4.2.2 Random Forest

Random forest is an ensemble learning technique that combines multiple decision trees and then considers their combined results to find the correct result. As it uses multiple weak single classifiers, it can reduce overfitting.

4.2.3 Naive Bayes

Naive Bayes is a statistical model which uses the assumption of conditionally independent features along with using Bayes theorem for predicting the class to which the model belongs. Despite the naive assumption the model gives reasonable results thus it is an accepted model in the field.

4.2.4 XG Boost

The XG Boost algorithm is an advanced boosting algorithm. Like Random Forests it also uses ensemble learning with the weak classifiers where each next classifiers correct the errors made by the next classifiers for continuous improvement of the model's accuracy.

4.2.5 Perceptron

The perceptron model is a linear classifier. It uses the the conepts of hyperplanes to adjust the weights of the input we take in the model while training. It is reasonably practical for binary classification but it's performance degrades as we move to more than two classes.

4.2.6 Multilayer Perceptron

Multilayer perceptron is a neural network consisting of multiple layers of neurons. It uses non linear activation function for capturing the complex patterns in the data of the dataset. MLPs can handle both classification and the regression tasks.

4.2.7 Support Vector Machine (SVM)

SVMs are used for both classification and regression tasks.It works by calculating the most optimal hyperplane separating data into the different classes. The SVM aims to maximize the margin between the data points and their margins which makes it effective in high dimension spaces.

4.2.8 K-Means Clustersing

K means clustering works by segregating the clusters into K different groups based on feature similarity, where each data point is assigned to nearest centroid and the coordinates of the centroids are updated in every iteration.

4.2.9 K-Nearest Neighbors

K Nearest Neighbors is a supervised learning algorithm. It classifies the data points based on the majority class of the data points which are the K closes to the point being worked on in the feature space.

5. Results and analysis

The following tables contain the accuracy value which we used as a metric for the evaluation of the different models with the different modifications on the data.

Model	Unscaled Data	Oversampled Data	Undersampled Data
Decision Tree	88	86	51
Random Forest	91	91	60
Naive Bayes	92	49	53
XGBoost	93	93	58
Perceptron	92	92	46

Table 1. Model Performance with Unscaled, Oversampled, and Undersampled Data on the Old Dataset

Model	K-Fold Oversampling	K-Fold with Undersampling
Decision Tree	93	59
Random Forest	96	59
Naive Bayes	27	53
XGBoost	71	58
Perceptron	25	27

Table 2. Model Performance with K-Fold Cross Validation on the Old Dataset.

Model	Oversampled Data	Undersampled Data
MLP	15	57
SVM	37	60
GBC	61	63
GM	52	15
KNN	54	60
CNN	15	57

Table 3. More Models Performance with Oversampled and Undersampled Data on the Updated Dataset.

Based on the above table of accuracies we can conclude that before updating our imbalanced dataset of the accuracies we had the most optimal ML model as our XG Boost model with the unscaled data. But after we modified our dataset based on the input from the invigilator in the mid-sem evaluation the model which had the best accuracy was the GBC model with the Undersampled data. This model showed the accuracy of 63 percent.

The reason for such big drop in the accuracy might be because in our old dataset we had highly imbalanceed data which caused the data in our testing set to be mostly "CERTIFIED" values causing the output to be mostly of one class only. This issue was solved in the updated dataset.

6. Conclusion

- We updated our pre-processing steps based on the Inputs from the Invigilator in the Mid-Evaluation and attempted to improve the balance in our highly imbalanced dataset.
- We were able to successfully extract the essential features from the dataset and trained multiple classification models.
- We were able to follow the proposed timeline tentatively as mentioned in the proposal and we were able to achieve all our goals mentioned in the propsal.
- We were able to successfully deploy our End-to-End model which we deployed on our own website using Node.js and Express.js for backend. For the frontend we used HTML/CSS.
- We were able to successfully tune the hyper parameters for the models to improve the accuracy of the models overall.

Individual Contributions

- **Sujal Soni:** Data Analysis and Processing,Problem Statement and Dataset Identification, Model Training (Major Contribution).

- **Mann Nariya:** Literature Review, Data Analysis, Model Deployment, Model Training
- **Guneet Pal Singh:** Literature Review, Data Processing, Model Training.
- **Dushyant Singh:** Data Analysis and Processing,Problem Statement and Dataset Identification, Model Training, Model Deployment (Major Contribution).

7. References

1. Raj, P. B. A. S., Piri, J., Reddy, S., & Eluri, S. B. (2023, May). An analytical study of regression techniques towards H-1B visa prediction. In *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 871-876). IEEE.
2. P. B. Aakash Sai Raj, J. Piri, S. B. Eluri, & S. R. S. (2023). Work visa analysis using machine learning techniques. In *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, Coimbatore, India (pp. 616-621). doi: 10.1109/ICAIS56108.2023.10073837. **Keywords:** Machine learning algorithms; Costs; Law; Employment; Machine learning; Forestry; Data science; H1B visa prediction; Random Forest algorithm; Synthetic Minority Oversampling Technique (SMOTE); Imbalanced dataset.
3. Dataset: https://www.dol.gov/sites/dolgov/files/ETA/oflc/pdfs/H-1B_Disclosure_Data_FY2019.xlsx
4. COLI'city CSV: https://advisorsmith.com/data/coli/#data
5. COLI state CSV: https://worldpopulationreview.com/state-rankings/cost-of-living-index-by-state
6. Longitude and latitude CSV: https://simplemaps.com/data/us-cities
7. GitHub repository for reference: https://github.com/sharan-naribole/H1B_visa_eda/tree/master
8. SMOTE oversampling for imbalanced classification: https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/