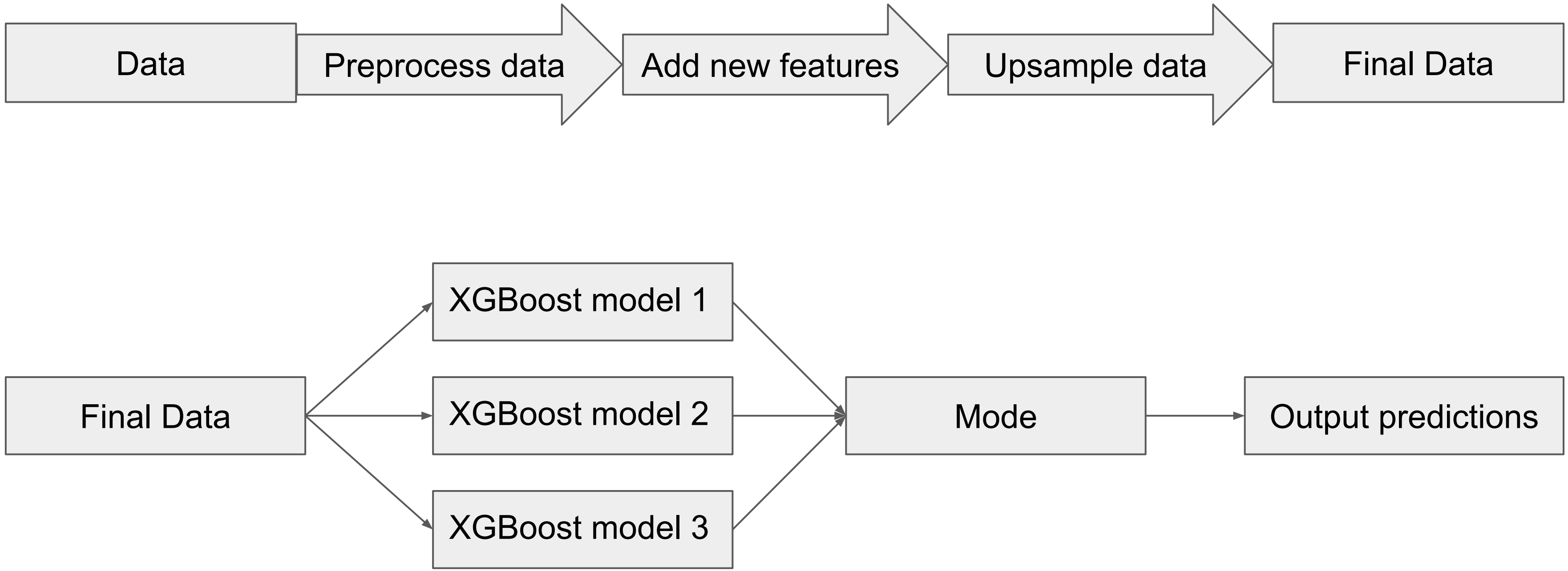
ZS Online Hackathon

M R Abhishek

Problem

- The H1-B visa is a type of Visa in United States that allows U.S. employers to temporarily employ foreign workers in speciality occupations.
- The given dataset has records from H1-B visa applications for the years 2007-2017. The data-set has ~4 lakh records with 27 features.
- The task is to predict the CASE_STATUS of H1B Visa with the given data-set.

Overview


Data
Preprocess data
Add new features
Upsample data
Final Data
Final Data
XGBoost model 1
XGBoost model 2
XGBoost model 3
Mode
Output predictions

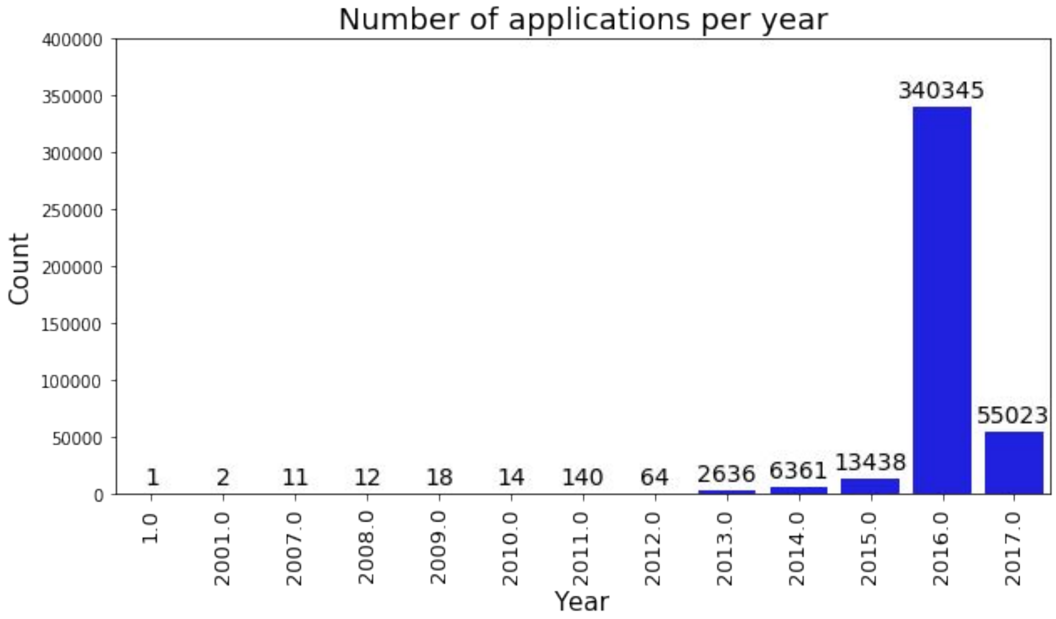
Data Analysis and Preprocessing

Dropped columns

- CASE_NO - This is a unique number for each CASE. Since, it does not capture any information, I dropped this column.
- EMPLOYER_NAME - This column has been removed because there are many categories(53,463). Most of the categories occur only once. So, it's better to drop this feature.
- EMPLOYER_COUNTRY - There are 4 categories(USA, CANADA, AUSTRALIA, CHINA) in this column. Out of all Visa cases in training set, only 9 belong to CANADA, AUSTRALIA and CHINA. In test data, all the cases belong to USA. As this feature is constant in test data, this feature has been dropped.
- WORKSITE_POSTAL_CODE - There are 17427 categories. I have compared the performance with and without this feature. Since, it didn't affect the score much, I dropped this feature.

Errors found in the dataset

- PW_SOURCE_YEAR has year information from 2007 - 2017 and 2001. One of the record has a value of ‘1’, which is not an year. Hence, I replaced that value with the mode of all the values in that feature.



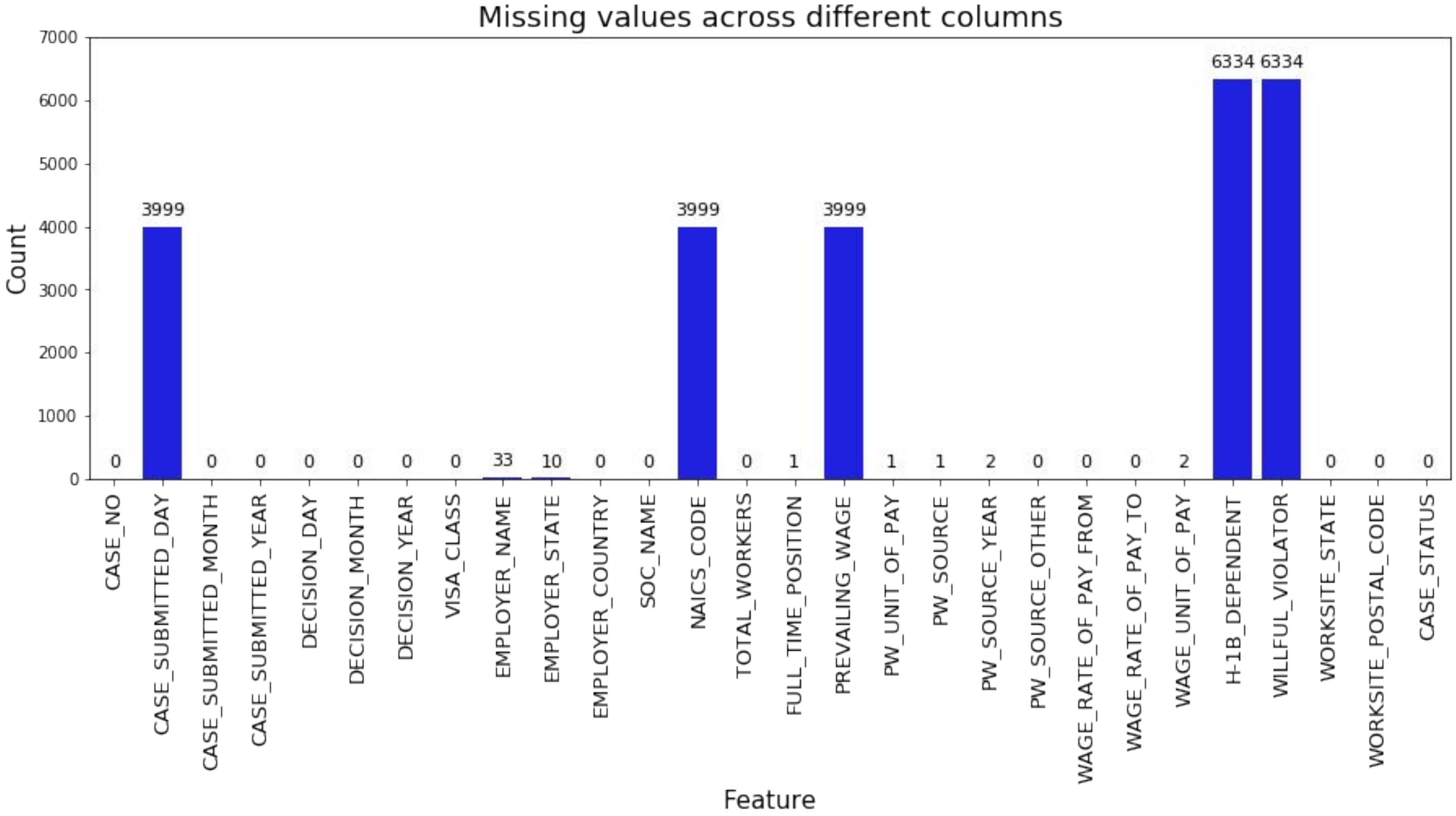
Errors found in the dataset

- It is obvious that WAGE_RATE_OF_PAY_FROM must be greater than WAGE_RATE_OF_PAY_TO. Some of the records in WAGE_RATE_OF_PAY_TO contains the value zero(0). So, I replaced those values with the values in WAGE_RATE_OF_PAY_FROM.

WAGE_RATE_OF_PAY_FROM	WAGE_RATE_OF_PAY_TO
71000.0	91000.0
51000.0	0.0
28455.0	0.0
62000.0	82000.0
72000.0	92000.0

WAGE_RATE_OF_PAY_FROM	WAGE_RATE_OF_PAY_TO
71000.0	91000.0
51000.0	51000.0
28455.0	28455.0
62000.0	82000.0
72000.0	92000.0

Missing values per column



Impute missing values

- CASE_SUBMITTED_DAY has day information. So, I imputed missing values by randomly sampling numbers between 1 to 28 (if it is between 1 to 31, some of the months may not have that many days).
- EMPLOYER_STATE, NAICS_CODE, H-1B_DEPENDENT and WILLFUL_VIOLATOR are categorical features. They were imputed with the mode of all the values in that feature.
- PREVAILING_WAGE is a numerical feature with 3999 null values. The null values were imputed with the mean of all the values in that feature.

Generation of new features

- Used (CASE_SUBMITTED_DAY, CASE_SUBMITTED_MONTH, CASE_SUBMITTED_YEAR) and (DECISION_DAY, DECISION_MONTH, DECISION_YEAR) to get CASE_SUBMITTED_DATE and DECISION_DATE date-time features respectively.
- Difference between CASE_SUBMITTED_DATE and DECISION_DATE date-time columns which carries information about the gap (in number of days) is labelled as DECISION_PERIOD.

	CASE_SUBMITTED_DATE	DECISION_DATE	DECISION_DAY	DECISION_MONTH	DECISION_YEAR	DECISION_PERIOD
0	2011-03-23	2017-04-14	14	4	2017	2214
1	2011-03-28	2017-03-10	10	3	2017	2174
2	2012-02-17	2016-10-18	18	10	2016	1705
3	2012-03-22	2017-04-14	14	4	2017	1849
4	2012-03-22	2017-04-14	14	4	2017	1849

Generation of new features

- A new boolean feature IS_ES_SAMEAS_WS is added which tells whether the EMPLOYER_STATE and WORKSITE_STATE are the same or not.

	EMPLOYER_STATE	WORKSITE_STATE	IS_ES_SAMEAS_WS
0	CA	CA	0
1	MD	MD	0
2	KY	KY	0
3	CA	CA	0
4	CA	CA	0
5	NJ	TX	1
6	NJ	WI	1

- PW_SOURCE_OTHER is a categorical feature that has several categories in which most of the categories have occurred less than 10 times. Such categories are clubbed into one category. Around 116 categories are combined into one category ('OTHER').

Generation of new features

- A new division feature RATIO_OF_PAY_FROM_TO (WAGE_RATE_OF_PAY_FROM / WAGE_RATE_OF_PAY_TO) has been created. Most of the methods capture information from the addition and subtraction of features. But, it is difficult to extract information from division of two features. So, I have created this feature to extract hidden information. This feature has increased the F1 score.

	WAGE_RATE_OF_PAY_FROM	WAGE_RATE_OF_PAY_TO	RATIO_OF_PAY_FROM_TO
0	71000.0	91000.0	0.780220
1	51000.0	51000.0	1.000000
2	28455.0	28455.0	1.000000
3	62000.0	82000.0	0.756098
4	72000.0	92000.0	0.782609

Generation of new features

- Infosys, Capgemini, IBM, TCS, Tech Mahindra, Google are the top companies submitting the applications for their employees.
- However, the application is most likely to be accepted if it is from an University. The below table shows the number of records per label that are submitted by a University(14369 records).
- So, a boolean feature 'IS_UNIVERSITY' is created which tells whether the application is from a University or not.
- EMPLOYER_NAME feature has been dropped.

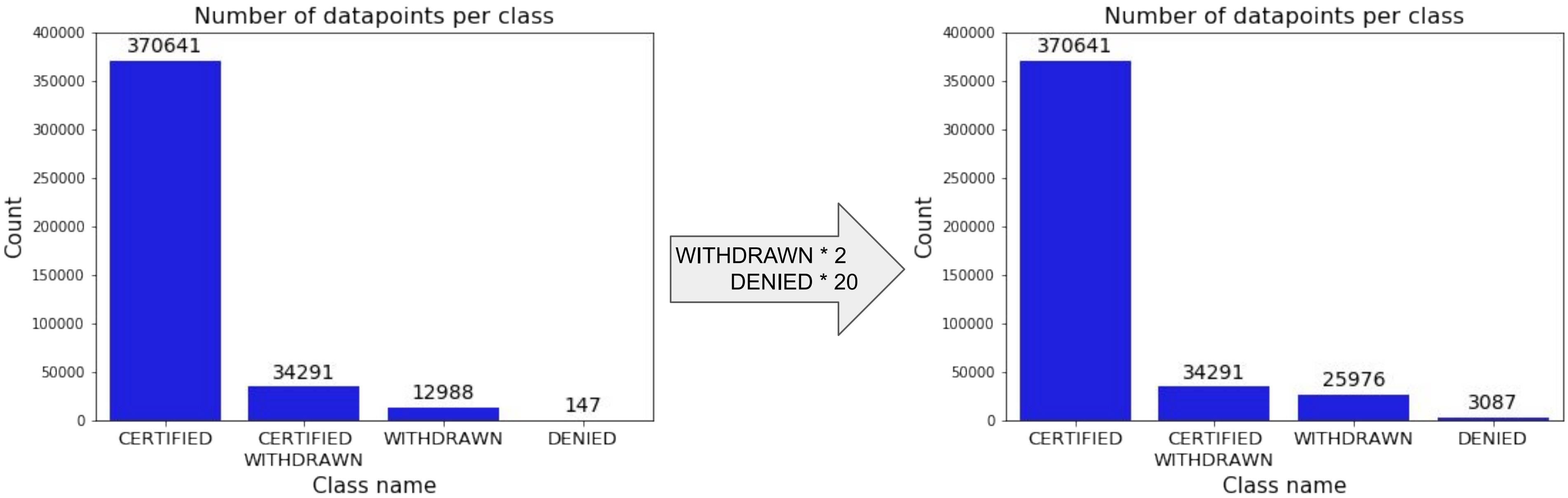
Label	Number of records
CERTIFIED	10680
CERTIFIED WITHDRAWN	3228
WITHDRAWN	459
DENIED	2

Modelling

Modelling

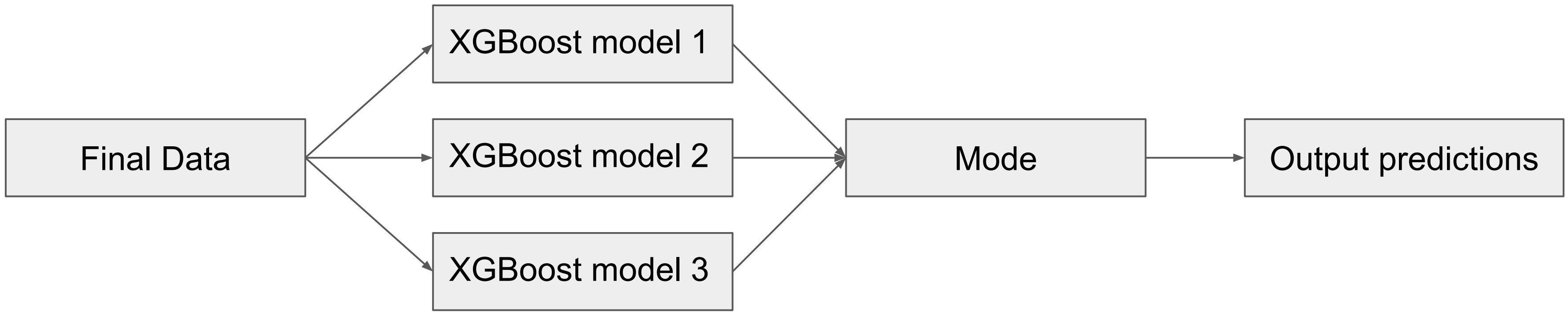
- After performing data cleaning and imputing NULL values, I applied XGBoost Classifier as a base model. This gave 96.58 score on LB.
- Later, I included additional features that were generated. These additional features boosted my score. I achieved 97.72 F1 score.
- Since the dataset provided is imbalanced, I manually up-sampled the data-points with lower class by 20 times and finally achieved a score of 97.78.

Imbalance data-set

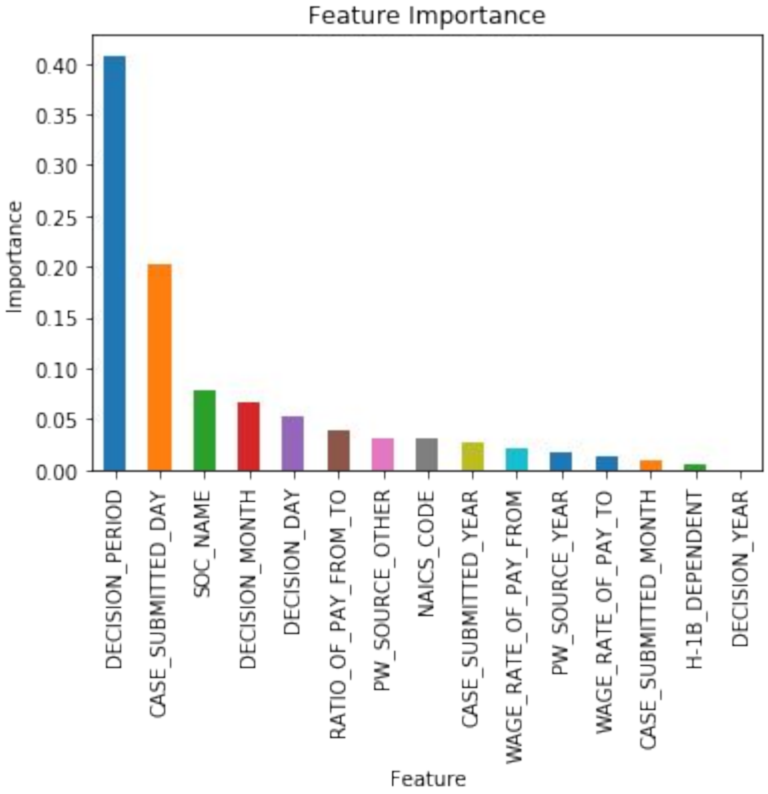


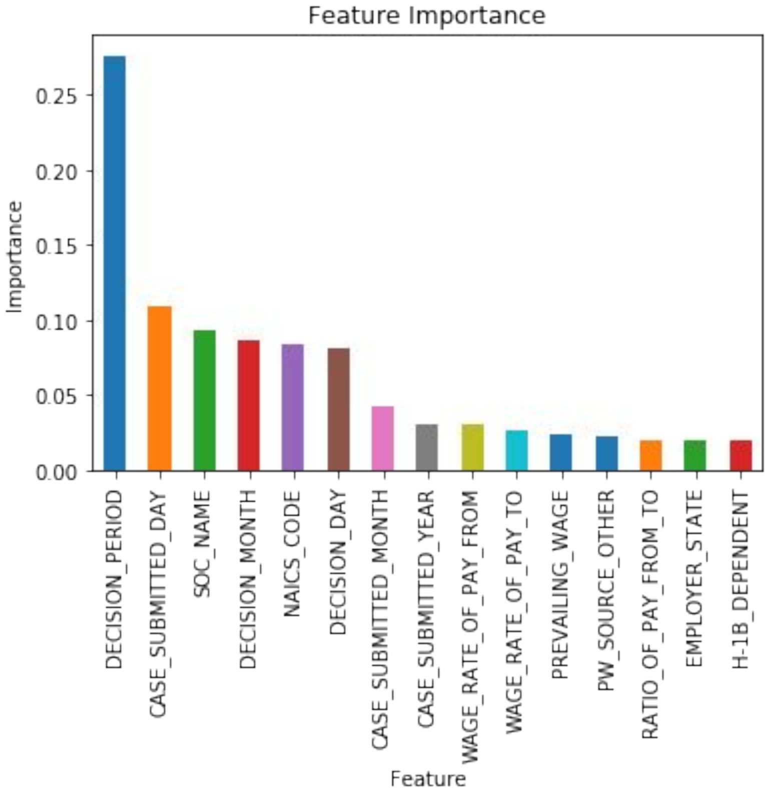
Modelling

- For my final submission, I applied 3 XGBoost models by varying n_estimators, max_depth, random_state, learning_rate parameters and ensembled the three. I took mode of the three outputs and finally achieved best score of 97.99 on LB.
- I also used other models (Light GBM and Random Forest Classifier) and made submissions. However, XGBoost outperformed other models.
- The final model was the mode of three XGBoost models.



Significant variables - Two XGBoost models





Thank you