Canvas group: 12

DATA MINING PROJECT

Title: Predicting Case Status of H1-B Petitions

Prof: David Anastasiu

San Jose State University

Spring 2018

Team Members:

V. Vineela 012445186
Suhani Vaishnav 012489191
Keerthi Veerabhadraiah 012466116
Canvas group Id : 12

Canvas group: 12

INTRODUCTION

Motivation:

The motivation for this project stemmed from the curiosity to understand the H1B visa petition process, H1B being the most sought out visa category. Since many factors affect the certification, the raw data (obtained from Office of Foreign Labor Certification (OFLC)) was challenging with maximum number of attributes being categorical and hence furthered our interest to work upon it.

Objective:

The aim of this project is to implement various data mining techniques, to understand how the petitions received for H1B visa category are analyzed and predict the status of application. Also, the factors mainly affecting selection are obtained, post implementation. The algorithm designed in the project can help both, H-1B applicants as well as the employers to gauge the likelihood of obtaining certification to further proceed with the process.

Literature Review:

The data for this project was obtained from the Office of Foreign Labor Certification (OFLC) website. This website generates data which is both useful as a measure of program effectiveness as well provides the external stakeholders the required information about OFLC's immigration programs. On the performance data page, there are four tabs, explained below:

a) Annual Report Performance Data: Gives an overview of statistics for each state in terms of applications processed and their percentage in overall applications by the nation.
b) Annual Performance Reports: This report presents employment-based cumulative immigration program data and analysis based on applications submitted to the Department by employers across the country.
c) Selected Statistics by Program: This section presents program factsheets, displaying key selected statistics about each of the major immigration programs. These factsheets include cumulative information and are updated on a quarterly basis.
d) Disclosure Data: This page allows the public to access the latest quarterly and annual disclosure data in easily accessible formats for the purpose of performing in-depth longitudinal research and analysis. OFLC case disclosure data is available for download by the federal fiscal year cycle covering the October 1 through September 30 period (all disclosure data sets are saved in the Microsoft Excel (.xls) file format).

There is a Kaggle (data-science competition platform) competition also for this dataset, wherein the data is already preprocessed and supplied. The submissions there, comprise of mostly analysis instead of prediction like, analysis of data science jobs over the years, wage distribution, demographics etc.

Canvas group: 12

SYSTEM DESIGN & IMPLEMENTATION

Algorithms:

For Implementation, the following algorithms were considered based on our in-class knowledge:

I) **Dimensionality Reduction:** For reducing the number of dimensions, various algorithms were used one by one to check for the change in accuracy and better computation times. They are as follows:

a. Singular Value Decomposition
b. Principal Component Analysis
c. Random Projection
d. Locally Linear Embedding
e. Fisher's Linear Discriminant Analysis
f. Feature selection: Variance Threshold
g. Feature selection : Select K-best

II) **Classification:** For predicting the application status, various classification algorithms were implemented depending on reasons listed below:

a. K-Nearest Neighbor:
It does not make any assumptions in data distribution; hence this was selected. The attributes which are not so important will have same influence as the important attributes.

b. Stochastic Gradient Descent:
It is a simple and efficient approach for linear classification. It considers only one random change with changing weights; hence it works faster with large dataset. Since our dataset had 1.7million rows, this algorithm was selected.

c. Decision trees:
It represents a decision situation and helps in obtaining all the possible outcomes for an upcoming choice. Branches of the tree explicitly show all the factors that are considered relevant to the decision. It is due to this property that it works well for linear classification and hence it was chosen.

d. Extra-trees:
Extra trees or extremely randomized trees is a variant of random tree, unlike which, it picks decision boundaries at random without having any criteria, hence it was chosen to be implemented.

e. Multi-layer perceptron (Neural network):
MLP is an universal approximator which works very well on linear or non-linear classification. With appropriate weight optimization method, it works best for data with thousands of training samples similar to our dataset.

f. Random Forest:
It handles thousands of input variables without variable deletion, giving an estimation of importance of each variable in the classification. It doesn't overfit the data and can handle large imbalanced dataset, hence this was chosen.

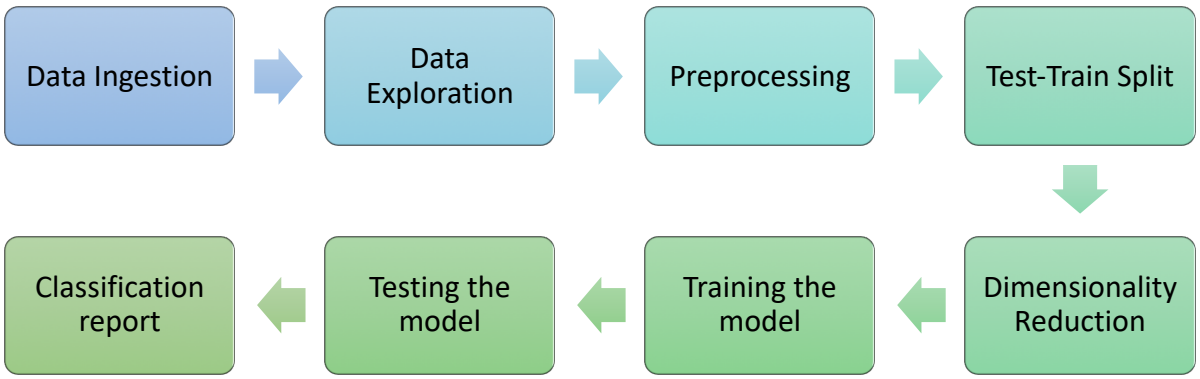
g. Adaboost:
Multiple weak learners can be made into a single strong learner. It gives better accuracy by using multiple instances of same classifier with different parameters, hence this was chosen to work along with the decision tree.

h. Bagging:
Bagging or Bootstrap aggregation works better with decision trees, compared to Boosting. Consistent behavior of the decision tree can be obtained from bootstrap aggregating, even without tuning the parameters. This helps in avoiding explicit fine-tuning.

i. Gradient Boost:
It is the best known technique for model improvement which works well for regression and classification problem. Decision trees can benefit from applying this algorithm ,hence the choice.

Architecture / Program Flow:

The program flow is as follows:



Tools & Technology used:

Software Tools	Technology
Anaconda/Jupyter notebook: for python programming	HPC @ SJSU: for execution of program
Tableau: for creating visualizations	

Canvas group: 12

EXPERIMENTS/ PROOF OF CONCEPT EVALUATION

Dataset:

The dataset for the project is taken from the website of United States Office of Foreign Labor Certification. It has year wise disclosure data for H-1B visa petitions.

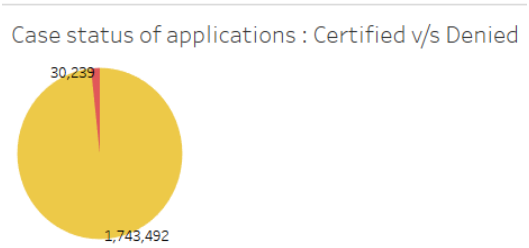
Ref: https://www.foreignlaborcert.doleta.gov/performancedata.cfm

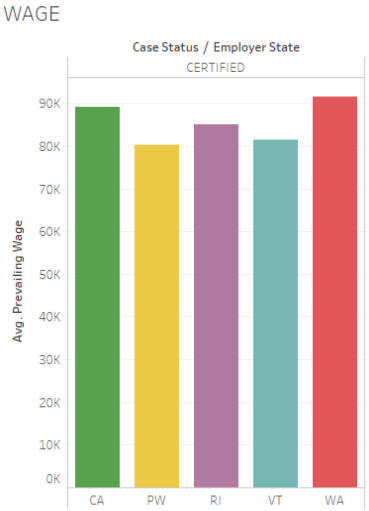
The data is in Excel format, year-wise each file. To encompass maximum data for training and testing, the sets for the years 2015-2018 were merged in to one, which resulted in 1.9 million rows and 53 attributes. This dataset was then used for further processing and prediction.

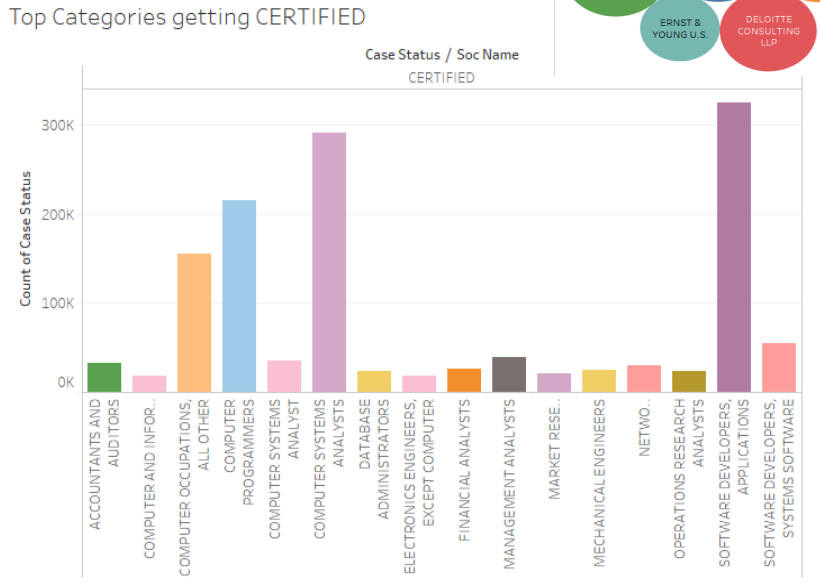
The attributes mainly categorize as follows:

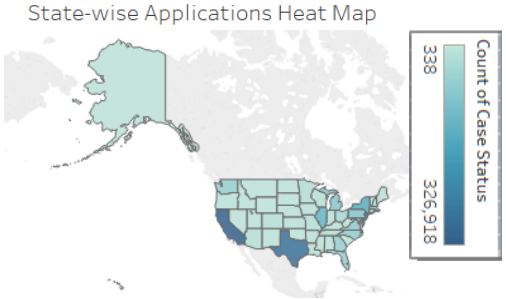
Application Case details	Employer details	Agent-attorney details	Applicant's employment details
• Case Number • Submission date, Decision date • Case Status	• Name • Address (City, Postal code, Sate, Country) • Employment Start and end date	• Name of the attorney • City and State	• Job title • SOC name and code • Wage : Prevailing wage for the SOC code as well as wage offered by employer • Full time position • Worksite location

Data Interpretation:









Canvas group: 12

Preprocessing:

The data was initially in the form of 4 excel files, each for the year 2015, 2016, 2017 & 2018. For each of the files, the concise table was tabulated to understand the data and attributes within. The common columns in these files were extracted and one master file was created. Also the null values for each column were obtained and based on the that, the columns with >50% of null values which did not affect prediction, were dropped. This number came to be 33.

The preprocessing was then applied to the remaining 20 columns and the steps are described below:

- **Dealing with Missing values:**
  - AGENT_REPRESENTING_EMPLOYER: This column was not present for the years 2015 & 2016 but it was present for the years 2017 & 2018. Using the attribute: AGENT_ATTORNEY_NAME (2015-16), the values for this column were imputed, based on presence of the attorney name in that column. The output was in the form of "Yes" or "NO".
    Subsequently the AGENT_ATTORNEY_NAME column was dropped.

  - EMPLOYER_COUNTRY: Based on the EMPLOYER_STATE value, if it indicated a state in the US, then the missing value for the EMPLOYER_COUNTRY was filled as "UNITED STATES".
    The EMPLOYER_COUNTRY column was then dropped.

- **Feature extraction:**
  - PREVAILING_WAGE: The PW_UNIT _OF _PAY had varying values from Weekly to Yearly and the corresponding wage values changed accordingly. To keep the same unit of pay, calculations were performed and the pay was transformed to Yearly value.

  - WAGE_RATE_OF_PAY_FROM: The WAGE_UNIT _OF _PAY had varying values from Weekly to Yearly and the corresponding wage values changed accordingly. To keep the same unit of pay, calculations were performed and the pay was transformed to Yearly value.

  - INCREMENT_FROM_PREV_WAGE: This column was created to represent the difference between WAGE_RATE_OF_PAY_FROM and PREVAILING_WAGE.

  - INCREMENT: This column was created to show the comparison between WAGE_RATE_OF_PAY_FROM and PREVAILING_WAGE. The output was in Boolean.
    Subsequently the following columns were dropped: WAGE_RATE_OF_PAY_FROM, WAGE_RATE_OF_PAY_TO, PW_UNIT_OF_PAY, WAGE_UNIT_OF_PAY, PREVAILING_WAGE.

  - APP_COUNT: A new column was created which showed the count of no. of applications submitted by each employer. This was created since creating dummies for

EMPLOYER_NAME would bring in the curse of dimensionality, leading to a vast increase in number of columns->time->memory ultimately.

- ACCEPTED_RATE: A new column was created to evaluate and show the acceptance rate for each employer. Subsequently the EMPLOYER_NAME column was dropped.

- DURATION: A new column was created to represent the contract time period for which the employer is hiring an employee. This was extracted as a difference of columns EMPLOYMENT_START_DATE & EMPLOYMENT_END_DATE and subsequently these were dropped.

- For the following categorical attributes: AGENT_REPRESENTING_EMPLOYER, H1-B DEPENDENT, WILLFUL_VIOLATER: There were many missing values and dropping them would have led to massive loss of data hence to compensate that a new "blank" (category)column was created while implementing dummies for all these columns to preserve the data within.
- For the following categorical attributes: EMPLOYER_STATE, PW_SOURCE, SOC_CODE, VISA_CLASS, WORKSITE STATE : dummies were created and new columns were appended.
- For the following binary attributes: INCREMENT and CASE_STATUS: the label binarizer was implemented.

Methodology:

- The test-train split ratio was chosen to be 20:80.
- Cross-fold validation was implemented with the number of folds(k) as 5 and 10 for the output of the top 3 classifiers.
- To improve the accuracy further, GridSearchCV was applied with various parameters.
- As the dataset is imbalanced, Down-sampling was carried out to check for better accuracy.

Algorithms Comparison:

As previously mentioned, a total of 7 dimensionality reduction algorithms and nine classifiers were considered. During implementation, 2 dimensionality reduction algorithms – LLE and Random Projection led to a very high computation time. Hence the two were discarded leading to a total of 45 combinations instead of 63. Below results are shown for 0.1million rows. This is to reduce computation time since results match the ones obtained for full dataset as well.

Below screenshot shows the F1 scores and computation time for all the combinations:

DR	SVD		PCA		Fisher's LDA		VarianceThreshold		Select Kbest	
Classifier	F1 Score	Time	F1 Score	Time	F1 Score	Time	F1 Score	Time	F1 Score	Time
KNN	97.05	137.25	97.05	24.8	96.83	3.82	97.05	26.86	96.99	63.05
Neural Network	97.05	11.19	97.05	10.1	97.52	0.79	97.05	8.18	96.99	6.08
SGD	97.86	0.21	97.86	0.15	97.48	0.03	97.86	0.18	97.76	0.15
Extra Trees	95.93	919.335	95.92	800.34	95.23	107.09	96	691.54	95.79	708.84
Decision Tree	97.18	38.32	97.23	23.58	97.13	0.24	97.37	4.18	97.25	4.61
Random Forest	96.34	243.98	96.33	110.31	95.48	98.93	96.46	115.97	96.49	106.03
Gradient Boost	97.87	452.47	97.89	150.27	97.33	7.87	97.82	90.96	97.65	61.68
Adaboost	95	4900.23	95.6	5203.23	95.51	5723.23	96.53	8333.05	96.25	1209.63
Bagging	97.55	900.23	97.2	1021.43	97.1	73.52	97.55	1722.61	97.49	883.66

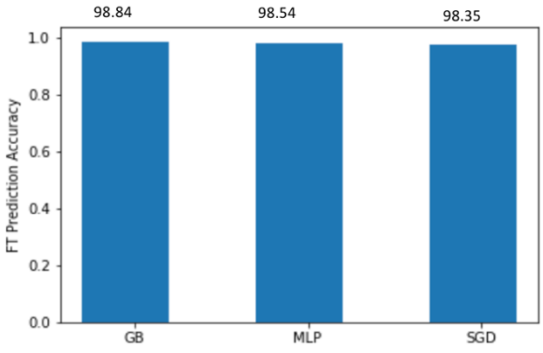
Canvas group: 12

Considering the trade-off between F1 scores and Computation time, top 3 classifiers are: Gradient Boost, MLP and SGD. Below screenshot shows the F1 scores for the combinations of top 3 algorithms without Dimensionality reduction. Also, we can observe the variation in accuracy when down sampling was applied to reduce imbalance in data. Downsampling is performed on the following two sets:

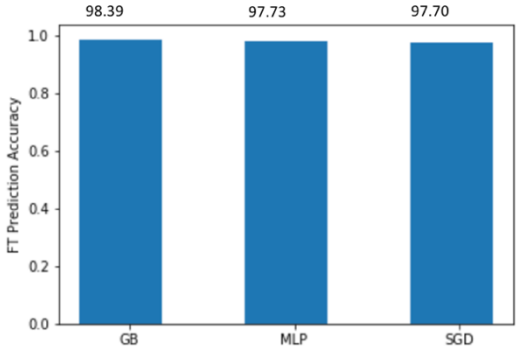
- Certified: 30.3k, Denied: 30.3k
- Certified: 90k, Denied: 30.3k

	No DR & Downsampling	Downsampling	
Sample Count		**65k rows**	**120k rows**
Gradient Boost	0.97	0.72	0.8
MLP	0.97	0.62	0.77
SGD	0.98	0.62	0.77

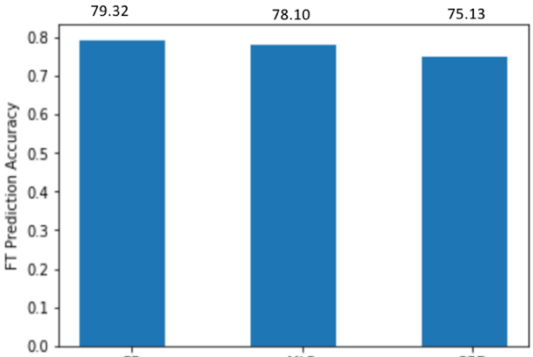
From the above statistics, 3 different approaches were formulated. Also, cross validation and Tuning were implemented in each approach. Final F1-scores(post-tuning) are represented in the bar graphs below.



1. Classifiers with Dimensionality Reduction



2. Classifiers with Feature Selection



3. Classifiers without Dimensionality Reduction and with Down sampling

Result Analysis:

From the graphs above, it can be concluded that Gradient Boost classifier when applied with Dimensionality Reduction (PCA) results in highest F1-score of 98.84%. Downsampling has reduced the F1-score as per our observation. Also, the top 5 features influencing the prediction are derived. (ACCEPTED_RATE; INCREMENT_FROM_PREV_WAGE; DURATION; INCREMENT; APP_COUNT)

Canvas group: 12

CONCLUSION

Difficulties faced:

1. Since the dataset had many categorical attributes, computations for them were difficult. After multiple conversions and feature extractions, the data was made in a feed-ready form for the classifiers.
2. Attribute: EMPLOYER_NAME had around 64k unique values and creating dummies was leading to the curse of dimensionality. Various methods like converting to sparse, feature hashing, DictVectorizer were implemented on trial & error basis but all of it was eventually leading to memory error. Ultimately two new columns were created using feature extraction and the problem was solved.
3. Also, being new to the HPC (Unix) system, it took time in figuring out how to upload and run the program using a bash script.

Things that did not go well:

1. The time allotted per batch was 24hrs and the program timed out without executing completely multiple times during trials. As a solution to this, the algorithms were applied on only a subset of data.
2. For the algorithm combinations implemented, the following algorithms took a lot of computation time:

Classification	Dimensionality Reduction
Bagging	Random Projection
Adaboost	LLE

Things that worked:

Once the data was in feed-ready form for the algorithms, the accuracy obtained was very high, indicating good prediction. The accuracy range was around 97-99% for the 47 combinations that we tried.

Conclusion:

The curiosity for understanding the complex process of H-1B visa petitions is what lead to taking up this project with a challenging dataset. Despite the difficulties faced, various solutions were executed which helped learn important concepts of dealing with large datasets and multiple categorical attributes. The prediction results demonstrate the effective implementation of pre-processing and classification algorithms. Hence, by using the model generated, the status of H-1B visa petitions can be predicted with an accuracy of 98%.

Canvas group: 12

PROJECT PLAN (TASK DISTRIBUTION)

The task was divided into two main parts:

Part I: The preprocessing was jointly done, so that the final results of accuracy can be compared. Each person then implemented combination of dimensionality reduction algorithms with all nine classifiers mentioned in earlier section. The distribution was as follows:

Student name: SJSU ID:	Vineela Velicheti 012445186	Suhani Vaishnav 012489191	Keerthi Veerabhadraiah 012466116
Preprocessing	Jointly done	Jointly done	Jointly done
Dimensionality Reduction	Variance Threshold	SVD	PCA
	Select K-best	Random Projection	Fisher's LDA
Classifiers	KNN	KNN	KNN
	SGD	SGD	SGD
	MLP	MLP	MLP
	Adaboost	Adaboost	Adaboost
	Gradient Boost	Gradient Boost	Gradient Boost
	Decision trees	Decision trees	Decision trees
	Extra trees	Extra trees	Extra trees
	Bagging	Bagging	Bagging
	Random Forest	Random Forest	Random Forest

Part II: Considering the trade-off between computation times and f1-score, three classifiers were selected and further cross validation and tuning were applied. Three different approaches were followed to understand the variation in accuracy score.

Student name: SJSU ID:		Classifier	
Keerthi Veerabhadraiah 012466116	Dimensionality reduction (PCA)	SGD Gradient Boost MLP	Cross Validation & Tuning
Suhani Vaishnav 012489191	Feature Selection (Variance threshold)	SGD Gradient Boost MLP	Cross Validation & Tuning
Vineela Velicheti 012445186	No dimensionality reduction Instead, Downsampling	SGD Gradient Boost MLP	Cross Validation & Tuning