Assignment 2 on Microsoft Malware Detection Team : The Hustlers3

Abhinav Tiwari
MT2020027
International Institute Of Information Technology
Bangalore
Abhinav.Tiwari@iiitb.org

Sandeep Kumar Jha
MT2020087
International Institute Of Information Technology
Bangalore
Sandeep.Jha@iiitb.org

Shanu Kumar
MT2020030
International Institute Of Information Technology
Bangalore
Shanu.Kumar@iiitb.org

***Abstract*—This is a report on our work on Microsoft Malware Detection in which we have to classify malware will be detected or not.**

***Index Terms*—Feature Engineering,Undersampling ,Oversampling,Grid Search,Logistic Regression,XGBOOST,LGBM**

I. Problem Statement

- The malware industry continues to be a well-organized, well-funded market dedicated to evading traditional security measures. Once a computer is infected by malware, criminals can hurt consumers and enterprises in many ways.
- This competition's dataset provides lakhs of System related data from microsoft that affect the presence or absence of malware.We have to predicted whether malware will be detected or not i.e 0 or 1 mathematically in **HasDetection** column.

II. DATASET

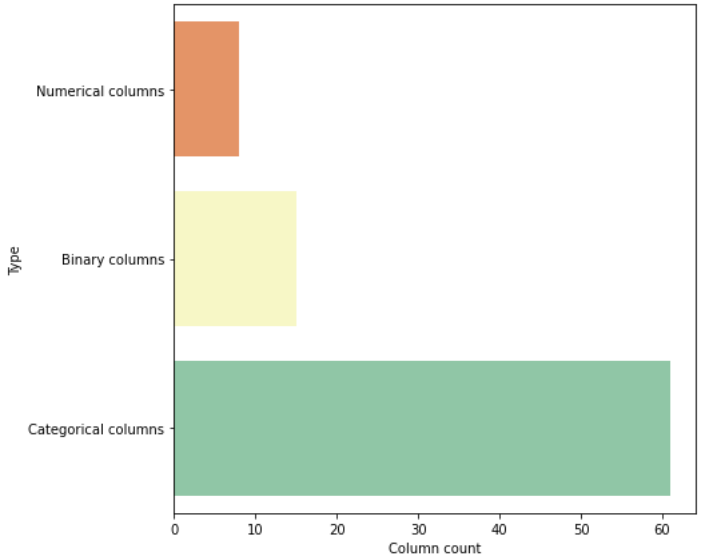
This dataset contains 567730 rows and 84 columns in training data provided, and 243313 rows of data in Test dataset. There is Machine Identifier which uniquely identifies each row in train and test data, and there is **HasDetection column** which contains values in either 0s or 1s through which we will train our ML model. Also we observed the data quite imbalanced in approx 6:1 ratio(for 0 and 1) in case of HasDetection column.

III. APPROACH TAKEN

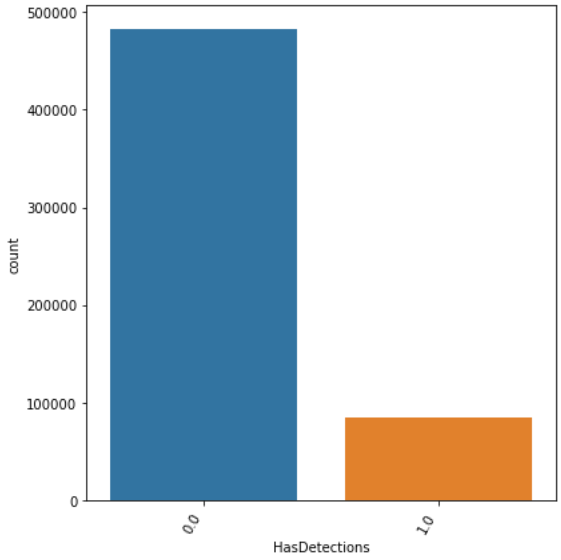
- We saw that the dataset is quite imbalanced so we performed sampling techniques undersampling gave better results than oversampling ultimately so we used that. determined categorical and numerical features, then we determined the columns which contains huge amount of null values or Nans and dropped them if they are more than a minimum threshold and other columns which has somewhat lesser amount of null values have those slot replaced with a unique value.As the features were quite technical we performed preprocessings in the ones we could comprehend.
- After this we trained our model using different algorithms and studied their AUC scores to get the ideal model on which we have to perform Grid search techniques for model tuning to get better hyperparameters and then finally predicted Results with that tuned model.

IV. DATA ANALYSIS and VISUALIZATIONS

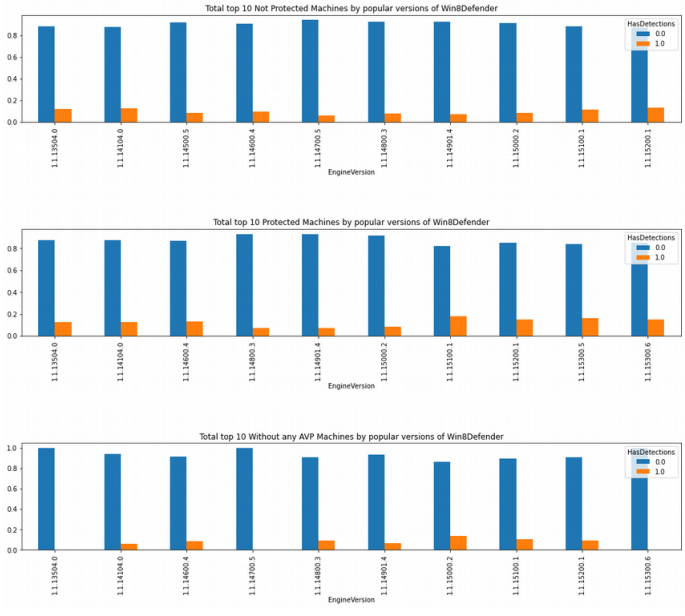
We observed number of different types of columns. Categorical features : 39 non categorical features : 45



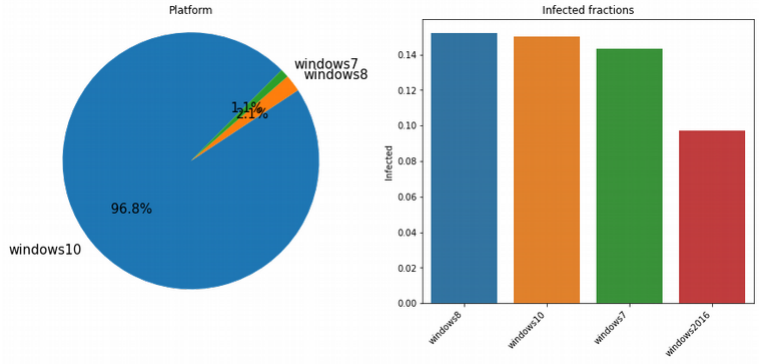
We observed the dataset given is imbalanced for ex in HasDetection case in the ratio approximately 6:1



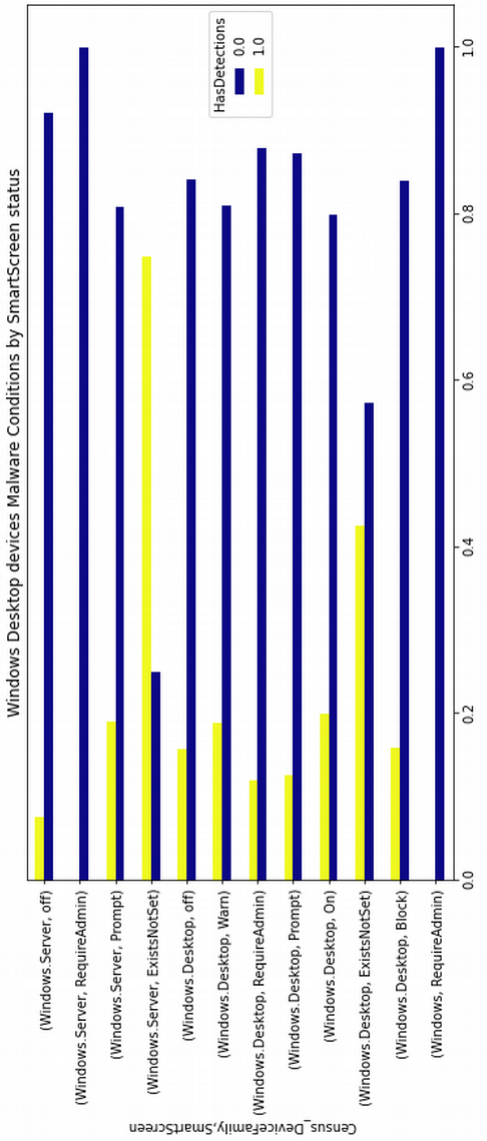
We can see some more imbalances when we were visualising behaviour of Engine Version of Win8Defender and other Antivirus products with occurence of Malwares



We have performed many more data visualisations like we have observed histogram plots of various numerical features to observe skewness and we found that 3 columns which are highly skewed namely **'IsBeta', 'CensusOSBuildRevision', 'CensusTotalPhysicalRAM'** But after dropping them we didn't experience better outcome at last so kept them. We visualised some features how many system with that feature are infected with malware by plotting a batch of pie and bar graph like this (Windows Platform):



We observed some feature like 'SmartScreen' and its relation with Malware infection.



We did some more EDAs redundantly on different features.

V. DATA PREPROCESSINGS INCLUDING FEATURE ENGINEERING

- We observed that there are columns which contains many null values.So we removed the columns having more than 70 percent null values which are: **Unnamed:83,**

PuaMode, CensusProcessorClass, DefaultBrowsersIdentifier, CensusIsFlightingInternal, CensusInternalBatteryType, CensusThresholdOptIn, Census IsWIMBootEnabled.

- We performed Imputations on our data which still contains null values. We imputed numerical features with mean of values of that column but ultimately got bad final results with it so later replaced it with value 0 and results got improved.
- Categorical features are imputed with new value 'other' but we got better result when we imputed the null values with '0'.
- We then performed encoding techniques since ML algorithms doesnt understand object data-type.We performed Label Encoding because in case of One Hot Encoding there were some columns which had more almost 1000 distinct values so it would make our dataset huge and sparse.
- As we observed earlier our data is imbalanced so we studied about sampling techniques to make it almost balanced.There are basically 2 types of sampling techniques 1) Oversampling and 2) Undersampling
- We at first performed Oversampling since it is generally preferred on less amount of data (her less than 1 million) then we performed Undersampling and compared ultimate results and found out that Undersampling gave us better results Hence we stayed with Undersampling technique to make our dataset balanced.
- While performing EDA we observed there were very only few continous features so performing outlier detction and removal didn't get us any signigicant output rather degraded our results.We also studied correlation matrix but didnt get any significant advancement in that either.

VI. MODEL SELECTION :

After preprocessings and feature engineering our dataset size got somewhat reduced so it became easy for Machine learning models to converge.

We trained our model on following algorithms :

- Logistic Regression
- XGBOOST
- Light Gradient Boosting Model (LGBM)

We trained our model using above algorithm in specified order and compared their AUC scores in test data and validation data after submitting our results. We checked AUC scores after some changes in preprocessings also and inculcated the changes only if they improved results. After getting predicted HasDetection output as ypred in the from of probabilities we have to convert it to either 0 or 1 class . We did that by taking 0.5 as threshold probability below which will be classified as 0 and above that will be classified as 1.

Following are the AUC scores we got from different models

Sno.	Algorithm	Validation Data	Test Data
1	Logistic Regression	0.55380	0.48767
2	XGBOOST	0.65033	0.61623
3	LGBM	0.63089	0.50855

The ultimate score on leaderboard may be different from above mentioned scores.

VII. CONCLUSION

From the previous table we found that XGBOOST gave us better AUC scores compared to others.LGBM although a better algorithm than XGBOOST didn't gave better result than the latter. Hence we move forward to tuning XGBOOST model.We implemented Grid search technique to decide on the best hyperparameters from a given set of parameter values. **The Hyperparameters selected for XGBOOST were : learningrate=0.1, n estimators=20000, max depth=8, min child weight=10, gamma=0.3, subsample=1, colsample bytree=0.3, scale pos weight=1, reg alpha = 0.6, reg lambda = 3**. We saw our scores gradually improving while we tuned our model. We ultimately got **around 0.62** AUC score on competition's test data which decided our position on the leaderboard.

VIII. ACKNOWLEDGEMENT

We would like to thank our teaching assistants for helping us out whenever we felt stucked or cannot understand errors or requirement. Special thanks to Raghavan sir, His highly detailed lectures on various topics helped us understand what we were actually doing, rather than just copy pasting code from tutorial sites.

REFERENCES

- https://towardsdatascience.com/
- https://www.analyticsvidhya.com/
- https://elitedatascience.com/
- '*Learning from class-imbalanced data: Review of methods and applications*',Paper by Guo Haixiang Yijing Li Jennifer Shang
- *Ensemble methods in machine learning* by TG Dietterich

and many more