Malware Detection
Team : OptiMen

Vaibhav Tandon
MT2020150
IIIT Bangalore
vaibhav.tandon@iiitb.org

Pawan Kumar Gupta
MT2020097
IIIT Bangalore
Pawan.Gupta@iiitb.org

Himanshu Singh
MT2020143
IIIT Bangalore
Himanshu.Singh@iiitb.org

***Abstract*—The given Machine Learning problem set comes under classification whose goal is to predict a Windows machine's probability of getting infected by various families of malware, based on different properties of that machine.**

***Index Terms*—Feature Engineering ,Cross Validation, Logistic Regression, LightGBM, Random Forest Classifier, SMOTE,**

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.

The goal of this assignment is to predict a Windows machine's probability of getting infected by various families of malware, based on different properties of that machine.

Dataset

The telemetry data containing these properties and the machine infections was generated by combining heartbeat and threat reports collected by Microsoft's endpoint protection solution, Windows Defender.

While the dataset provided here has been roughly split by time, the complications and sampling requirements mentioned above may mean we may see imperfect agreement between your cross validation, public, and private scores. Additionally, this dataset is not representative of Microsoft customers' machines in the wild; it has been sampled to include a much larger proportion of malware machines.

There are 83 columns in the dataset, amongst which some are marked with "NA" Each row in this dataset corresponds to a machine, uniquely identified by a MachineIdentifier. HasDetections is the ground truth and indicates that Malware was detected on the machine.

I. Introduction

With the advancement in technology, there has been a continous increase in the number of malware attacks. Despite numerous anti-malware measures, cybercriminals and hackers continue their misdeeds. Global malware detections for both buisnesses and consumer has increased year by year.

Given a set of features of a windows machine, we are to predict whether that particular machine will get infected or not. The model we'll generate will predict based on the engineered features the same.

II. Data Visualization

The dataset consists of numerical and object (categorical) type of features. The numerical data is of type either int64 or float64. There seem to be a high amount of missing values amongst the features of the dataset (upto 99% missing values).There is a high amount of correlation (0.6 and greater) amongst the numerical features. The object data types seem to have enormously large number of categories. Now on visualizing the training label it is clear that the dataset is highly unbalanced.

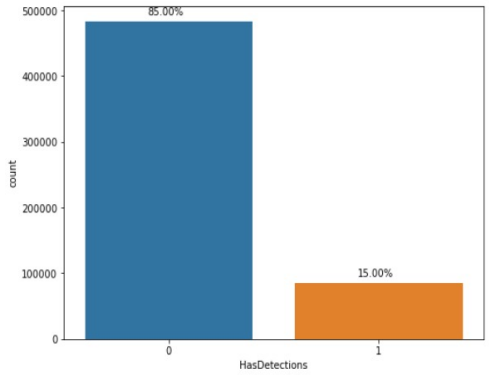


Fig. 1. Target Analysis

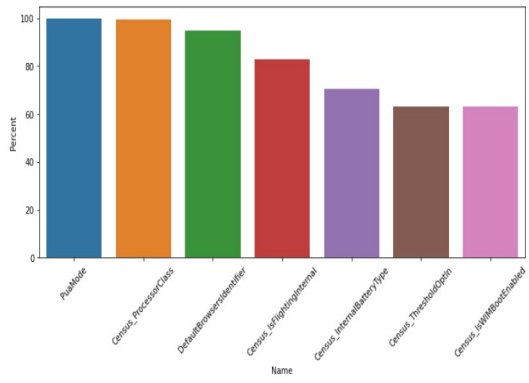


Fig. 2. Missing Value Analysis

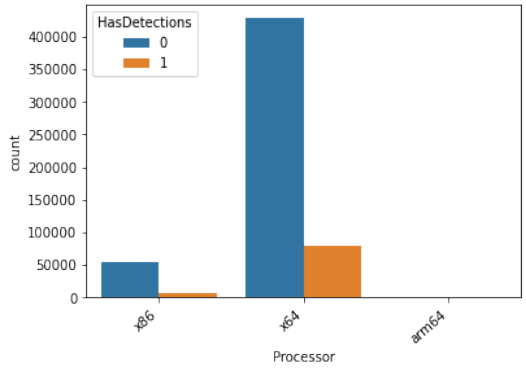


Fig. 3. Processor Count Plot

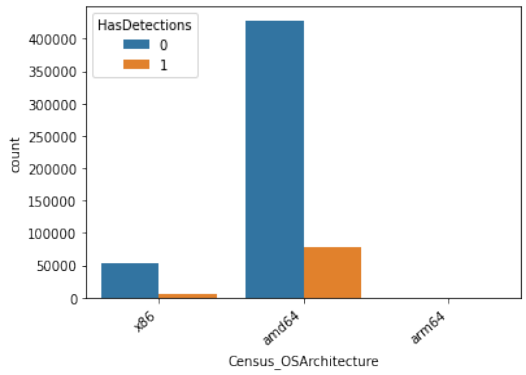


Fig. 4. Census_OSArchitecture Count Plot

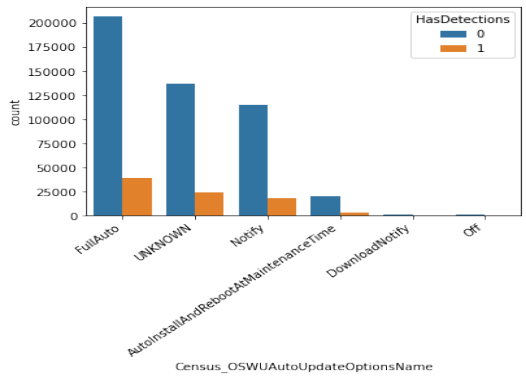


Fig. 5. Census_OSInstallTypeName Count Plot

III. Data Preprocessing

Data preprocessing is a crucial step that helps in enhancing the quality of data to promote the extraction of meaningful insights from the data. Basically it is a process of preparing (cleaning and organizing) the raw data into an understandable and readable format to make it suitable for a building and training Machine Learning models. Models trained on meaningful data always have a upper hand to those trained on raw/less meaningful data.

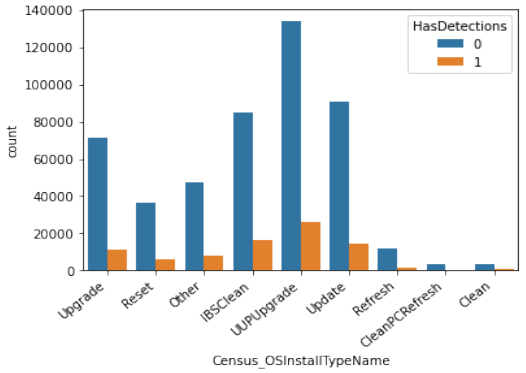
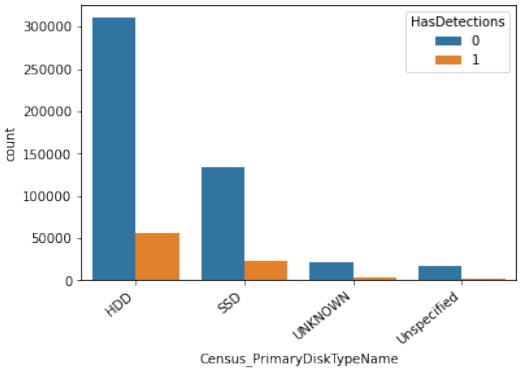


Fig. 6. Census_PrimaryDiskTypeName Count Plot



]

Fig. 7. Census_OSWUAutoUpdateOptionsName Count Plot

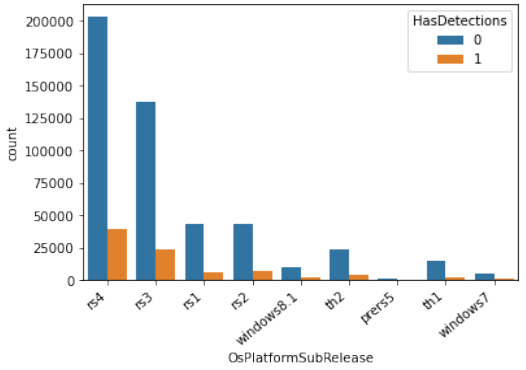


Fig. 8. OsPlatformSubRelease Count Plot

A. Handling Missing Data

We are removing the features having more than 40% of missing data since imputing such columns will have undefined behavior on the target variable.

B. Feature Selection

For numeric data we are removing highly correlated columns i,e having Pearson correlation value greater than 0.6 so as to avoid multidimentionality in the dataset.

For categorical data we are removing features having large number of categories since as the number of categories in-

creases, the complexity of the model increases which will unnecessarily lead to large amount of time in predicting the results which may not be accurate.

C. Encoding

Most of the categorical data is of string data type. So we used Label Encoder to to convert the same into numerical ones so that these columns can be mathematically trained by the model.

D. Imputation

To handle missing data amongst the numeric features we employ mean strategy of imputation. For categorical features Label Encoder automatically assigns a numeric category to missing (NaN) values.

E. Oversampling

Since our target variable is highly unbalanced so we employ SMOTE oversampling technique to equate the number of binary classes.

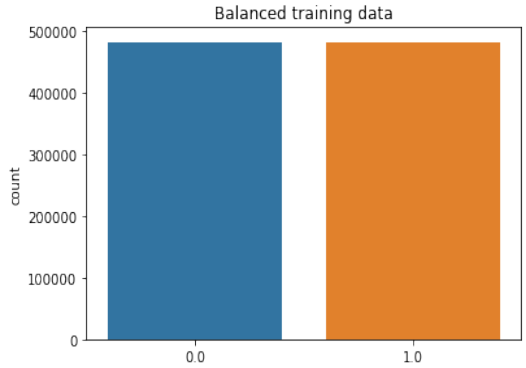


Fig. 9. Target Analysis after oversampling

IV. Training the model

After feature engineering, our number of useful features reduced to 45 We trained our data on the following algorithms:

- Logistic Regression
- RandomForest
- XGBoost
- lightGBM

S.No.	Algorithm	With SMOTE	Without SMOTE
1	Logistic Regression	0.52794	0.52753
2	RandomForest	0.57621	0.62955
3	XGBoost	0.61867	0.65769
4	lightGBM	0.58668	0.62898

TABLE I
ROC_AUC SCORES

V. Conclusion

With the roc_auc scores score of 0.65769, XGBoost model (without SMOTE) best predicted whether the malware was detected by the system or not.

References

[1] https://medium.com/@taplapinger/tuning-a-random-forest-classifier-1b252d1dde92
[2] https://machinelearningmastery.com/gradient-boosting-with-scikit-learn-xgboost-lightgbm-and-catboost/
[3] https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc
[4] https://www.kaggle.com/stuarthallows/using-xgboost-with-scikit-learn
[5] https://www.analyticsvidhya.com/blog/2017/03/imbalanced-data-classification/