Cyber Security

Using Predictive and Descriptive Models in Social Engineering Attacks

Contents

Executive Summary 2

The Situation Room 2

Dataset Description 4

Data Preprocessing 5

Summary Stats and Descriptive Analysis 6

Intended Algorithms and Rational 8

Model Results and Evaluation 10

Conclusions 11

Resources 13

Appendix 14

Executive Summary

Cyber Security is an ever-changing landscape, filled with malicious and stealthy methods of attacking people, process, and technology. The duty of protecting a nations information systems and assets remain a top concern and priority around the world. This situation becomes apparent when you consider the extent of data breaches that occurred in 2017 alone. Often, breaches such as Equifax, Yahoo, and NSA leaks have become a part of the daily news, with little to no end in sight. One rapidly growing defense strategy for this phenomenon is data mining and machine learning. These concepts and practices are considered the next wave of defense for things such as insider threat, user behavior analytics, social engineering, firewalls and more. This project breaks down some of the fundamental practices of data mining and machine learning in effort to predict social engineering attacks from succeeding. Both descriptive and predictive models were utilized to eventually achieve a 94% accuracy, 98% precision, and 89% f-score; to name a few. Moreover, some concerns were found in the 19% False negative rate. This means that the most promising predictive model inaccurately predicted non-malicious when it was malicious. For neural networks false negative rate was the lowest at 17%. In Cyber Security this is always a major concern, since companies are only as strong as their weakest link, missing only 1 attack can cause permeant damage. Additional findings proved that most received and reported emails fall between 13:00 (afternoon) to 17:00 (evening) hours, while most processing times occurred between morning and afternoon, arming leadership with the details they requested for scheduling staff. Missing data and the correlation of missing data is of slight concern for key attribute “From”. This is a concern if the business is looking to improve auditability of the tool. In addition, major key drivers for predicting malicious and non-malicious class labels were “Recipes”, “From.Domain”, “Contains.URL”, “Score”, and “URLs”. This project provides the insight on how predictive analytics can provide meaningful and actionable intelligence in a real-world application.

The Situation Room

Data breaches are complex situations which often involve a combination of factors, one of which is a human factor known as Social Engineering. Phishing, the most common social engineering technique used today is often used as a foothold to launch a variety of attacks against a victim. According to Verizon’s 2017 Data Breach Report, 95% of phishing attacks that led to a breach were followed by some sort of software installed. Verizon also found that of their sample size, 1 in 14 users where tricked into following an attachment or link within a phishing email (Verizon, 2017). The Topic of phishing is still a lingering issue today. Companies often deploy large 24/7 incident response teams to handling these complex attacks. People leaders are then left to determine low and peak time staffing schedules, not to mention many teams experience incident fatigue which can lead to mis-diagnosis and/or job dissatisfaction. This project will aim to identify areas where descriptive and predictive analytics can reduce this side effect. Table 1 below list all business user stories/requirements this project will answer.

Table 1. Business User Stories

#	User Story	Technique/Method/Algorithm	Model
1	Management wants to know the summary description of all reported phishes with category of 2 and 4 since tool was implemented.	Measures of Central Tendencies / Measures of Dispersion (preprocessing)	Descriptive
2	Users don't want any missing data. Identify any missing data and or gaps in this tools process.	Data Preprocessing	Descriptive
3	Management wants to know if there any correlations in the missing data	Data Preprocessing	Descriptive
4	Management needs to know what time of day most phishing attacks occur. (i.e Morning, Afternoon, Night)	Measures of Central Tendencies / Measures of Dispersion (preprocessing)	Descriptive
5	Management wants to accurately Predict email category given N variables	Naïve Bayes	Predictive
6	Management wants to accurately Predict email category given N variables with another predictive model	Decision Trees (ID3/C4.5)	Predictive
7	Management wants to run similar scenarios in an alternative model (if applicable)	Neural Networks	Predictive
8	Management wants to know What are some of the main drivers for accurately predicting email category?	Decision Trees (ID3/C4.5)	Predictive
9	Managements want to compare the accuracy of each model for best fit	Model Evaluation	Descriptive

Dataset Description

THE COFENSE TRIAGE DATASET

The Cofense Triage data is a part of a suite of tools that enables companies to recognize, report and analyze malicious phishing emails. The data acquired for this project is a subset of email data that has already been categorized by professionals as either Non-Malicious (category 2) or Malicious-crimeware (category 4). The dataset consists of nearly 100k observations from the last 10 years (2008-2018). The triage tool is built with a set of rich features that help to enrich normal email content and metadata (i.e. from, to, subject, received). In total there are 14 original attributes from the initial export which includes features such as reporter, category, tags, recipes, URL's, Attachments, and more. Immediately, this is considered a multivariate scenario. The class label was called "Category" which contained two levels of values: "2" indicates non-malicious, and "4" indicates malicious-crimeware. Table 2 below provides a hybrid of the data dictionary for all original variables.

Table 2. Data Dictionary (Original)

#	Attribute Name	Description	Dependency	Scale
1	Attachments	Number of attachments in the email	IV	Ratio
2	Category	The Analysts final fidelity score of the reported phish (i.e 2, 4)	DV	Nominal (BA)
3	From	The original sender	IV	Nominal
4	Processed	Time the email was processed by an analyst (in ISO format)	IV	Ratio
5	Received	Time the end user received the email (in ISO format)	IV	Ratio
6	Recipes.Analyst.Matched	Name(s) of predefined rule that match against certain email contents. Or analysts nickname	IV	Nominal
7	Reported	Time the end user reported or submitted the email as phishing (in ISO format)	IV	Ratio
8	Reporter.Email	The Reporters email address	IV	Nominal
9	ReportID	Unique ID for each event	IV	Nominal
10	Score	Reputation score of the reporter	IV	Ratio
11	Subject	Subject of email	IV	Nominal
12	Tags	Additional custom descriptors about the email and/or matched recipes/rule names	IV	Nominal
13	URLs	Number of links in the email	IV	Ratio
14	VIP	Very Important Person. Indicates if Reporter is a high profile target	IV	Nominal (BA)

** IV = Independent Variable, DV = Dependent Variable or Class Label*
** BS = Binary Symmetric, BA = Binary Asymmetric*

Data Preprocessing

Initial dataset observations proved to be a worthy candidate for applying predictive and/or descriptive models, although it would need additional transformations before it can be ready for any machine learning models.

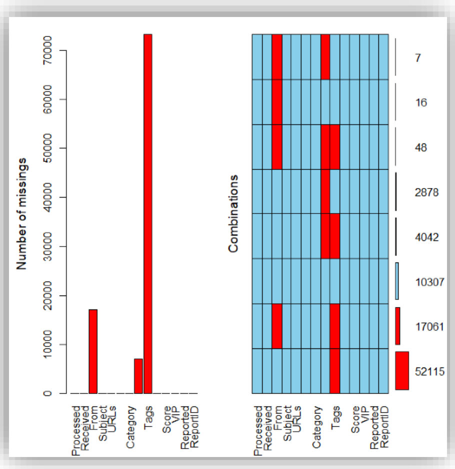
Data Cleansing

Upon initial import of the data, it was observed to have missing values and data conformity issues. For example, missing values consisted of both empty strings and single space characters. This was later resolved by performing a sequence of data read and write passes to get the data to properly identify itself with “NA” values using R’s native na.strings parameter. NA values were needed for user stories 2 and 3. In addition, data conformity for quoting text and recognizing numbers were mismatched, which was expected given the open range of allowed text within the Subject, Recipe, and Tag variables. This was fixed using the built-in regex within R. Regex was used heavily throughout data preprocessing.

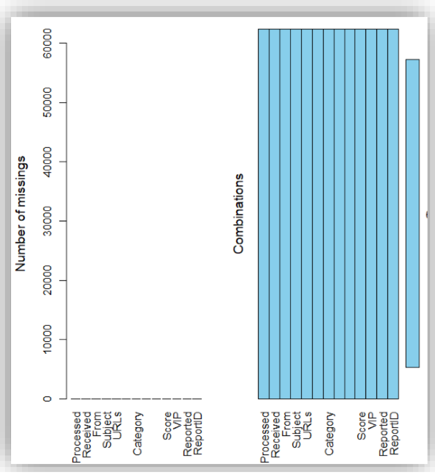
Data Reduction

Next steps aimed to eliminate some irrelevant features of this dataset. By using principle component analysis along with subject matter expertise, I was able to hypothesize which attributes could be dismissed and/or which attributes would have more impact on the potential outcome. In later sections, this hypothesis would soon be tested pragmatically using the ID3 algorithm. The variable “Tags” was eliminated from this dataset since this column was missing 84.72% of data and was not widely adopted in day to day operations. The remaining missing values came from variables “From” and “Recipes” which only amounted to 25.87% of the total data. This was a difficult decision but left nearly 70K complete observations for model processing. All other missing data points did not contain any correlation amongst each other. The highest missing data correlation observed was less than 25% which was deemed to have no high correlation. See Figure 1 set below for missing data visualization.

Figure 1. Missing Data (before)



Missing Data (after)



User Story 2&3 Stats

Attribute/Feature Construction, Discretization, and Transformation

Initially, the original dataset started off with 14 variables, all of which were a mix between nominal and ratio scale data. It was determined early on, that feature construction was needed to properly answer user stories 4-8. A combination of regex, custom functions, and multiple 3rd party libraries were used for preprocessing. For example, feature construct used "Received" to construct "Received.TOD", and discretization used "Reported" to obtain "Reported.Time.hh".

Attribute/Feature Selection

The original 14 variables grew to an enhanced view of 30 total variables comprised of the following additional variables in table 3 below, note: the table does not include the original 14 variables.

Table 3. Attribute/Feature Construct

Attribute Name	Description	Dependency	Scale
Received.Time.hhmm	*Derived - Received Time in format hh:mm*	*IV*	*Ratio*
Received.Time.hh	*Derived - Received Time in format hh*	*IV*	*Ratio*
Received.TOD	*Derived - Time of Day email was received*	*IV*	*Nominal*
Reported.Time	*Derived - Time the email was reported from the end user in format YY:MM:SS*	*IV*	*Ratio*
Reported.Time.hhmm	*Derived - Reported Time in format hh:mm*	*IV*	*Ratio*
Reported.Time.hh	*Derived - Reported Time in format hh*	*IV*	*Ratio*
Reported.TOD	*Derived - Time of Day email was reported as a phish*	*IV*	*Nominal*
Processed.Time	*Derived - Time the email was processed by an analyst in YY:MM:SS*	*IV*	*Ratio*
Processed.Time.hhmm	*Derived - Processed Time in format hh:mm*	*IV*	*Ratio*
Processed.Time.hh	*Derived - Processed Time in format hh*	*IV*	*Ratio*
Processed.TOD	*Derived - Time of Day analysts confirm the fidelity of the email*	*IV*	*Nominal*
Contains.Url	*Derived - Indicates if email contains any URLs*	*IV*	*Nominal (BS)*
Contains.Attachment	*Derived - Indicates if email contains any attachments*	*IV*	*Nominal (BS)*
Subject.Length	*Derived - Character space length of the Subject title*	*IV*	*Ratio*
From.Domain	*Derived - Email domain from original sender's email address*	*IV*	*Nominal*

** IV = Independent Variable, DV = Dependent Variable or Class Label*
** BS = Binary Symmetric, BA = Binary Asymmetric*

Overall, data preprocessing was by far the most rigorous and time-consuming throughout the data mining process taking up at least 80% of the total time. Nevertheless, it produced a pristine dataset for the proceeding summary stats, and classification models to come.

Summary Stats and Descriptive Analysis

After preprocessing steps were complete, a more rounded summary was obtained of the newly enriched dataset. The dataset proved to be normally distributed across all targeted data points and did not contain any high correlations for the targeted variables. See table 4 for basic summary statistics.

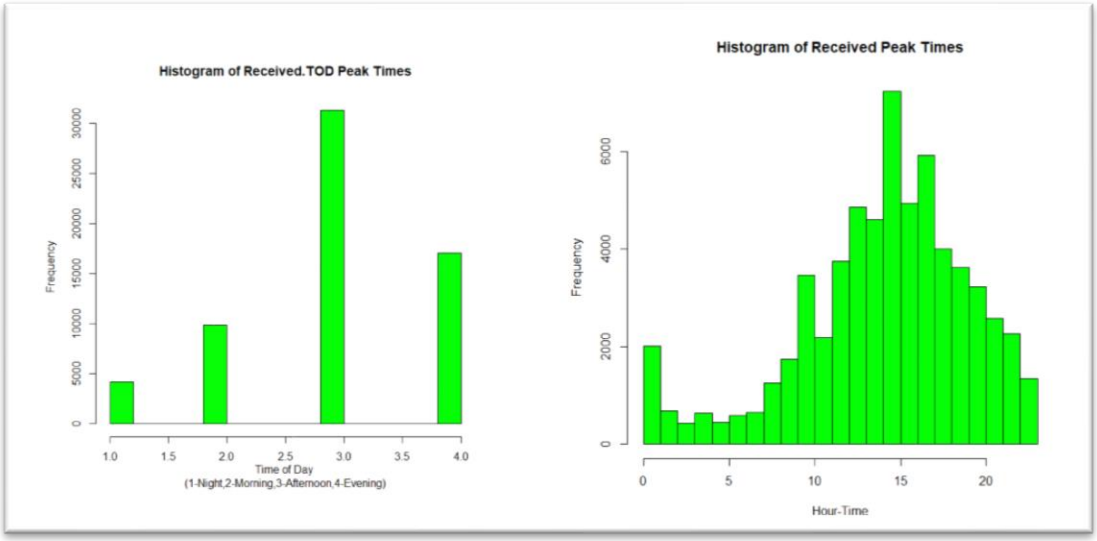
Table 4. Dataset Summary Stats

#	Attribute	Mean	Min	Max	Median	Mode	Sd()
1	*URLs*	7.143	0	1402	2	1	17.84
2	*Attachments*	1.577	0	169	0	0	4.84
3	*Score*	10.26	-615	2503	5	-5	115.06
4	*Processed.Time.hh*	13.05	0	23	13	9	5.13
5	*Reported.Time.hh*	15.74	0	23	16	15	3.68
6	*Received.Time.hh*	14.44	0	23	15	15	5.05
7	*Subject.Length*	37.94	1	521	33	23	24.64
8	*Category*	1.294	1	2	1	2	0.455
9	*VIP*	1.009	1	2	1	1	0.095
10	*Contains.URL*	1.772	1	2	2	2	0.419
11	*Contains.Attachment*	1.31	1	2	1		0.462
12	*Processed.TOD*	2.702	1	4	3	3	0.9
13	*Received.TOD*	2.98	1	4	3	3	0.836
14	*Reported.TOD*	3.218	1	4	3	3	0.606

***User Story 1 Stats*

At this time, it seemed appropriate to answer user story # 4. Determining the time of day most potential attacks occur (Received.TOD attribute) was determined by using a classic histogram plot. This resulted in most occurring in the afternoon, shortly followed by evening and then mornings. More could be derived from this dataset to determine exact hours; thus, further analysis was done to include peak time hours(Received.Time.hh). Peak time hours landed between 13:00 -17:00 hours. This armed management with the granularity they needed to understand when potential attacks occurred most frequently. Figure 2 below shows the plot of frequencies.

Figure 2. Received Peak Time Histogram



It was not enough to just answer these basis questions, so further analysis was conducted for variables “Reported.Time|TOD” and “Processed.Time|TOD” which described when users reported the most phishes to the team and when most processing times occur. It was determined that users reported nearly the same time of day and hours as they had received the email; afternoon and evenings between 13:00 – 17:00 hours. There was a slight difference when most attacks were processed, with most processing times occurring in the morning and afternoon between 09:00-12:00 hours. All in all, this empowered management with the granularity they needed to properly schedule staff.

Intended Algorithms and Rational

This leads to the remaining user stories 5-8, which can be satisfied by using a mix of classification and association rule mining models. The overall goal of these algorithms is to predict the category (dependent variable) of email(s) as non-malicious (Category 2) or malicious-crimeware (Category 4). The intended classification algorithms for this project will be Decision Tree ID3 with C4.5, Naïve Bayes, and Neural Networks. All algorithms will use supervised learning techniques containing proper fitting, training, and testing techniques until a reasonably desirable output is produced. Finally, all models will be evaluated for its performance. Table X below contains a list of variables considered for these algorithms along with their dependency states.

Table 5. Attributes Pool for Models

#	Attribute Name	Description	Dependency	Scale
1	***Category***	*The Analysts final fidelity score of the reported phish (i.e 2, 4)*	*DV*	*Nominal (BA)*
2	***Received.Time.hh***	*Derived - Received Time in format hh*	*IV*	*Ratio*
3	***Received.TOD***	*Derived - Time of Day email was received*	*IV*	*Nominal*
4	***VIP***	*Very Important Person. Indicates if Reporter is a high-profile target*	*IV*	*Nominal (BA)*
5	***Score***	*Reputation score of the reporter*	*IV*	*Ratio*
6	***Recipes.Analyst.Matched***	*Name(s) of predefined rule that match against certain email contents. Or analysts nickname*	*IV*	*Nominal*
7	***Attachments***	*Number of attachments in the email*	*IV*	*Ratio*
8	**Contains.Url**	*Derived - Indicates if email contains any URLs*	*IV*	*Nominal (BS)*
9	***Contains.Attachment***	*Derived - Indicates if email contains any attachments*	*IV*	*Nominal (BS)*
10	***Subject.Length***	*Derived - Character space length of the Subject title*	*IV*	*Ratio*
11	***From.Domain***	*Derived - Email domain from original sender’s email address*	*IV*	*Nominal*

** IV = Independent Variable, DV = Dependent Variable or Class Label*
** BS = Binary Symmetric, BA = Binary Asymmetric*

Decision Tree Classification Model

Decision tree classification model will be the first algorithm to start with since it also contains a built-in function for feature selection via its entropy reduction stage. It would be critical to understand which attributes reduced the most uncertainty and which attributes could influence the dependent variable more. Using R, a training set of 70% (43,695 observations) and test set of 30% (18,727 observations) was created for ID3. The C4.5 algorithm was also implemented prior to testing to ensure any factors that resulted in continuous variables would be handled appropriately. The initial variables used for this model were: "VIP", "Score", "Received.TOD","Recipes.Analyst.Matched", “Subject.Length", "Contains.Attachment", "Contains.Url", "Attachments", "URLs", and "From.Domain". The model produced the following decision tree which ranked variables with the most entropy reduction. Table 6 below provides IG values.

Table 6. Information Gain Table

#	Attribute	Information Gain/Entropy Reduction
1	Recipes.Analyst.Matched	100%
2	From.Domain	24.76%
3	Contains.Url	12.23%
4	Score	12.12%
5	URLs	10.69%
6	VIP	1.82%
7	Attachments	1.81%
8	Contains.Attachment	1.61%
9	Received.TOD	1.13%
10	Subject.Length	1.02%

Naïve Bayesian Classification Model

The second classification model used will be Naïve Bayes. It was determined prior to processing that all intended independent variables contained no correlation with each other. A correlation matrix was produced to identify any correlations. This lead to eliminate variable “Received.Time.hh” as this was highly correlated with “Received.TOD”. Eliminating this variable would not lead to significant changes in the output.

Neural Network Classification Model

The Final classification model used was neural networks. The number of hidden nodes started with 5, which was half the input node count, and later adjustments were made to recognize changes in performance. Supervised learning was performed with Class label “Category” and the following training/testing set variables: “vip + score + received.tod + received.time.hh + recipes + subject + urls + contains.attach + contains.url + attachments”. Before supervised learning and testing could be performed, all variables, including the class label were converted to numeric data. In addition, normalization was done using scale functions in r, and setting the stepmax to “1e6” to allow more iterations for the large dataset size and number of input nodes.

Model Results and Evaluation

Decision Tree

Confusion Matrix

	Predicted	
Actual	Malicious	Non-Malicious
Malicious	4424	1042
Non-Malicious	97	13164

Evaluation Metrics

Accuracy	Precision	Recall	F-Score	Sensitivity (TPR)	Specificity (TNR)	False Positive Rate	False Negative Rate
93.91%	97.85%	80.93%	88.59%	80.93%	99.26%	0.73%	19.06%

Naïve Bayes

Confusion Matrix

	Predicted	
Actual	Malicious	NON-Malicious
Malicious	4143	1323
Non-Malicious	3915	9346

Evaluation Metrics

Accuracy	Precision	Recall	F-Score	Sensitivity (TPR)	Specificity (TNR)	False Positive Rate	False Negative Rate
72.02%	51.41%	75.79%	61.26%	75.79%	70.47%	29.52%	24.20%

Neural Network

Confusion Matrix

	Predicted	
Actual	Malicious	NON-Malicious
Malicious	4547	958
Non-Malicious	2603	10619

Evaluation Metrics

Accuracy	Precision	Recall	F-Score	Sensitivity (TPR)	Specificity (TNR)	False Positive Rate	False Negative Rate
80.98%	63.59%	82.59%	71.86%	82.59%	80.31%	19.68%	17.40%

Conclusions

1. Peak potential attack times were Afternoon and Evening: At beginning I hypothesized mornings to be the most frequent time when potential phishes were sent to the user. This did not hold true when summary statistics were collected. Between 13:00 to 17:00 hours in the afternoon/evenings were the busiest. This would imply that afternoon and evening could potentially needing the most staffing, but that would not be confirmed until determining Report frequencies. Additional insights were soon gathered to confirm if users were reacting fast enough to phishing emails. This satisfies user story 4.

2. Received.Time and Reported.Time were similar: Contrary to popular belief, mornings were not the most frequent received times, but rather afternoon and evenings. This same fact also held true for Reported time. Between 13:00 to 17:00 hours in the afternoon/evenings were the busiest. This further satisfies user story 4.

3. Neural Networks training set needed to be smaller for this project: Initial training was done with the original training size of 70% of the data, which did not suit well for computer resources (Big-O notation). This is often the tradeoff observed when trying to develop higher performing model. Nevertheless, a smaller 5k training set still placed this model in 2nd place in terms of performance metrics. This satisfies user story 7.

4. Information Gain was valuable for systematically finding key drivers: Key drivers according to the ID3 Training method, from descending order were Recipes + From.Domain + Score + Contains.Url + URLs + Attachements + Subject.Length + Contains.Attachments + Received.TOD. This satisfies user story 8.

5. Decision Tree ID3/C4.5 classification model produced the best evaluation metrics: The ID3 classification model produced the highest evaluation score across all metrics. Accuracy was nearly 94% and precision nearly 98%. Some concerns can be found in that the False negative rate of 19%. This means that it inaccurately predicted good when it was bad. In Cyber Security this is a major concern since companies are only as strong as their weakest link, and all it takes is the right attack to penetrate to cause permeant damage. this satisfies user stories 5,6&9.

6. There was nearly 28% of data missing: Majority of missing data came from an under-utilized variable ("Tags") which often occurs in large, feature rich applications. The remaining came from attributes unanticipated, such as "From" and "Recipe" attributes. This feedback and insight could be valuable to the business if business rules relied heavily on it or if audits were a concern. This satisfies user story 2&3.

7. There was little to no correlation in missing data: Aside from datetime variables, each variable did not contain any highly correlated values, which served well for Naïve Bayesian. The highest correlations found were just under 25% correlation. This satisfied user story 2&3.

8. C4.5 was very useful in decision tree classification model: This model converted nominal/continuous data into the proper numerical form which served the entropy reduction stage and subsequent learning/testing stages very well.

8. Typically, the more supervised learning the better predictions it can produced: It was observed that the more supervised learning the better each model performed, granted your supervised learning was not taught wrong. The side affect of this is can be slower training/testing times along with increased resources, also known as Big-O notation.

Resources

Verizon Enterprise (2017) Data Breach Investigation Report Executive Summary. Retrieved from www.verizonenterprise.com/verizon-insights-lab/dbir/2017/.

Appendix

Figure 3. Decision Tree Visual

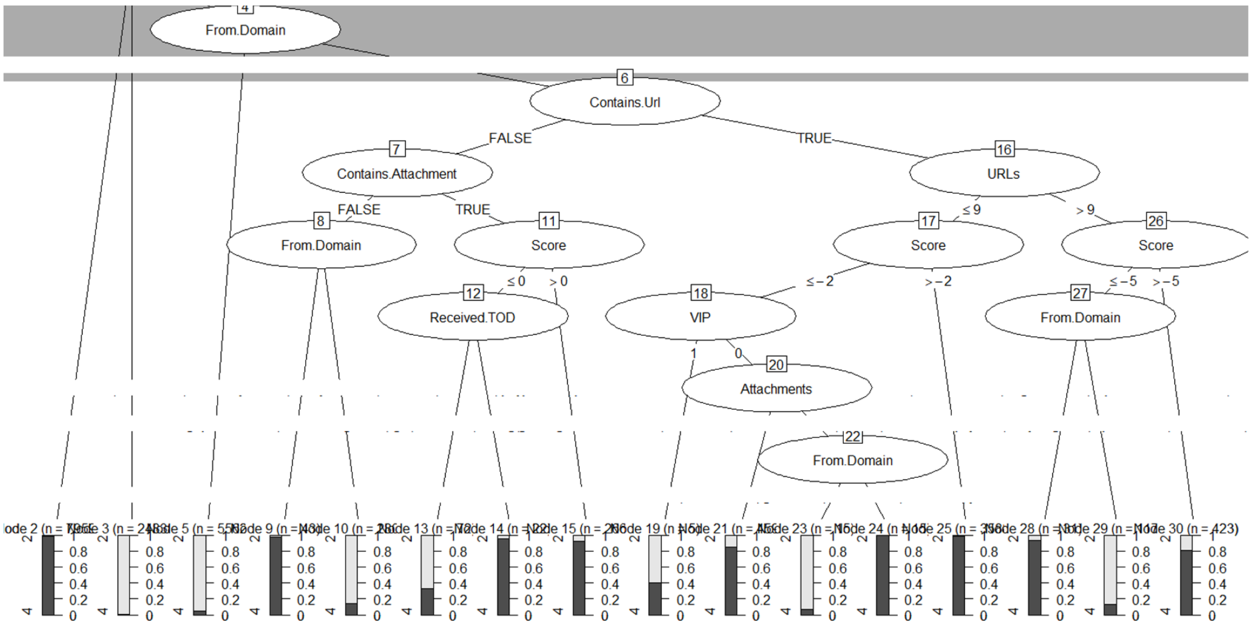
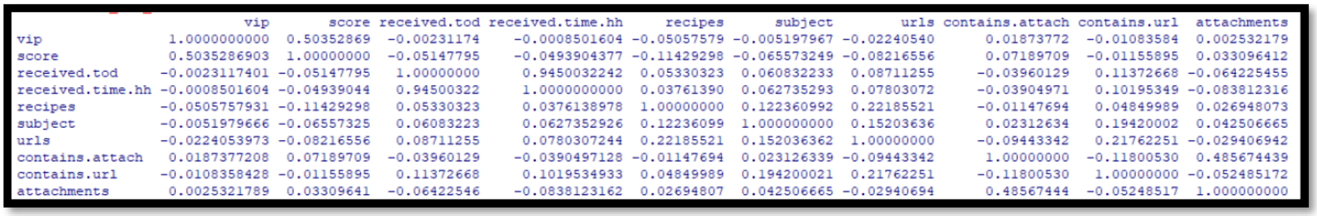
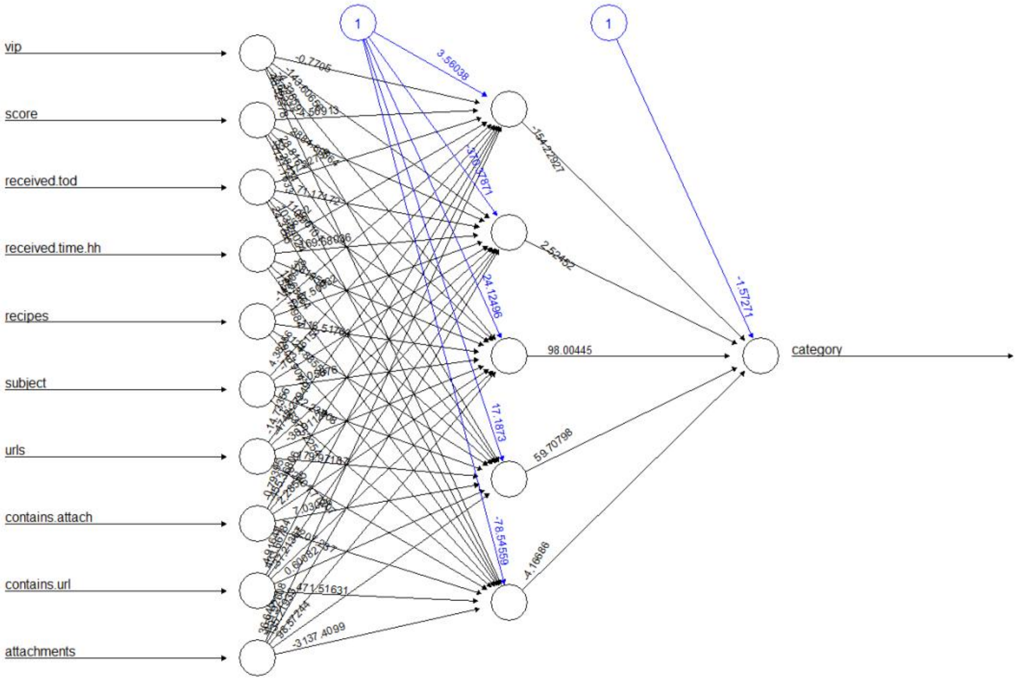


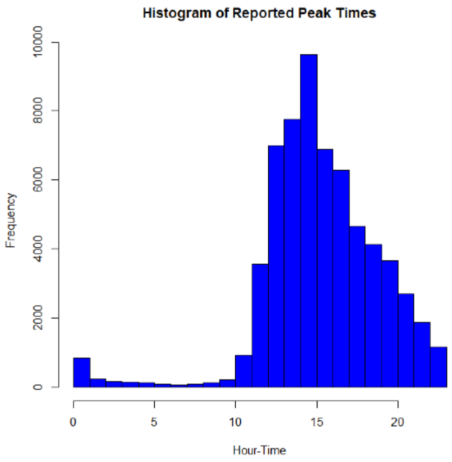
Figure 4. Correlation Matrix of Given Attributes

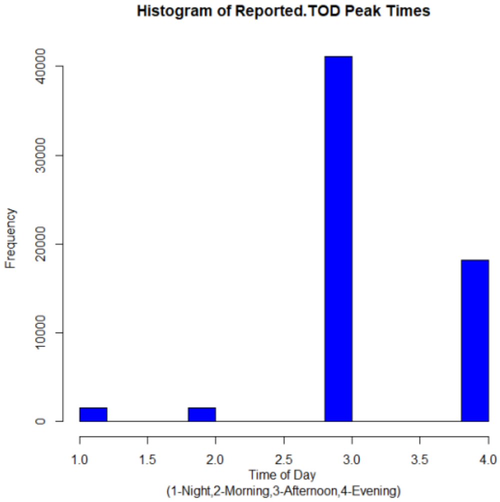


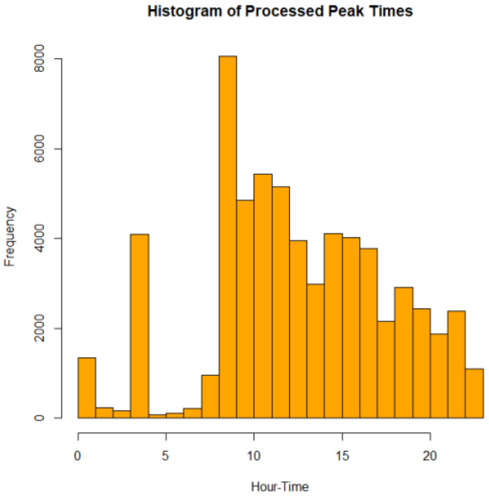
```
                          vip       score received.tod received.time.hh     recipes      subject        urls contains.attach contains.url  attachments
vip                1.0000000000  0.50352869  -0.00231174    -0.0008501604 -0.05057579 -0.005197967 -0.02240540      0.01873772  -0.01083584  0.002532179
score              0.5035286903  1.00000000  -0.05147795    -0.0493904377 -0.11429298 -0.065573249 -0.08216556      0.07189709  -0.01155895  0.033096412
received.tod      -0.0023117401 -0.05147795   1.00000000     0.9450032242  0.05330323  0.060832233  0.08711255     -0.03960129   0.11372668 -0.064225455
received.time.hh -0.0008501604 -0.04939044   0.94500322     1.0000000000  0.03761390  0.062735293  0.07803072     -0.03904971   0.10195349 -0.083812316
recipes           -0.0505757931 -0.11429298   0.05330323     0.0376138978  1.00000000  0.122360992  0.22185521     -0.01147694   0.04849989  0.026948073
subject           -0.0051979666 -0.06557325   0.06083223     0.0627352926  0.12236099  1.000000000  0.15203636      0.02312634   0.19420002  0.042506665
urls              -0.0224053973 -0.08216556   0.08711255     0.0780307244  0.22185521  0.152036362  1.00000000     -0.09443342   0.21762251 -0.029406942
contains.attach    0.0187377208  0.07189709  -0.03960129    -0.0390497128 -0.01147694  0.023126339 -0.09443342      1.00000000  -0.11800530  0.485674439
contains.url      -0.0108358428 -0.01155895   0.11372668     0.1019534933  0.04849989  0.194200021  0.21762251     -0.11800530   1.00000000 -0.052485172
attachments        0.0025321789  0.03309641  -0.06422546    -0.0838123162  0.02694807  0.042506665 -0.02940694      0.48567444  -0.05248517  1.000000000
```

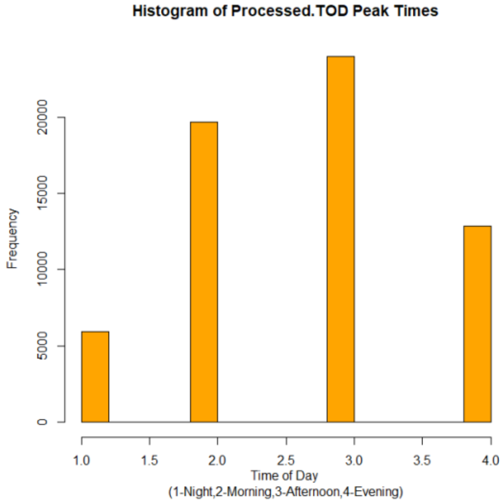
Figure 5. Neural Network Visual

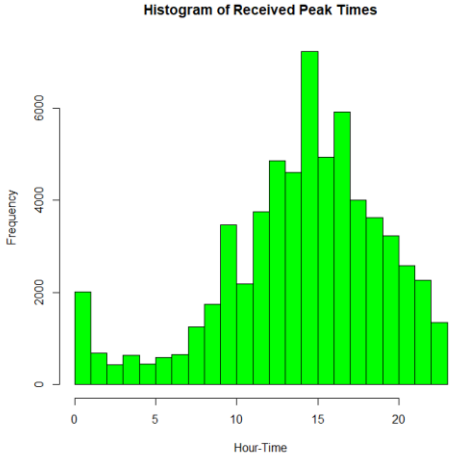


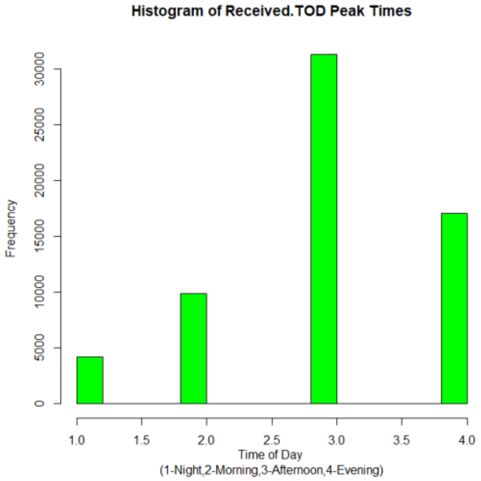

Histogram of Reported Peak Times


Histogram of Reported.TOD Peak Times


Histogram of Processed Peak Times


Histogram of Processed.TOD Peak Times


Histogram of Received Peak Times


Histogram of Received.TOD Peak Times