Team 10 Report

Jacob Mulford, Brendan Schell, Wilson Cheung, Carlos White, Manu Bhardwaj

Introduction And Motivation

Spam calls are a menace that have increased in recent years because advances in Voice over IP and auto dialer technologies have made it cheaper and easier for spammers and telemarketers to run their operations from anywhere in the world. These calls go beyond just playing recorded messages - they can be active phishing campaigns or outright fraud.

Problem Definition

We want to build a visualization of spam calls in order to allow users to better understand attributes of spam calls, such as their location and time of call.

We also want to build a predictive model that, given attributes of an incoming phone call, can identify the type of spam call and present probabilities of each type of potential spam call.

Survey

Proposed solutions

References: [2], [5], [6], [7], [8], [9], [11], [13]

These papers propose solutions to spam detection in VOIP and regular calls through both machine learning ([2], [5], [9], [13]) and rules-based approaches ([6], [7], [8], [11]) with input features such as call content [3], geographical [1], and time data [1].

These papers are mostly useful for determining features that can be used for our model, the relative success of the different approaches, as well as for learning of any assumptions and limitations in the solutions.

The shortcomings of these previous approaches are that most of them use information which is not known ahead of time [3], they are localized to specific countries ([2], [5], [6], [7], [8], [9], [11], [13]), and none of them are resistant to spoofing. There are also many assumptions which may not translate to use in the real world. Our proposed solution will only use information that would be known at the time of call and should therefore be more feasible as a real-world approach.

Suggested solutions

References: [1], [3], [4], [10], [12], [14], [15]

These papers provide many theoretical and applied approaches to aid us in developing a solution for our problem of interest. They dive into methods of preventing [3], combating [14], and identifying spam calls [15] through various machine learning techniques.

These papers enable us to further understand the problem of interest through social networks [12], time series analysis [1] and data visualization [10], and identify which analytical and non-analytical methods of spam detection are most effective [3].

Shortcomings present in these papers include the making of unwanted assumptions in formulating a model [1, 10, 12], the vagueness of the defining metrics in the presented analysis [4], and the measuring of the practicality of results to the targeted problem [3].

Referenced Literature

[1]
M. R. Islam, N. Sultana, M. A. Moni, P. C. Sarkar, and B. Rahman, “A Comprehensive Survey of Time Series Anomaly Detection in Online Social Network Data,” International Journal of Computer Applications, vol. 180, no. 3, pp. 13–22, Dec. 2017.
[2]
H. Li et al., “A Machine Learning Approach To Prevent Malicious Calls Over Telephony Networks,” CoRR, vol. abs/1804.02566, 2018.
[3]
V. M. Quinten, R. Van De Meent, and A. Pras, “Analysis of techniques for protection against spam over internet telephony,” in Meeting of the European Network of Universities and Companies in Information and Communication Engineering, 2007, pp. 70–77.
[4]
I. Orife, S. Walker, and J. Flaks, “Audio Spectrogram Factorization for Classification of Telephony Signals below the Auditory Threshold,” arXiv preprint arXiv:1811.04139, 2018.
[5]

T. H.-D. Huang, C.-M. Yu, and H.-Y. Kao, “Data-Driven and Deep Learning Methodology for Deceptive Advertising and Phone Scams Detection,” arXiv preprint arXiv:1710.05305, 2017.
[6]
H. E. Bordjiba, E. B. Karbab, and M. Debbabi, “Data-driven approach for automatic telephony threat analysis and campaign detection,” Digital Investigation, vol. 24, pp. S131–S141, 2018.
[7]
R. Dantu and P. Kolan, “Detecting Spam in VoIP Networks.,” SRUTI, vol. 5, pp. 5–5, 2005.
[8]
A. S. Kessler and T. Cornwall, “Does Misinformation Demobilize the Electorate? Measuring the Impact of Alleged Robocalls in the 2011 Canadian Election,” CEPR Discussion Paper No. DP8945, SSRN: http://ssrn. com/abstract= 2066318 …, 2013.
[9]
G. Vennila, M. S. K. Manikandan, and M. N. Suresh, “Dynamic voice spammers detection using Hidden Markov Model for Voice over Internet Protocol network,” Computers & Security, vol. 73, pp. 1–16, 2018.
[10]
L. Kharb, “Exploration of Social Networks with Visualization Tools.”
[11]
R. B. Prayaga, E. W. Jeong, E. Feger, H. K. Noble, M. Kmiec, and R. S. Prayaga, “Improving refill adherence in Medicare patients with tailored and interactive mobile text messaging: pilot study,” JMIR mHealth and uHealth, vol. 6, no. 1, 2018.
[12]
R. A. Hanneman and M. Riddle, Introduction to social network methods. Riverside, CA: University of California, Riverside, 2005.
[13]
S. Phithakkitnukoon and R. Dantu, “Predicting calls–new service for an intelligent phone,” in IFIP/IEEE International Conference on Management of Multimedia Networks and Services, 2007, pp. 26–37.
[14]
H. Tu, A. Doupé, Z. Zhao, and G. Ahn, “SoK: Everyone Hates Robocalls: A Survey of Techniques Against Telephone Spam,” in 2016 IEEE Symposium on Security and Privacy (SP), 2016, pp. 320–338.
[15]
M. A. Azad, R. Morla, and K. Salah, “Systems and methods for SPIT detection in VoIP: Survey and future directions,” Computers & Security, vol. 77, pp. 1–20, 2018.

Proposed Method

Intuition

This method should be better than the state of the art for a few reasons. We are building our predictive model from historic reports of spam calls. By using data from several years of reported spam calls, we aim to predict the type of spam call based on the date and time of the call, the phone number making the call, and the location of the recipient of the call. This allows us to answer questions like “Do certain types of spam calls follow an hourly/daily/weekly/monthly pattern?” or “Are certain types of spam calls more likely to come from, or target, a certain area code?”.

For the visualization, we want to build an interface that allows the user to enter details of their call and determine the type of spam call, and a visualization explaining the prediction. This allows users to understand potential patterns in type of spam calls. We also have an interactive map of the US that highlights the areas with most and least amount of spam calls based on the recipient’s location. This can potentially help regulators focus on certain cities or states that have a high degree of spam calls.

Description Of Approaches

Model Development

To tackle the problems of interest outlined in the report, we broke down our modeling development process into two subcategories: spam call classification, and forecasting when the next spam call will be made.

In both subcategories, we need to clean and preprocess the provided data for our modeling purposes. The data contains many features ranging from date and time of incident, phone number(s) involved, method of calling, location of the recipient, and type of spam call.

Spam Call Classification

For the spam call classification task, it is assumed that we have no prior knowledge of the receiving call. Under this assumption, much of this data, such as method of calling, is not relevant for the development of our classification model.

We need to categorize phone numbers and location of recipient. Since there are billions of valid phone numbers, we decided to categorize them by area code (roughly 300 valid area codes in the USA). For location of recipient, we decided to map coordinate points to the closest area code, and categorize the location by these area codes.

For time data, we looked at time of day, day of month, day of week, and month. Day of month and month are already represented as numeric values, so we kept them as is. Day of week also was provided as numeric values so we kept those as is as well. In order to stay consistent with our representation of date information as numeric values, we defined time of day as seconds past midnight and made the necessary conversions.

We used the dataset that we have to generate a classifier using SKLearn's Random Forest Classifier. Currently we are using the area code of the phone call (categorical), the area code of the recipient (categorical), and date and time (continuous) values to build our model. The value we are predicting is the type of spam call (categorical). Since our end goal is to predict whether or not the incoming call is a spam call, we are using the confidence of the predicted type of spam call to evaluate the likelihood of whether or not the call is a spam call. As an example, our model may report that an incoming call if 70% likely to be a spam call of type "Robocall" or 30% likely to be a spam call of type "Telemarketing".

Forecasting Future Spam Calls

Based on the results of our previous classifier, we have determined that the area codes of both source and target locations play a significant role in determining the type of spam call. We use this found knowledge to impute missing values in the columns representing the type of spam call. However, from our data exploration phase, we noticed that approximately half of the dataset contains missing values for the area code of the source location.

To tackle this, we developed another classifier to see if we can predict this area code using a subset of the categorical features provided in the dataset: call method, reported issue, and targeted area code. We built a data pipeline that involves using a customized transformer using SKLearn's TransformerMixin, one hot encoding to translate categorical variables into boolean features, TruncatedSVD for dimensionality reduction, and finally, a multi-class random forest to predict the area code.

With this information, we construct a new boolean feature representing whether the source area code is known and use that information to detect the likelihood of a call

being made in some time interval. This time interval is constructed using the hyperparameter representing the number of bins to equally partition the known dates into subintervals, and categorize each entry with respect to time and date of issue. As the number of bins increase, the length of each subinterval decreases. From this, we receive more accurate “confidence intervals” at the expense of longer training times.

The resulting model to predict the category assigned by the corresponding bin, follows a very similar structure to the previous classifier we have developed to classify the source area code.

Visualization

A web application was created using the Flask framework in python. It consists of three modules: a home page, a real-time prediction system for spam calls, and a US spam call exploration utility.

The home page acts as a portal to the two other modules of the application. It also contains explanations for the other two modules.

The exploration utility allows users to view prior US spam calls geographically, by type of issue, and using a customizable geographical sizing and colour interpolator. It is visualized as hexagonal elements within a US map with visual encodings to demonstrate the number of spam calls. It is visualized using v5 of d3.js.

The real-time prediction system allows users to enter information about a call they are receiving into a form. A request is then made to a Flask API running on the same server that executes the prediction using an interface class to the serialized sklearn model and returns a JSON response with the prediction details and an explanation of the mostly likely class. The prediction probabilities are an output of the sklearn classifier while the explanations of the prediction are created using the lime library which fits linear models to the decision surface in order to create an explainable interpretation of a prediction instance. Once the response is received from the API, the result is visualized to explain what type of spam call it is predicted to be, the confidence of the prediction, and an explanation of the features contributing to the prediction. The explanation is in the form of two bar charts, one for visualizing the prediction probabilities of the different classes and one for the explanation of the top class, rendered using v3 of d3.js.

Flask was used to develop the application since it allows for a simple and maintainable web application while being easy to upgrade to a production server. It also allows for

dependencies to be containerized per module which is convenient for ensuring that there are no conflicts in dependencies between modules. Lastly, it provides many convenient modules and functions for building both APIs and templatable web applications, greatly reducing time to build.

The d3.js library was used for its customizability in creating web visuals which was necessary given the complexity and uniqueness of the visuals, particularly the exploration module. The bootstrap library was used to create a simple and clean web front-end that is responsive and behaves as expected.

Experiments And Evaluations

Testbed Questions

- Do certain types of spam calls follow a predictable pattern based on features of the call?
- Are certain area codes more likely to receive spam calls?
- Are certain area codes more likely to make spam calls?
- Is it possible to predict when a person will receive a future spam call?

Details And Observations

Our predictive model used features such as area code of the spam call, location of the recipient, time of day, day of month, day of week, and month to identify the type of spam call. Using a RandomForestClassifier, we found it had roughly 70% accuracy in identifying the type of spam call based on the features. In addition, roughly 70% of predictions had a confidence value greater than 60% in the predictive model. Also, the most important features in order were time of day, location of recipient, and area code of the spam call. The rest of the features were insignificant.

The resulting classifier developed to predict source area code from other categorical features in our dataset performed poorly. Specifically, our classifier received training score of 0.126 and test score of 0.016. We can conclude that the source area code exhibits little to no relationship from the categorical features present in our dataset.

For the choice of 5 bins in our attempt to develop a reliable forecasting model, we see that the classifier performs with a training and test score of approximately 0.501. Although the model seemingly performs well in the classification scheme, the length of

each subinterval is approximately a year, so the information we gain from the model is not very useful. For larger number of bins, we observe that the model scores deflate abysmally, leading us to conclude that we need to engineer features that provides more predictive power for this task. This is additionally supported by the results of the multicollinearity test on the provided categorical features in our dataset.

Based on the visualizations, we were able to conclude that certain area codes were more likely to receive spam calls. The interactive US map shows that urban areas suffer most from spam calls. The Northeastern region, from Virginia, passing through Maryland, New York, Connecticut, to Massachusetts has the most amount of spam calls, followed by California and Florida. It is interesting to note that there are many areas in the Mountain West that do not have a single reported spam call. In general when filtered by issue, the data shows the same concentrations of spam calls as the total dataset suggests.

Conclusions And Discussions

We concluded that certain types of spam calls may follow a predictable pattern. Certain types of spam calls could follow a trend based on the time of day, the location of the recipient, and the area code of the phone number making the spam call. This information can be used to build a predictive model with reasonable performance in identifying the type of spam call.

We concluded that building a time series analysis to predict when someone will receive a future spam call will occur is impractical with the given information. This is not saying that this is an impossible task; more information could lead to this prediction becoming a reality.

The prediction module of the web application allows the user to both predict the type of spam call using time of call details and to provide an explanation of its prediction. This type of explanatory application for machine learning products is useful for reducing the blackbox effect for nonlinear models and could be extended to other machine learning applications outside of spam calls.

The exploration module of the web application allows users to analyze historical spam calls by type and geographical region. This could be further extended by clustering and network analysis along with other dimensions such as temporal analysis.

Overall, more research needs to be done to conclude if our model is effective or not in combating spam calls, both in real time classification and in determining future occurrences. The model is effective in classifying spam calls based on the historical dataset, but further analysis would be required to determine whether this performance extends to external datasets. While unable to predict future spam call time of occurence based on the current model, it is believed that with additional features it might be possible to improve upon this area as well.

Distribution Of Effort

All team members have contributed similar amounts of effort.