Predicting Spam from SMS Message Board

Group member: Winnie Shao, Yihui Zhang, Wenjia Dou

Summary of research problems

1. What information can we get to distinguish the difference between ham and spam messages if we only perform basic exploratory data analysis?

   In this part, we want to perform the Exploratory Data Analysis on our dataset. Because the dataset we used only has 2 columns, we want to find more information from those text messages. By doing that, we want to compare the length of two categories and also intend to explore the top 10-20 words used in spam and ham messages. We may plot a word cloud to visualize different word choices in spam and ham SMS.

2. How can we process the text message to make them machine-readable?

   In order to predict spam from the message board, we want our data machine-readable so that we need to transform our raw SMS messages into numerical. We plan to develop 5 features (word count / average word length / URLs count / special punctuation count / TF-IDF) that can define these messages. By converting the raw SMS messages into numerical language features, the messages will be machine-readable so that we can use the dataset to make predictions.

3. What model is most effective in predicting spam messages?

   For this problem, we will try some different models, such as Decision Tree Classifier, Naive Bayes Classifier, SVM Classifier, and XGBoost Classifier, to find the most optimized one. To do this, we divide the language feature generated data into training and testing sets and then try to find the minimum test mean squared error and the most accurate train method. Finally, we can plot out the internal decision tree of our model so that we can easily find out the most proper model and most important language features in defining spam messages.

4. What features are most important in predicting spam messages?

   After we figure out which model is the most effective one, we can then choose which features are the most important. Since we have 5 features, including word count, average word length, URLs count, special punctuation count, TF-IDF, we choose each of them one by one as the selected feature, and then use the model we find to see which feature shows higher accuracy to predict whether the message is spam or ham. By doing these, we can finally figure out the most important features in finding spam messages.

Motivation and Backgrounds

Due to the rapid development of technology, companies or some criminals can immediately send pre-written text messages to thousands of personal mobile phones. According to ABC News, based on the data released in 2012 by the Pew Research Center, 69% of those who text said that they got unwanted spam messages. Additionally, 25 percent of them admitted to getting spam messages at least once a week. These spam emails usually include advertisements, unsafe links, fraudulent emails, etc. This not only makes it inconvenient for people to view important text messages but also puts people at risk of being scammed. Our team members are also annoyed by the frequent receipt of various types of spam messages, so our motivation is to help people reduce the trouble and the risk caused by spam messages. Also, we notice that although there are lots of spam message blocker apps available on phones, those apps cannot completely prevent spam messages and sometimes mistreat important messages as spam. Therefore, during the journey to find an efficient model for eliminating spam, we want to understand the decision factors and parameters in those predicting models. After this project, we can find the most efficient model to automatically identify spam messages and reduce the harassment caused by spam messages.

Dataset

URL: https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection

We obtained this dataset from the UCI Machine Learning Archive. In this dataset, we have 5574 text messages. The dataset only has 2 columns: the first column indicates if the text is spam by using string value ‘spam’ and ‘ham’, and the second column contains the text content.

Challenge Goals

Two of the Challenge Goals we plan to use are **Machine Learning** and **New Library**. Our project will meet the goal of Machine Learning because we will look through various model types and the different settings of hyperparameters to decide which model possesses higher accuracy in predicting spam. In addition, we will use a new library of advanced visualizations to perform the analysis of the difference between spam and ham messages (machine learning models performance) and use a new library of natural language processing to process the large volume of the text in our dataset.

Methodology

Our goal is to create several machine learning models to predict which message is spam or ham and figure out which model is the most effective one to use. Also, we want to find the most important features.

Step 1: Exploratory Data Analysis

From our original dataset, we need to grab some basic information about the messages. First, we will clean the data to make sure there are no missing values in the dataset, and also we will add a binary column representing the category of each message(spam = 1 and ham = 0). We will compare the message length distribution for 'ham' and spam messages to see if there is a difference between the mean, median, and standard deviation of different word lengths. Then we will use the Wordcloud package to create word cloud plots so that we could have a better understanding of the word-choice difference between 'ham' and spam messages.

Step 2: Natural Language Processing

After cleaning the dataset and grabbing basic information of messages, we will process our data to make them machine-readable by doing natural language processing. First, we will tokenize the messages in the SMS column and break the words. Next, we will create new columns to contain five numerical language features, including word count, average word length, URLs count, special punctuation count, and TF-IDF, to make the messages machine-readable. The word count column saves the total word without punctuations inside each message; The average word length contains the value of total words length divided by the total word count; The URLs count is the number of the URL links for each message starting with "http"; The special punctuation count represents the number of the special punctuations except periods and commas; The TF-IDF means the term frequency-inverse document frequency between each message and each term in the vocabulary.

Step 3: Model Fitting

Next step we need to figure out which model is the most effective one in predicting spam messages for our dataset and so we can build our machine learning algorithm with it. First, we will choose some models, such as Decision Tree Classifier, Naive Bayes Classifier, SVM Classifier, and XGBoost Classifier, and then we will build a machine learning algorithm with each of them and see which one has the highest accuracy. Specifically, we will divide the language feature generated data into a 20% training set and an 80% testing set and then find the model with the minimum test mean squared error and the most accurate train method. Then we will draw the internal decision tree of our model so that we can easily find out the most proper model in defining spam messages.

Step 4: Finding the Most Important Feature

In this step, we will focus on finding the most significant predictor for spam messages by using the 5 features and the most effective model we obtained from the last steps. For testing each feature, we are planning on building the model with only this feature and then calculate the accuracy of the model prediction. We assume the most accurate model will use the most important feature. Also, we are thinking about implementing the Permutation Importance test so that we could find the most important feature(s) with an amount of randomness.

Work Plan

Task 1: Process Data

In this task our three group members will work together for about 3 days to process our dataset. We will discuss and analyze the data together, and then we can figure out how to build our dataset that will be used in the following tasks.

We will clean the data and build the five features we want for our dataset. Specifically, we will use WeChat to have a group chat to discuss together and use visual studio live share to write code together.

Task 2: Finding Effective Model

For finding the most effective model, we will separately write code in visual studio to test which model is the best one with the highest accuracy for predicting spam messages. Each of us will test at least 2 models and after that we will share our results and choose the best model to use. We will spend about 4 days completing this task.

Task 3: Finding Important Features

Finally, for finding important features, we will separately write code in visual studio to test the importance of each of the 5 features (word count / average word length / URLs count / special punctuation count / TF-IDF). Then we will come together and choose some important features with higher accuracy in predicting spam messages to build our final predicting model. This whole process will take about 4 days to finish.

Citation

ABC News, ABC News Network, abcnews.go.com/blogs/technology/2012/08/69-of-mobile-phone-users-get-text-spam.