CS 267A Project Proposal

Huo, Jianfan
jihuo1116@g.ucla.edu

Du, Yunhao
yudwn010@g.ucla.edu

Wang, Yuchen
ycw509@g.ucla.edu

May 2023

Abstract

Our project aims to create an efficient and accurate spam classification model that will significantly enhance the filtering capabilities of messaging platforms. With the exponential growth of online communication, the number of spam message has become a serious issue, leading to stress and potential safety for users. To address this challenge, we will use the power of modeling data with a probabilistic programming language. Our model will consider specific words within messages as conditional parameters for classification. We considered putting some specific words within the message to our model as the conditional parameter for classification. The dataset will include approximately 87% non-spam and 13% spam messages. We will carefully split the 80% dataset into training and 20% dataset into test to ensure unbiased evaluation. During the training process, we will construct a vocabulary table, which will include words commonly associated with spam, such as "prize" and "claims." These words will be the crucial features for training our model to accurately identify the spam message. For the creation of this glossary, we are still exploring its feasibility, which is the biggest challenge for our project to consider now. After the training is complete, we will check if it achieves a predetermined baseline of accuracy. And we will also use the Problog to refine the model's predictions and further improve its efficiency.

1 Motivation and Introduction

In the dynamic landscape of today's rapidly evolving information, the alarming prevalence of personal data breaches has reached new heights. Protecting our cherished information against such breaches is of paramount importance. However, it is equally important that once our private information is leaked, we must protect ourselves for dealing with the far-reaching effects. Amidst these perils, the nefarious realms of spam and deceitful information cast an indelible impact on our lives. Though the dangers posed by the private information leaks, fraudulent messages have a huge impact on our lives. While sometimes these dangers may seem insignificant, but if we don't pay attention on them, the consequences can quickly escalate to terrible levels, including finances, time and even our lives. Even if we are careful in our lives to avoid becoming victims of these scams, the presence of these massive scams and spam messages can cause us psychological pain. We are keen to reduce the impact of spam on people's lives by implementing a keyword filtering system. During the project, we will develop deeper into the usage of ProbLog, extensively explore the usefulness of the Probabilistic Programming language, and gain a comprehensive understanding of how it benefits data analysis. This groundbreaking solution is designed to protect individuals from the relentless plague of fraudulent messages.

2 Course relevance

In this project, we will cover the topic of Discrete Probabilistic Program Inference we have studied in class. We will explore the use of the probabilistic programming language Problog to calculate the probability that the given message is spam or non-spam given some of the words. The idea is based on the following equations:

$$Pr(Spam|w_1, w_2, ...) = Pr(Spam) * \prod_{i=1}^{n} Pr(w_i|Spam)$$

$$Pr(Nonspam|w_1, w_2, ...) = Pr(Spam) * \prod_{i=1}^{n} Pr(w_i|Nonspam)$$

where wi represents the word in the message sentence.

Before that, we will use Model Counting to calculate $Pr(wi|Spam)$ and $Pr(wi|Nonspam)$ insides the formula above. We need to use the following equations:

$$Pr(w_i|Spam) = \frac{N(wi|Spam)}{N(Spam)}$$

$$Pr(w_i|Nonspam) = \frac{N(wi|Nonspam)}{N(Nonspam)}$$

$N(wi|Spam)$ represents the number of times the words wi occurs in spam messages
$N(Spam)$ represents the total number of words in spam messages
$N(wi|Nonspam)$ represents the number of times the words wi occurs in non-spam messages
$N(Nonspam)$ represents the total number of words in non-spam messages

On the basis of calculating these probabilities, we can further explore the relationship between these words and scam information. In other words, we can try to find out which words are dependent on scam messages in order to help us more accurately identify the spam message in our training model.

3 Background

For exploring the dataset and creating the spam filter model, we mainly use a Python environment and a Probabilistic Programming language ProbLog which we don't know yet. Therefore, we will consider downloading the Problog package in our Python environment and studying ProbLog based on the following relevant references which provide the guideline for installation and tutorial:

`https://github.com/ML-KULeuven/problog`
`https://problog.readthedocs.io/en/latest/python.html`

The dataset used was from The UCI Machine Learning Repository. It includes a total of 5574 SMS messages that were collected from free or free for research sources on the Internet. You can download the dataset from the following link:

`https://archive.ics.uci.edu/ml/datasets/sms+spam+collection#`

4 Measures of success

- Baseline:
  Our minimum requirement is to be able to explore the entire dataset at the end of the quarter, calculate all the probabilities we need, and use our model to achieve at least 80% filter accuracy.

- Medium:
  For the higher level, we will adjust our Naive Bayes and model to achieve more than 95% accuracy for spam classification. In addition, we hope that our model not only has high test accuracy in the dataset we trained but also can accurately filter it when we get a brand new message.

- Strech:
  We all know that if you want to spam a message, there must be a reason, such as fraud, false information, or advertising. If we have extra time, we hope that on the existing basis, our model could provide the corresponding reason based on the key words if it predicts a message needed to be spam.

5 Planning and Timeline

Based on the expected output of our project and the remaining time for this quarter, we plan to divide and arrange tasks on a weekly basis.

- Week 7:
  - Learn ProbLog and get familiar with the dataset (everyone)
  - Pre-processing the dataset for future analysis
    * Extract the useful information (messages and their corresponding labels) from the dataset (Yuchen)
    * Split the dataset into the training set and the testing set for building our model and making the predictions (Yunhao)
- Week 8:
  - Train the model
    * Calculate the frequency at which each word appears on two labels after removing the punctuations and capitalizations (Jianfan)
    * For different combinations of words, calculate the probability that they will form a spam message and the probability that they will form a non-spam message based on the Naive Bayes algorithm (Yuchen)
    * For each label, calculate the probability for each word in that label (Yunhao)
- Week 9:
  - Use the probabilities we calculated to predict the label for a new message (Jianfan)
  - Measure the accuracy of our filter and try to improve it (Yuchen)
  - Write Project Check-in (everyone)
- Week 10:
  - Continue improving our filter (Yunhao & Jianfan)
  - Try to classify the spam messages based on the spam reasons (fake news/fraud messages/advertisement) (everyone)
    * Come up with different reasons for labeling a message to be "spam"
    * Group the keywords for each reason
    * Calculate the frequency at which each keyword appears in each reason
    * For each reason, calculate the probability for each keyword in that reason
- Week 11:
  - Continue working on classifying the spam messages into different reasons and try to achieve the best accuracy (everyone)
  - Final Writeup (everyone)