Machine Learning Engineer Nanodegree

Capstone Proposal

Ramkumar Singh
August 7th, 2017

ML Spam Filter

Domain Background:

In February 2012 Microsoft boasted that its spam filters were removing all but 3 percent of the junk messages from Hotmail. Google claimed that its service, Gmail, removed all but about one percent of spam messages, adding that its false positives rate. The initial implementation of spam filters were based on heuristic technologies—which identify spam based on pre-defined rules.

Although these spam filters were considered as success, they still weren't working well enough. One percent spam is still pretty annoying and a one percent false-positive rate is also not good for user either.

Problem Statement:

Implement a spam filter which could adapt to the changing situation, improve itself by time, and could perform better than the filters based on the heuristic approach mentioned in the above section.

Data Source :

Apache SpamAssasin public corpus has spam and ham mails data. Below is the overview of data

URL:

- http://spamassassin.apache.org/old/publiccorpus/

Content Folder:

- spam: 500 spam messages, all received from non-spam-trap sources.
- easy_ham: 2500 non-spam messages. These are typically quite easy to differentiate from spam, since they frequently do not contain any spammish signatures (like HTML etc).

- hard_ham: 250 non-spam messages which are closer in many respects to typical spam: use of HTML, unusual HTML markup, coloured text, "spammish-sounding" phrases etc.
- easy_ham_2: 1400 non-spam messages. A more recent addition to the set.
- spam_2: 1397 spam messages. Again, more recent.

Total count: **6047** messages, with about a **31%** spam ratio.

Solution Statement:

Create a solution which could generate “Features” to be analysed, on the fly. Use these features to create training/test sets and train a machine learning classifier which could classify the mails as spam and ham with high accuracy.

Benchmark Model :

As per 2015 Gmail's spam stats (spam filter based on machine learning), its false negative is 0.1 percent, and its false positive is 0.05 percent. It detects 99.9 percent of spam mail. On calculating the f1score by above data, the f1score would be 0.9992. The goal is to get close to this benchmark.

Evaluation Metrics:

In real world scenario volume of ham males are generally higher that the spam mails. For such problem f1score is a better metric to analyse performance of a classifier. In the given dataset also the spam ratio is 31% only, so f1score is applicable for the given dataset as well. The solution will evaluate f1score of ML model against the benchmark f1score

Project Design:

Implement a supervised learning classifier model using the SpamAssasin public corpus data to train and test the model.

The email contents need to be processes using some natural language processing library. High frequency words in spam and ham would be considered for the feature.

The solution would try different ML Classifiers like Logistic Regression, K-Neighbour Classifier, SVM with different kernels etc and compare their time efficiency and performance. We would finally fine tune the best classifier to get the high accuracy/f1score