Spam Detection

Connor Moore | Gaury Nagaraju

09 May 2016

Agenda

- Domain: Dataset Selection and Structure
- Data Wrangling Techniques
- Email Classification Techniques
- Spam Assassin
- Future Actionable Steps

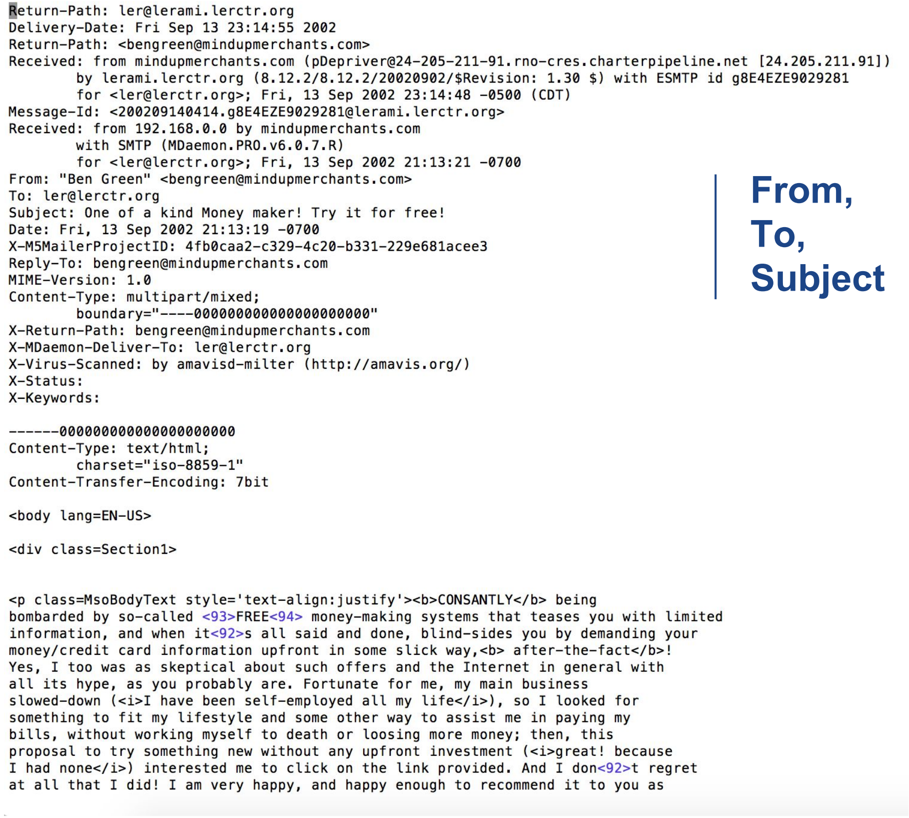
Problem

- Phishing is one of the leading sources of viruses
- Hard to beat b/c taking advantage of something doing it’s job correctly
- Spam filtering is one of the main ways people fight spam

Domain: Dataset Selection & Structure

- Sources Explored: South African Hindawai Journal Research, Phishtank, Enron datasets
- Source Selected: CSMining Group (also referred to by Kaggle)
- Data Structure:
    - Training: Labeled Emails - 4327 messages (2949 non-spam, 1378 spam)
    - Testing: Unlabeled Emails - 4292 messages

Sample Email File: TRAIN_00000.eml



```
Return-Path: ler@lerami.lerctr.org
Delivery-Date: Fri Sep 13 23:14:55 2002
Return-Path: <bengreen@mindupmerchants.com>
Received: from mindupmerchants.com (pDepriver@24-205-211-91.rno-cres.charterpipeline.net [24.205.211.91])
        by lerami.lerctr.org (8.12.2/8.12.2/20020902/$Revision: 1.30 $) with ESMTP id g8E4EZE9029281
        for <ler@lerctr.org>; Fri, 13 Sep 2002 23:14:48 -0500 (CDT)
Message-Id: <200209140414.g8E4EZE9029281@lerami.lerctr.org>
Received: from 192.168.0.0 by mindupmerchants.com
        with SMTP (MDaemon.PRO.v6.0.7.R)
        for <ler@lerctr.org>; Fri, 13 Sep 2002 21:13:21 -0700
From: "Ben Green" <bengreen@mindupmerchants.com>
To: ler@lerctr.org
Subject: One of a kind Money maker! Try it for free!
Date: Fri, 13 Sep 2002 21:13:19 -0700
X-M5MailerProjectID: 4fb0caa2-c329-4c20-b331-229e681acee3
Reply-To: bengreen@mindupmerchants.com
MIME-Version: 1.0
Content-Type: multipart/mixed;
        boundary="----00000000000000000000"
X-Return-Path: bengreen@mindupmerchants.com
X-MDaemon-Deliver-To: ler@lerctr.org
X-Virus-Scanned: by amavisd-milter (http://amavis.org/)
X-Status:
X-Keywords:

------00000000000000000000
Content-Type: text/html;
        charset="iso-8859-1"
Content-Transfer-Encoding: 7bit

<body lang=EN-US>

<div class=Section1>


<p class=MsoBodyText style='text-align:justify'><b>CONSANTLY</b> being
bombarded by so-called <93>FREE<94> money-making systems that teases you with limited
information, and when it<92>s all said and done, blind-sides you by demanding your
money/credit card information upfront in some slick way,<b> after-the-fact</b>!
Yes, I too was as skeptical about such offers and the Internet in general with
all its hype, as you probably are. Fortunate for me, my main business
slowed-down (<i>I have been self-employed all my life</i>), so I looked for
something to fit my lifestyle and some other way to assist me in paying my
bills, without working myself to death or loosing more money; then, this
proposal to try something new without any upfront investment (<i>great! because
I had none</i>) interested me to click on the link provided. And I don<92>t regret
at all that I did! I am very happy, and happy enough to recommend it to you as
```

From,
To,
Subject

Header

Body

Data Wrangling

- Goal: Email → Body Text
- Tools Used:
  - Email Parser (.eml to .json)
  - Beautiful Soup and HTML Parser
  - Pickle
- JSON Object: charset, body (in ascii format), label (is spam or not)
- Went from 4327 to 3721 emails

Research Methods

- Naive Bayes
- Paul Graham Algorithm
- Content filter
- SpamAssassin (comprehensive set of rules)

Email Classification: is it spam?

- CountVectorizer: count word occurrence in each email and associate with spam label
- TfidfVectorizer: take into account term frequency and inverse document frequency
- Spacy Tokenizer: better tokenization with respect to identifying punctuation, digits, urls, lemma

Additional enhancements

- Blacklist: SpamAssassin list of words + word length<3
- N-grams of range 1-3, analyze by word vs. char
- Feature Selection Technique:
  - ExtraTreesClassifier → SelectFromModel
- Email Classification Model:
  - BernoulliNB vs. MultinomialNB

Spam Assassin

- Script: Request to SpamAssassin for spam score for each mail
- Score > 5 ⇒ Spam
- Provides a set for comparison since test data is unlabeled

Model Metrics

- Important Metrics:
  - Recall: positive identification of spam from all of test data
  - Precision: positive identification of spam from those identified as spam
  - Accuracy: model accuracy
- Paul Graham Algorithm rates:
  - False Positive: .05%
  - True Positive: 99.5%

Comparison of Model Metrics

		Spam		Ham	
Feature	**Accuracy**	**Recall = 1-FP**	**Precision**	**Recall = 1-FP**	**Precision**
CountVectorizer with TreeTokenizer	0.958	0.9	0.96	0.98	0.96
CountVectorizer with ngrams, Spacy Tokenization and SpamAssassin Exclusive List	**0.965**	**0.95**	**0.94**	**0.97**	**0.98**
CountVectorizer with ngrams and feature selector	**0.957**	**0.96**	**0.9**	**0.95**	**0.98**
CountVectorizer with Spacy Lemma Tokens	0.763	0.33	0.76	0.95	0.76
Tfidf Vectorizer	0.905	0.69	1	1	0.88
Tfidf Vectorizer with ngrams	0.894	0.51	1	0.82	1

Future Actionable Steps

- Better Data Sanitation
- Message header data unhelpful
- Identify Significant Features for Spam Email: Spam Words, Links, Length of messages
- With more information you can do more (gmail)
- More clever feature engineering