Project 2A Specification

Goals

The goal of this project is to develop—through experience—your ability to:

1. Make technical choices.
2. Bring multiple ideas together to solve a problem.
3. See how one problem can serve as a precedent for the solution of another.
4. Deal with problems that have less specification than that typical of a problem set.
5. Organize and perform work with a partner, if you choose to work with a partner.
6. Write about your work.

The General Email Sorting Problem

Every day we are deluged by email. Some we want to read, some we don't care about, and some is spam: unsolicited email advertising services we don't want. Here at MIT, we are relatively sheltered from spam, but nevertheless, a large amount still gets through. We want you to develop software that will assist in solving the problem.

Content Based Sorting

People have developed many ways to deal with spam. Some systems have blacklists that block out known spamming domains. Others look at the email headers and reject the faked addresses that many spammers use. Yet others look for incriminating evidence like too many exclamation points, or lots of capital letters.

Unfortunately, as spam filters have been getting more sophisticated, the techniques used to avoid them have been improving as well. As filters get better and better, spam looks more and more like real email. How, then, can we detect it?

Fortunately, there is one thing that never changes about spam: it's fundamentally a sales pitch. Content based filtering means that we are trying to detect sales-pitch language, rather than surface characteristics such as how many times the email uses "h0t chixxx!".

Another important consideration in spam filtering is the cost of false positives. If a filter rejects one email which is both real and important, that is **much** worse than allowing a spam to filter through. On the other hand, if too much spam gets through, the filter is not useful.

For some nice reference material on spam and content-based filtering, check out paulgraham.com

Your job

Your job is to build a system that decides whether an email message is spam based on the content (subject line and body of email). Your system is to be trained by examples of spam and non-spam email.

More precisely, you should do the following:

Basic level:

1. Select an approach for determining "spamness".
2. Implement a program based on that approach.
3. Test and evaluate your implemented program.

Optional improvements:

1. Allow user feedback to refine the definition of spam
2. Detect similarity in groups of non-spam emails and categorize them by type.

What you start with

Two files are provided for your use, containing selected portions of an email over the past week. The emails have been stripped of HTML and some junk characters, and only the subject line and body are left. Additionally, some very long emails have been truncated.

1. spam.txt (60 spam emails)
2. notspam.txt (99 non-spam emails)

Note that you will want to separate out test data from training data.

Check points

There is only one checkpoint: you are to provide us with evidence of working code and your final report on the latest day MIT rules permit us to ask for it.

Report Length

The right length for a paper is always the shortest length that covers what you want to say clearly. As a rough guide, we do not want you to write more than **five pages**, exclusive of illustrations, code, run printout, and the like. We would be surprised if you can say what needs to be said in much less.