Contents

Introduction

I don't have to tell you that spam is a huge problem. The fact that you're looking that this tool already confirms you have experienced that. This tool is unique in the sense that it is extremely easy to use if you have an IMAP server.

I used to combat spam with SpamAssassin, but there were three reasons why I started looking for a different tool:

1. It became increasingly ineffective. Upgrading to the latest version helped, but not enough.
2. I started to work for a company that had a Microsoft Exchange Server. So I couldn't plug my SpamAssassin into a `.procmailrc` anymore.
3. I became intrigued by Paul Graham's "Plan for Spam". It looked easy to implement, and very effective.

After a while people picked up Paul Graham's idea and **Bayesian anti-spam tools started to appear**. I looked at **CRM114** for example. But it was hard to use. It didn't work at all for me. Just saving your spam into a text file is much harder than people think. You can't just save a raw email, because it might have an encoded subject, or BASE64 encoded body. Tools that expect email in a text file get very upset by that. Also saving every message into a text file is time consuming. I had a large body of spam, saved over the years, and I wanted to use that.

And even when I tried to feed properly saved messages, those tools just didn't work for me. Maybe because of all those Chinese messages, or because I receive lots of Dutch email.

So I decided it was time to write my own anti-spam tool. The requirements were:

1. Should be very easy to use. I became frustrated with the tools I tried. But I have to admit that your own tools are always the easier to use. . .
2. Work with any email client. I use Emacs **Gnus** for example, can't and won't want a tool that will tie me to a particular email client, Mozilla or Microsoft.
3. It should work for people that have an IMAP server and have no control over message delivery.

My anti-spam solution consists of two utilities:

1. `emc_scan` can read messages stored in IMAP archives. Just specify folder names in the configuration file `emc.config`. No need to save or decode messages.
   The output of this tool is a file called `probabilities.txt` that the feed for the second utility.
2. `emc_classify` can scan an archive for new messages and calculate their spamminess. It can move spam to a different folder for example.

I'm very interested to hear about your experiences with emc. Have fun and no spam!

1 Installation

This chapter describes, very shortly, how to setup emc. Subsequent chapters give more detail.

1.1 Making the binaries available

Copy the following two binaries to a location in your path:

1. `emc_scan`: creates a list of tokens which occur in spam and non-spam.
2. `emc_classify`: classifies email messages and moves them to a special folder if necessary.

1.2 Copy the configuration file

Copy the sample configuration file `emc.config.sample` to one of the following locations:

1. To the `.emc` directory in your home directory.
2. To the `/usr/local/etc/emc` directory.
3. To the directory where emc's binaries reside. With this setup, you can unpack emc and run everything from the same directory if you so desire.

After you have copied it, change the name to `emc.config`.

1.3 Edit the configuration file

Next edit `emc.config`. The comments should give useful guidance. And consult **chapter 3** for full details.

2 How to use emc

This chapter gives a general introduction to working with spam filtering software, and gives particular hints for emc.

2.1 Building a spam corpus

Before you can employ emc, you must have a folder with spam messages. This is your spam corpus. These days building such a thing isn't that difficult unfortunately. Just move every spam you receive to a folder called `spam`. Or any other name you want to give such a folder. Your spam folder should have a few hundred messages before emc can get effective.

You also need one or more folders with non-spam messages. This is your non-spam corpus. In most cases you probably also have these. For example your `Received` folder and your `Sent Items` folders. But any folder with non-spam messages will do. Another useful non-spam folder is one with messages emc has classified wrongly as spam.

Preferably your spam and non-spam corpus have more or less the same number of messages. It doesn't matter much if you have twice as much spam as non-spam for example, but if you have a couple of times more messages in one corpus as in the other, emc will get less effective.

The folders with spam and the folders with non-spam should be specified in emc's configuration file. They appear inside the <scan> element. More information about emc's configuration can be found in **chapter 3**. The following example is an extract of this configuration file and shows how the spam and non-spam folders are specified:

```
<emc xmlns="http://www.pobox.com/~berend/emc/">

  ...

  <scan>
    <spam>
      <mailbox>INBOX.spam</mailbox>
    </spam>

    <nonspam>
      <mailbox>INBOX.Delphi</mailbox>
      <mailbox>INBOX.Eiffel</mailbox>
      <mailbox>INBOX.nonspam</mailbox>
      <mailbox>INBOX.Sent Items</mailbox>
    </nonspam>
  </scan>

  ...

</emc>
```

2.2 Building the word probability table

After you have setup your spam and on-spam corpus, you have to build a table that lists every word in these messages. Every token really, as it considers everything separated by white space as words. This is done with the `emc_scan` tool. Run this tool to build the file `probabilities.txt`.

A typical run looks like this:

```
# ./emc_scan
mailbox scanner 0.6  (c) 2003 by Berend de Boer.
Reading configuration...
Scanning 1 mailboxes for spam tokens...
Scanning INBOX.spam (1 of 1)...
Number of messages: 12261 (learning only 8262)

Completed 1%
...
```

After you have run this tool, you have to move `probabilities.txt` to the directory where the configuration `emc.config` resides.

2.3 Classifying messages

If you have a proper `emc.config` configuration file, and if you have build the token table `probabilities.txt` you are ready to start classifying messages as spam or non-spam. This is done with the `emc_classify` too.

Make sure the <classify> element in the configuration file is specified properly. There are three things you want to specify there:

1. The <mailbox> element contains the name of the mailbox you want to classify.
2. Optionally, you can specify the <move-spam-to> element. This element contains the name of the IMAP folder a message that is classified as spam is moved to.
   Don't add messages that are classified as spam to your spam corpus! This will unnecessarily increase your spam corpus as the message is already recognized as spam. Also if your spam corpus grows much larger as your non-spam corpus, emc's effectiveness will be reduced.
3. Also optionally, you can specify the <move-nonspam-to> element. This element contains the name of the IMAP folder a message that is classified as non-spam is moved to.

If you're unsure if `emc_classify` will do the right thing, just run it with the `-dry-run` option like this:

```
emc_classify -vv --dry-run
```

The `-vv` option makes `emc_classify` more verbose. With the `-dry-run` option, `emc_classify` will only specify what it will do, it will not actually do it.

A typical run looks like this:

```
# ./emc_classify --dry-run
mail classifier 0.6  (c) 2003 by Berend de Boer.
Reading configuration...
```

```
(dry-run mode)
Reading probabilities...
Starting classification...
Classifying INBOX
Connecting to server mail as joe...
Number of messages: 46

Subject: bericht uit Gameren
Spam probability: -0.633749

Subject: Re: [gobo-eiffel-develop] DS_CURSOR.start for filtered cursors
Spam probability: -0.324402

...
```

2.4 Maintaining your spam corpus

You need to maintain your spam and non-spam corpus so the classification process can improve. This is especially important in the beginning. Maintenance consists of two activities:

1. Any time emc does not recognize a message a spam while it should have, you add that message to your spam corpus.
2. Any time when emc classifies a messages as spam while it is not spam, you add it to your non-spam folder.

3 Configuration

This chapter explains the details of emc's configuration file. This file is used by all emc's utilities.

3.1 The configuration file

EMC needs a single configuration file. The name of this file is `emc.config`. The tools will search for this file in the current directory only.

Configuration consists of three steps:

1. Define your IMAP server and login information.
2. List the archives of spam and non-spam messages.
3. Define the folder that contains your new messages. Messages in this folder can be moved to a spam or non-spam folder.

3.2 Configuration is XML file

The configuration file of EMC is XML based. It should obey the syntax defined in `emc.rng` or `emc.dtd`.

3.3 Configuration of your IMAP server

To connect to your IMAP server, you have to specify three things:

1. The name of your server.
2. Your user name.
3. Optionally your password. If you don't provide a password, you're asked for one at the command-line. Another option is to set the `IMAP4_PASSWORD` environment variable.
   Providing a password in `emc.config` is safe, as long as this file isn't world-readable[1]

This information must be provided within the <server> tag as follows:

```
<emc xmlns="http://www.pobox.com/~berend/emc/">

  <server>
    <host>mail</host>
    <login-name>joe</login-name>
    <password>secret</password>
  </server>

  ...

</emc>
```

[1] EMC should check for this, doesn't do this yet.

3.4 Configuration of your spam and non-spam archives

`emc_scan` will scan folders for spam and non-spam tokens. This information must be provided within the <scan> tag. The folders with spam are listed within the <spam> tag. The folders with non-spam are listed within the <nonspam> tag. An example is:

```
<emc xmlns="http://www.pobox.com/~berend/emc/">

  ...

  <scan>
    <spam max-message-size="65536">
      <mailbox>INBOX.spam</mailbox>
    </spam>
    <nonspam max-message-size="65536">
      <mailbox>INBOX.nonspam</mailbox>
      <mailbox>INBOX.Eiffel</mailbox>
      <mailbox>INBOX.TeX</mailbox>
      <mailbox>INBOX.TeX.ConTeXt</mailbox>
      <mailbox>INBOX.Xplain</mailbox>
      <mailbox>INBOX.Received mail</mailbox>
    </nonspam>
  </scan>

  ...

</emc>
```

Within the <spam> and <nonspam> tags one or more occurrences of the <mailbox> tag must be provided. The contents of this tag is the name of a mailbox.

You can specify the following attributes for the <spam> , <nonspam> and <mailbox> elements:

1. max-message-size: skip messages above a certain size. Usually they contain binary attachments and such, so they're not very useful for scanning anyway. And it might take a while before a 1MB email is retrieved.
2. max-messages: the maximum number of messages to retrieve from an archive. If you have really large archives, you might want to use only the latest 10,000 or so.

3.5 Configuration of email to classify

Within the <classify> tag, you should provide the name of the mailbox to scan for spam. Only new messages are scanned (In IMAP terms, messages without the `\Seen` flag). Depending on your threshold, the messages is classified as spam or non-spam. The threshold is a value between -1 and 1 and indicates the spamminess of the message. A value of -1 indicates with 100% certainty that the message is spam. A value of 1 indicates with 100% certainty that it is not spam. A value of 0 indicates that there is equal evidence both ways. You usually set it between a value of 0 and 1.

```
<emc xmlns="http://www.pobox.com/~berend/emc/">

  ...

  <classify>
    <mailbox>INBOX</mailbox>
<!--      <move-nonspam-to>INBOX.test.nonspam</move-nonspam-to> -->
    <move-spam-to>INBOX.spam</move-spam-to>
    <threshold>0.4</threshold>
  </classify>

  ...

</emc>
```

You also want to provide the <move-spam-to> tag to move spam from the folder with new messages to another folder. You can also provide a <move-nonspam-to> tag to move non-spam. The latter is useful if your messages initially are placed into a folder INBOX.unclassified for example and you want to move spam to INBOX.spam and non-spam to your normal INBOX folder.

3.6 Options

The <mailbox> tag has several optional attributes:

1. max-messages: the maximum number of messages to read from a mailbox. Useful if you have a big archive of spam and you wish to process only a few thousand of it.
2. max-message-size: the maximum size of a message to be processed. If you have messages of several MBs which usually are Microsoft Word or image attachments, you don't want to read those messages as images or Word files are ignored anyway. Reading just slows down the processing.
3. percentage: like max-messages, but a percentage of the total messages in the IMAP folder.
4. lowest-sequence: minimum sequence number of messages that will be processed. Messages in an IMAP archive are numbered sequentially, starting from 1. By setting this attribute, you can skip the first messages in the archive.
5. highest-sequence: maximum sequence number of messages that will be processed.

4 Token probability file

This chapter describes the format of the `probabilities.txt` file. This file is generated by `emc_scan` and used by `emc_classify`. The format of this file is described for the curious, it never needs to be inspected or corrected by humans.

4.1 Overview

The `probabilities.txt` contains of two sections: the header, containing the number of tokens and such, and the list of tokens themselves.

4.2 Header

The header consists of three lines:

1. The first line contains the number of spam messages and the number of non-spam messages in the corpus.
2. The second line contains the spamminess probability to assign to new words. Often the classification tool will encounter words that are not in `probabilities.txt`. It does not know if they are spam or not. It will assign these new words the probability listed on this line.
3. The third line contains the total number of tokens in the file.
   It might not actually be the number of tokens written to the file if the minimum-occurrences attribute of the <scan> element is set, see **chapter 3**.

5 Building from the source

Difficult, not yet recommended.

5.1 Eiffel compiler

ISE Eiffel recommended at this moment.

- ISE 5.2 or later.
- SmartEiffel 1.1rc2 or later. Note that although the utilities compile with SmartEiffel, they do not work correctly with it, due to a problem with SmartEiffel's garbage collector.

Due to use of agents, it does not work with VisualEiffel.

5.2 Eiffel libraries

You will need the following libraries:

1. Gobo, 3.4.
2. eposix 2.0.
3. xip (XML validation).
4. Formatter library.
5. Pipeworks library.

The last three libraries are available from my Eiffel page, **http://www.pobox.com/~berend/eiffel/**.

A Default configuration file

This is the default configuration file distributed with the release.

```
<?xml version="1.0"?>
<emc xmlns="http://www.pobox.com/~berend/emc/" version="1">

  <!-- login information to your mail server -->
  <!-- password is optional, if not set, you are queried for it on
       the command-line -->

  <server>
    <host>mail</host>
    <login-name>joe</login-name>
    <password>secret</password>
  </server>

  <!-- archives to scan for spam and nonspam tokens -->

  <scan>
    <spam max-message-size="262144">
      <mailbox>INBOX.spam</mailbox>
    </spam>
    <nonspam max-message-size="65536">
      <mailbox>INBOX.nonspam</mailbox>
      <mailbox>INBOX.Delphi</mailbox>
      <mailbox>INBOX.Eiffel</mailbox>
      <mailbox>INBOX.TeX</mailbox>
      <mailbox>INBOX.TeX.ConTeXt</mailbox>
      <mailbox>INBOX.Received Items</mailbox>
    </nonspam>
  </scan>

  <!-- options when classifying messages as spam or not -->

  <classify>

    <!-- Folder with new messages to classify. Only unseen/unread
         messages in this folder are classified. -->
    <mailbox>INBOX</mailbox>
```

```
    <!-- If you provide a <move-nonspam-to> tag, nonspam messages are
         moved to the indicated folder.
         For example you can put all your new mail into
         INBOX.unclassified and move it to INBOX only when it is
         nonspam. -->
<!--     <move-nonspam-to>INBOX.test.nonspam</move-nonspam-to> -->

    <!-- If you provide a <move-spam-to> tag, spam messages are moved
         to the indicated folder. It is usually not a good idea to
         move messages classified as spam to the spam corpus you use
         for learning. It is already classified properly so it will
         only unnecessarily increase your spam corpus. -->
    <move-spam-to>INBOX.test.spam</move-spam-to>

    <!-- Threshold is a value between -1 and 1 and indicates the
         spaminess of a message. The bigger the number, the more
         likely it is to be spam. 1 is 100% spam, -1 is 100% nonspam. 0
         is evidence both ways.

         If a message has a spaminess above or equal to <threshold> it
         is moved to the <move-spam-to> folder, if this folder is
         set.
         If a message has a spaminess lower than <threshold> it is
         moved to the <move-nonspam-to> folder, if this folder is set.

         Set it to a value that works for you (I need 0.2 for example). -->
    <threshold>0.4</threshold>

    <!-- The classification might encounter new words, words it has
         not seen before. In that case it doesn't know what
         probability to assign to such words. Is a new word likely to
         be spam, or non-spam?
         This probability is automatically calculated. by the scan tool,
         use the -vv parameter on emc_classify to show this value.

         The following parameter is the strength of your believe that
         this automatically calculated value is correct. The strength
         is a value between 0 and 1.

         If you have much more ham than spam, you might want to have a
         low value here (for example 0.30), if you have an equal
```

```
            amount of spam and ham, or more spam than ham, you might want
            to use a higher value (for example 0.60). If you believe the
            spam probability for new words is correct, just give it the
            value 1. -->
        <strength>1</strength>

        <!-- The minimum-deviation is a range of probability values. Words
            that have a spaminess probabilityy within this range are
            discarded and not used to calculate the spaminess of a
            mail. Some spammers add lots of innocent words to a mail to
            lower the spaminess of their email. By setting the
            minimum-deviation you tend to discard those words, and only
            count the spam or non-spam words that are really
            important.
            The value must be between 0 and 1. -->
        <lower-minimum-deviation>0.40</lower-minimum-deviation>
        <upper-minimum-deviation>0.60</upper-minimum-deviation>

    </classify>

</emc>
```