Canning more than SPAM with Bayesian Filtering.

Martin Overton, IBM Global Services, UK

Email: overtonm@uk.ibm.com
WWW: http://www.ibm.com/uk
Tel: +44 (0) 2392 563442

Abstract:

When most people think of tools to combat malware, very few will give a passing thought to Bayesian Filtering, why?

Common reasons include:
- They don’t realize that Bayesian Filtering can be used against malware (viruses, Trojans, worms, etc.)
- They are just for spam.
- They don’t know how to train them for malware.

This paper will investigate the use of Bayesian Filtering, specifically to counter/block/detect malware. What’s more, this paper will focus on tools such as POPfile and SpamPal (which are free anti-spam systems available for both UNIX and Windows).

The use of Bayesian Filtering systems can be extremely useful in cases of fast burning or very complex malware outbreaks as a stop-gap until the anti-virus vendors manage to get reliable updates out to their customers.

Bayesian Filtering of internal mail can also be useful in identifying infected systems in your organization that need remedial action before the ‘trickle’ of infections become a ‘torrent’ and you are left fighting to keep your head above the rising waters.

The paper will include statistics clearly showing the accuracy of Bayesian Filtering, not just for malware, but also SPAM and 419 advance-fee-frauds too.

This paper was written for, and presented at, the 2004 Virus Bulletin conference at the Fairmont Hotel, Chicago, USA on September 29th – 1st October 2004.

I would welcome any constructive feedback on this paper and its content.
1 Introduction
This paper is the result of ongoing research into Bayesian Filtering as a tool for detecting e-mail borne malware and other unwanted threats.

Research has been ongoing since October 2003 and the results so far look very promising. Bayesian filtering appears to have a place in the anti-malware technician’s toolbox, as not only does it (once trained) detect known malware, but during the research it has also detected (classified) many of the Netsky, Bagle and MyDoom (and many other e-mail worm) samples before the anti-virus companies knew about them (and more importantly, before they could detect them, even heuristically).

The research has been carried out using an Internet facing mail server first using POPfile as a proxy and now directly called via a daemon (POPfiled) by the mailserver (Mercury/32). Other spam filtering techniques and virus scanning of mail is also implemented on the same mail server.

For full details of the setup, please see the network diagram in the ‘Putting it all together’ section later in this paper.

2 Definitions
Let’s get a few basic definitions out of the way first so that we all understand what is meant by the following terms:

2.1 SPAM
SPAM is:

1. A meat product sold in tins (Spiced Pork And Ham, like luncheon meat).\(^1\)
2. Slang for Unsolicited Commercial E-mail aka UCE

Use of the term "spam" was adopted as a result of the Monty Python sketch\(^2\) in which the SPAM meat product was featured. In the Monty Python sketch, a group of Vikings sing a chorus of "spam, spam, spam..." in an increasing crescendo, drowning out other conversation. Hence, the analogy applied because UCE was drowning out normal discourse on the Internet.

2.2 Ham
If it ain’t SPAM it must be Ham!

In other words, this is e-mail you want to receive. My POPfile installation calls the bucket for this ‘inbox’.

2.3 Malware
Malicious software.
“Code that causes unwanted effects: Such as viruses, Trojans (including Remote Access Trojans (RATS)), worms and the side-effects thereof.”\(^3\)

Want to know more?
Viruses and Lotus Notes:- Have the Virus Writers Finally Met Their Match

---

\(^1\) Home page: [http://www.spam.com](http://www.spam.com)

\(^2\) Sketch script can be found here: [http://w3.informatik.gu.se/~dixi/spam.htm](http://w3.informatik.gu.se/~dixi/spam.htm)

\(^3\) This definition was created by the author in 1999 and referenced in the "Viruses and Lotus Notes:- Have the Virus Writers Finally Met Their Match" paper presented at the VB1999 conference.
2.4 419s aka Advance-Fee-Fraud

419 frauds combine the threat of impersonation fraud with a variation of an advance fee scheme in which a letter or e-mail from Nigeria [or just about anywhere now!], offers the recipient the "opportunity" to share in a percentage of millions of dollars that the author, quite often a self-proclaimed government official, doctor, engineer, bank official, etc., is trying to transfer illegally out of wherever. The recipient is encouraged to send information to the author, such as blank letterhead stationary, bank name and account numbers and other identifying information using a facsimile number provided in the letter.

The scheme relies on convincing a willing victim, who has demonstrated a "propensity for larceny" by responding to the invitation, to send money to the author of the letter in Nigeria in several instalments of increasing amounts for a variety of reasons.

Payment of taxes, bribes to government officials, and legal fees are often described in great detail with the promise that all expenses will be reimbursed as soon as the funds are spirited out of Nigeria [or wherever]. In fact the millions of dollars do not exist and the victim eventually ends up with nothing but loss.

Once the victim stops sending money, the perpetrators have been known to use the personal information and checks that they received to impersonate the victim, draining bank accounts and credit card balances until the victim's assets are completely exhausted. For most law-abiding citizens the 419 e-mails/letters are seen through as a hoax/scam, however, millions of dollars in losses are caused by these schemes annually around the world.

The Nigerian government is not sympathetic to victims of these schemes, since the victim actually conspires to remove funds from Nigeria in a manner that is contrary to Nigerian law.

The scheme violates section 419 of the Nigerian criminal code, hence the label "419 fraud." although the fraud is now commonplace outside of Nigeria too.

There are many reports from both the UK and the USA that a surprising number of mugs…er….I mean unsuspecting victims have lost a significant amount of money, been lured to the originating country where they have been imprisoned, tortured and occasionally lost their life too (so much so that it has been subject to an FBI warning as well one from the US Secret Service).  

Want to know more?  

2.5 What is Bayesian Filtering?

In brief: Bayesian filtering algorithms (yes, it is a mathematical theorem) calculate the probability of a message being classified as a pre-determined ‘type’ based on its contents. Unlike simple content-based filters, Bayesian filtering learns from ‘spam’ and from good mail (known as ‘ham’), and any other categories (types) you assign. The result is a very robust, adaptive and efficient classification approach for classifying a message by its content. Once properly trained Bayesian filtering produces very few false positives or negatives.

“Bayesian filtering is based on the principle that most events are dependent and that the probability of an event occurring in the future can be inferred from the previous occurrences of that event.”

Want to know more?  
http://email.about.com/cs/bayesianfilters/a/bayesian_filter.htm  

Source: http://cluestick.me.uk  
1 A reasonably easy to follow example of the theorem can be found here - http://www.nwfusion.com/columnists/2003/0922gearhead.html  
2 From ‘Why Bayesian filtering is the most effective anti-spam technology’ - http://www.gfi.com/whitepapers/why-bayesian-filtering.pdf
http://www-ccrma.stanford.edu/~jos/bayes/Bayesian_Parameter_Estimation.html

3  Who was Bayes?

In brief: “Thomas Bayes was born in 1702 in London, England and was an English theologian and mathematician Thomas Bayes has greatly contributed to the field of probability and statistics. His ideas have created much controversy and debate among statisticians over the years.

Throughout his life, Bayes was also very interested in the field of mathematics, more specifically, the area of probability and statistics. Bayes is believed to be the first to use probability inductively. He also established a mathematical basis for probability inference. Probability inference is the means of calculating, from the frequency with which an event has occurred in prior trials, the probability that this event will occur in the future. According to this Bayesian view, all quantities are one of two kinds: known and unknown to the person making the inference. Known quantities are obviously defined by their known values. Unknown quantities are described by a joint probability distribution. Bayesian inference is seen not as a branch of statistics, but instead as a new way of looking at the complete view of statistics.

Bayes wrote a number of papers that discussed his work. Perhaps Bayes's most well known paper is his Essay Towards Solving a Problem in the Doctrine of Chances. This paper was published in the Philosophical Transactions of the Royal Society of London in 1764. This paper described Bayes's statistical technique known as Bayesian estimation. This technique based the probability of an event that has to happen in a given circumstance on a prior estimate of its probability under these circumstances. This paper was sent to the Royal Society by Bayes's friend Richard Price. Price had found it among Bayes's papers after he died. Bayes's findings were accepted by Laplace in a 1781 memoir."

Want to know more?
http://www-gap.dcs.st-and.ac.uk/~history/Mathematicians/Bayes.html
http://www.bayesian.org/bayesian/bayes.html
http://en.wikipedia.org/wiki/Thomas_Bayes

4  FREE anti-spam products.

So, now you know who Bayes was, what he was responsible for, and that a statistical method (or model) he devised over 300 years ago can be used to address the electronic scourge known as ‘Spam’ or ‘UCE’.

There are a number of anti-spam products available which use Bayesian filtering. These are available in Freeware9, Shareware10, Open Source11 and commercial flavours. To make this paper useful to as wide an audience as possible I will focus on just two, both of which are available at no cost on their respective home pages on the internet.

Most of the paper will use data and techniques from POPfile, as this is the most feature-rich when focusing on Bayesian filtering. It is also available on the widest number of platforms.

---

9 Edited excerpt from: http://www.mrs.umn.edu/~sungurea/introstat/history/w98/Bayes.html
10 Unsolicited Commercial Email
11 Software which is offered at no cost but still copyrighted by the author.
12 Software which allows you to try it out before you purchase it (often 30 days use is allowed before purchase).
13 Software for which the source code is available for users to look at and/or modify and use at no cost.
4.1 POPfile

Here’s a brief description of POPfile from the website:\(^{12}\):

**What is POPfile?**

POPfile classifies email into categories you define. It can sort into spam and not spam or into any number of categories you like (e.g. work, personal, important project, hobby, etc.).

The classification is done using a naïve Bayes algorithm. In other words, POPfile uses statistics to track which words are likely to appear in which messages. This means that POPfile will adapt to the kind of mail you receive and needs to be trained. Out of the box, it doesn't know anything about spam or how messages from your mother differ from those your friends send you. However, if you train it, it will soon learn how to tell these different kinds of messages apart.

**Was POPfile based on Paul Graham's research?**

No, it was inspired by another project called Ifile, that predates Graham's now famous paper [*A Plan For Spam*]. It should be said however, that since the statistical approach to spam filtering became popular in the summer of 2002, a lot of ideas have been shared across projects. This even led to the creation of a [Spam Conference] that held its first meeting in Boston at MIT on January 17th, 2003 where many of the leaders of various filtering projects got together to discuss how to tackle this problem and where both Paul Graham and John Graham-Cumming presented their work.

POPfile is written in Perl, so it works on most (if not all platforms) that support Perl. This includes Windows, in fact they offer a very nice ‘Easy-to-install Windows Version’, which installs all the required parts automatically.

Once installed and POPfile is running, all configuration is done via the Web interface (usually accessed at http://127.0.0.1:8080).

More details on POPfile, what it looks like and how it works can be found in section entitled ‘POPfile Configuration’.

---

4.2 SpamPal

Here’s a brief description of SpamPal from the website.\(^\text{13}\)

**How does SpamPal work?**

SpamPal sits between your email program and your mailbox, checking your email as you retrieve it. Any email messages that SpamPal considers to be spam will be "tagged" with a special header; you simply configure your email client to filter anything with this header into a separate folder and your spam won't be mixed up with the rest of your email anymore!

But how does SpamPal know what is spam and what isn't? Well, it uses what are called DNSBL lists. Patterned after the famous MAPS RBL, these are lists of parts of the Internet that in one way or another facilitate spam. Any email you get from a machine on one of these lists has an increased probability of being spam. Some ISPs already block all email from machines on some of these DNSBL lists, but many do not. This is where SpamPal comes in.

You can choose to use any or all of the freely-usable DNSBL lists; SpamPal will look at the machines each email message passed through on its way to your mailbox, and if any of those machines is on one of the DNSBL lists you have chosen then that message will be tagged as spam.

Because not everyone who uses a machine on a DNSBL list will be a spammer, SpamPal has a powerful whitelisting feature that allows you to ignore DNSBL listings for certain senders or for parts of the Internet.

As you can see from the description above, SpamPal is an anti-spam system which uses Whitelists, Blacklists and RBLs\(^\text{14}\). SpamPal can be extended via a number of free plugins, one of which is a Bayesian Filter. However, this implementation is rather limited at the moment, as it only supports two classifications; SPAM and HAM.

SpamPal only works on Windows.

5 It's all about Training!

As with most things in life, Bayesian filtering only works well when it is properly trained. This is usually done in two ways:

1. You feed it at least several hundred messages for each ‘bucket’ or classification you wish to use. You can automate this task using the supplied ‘insert.pl’ script for POPfile.
2. You manually ‘fix’ classification errors as you use it (which you will need to do to maintain the accuracy).

Option 1 improves the speed of training a new installation which doesn’t yet know what each classification should be composed of. It is the quick and somewhat dirty method. Whereas option 2, is slower, more manpower intensive but less prone to errors, as each misclassified (false positive or negative) is dealt with as they occur.

There is a third option, especially if you use POPfile, you can use another persons or companies word-list (corpus database). This combines the best of both options 1 and 2, without you having to do all the hard-work.

This ‘third’ option would be most beneficial when dealing exclusively with malware as it can seriously shorten the ‘time-to-train’ to almost zero, and the classification accuracy to 95 percent (or more), although some training will still be required to maintain the high-level of accuracy.

---

\(^{13}\) SpamPal Home Page - [http://www.spampal.org/](http://www.spampal.org/)

\(^{14}\) Realtime Blackhole List. More details can be found here – [http://www.exim.org/howto/rbl.html](http://www.exim.org/howto/rbl.html)
5.1 Filtering SPAM from Ham.
This is the simplest of the classifications to make; normal mail that you want or are expecting should be
classified as ‘inbox’, everything else should be classified as ‘spam’.

Full statistics and bucket breakdowns can be found later on in this paper in the section entitled
‘Statistics’.

5.2 Filtering 419s from SPAM
This is the next step, if you are interested in how many of these Advance-Fee Frauds you receive. They
are trivial to spot and therefore it is a simple task for a Bayesian filter to tag them as such.

Full statistics and bucket breakdowns can be found later on in this paper in the section entitled
‘Statistics’.

5.3 Filtering Malware
This is the trickiest one to achieve as most e-mail based malware has little or no textual data, only
MIME encoded data. But fear not, they can still be classified quite successfully.

Also, please remember that POPfile includes mail header data in its classification database (corpus), so
even that data is useful and can be used to help classify e-mail borne malware, as is any textual
information in the e-mail body.

Initial training of POPfile to detect malware is not difficult, but in most cases should be carried out by
anti-malware staff that know, or have been trained, how to safely handle malware infected e-mails.

Once POPfile has been trained with several samples of each common e-mail borne malware it will
know what they look like and should now be able to detect other e-mail borne malware that it hasn’t
seen before, without being retrained.

However, the more malware it sees (and classifies) the smaller the probability of either a false negative
or false positive being generated, and the better the probability of it correctly classifying new
(unknown\footnote{One that the anti-virus companies haven’t yet added detection for, to their products.})
malware when it sees it.

It makes little difference to POPfile if the malware is encrypted\footnote{Such as password-protected ZIP files.}, packed or otherwise obfuscated
(including multiple passes through the same or different packers/compressors\footnote{These could include: RAR, ZIP, ACE, UPX, Yoda, PeX, PePack, FSG, and so on.}) before it in MIME
encoded.

As you will see from the statistics in the section entitled ‘Statistics’ later in this paper, malware makes
up a significant percentage of my e-mail totals per month (at least 10%, and as much as 25%).
6 POPfile Configuration

This section will take a quick walk through the web interface of POPfile. More detailed information and the manuals for POPfile can be found on the POPfile website.

6.1 Someone, please hand me a bucket!

The above screenshot shows the ‘Buckets’ tab of the POPfile web interface. You can see that I’ve created four ‘Buckets’; 419-aff, inbox, malware and spam. The unclassified one is created by POPfile and is used to classify e-mails that didn’t meet the classification requirements for any of the other four buckets.

As you can see from the screenshot, each bucket is colour coded and you can specify other actions for each bucket, such as ‘Turn the Subject Header Modification’ on or off for each bucket. This actually causes the message Subject to be prepended with the bucket name, e.g. ‘[inbox] <original subject line here>’.

The ‘X-Text Classification Header’ option should always be turned on for each bucket, as this is the key to getting your mail server/client to filter out Spam, Malware, etc. so that your end-users don’t even see it.

I would also suggest that the ‘X-POPfile-Link Header’ be enabled for all buckets too. This allows administrators to click a link in the classified e-mail which will take them to the e-mail which is stored in POPfile so that any re-training can be carried out to fix false-positives and/or negatives.

You can modify the colour of each bucket, this shows up in e-mails stored in POPfile, and you can see at a glance which words are part of which bucket.

The ‘Quarantine Message’ option for a bucket is very useful where you want to ‘de-fang’ HTML messages. It basically takes the original body of the mail and appends it as an attachment, then replaces the original body with some explanatory text.

The bottom of the web page shows the current ‘Classification Accuracy’, The bucket details of messages that have been classified, and finally the ‘Word Counts’ for each classification bucket.
6.2 Let us learn from history

The above screenshot shows the ‘History’ tab of the POPfile web interface. You can see the last 17 e-mails stored by POPfile. Furthermore, you can see that all four buckets are represented in this screenshot.

Each e-mail can be re-classified, opened or deleted from the POPfile database via this screen.

Below is an example of an e-mail stored in POPfile.

You can see that many of the words in the e-mail are colour coded, and that this e-mail has been rightly classified as ‘spam’.
6.3 What a magnetic personality!

The above screenshot shows the ‘Magnets’ tab of the POPfile web interface. You can see that I currently do not use this feature.

Magnets should be considered as a Whitelist and should be used with caution as they completely bypass the Bayesian classification engine.

6.4 Configuration

The above screenshot shows the ‘Configuration’ tab of the POPfile web interface. From here you can change how POPfile looks (Skin), which ports it uses for POP3 and the web interface.
6.5 Security

The above screenshot shows the ‘Security’ tab of the POPfile web interface. From here you can allow/disallow remote access and enable/disable password protection of the web interface. You can also enable/disable Statistics reporting, Update checking, and allow/disallow remote POP connections too.

6.6 Advanced

The above screenshot shows the ‘Advanced’ tab of the POPfile web interface. From here you can tweak lots of internal settings and add/remove words to be ignored by POPfile.
### 7 Putting it all together

This section will cover how I use POPfile and how it was integrated into my WormCharmer\(^{18}\).

#### 7.1 My Setup

I have the following personal mail server running on an ADSL link from my ISP. All the following are running on a single PC (Pentium III 500Mhz) running Windows 2000.

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
<th>Home Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mercury/32</td>
<td>Mercury/32 acts as my SMTP/POP3 server, all inbound/outbound mail passes through this server.</td>
<td><a href="http://www.pmail.com">http://www.pmail.com</a></td>
</tr>
<tr>
<td>Web2Pop</td>
<td>This application I use to get mail from free web-based mail accounts which I have created to act as malware, scammer and spammer bait.</td>
<td><a href="http://jmasoft.free.fr/">http://jmasoft.free.fr/</a></td>
</tr>
<tr>
<td>POPfiled</td>
<td>This is the daemon for Mercury/32 to allow it to call POPfile directly rather than as a POP or SMTP proxy.</td>
<td><a href="http://home.pacbell.net/mmiinc/POPfiled/">http://home.pacbell.net/mmiinc/POPfiled/</a></td>
</tr>
</tbody>
</table>

This is what the mail server flow diagram looks like before it is integrated into WormCharmer:

All mail regardless of being inbound or outbound passes through Mercury/32 and POPfile via POPfiled, and therefore all mail gets classified.

7.2 **POPfile and WormCharmer**

This section will show how POPfile has been enabled to feed samples into the WormCharmer system (details of which can be found in the proceedings of VB2003, and also here: http://arachnid.homeip.net/papers/).

This integration means that the real-time statistics produced by WormCharmer are now more ‘rounded’ and offer a more realistic ‘World-view’ of the current in-the-wild and ‘breaking’ malware.

The above chart shows a ‘Simplified’ view of the WormCharmer system, clearly showing how e-mail has been integrated into it. You can see that POPfile and a virus scanner have been implemented to classify/scan all e-mail (inbound and outbound).

Mail gets both virus scanned and classified by POPfile, this ensures that:

1. New malware gets classified by POPfile, even if it gets missed by the virus scanner.
2. All identified malware (by the virus scanner) is automatically moved to a folder which is processed by WormCharmer.

So, this means that WormCharmer now processes all SMB (445,139) and E-mail (25 [SMTP], 110 [POP3]) captured/classified samples.

Further developments of WormCharmer are planned, including the addition of capture feeds from Honeypots and other sensors.
The above flowchart shows the logic and flow of a trapped sample through the system, either from SMB or E-mail source. Further integration of mail sample capture into the main WormCharmer application is planned.
8  Statistics
Let us look at what all this means where it matters; the statistics.

From these statistics you can estimate the potential saving to your company by using Bayesian filtering technologies to classify e-mail, and then discard or quarantine those, such as 419s and Malware, which have to potential to cost your company the most in both lost manpower, and business.

8.1  Spam vs Inbox
In this section of the paper you will find a series of charts. These clearly show the percentage for each ‘bucket’ for each month since January 2004. These statistics will be updated each month and will be made available on my personal web site for all to see and/or use.

This chart shows the statistics for the ‘simplest’ POPfile bucket configuration; just SPAM and Inbox. The data is ‘real’ data from my own mail server. It shows that in February 2004 that 70% of all e-mail I received was SPAM.

The next set of charts will drill down from this, splitting out the other buckets (419-aff and Malware) so that you can see the relevant proportions of the SPAM bucket data they are each responsible for.

This chart shows the statistics for the 419-aff bucket broken out of the overall SPAM statistics shown previously. As you can see, this buckets overall affect on the total SPAM percentage is pretty small.
This is the chart you all want to see, as it shows all the four buckets I use, including the most important one (for most of us anyway); malware. As you can see from April on to the end of May 2004 was rather busy on the malware front. This was mainly due to the fight between the Netsky and Bagle authors; however, Mydoom and MiMail also played their part.

The main thing of note here is that POPfile correctly classified many of the new variants (and other e-mail borne malware) during this period without needing to be retrained in many cases. This is before many anti-virus companies had detection for those new variants and other new malware.

8.2 Accuracy

Right, I’ve shown that Bayesian filtering can be useful in the war against the dreaded Spammer, and even the Scammer and Malware authors too. But, I hear you cry “How many false alarms am I going to have to put up with?”

Good question, and below you will find the answer.

The above chart shows the level of classification accuracy I have experienced with POPfile between January and the end of May 2004. As you can clearly see it hasn’t fallen below 99.71%. That is the equivalent of only 29 classification errors per 10,000 e-mails. The best is only 26 classification errors per 10,000 e-mails. When this is compared to standard ‘Weighted keyword filtering’ based anti-spam tools the results are rather surprising [19].

In summary Bayesian filtering improves with age (the more spam/non-spam it sees the better its accuracy gets). The ‘Weighted keyword filtering’ is good in the short-term as it needs no training, however it does need constant reviewing and updating of the keyword lists that it depends on to maintain an acceptable level of false positives and negatives.

8.3 One Bucket at a time

Now the next stage is for us to look at each bucket in turn. So here goes!

This chart shows the statistics for just the ‘inbox bucket’. This is the mail I want to receive and read.

This chart shows the statistics for just the ‘spam’ bucket. This is the mail I don’t want to receive at all.
This chart shows the statistics for just the ‘419-aff’ bucket. This is the mail that I index and add to a database.

Any last but not least, this chart shows the statistics for just the ‘malware’ bucket. This is the mail that I index and gets processed by the WormCharmer application, which in turn creates charts and statistics which are used for trend analysis.
9 What else can I do with it?

You could train it to classify the following into their own buckets:

- Phishing Scams
- Hoaxes
- Urban legends

You may also be able to train it to classify:

- non-English mail (e.g. Big5)
- Graphics
- HTML based mail
- Anything that you want….

10 So, is the end of SPAM?

No. Standard Bayesian aka ‘Naïve Bayes’ is very accurate when well trained but it is not perfect.

Other techniques:
Markovian matching filter appears to be a very strong contender, as discussed in ‘The Spam-Filtering Accuracy Plateau at 99.9% Accuracy and How to Get Past It’ by William S. Yerazunis, PhD Presented at the 2004 MIT Spam Conference January 18, 2004 (http://crm114.sourceforge.net/Plateau_Paper.pdf)

10.1 Common tricks to try and fool Bayesian Filters:

So now you know how well Bayesian filtering works, what are the spammers doing to try and bypass/fool it into allowing their offspring in?

10.1.1 Wordlists aka ‘Dictionary Salad’

This works (so the spammers believe) by drowning out the ‘spam’ words/tokens and therefore adjusting the classification score to, they hope, make it appear not as spam but as ham.

Below is an example of what this word list may look like. This is taken from real spam, and no it didn’t fool POPfile into classifying it to go into the ‘inbox’ bucket.

```
eerily indigene charta descriptive gunpowder crystal carrie parentage smack anus thrive heisenberg halite desperado spyglass hovel variable blastula collage effectuate torsion earthmove pitchfork bristle burro transom al embassy brand despot zoroaster grotesque rotarian conscience bernhard degumming crotalia auditor tic cyclopean impel sunspot melanoma fifty transcribe husky sack styli alleviate captor censor lest armoire logarithmic macon broomcorn errancy eleanor dyspeptic artie haberdashery berserk cyclades servomechanism castillo appetite biggs dominic bergamot ncr elbow headquarters matricies judiciary macedon bronco castor gentlemen often yapping wasteland wrack delft area ranier preparatory chalkboard brainwash antigen cruelty motive parisian bluebonnet prodigal indiana loot statutory enunciable chinaman gusset stylish margin marten hoop minefield blithe teleprompter ephemeresides halverson anthracite bunk conant redemptive ottawa cache irresistible cobalt illicit coil lookup des clap upkeep dickerson maledict centenary donkey skirmish meredith gaggle heburn necromancy integral isotropic dolce rhodolite embroil aficionado abetting rothschild draftsperson coulomb avoidance cretaceous bantam bertha bronco calendrical hydride consul agave canyon westerly paranoid verse carpet conley gadfly assault he'd ivanhoe committable singapore radian dryden eschew dauphine gulf waveform argumentative charleston emanate weren't curia prow rumpus dusky merrymake ember attach citron belying
```

20 As I was finishing up this paper I added a ‘bucket’ for Phishing to my POPfile configuration. Early test results look promising, and I’ll have statistics available in time for the paper to be presented at the conference.
10.1.2 *Text passages from books, etc. aka ‘Long Story or News Story attack’*: 
This works (so the spammers believe) by drowning out the ‘spam’ words/tokens and therefore adjusting the classification score to, they hope, make it appear not as spam but as ham.

Below is an example of what this may look like. This is taken from real spam, and no it didn’t fool POPfile into classifying it to go into the ‘inbox’ bucket.

It also would create a ?one-stop? point of access to grant funding, technical assistance, threat assessments and information on best practices and equipment. It further would allow more flexibility in the use of federal dollars to address homeland security needs and provide first responders with funding in a more coordinated and timely manner. <br>WASHINGTON, D.C.?A new General Accounting Office (GAO) report examining the impact of the Government Performance and Results Act of 1993 (GPRA) 10 years after its enactment shows that the federal agencies have made steady improvement toward producing strategic plans, annual plans for the upcoming year, and then reporting on their success in meeting those goals. <br>They point to Susan Seidelman's "Desperately Seeking Susan." The 1985 film starred Madonna in all her 1980s black-lace glory, and showed that a story centered on two female protagonists could appeal to the masses. It's one of the most successful independent films to date.

10.1.3 *“Habeas Haiku”*
The Habeas Haiku is a short poem whose copyright is owned by Habeas Inc, who are a group of lawyers. Initially the use of the Habeas Haiku was to be under license only to companies that agree to abide by Habeas' code of email conduct. However the ‘Habeas Haiku’ has now been ‘borrowed’ by the very people it was supposed to protect against; the spammers! So much so that receiving an e-mail with the Habeas Haiku in the body of the e-mail is now a strong statistical predictor of the e-mail actually being spam.

Below is an example of the Habeas Haiku X-Headers that are added to e-mail, this one is actually taken from a SPAM e-mail.

X-Habeas-SWE-1: winter into spring
X-Habeas-SWE-2: brightly anticipated
X-Habeas-SWE-3: like Habeas SWE (tm)
X-Habeas-SWE-4: Copyright 2002 Habeas (tm)
X-Habeas-SWE-5: Sender Warranted Email (SWE) (tm). The sender of this
X-Habeas-SWE-6: email in exchange for a license for this Habeas
X-Habeas-SWE-7: warrant mark warrants that this is a Habeas Compliant
X-Habeas-SWE-8: Message (HCM) and not spam. Please report use of this
X-Habeas-SWE-9: mark in spam to .

10.1.4 *The Specialist ‘Mailing-List’ Trick*
The trick here is for spammers to target specialised mailing lists. They believe that these lists are very likely to be whitelisted by the list memberships companies, and so will bypass the ‘anti-spam’ filters.

This is why I do not approve of using the ‘magnets’ feature of POPfile, or indeed any ‘Whitelist’ based filter as this falls right into the spammers trap.

---

21 More detail about the Habeas Haiku system can be found here - [http://www.habeas.com/](http://www.habeas.com/)
10.1.5 This mail is from me and to me!

Spammers have learnt from the malware authors just how easy it is to forge e-mail headers, not just the To: and From: headers but also the HELO string of the SMTP mail server, as these tricks will fool many simple spam filtering systems and also, yet again, are aided and abetted by ‘Whitelist’ based e-mail authentication systems.

Below is an example of the headers of a ‘forged’ e-mail, this one is actually taken from a SPAM e-mail that is allegedly from ‘me’ to ‘me’.

```
Return-path: <martin@arachnophiliac.com>
Received: from c*****.com.** (XX.XX.XXX.XXX) by arachnid.homeip.net
(Mercury/32 v4.01a) ID MG001691;
X-Message-Info: 2UK10TIUxyk31fozIM928A53whzCFvBTL656
Received: from mail pickup service by XX.XX.XXX.XXX with Microsoft SMTPSVC;
Thu, 17 Jun 2004 16:53:13 +0300
Content-Class: urn:content-classes:message
Reply-To: "Sheldon Blount" <martin@arachnophiliac.com>
From: "Sheldon Blount" <martin@arachnophiliac.com>
To: "Martin" <martin@arachnophiliac.com>
Subject: Martin eggplant maestros of 73
Date: Thu, 17 Jun 2004 19:50:13 +0600
MIME-Version: 1.0
```

Below is an example of the headers of a ‘forged’ e-mail, this one is actually taken from an e-mail borne worm that is allegedly from ‘me’ to one of my ‘dummy’ e-mail accounts. Notice that this one has also forged the HELO string to appear that it has come from my own mail server (arachnid.homeip.net), however the IP address is wrong.

```
Return-path: <martin.overton@arachnophiliac.com>
Received: from arachnid.homeip.net (XX.XX.81.190) by arachnid.homeip.net
(Mercury/32 v4.01a) with ESMTP ID MG00154E;
15 Jun 2004 03:44:10 +0100
From: martin.overton@arachnophiliac.com
To: lure@arachnid.homeip.net
Subject: Re: Secure SMTP Message
Date: Mon, 14 Jun 2004 19:44:57 -0700
MIME-Version: 1.0
Content-Type: multipart/mixed;
boundary="----=_NextPart_000_0016=_NextPart_000_0016"
X-Priority: 3
X-MSMail-Priority: Normal
```
10.1.6 The usual suspects
The spammers also use the usual list of tricks which they employ to try and defeat ‘Weighted-keyword’
based anti-spam filters, such as:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>S P A C E S</td>
</tr>
<tr>
<td>2.</td>
<td>R.a-n£dˆo’m P?u&lt;n&gt;a-t#:i_o_n, etc.</td>
</tr>
<tr>
<td>3.</td>
<td>Characters r3placed by numb3r5.</td>
</tr>
</tbody>
</table>
| 4. | Characters replaced by non-alphabetical ch@ra(t£r$.
| 5. | Accented or other non-English alphabet characters. |
| 6. | Lots of html comments and invalid html commands. |
| 7. | Diferent colour HTML text to hide the ‘spammy’ words from all but your eyes. |
| 8. | Invisible HTML ink, white text, white background. |
| 9. | Web-bugs and HTML links only in e-mail body. |
| 10. | Too many others to list …… |

10.2 Sneakier tricks to try and fool Bayesian Filters:
So, what other tricks can/have the spammers employed?

10.2.1 Gamekeeper turned Poacher
Bayesian filters used to defeat Bayesian filtering. Effectively using an ‘Evil’ Bayesian against a ‘Good’
Bayesian to try and find word lists that are in the ‘inbox’ word list of the ‘Good’ system. This is easily
defeated by not bouncing bad mail, but quarantining or deleting it instead.

10.2.2 A picture is worth a thousand words
In this method a spammer will usually send just a graphic file. Just like detecting malware
detecting/blocking/filtering is based on the MIME-encoded binary data.

10.3 What else can I do to improve early detection?

10.3.1 I’m a dummy, dummy!
Seed ‘dummy’ e-mail accounts within your own internal mail servers, web server pages and also
external web sites too. These can then be seen by systems ‘spidering’ your infrastructure for new
‘victim’ spam, scam and malware addresses to add to their mailing list/database.

These ‘dummy’ accounts should be monitored regularly to ensure that you catch new issues as soon as
possible. This could include a pop-up of other automated alert system when mail arrives in the mailbox
of these ‘sensor’ accounts.
11 Detecting Malware

Right, let us now move on to the subject of malware detection/blocking/filtering by POPfile.

The example below is from an e-mail that was generated by one of the W32.Bagle variants and as you can clearly see it has yet again, forged the To: and From: addresses to make it appear that it came from me.

This particular variant was one of the many that used password-protected zip files as a container to hold the infected executable file, and it also used the method of holding the ‘password’ in a graphical format rather than ASCII text in the e-mail body.

Malware files sent this way were at this time causing some ‘problems’ for many anti-virus firms (and many vendors are still not able to detect this sample accurately). POPfile, however had already classified other variants of this family and was able to correctly classify this sample correctly without having to do anything more than compare the content of the mail against its ‘corpus’ for each bucket until it was classified into one of them, in this case the ‘malware’ bucket.

You can clearly see the X headers added by POPfile when it classified this infected e-mail.

```
Received: from spooler by arachnid.homeip.net (Mercury/32 v4.01a); 4 Apr 2004 02:40:36 +0100
X-Envelope-To: <lure@arachnid.homeip.net>
Return-path: <martin.overton@arachnophiliac.com>
Received: from ********.org (XXX.XXX.80.145) by arachnid.homeip.net (Mercury/32 v4.01a) ID MG000D98;
   4 Apr 2004 02:40:20 +0100
Date: Sat, 03 Apr 2004 17:40:21 -0800
To: lure@arachnid.homeip.net
Subject: Protected message
From: martin.overton@arachnophiliac.com
Message-ID: <rurhamcaomwjl7ivhie@arachnid.homeip.net>
MIME-Version: 1.0
Content-Type: multipart/mixed;
   boundary="--------jyhwaonxrhksmxvxcilm"
X-Text-Classification: malware
X-Recipient: <lure@arachnid.homeip.net>
X-Graphics: YES
```

The X header ‘X-Text-Classification: malware’ can be used to filter infected mail, either deleting it or forwarding/redirecting it to another, suitable quarantine account for further analysis. This means that the intended ‘victim’ won’t even see the infected e-mail, which means they can’t even try and run the attachment after unzipping it (after using the supplied password), and then infecting themselves.

I am not suggesting that virus scanning at the mail gateways is no longer required, but that we have another tool to ‘shore-up’ the defences when new or ‘different’ malware threats appear before the anti-virus companies can get detection/cleaning out to us.

Bayesian filtering will also work and correctly classify partial, bounced, or otherwise corrupted malware samples too. This is not so much of a security risk, but it will help to keep the ‘this-virus-got-past-our-protection’ brigade happy as they won’t see these non-viable samples that may evade the mail scanner(s).

Bayesian filtering can also (and should) be used on internal mail servers too, as this will then allow containment of new outbreaks, or at very least an ‘early-warning’ that something is amiss on your network.
12 Conclusions

So what conclusions can be drawn from using Bayesian filtering to detect not just SPAM, but 419s and last but not least malware?

While I have been finishing off this paper I have added a new ‘bucket’ for Phishing, statistics and results from this new bucket will be made available when the paper is presented at the conference.

- Bayesian filtering is very effective at identifying almost any type of content, and this facility therefore should be used by companies to classify malware (and use suitable filtering to act on these classified e-mails) which will assist in minimising the chance of it getting to the “weakest-link-in-your-security”; the human being behind the keyboard.

- Virus scanning at the perimeter and internal mail-servers still has its place, but adding Bayesian filtering technology will help to close that elusive window-of-opportunity that is otherwise left open when new (unknown) malware is unleashed on to the internet.

- The use of anti-virus, extension blocking/content filtering together with Bayesian Filtering will make the job of the internal <insert the person/team that deals with AV here> less reactive and start towards the more proactive approach that has been needed in most organizations for a number of years to counter the threat from e-mail borne malware.

- Scammers, spammers and malware authors will look for way to try and defeat Bayesian Filtering and so try and neutralize its effectiveness; however at the moment they seem to be unable to reliably and consistently defeat it. In some ways the more they try and defeat it the better it gets at detecting their wares.

- HTTP and NNTP filtering is also possible using Bayesian classification techniques to filter out unwanted content. There is a prototype NNTP proxy already for POPfile and there is at least one project looking into creating a HTTP proxy based on POPfile too.

- Bayesian Filtering is not language dependent this means it works just as well when trained in Chinese, Japanese, French, Portuguese, Spanish, Russian, Arabic and many other languages. In fact it can handle multiple languages with ease.

- In conclusion, Bayesian Filtering can be used to classify malware. However, more work and research is still needed into filtering and classification techniques. New (and old) classification techniques should also be investigated for their suitability for use in malware classification. It seems that no matter what the malware authors throw at us, be it compressed, packed, encoded, encrypted or other obfuscated files and content we will find a way to counter it even if we have to ‘think-outside-the-box’ to find a way to address the problem.