RescoreMassCheck310

Rescore Mass-Check

This is the procedure we use to generate new scores. It takes quite a while and is labour-intensive, so we do it infrequently.

We generate new scores by analyzing a massive collection of mail (a "corpus"), and running software to create a score-set that gets the best possible set of scores, so that the maximum possible number of mails in that corpus are correctly classified (ie. so that SA thinks the ham messages are nearly all ham, and the spam messages are nearly all spam).

Summary

The corpus consists of many (approximately 1 million) pieces of real-world, hand sorted mail.

A smallish number of people (about 15), including the developers themselves, work as volunteer "corpus submitters". They hand-classify their mail and then run mass-check over it. They submit the output logs mass-check generates. Occasionally people review the submitted logs for obvious mistakes, but it is largely a trust system.

If you want to see the statistics from the last corpus run, check the STATISTICS.txt files that come in the SA tarball. It will tell you how many emails were used, and what the hit rates of all the rules were.

Procedure

Here's the process for generating the scores as of SpamAssassin 3.1.0:

1. heads-up

Inform everyone in advance on the users and dev lists that we will be starting mass-checks shortly, and they should get their corpora nice and clean (see CorpusCleaning) and sign up for RsyncAccounts.

Enable all rules using the helper script to do this:

```
  masses/enable-all-evolved-rules < rules/50_scores.cf  \
                      > rules/51_newscores.cf
  mv rules/51_newscores.cf rules/50_scores.cf
  svn diff     [and ensure it looks sane]
  svn commit   [create a new bug attachment for review if in R-T-C mode]
```

Build a prerelease tarball using `build/update_stable`. See `build/README` for details on the build process.

2. announce mass-check

RescoreDetails is the full announcement text (and instructions) for this phase. It's sufficient just to send out a mail something like the one we used in 3.1.0:

```
To: users
Cc: dev
Subject: NOTICE: 3.1.0 rescoring mass-checks

OK, if you're planning to send us mass-check logs for the 3.1.0
rescoring, now's the time!

http://wiki.apache.org/spamassassin/RescoreDetails has all the
details.

cheers!

--j.
```

We then take the log files rsync'd up to the server, and use those logs for all 4 score sets. The initial logs are for score set 3 (the fourth), sets 0, 1, and 2 can be generated from set 4 by stripping out the network tests and/or the Bayes tests.

3. allow several days to complete (it takes a really long time!)

Provide enough time, at least a week including a weekend if possible, giving people enough time to get around to running it given that they may be busy with day-job stuff. 😉

4. close up the rsync site

Before the deadline, send out reminder mails to users and dev – if anyone's forgotten all about it, they have a final chance to pipe up at that point. finally, clean up the backup files and make the submission area read-only (in other words, closed for new data):

```
ssh spamassassin.zones.apache.org
cd /home/corpus-rsync/corpus/submit
sudo rm -f *~           [clean up backups]
sudo chmod a-w *.log .  [make the dir and files read-only]
```

That ensures that the data isn't going to change under your feet.

5. generate scores for score sets

See RunningPerceptron.

Once this is complete, rules/50_scores.cf will have the generated scores, created by runGA. (TODO: I think.)

Set aside the randomized logs set created by runGA, for use in later statistics-generation steps, because they are effectively the "source code" for the rescore run:

```
cd masses
tar cvfz rescore-logs.tgz gen-set{0,1,2,3}-*
```

Warning: these are *BIG* – 4.6GB uncompressed for 3.1.0, for example, 400MB compressed.

6. upload the test logs to zone

Since stuff like the STATISTICS cannot ever be regenerated without the (randomised) test logs, these need to be saved, too. Currently, I think the best bet is to upload the `rescore-logs.tgz` file somewhere on spamassassin.zones.apache.org; it doesn't have to be in a public place, ASF-committer-account-required is fine.

7. upload proposed new scores

Attach the new proposed 50_scores.cf as a patch to the rescoring bug on the bugzilla, for voting and comments. There will always be comments 😉

```
cd ..
svn diff rules/50_scores.cf > ~/newscores.diff
[upload ~/newscores.diff in your web browser]
```

Then wait for votes...

8. Make the stats files

Once the scores are voted on and tweaked to everyone's satisfaction, you'll need to rebuild STATISTICS files with the new scores. First, (just to make sure you're in sync!) repatch your scores file to match what's been voted on:

```
svn revert rules/50_scores.cf
wget -o newscores.diff http://bugzilla.spamassassin.org/....attachment?id=....
patch -p0 < newscores.diff
```

then, a little configuration; replace these with the paths to the correct gen-setN-* directories for the 4 score sets... the test logs the stats are measured against will be taken from these directories. NOTE: don't cut and paste these! they *will* be different for your runs.

```
genset0=/home/corpus-rsync/corpus/scoregen-3.1/gen-set0-2.0-4.0-100-nobob
genset1=/home/corpus-rsync/corpus/scoregen-3.1/gen-set1-2.0-4.0-100-nobob
genset2=/home/corpus-rsync/corpus/scoregen-3.1/gen-set2-2.0-4.625-100-nobob
genset3=/home/corpus-rsync/corpus/scoregen-3.1/gen-set3-2.0-5.0-100-nobob
```

Once those vars are set, run these commands:

```
  cd masses

  rm ham*.log spam*.log ; touch ham.log spam.log
  ln -s $genset0/NSBASE/ham-test.log ham-test.log
  ln -s $genset0/SPBASE/spam-test.log spam-test.log
  bash ./mk-baseline-results 0 > ../rules/STATISTICS-set0.txt

  rm ham*.log spam*.log ; touch ham.log spam.log
  ln -s $genset1/NSBASE/ham-test.log ham-test.log
  ln -s $genset1/SPBASE/spam-test.log spam-test.log
  bash ./mk-baseline-results 1 > ../rules/STATISTICS-set1.txt

  rm ham*.log spam*.log ; touch ham.log spam.log
  ln -s $genset2/NSBASE/ham-test.log ham-test.log
  ln -s $genset2/SPBASE/spam-test.log spam-test.log
  bash ./mk-baseline-results 2 > ../rules/STATISTICS-set2.txt

  rm ham*.log spam*.log ; touch ham.log spam.log
  ln -s $genset3/NSBASE/ham-test.log ham-test.log
  ln -s $genset3/SPBASE/spam-test.log spam-test.log
  bash ./mk-baseline-results 3 > ../rules/STATISTICS-set3.txt
```

There'll be a lot of output along these lines:

```
ignoring 'TO_ADDRESS_EQ_REAL': immutable and score == 0
```

But that can be ignored. (TODO: it'd be nice to make this step a little less labour-intensive.)

8. upload new stats files

Attach the new proposed STATISTICS*.txt as a patch to the rescoring bug on the bugzilla:

```
  cd ..
  svn diff rules/STAT* > ~/newstats.diff
  [upload ~/newstats.diff in your web browser]
```

And let all and sundry vote on that, too. Once the new scores and STATS files are approved and into SVN, and the log data is in a safe archival spot on the zone, you're done.