CRM114 vs. Mr X: Notes for the NIST TREC 2005 Spam Track

Fidelis Assis, *William S. Yerazunis*,*
Christian Seifkes, Shalendra Chhabra

TREC 2005 Closing Plenary

(CRM114 is 100% GPLed open source, and flourishes under the benevolent neglect of numerous companies and universities.)

* Presenter Affiliation: MERL – Mitsubishi Electric Research Laboratories, Cambridge, MA

The Real Goal

- The real goal is not to “get the best numbers” (although that’s an amusing game)
- The real goal is to destroy the spammer’s business model.
- Remember that when you make your engineering trade-off decisions....

CRM114 in One Slide

- It’s not a filter- it’s a language that lets you ***design*** a filter and JIT-compile it.
- The language has one data type – the overlapping string.
- The language allows mix-and-match of processing and N-way classifier options.

CRM114 as a Spam Filter

- People have created configurations for Linux, BSD, MacX, SunOS... Windows (!)
- Companies have integrated CRM114 into solutions for Eudora, Outlook, Webmails...
- CRM114 is also used for other than tasks – web filtering, Usenet monitoring...
- Typical filtering speed : 1 Megabyte/sec on a 1.6 Ghz laptop

The four CRM114 configurations tested:

- **OSBF** – OSB with “double extra voodoo” (a very fast approximate TF-IDF probability modifier)
- **Winnow** – an implementation of Nick Littlestone’s Winnow algorithm (basically a wide perceptron with back-propagation learning)
- **OSB** and **OSB-Unique** – naïve Bayesian classifiers; the only difference is Unique disregards all but the first appearance of any feature.

The four CRM114 configurations tested:

- OSBF – OSB with “double extra voodoo” (a very fast approximate TF-IDF probability modifier **– and a bug!**)
- ... so please ignore the OSBF data. :-(

If you look at numbers for the 44 filter setups tested at TREC:

- CRM114 and the Jozef Stefan filter ROC curves cross each other (though IJS is beautifully flat out at the limits and so IJS gets the best 1-ROCA% , with CRM114 at #2)
- CRM114 has best aggregate h=.1 (3.46) and LAM% (0.62)
- Of the eight “sweet spots” (error rates with a fixed 1% error in the opposite class x 4 test corpora) at least one CRM114 configuration is ***always either best, or statistically indistinguishable from the best filter configuration tested*.**

So, CRM114 does something right.

What does CRM114 do that's different?

What does CRM114 do that's the same?

What part of that is portable to other filters?

.....

"What's in the CRM114 Secret Sauce?"

What Does CRM114 Do Similarly to IJS?

- Tuples to form a Markov Random Field ! [Seifkes et al, ECML/PKDD 2004]
- Note that IJS is very similar with a Markov model; IJS defines single characters as individual MRF transitions; CRM114 uses an arbitrary regex to define each MRF transition.
- So, use a Markov model!

What Does CRM114 Do Differently from Everybody?

- No decoding.
- Of anything.
- Not even MIME or BASE64 encodings of attachments.
- (IJS does do decoding- so maybe decoding attachments is a good idea after all)

Words Are Not Features

Tuple-based features (such as OSB) are much better than single-word features.

Example: the string “foo bar baz wugga” yields this feature stream:

- foo bar
- foo <*skip*> baz
- foo <*skip skip*> wugga
- bar baz
- bar <*skip*> wugga

Words In Context:

CRM114 uses up to 5-word tuple features.

(note that some other word-based filters like DSPAM and SpamBayes have now added options or even default to use 2-gram tuple-based features instead of single words)

What Does CRM114 Do Differently By Design?

- Speed matters!
- Don’t throw away information
- Let the computer do the hard parts
- Openness matters- open source, open mind

Speed Matters!

- Whatever you do, think about the impact of your new gem of coding.
- If your filter is too slow, it will never get wide deployment, which means it won’t impact the spam business model and thus, the spam filter fails in its real goal.

Speed Matters!

- Whatever you do, think about the impact of your new gem of coding.
- If your filter is too slow, it will never get wide deployment, which means it won't impact the spam business model and thus, the spam filter fails in its real goal.
- Corollary: A slow filter means you won't be able to test many variations of the filter.

Avoid Throwing Away Information

- Unlike most Graham-esque filters, CRM114 has no “significance window” of the most extreme N words. Every feature counts, but only a little..
- No word or feature can have an overriding impact.
- There’s no “ten nonspammy words” that can sneak a spam past the filter.
- This totally violates the Bayesian assumption of statistical independence.... but it still works just fine.

Avoid Throwing Away Information

- Because everything counts, CRM114 can use a very gentle conditional probability formula, so statistical outlier features have low impact.
- CRM114's per-feature conditional probabilities are limited to roughly the range:

[.47 ... .53] for hapaxes

[.44 ... .56] for 10 occurrences

[.43 ... .57] for 1000 occurrences

Let the Computer do the Work

- Stop Thinking So Hard!
  - Tuple-based (Markov) features are much more robust than ad-hoc features like “__header:”
  - Don’t bother trying to guess heuristics.
  - Computers are good at accounting- so let the computer figure out what’s significant.

Humans have better things to do with their time.

Observation:

If your feature set is rich, you can use just about any combining rule or database and get publishable results.

(bayesian, winnow, KNN... everything we've tried works decently.)

Speed Matters!!!

- How you store your statistics matters (from the engineering point of view)
- Special-purpose hashing systems can be much faster than relational databases.
- Not just a little faster – hundreds of times faster.
- This really matters when you are doing testing.

Test like crazy.

(yes, I am preaching to the choir)

- Have multiple test corpora.
- There are huge disparities across corpora
- There are huge disparities across shuffles within a given corpus.

(thanks, Gordon! :-))

No Free Lunches?

- The No Free Lunch theorem (Wolpert and Macready, 1997) hits with a vengeance here.
  - “There is no best classifier. Beyond some limit, performance improvement in one dimension will always exact an equal performance penalty somewhere else.”

No Free Lunches !

- Expect 2:1 or worse disparities on an everyday basis, and 10:1 disparities on a monthly basis. The “sweet spot” analysis of the TREC corpora have 10:1 ratios for different corpora.
- You will learn something useful with every rude surprise.

Speed Matters!!!

- You will make far better progress if you can test against a 5,000 message corpus in 10 minutes than if you have to take a weekend to let your test run.
  - The fastest CRM114 configurations are capable of running the entire 4147 message (2003 SpamAssassin corpus) in under 1 minute!

Stop-words and Stemming

- **Don't** do "stop-word elimination".
  - Stop words may be common, but they still carry information, especially in a tuple-feature system.
  - Remember: it’s not bits – it’s bits in context.
- **Don’t** do stemming.
  - It’s slow
  - It’s dictionary-dependent
  - It doesn’t seem to help.

Symmetry?

- Design Decision: Should a filter err on the side of “accept as good”?
- **OPINION:** A good text filter should be completely symmetrical (at least by default).
  - Hypothesis: a falsely “accepted” spam can be as costly as a falsely “rejected” legitimate mail.

Symmetry?

- **OPINION:** A good text filter should be completely symmetrical (at least by default).
  - Consider the cost in time and dollars of a bank-authentication spam that your Grandma “falls for”...
  - If that doesn’t convince you, consider

    S / bank authentication / pedophile /

    S / grandma / 13-year-old with separated parents /

Hapaxes and Grooming

- Hapaxes carry information- but that information only becomes useful when the hapax is seen again (and thus proven to not really be a hapax).
- Don’t discard hapaxes until the last possible instant, when you really need to re-use that memory.

Hapaxes and Grooming

- Use a heuristic to groom out the oldest hapaxes
  - Your database may well have “hidden channel” information on features such as time-last-seen.
  - Even if it’s only relative, or approximate, (such as the hapax position within a hash overflow chain) this information is still valuable for grooming.

No Free Lunches – Part 2

- The No Free Lunch theorem proves that there is no one "best" classifier.
- Empirical observation:

There is no one best classifier author, either.

No Free Lunches – Part 2

- The No Free Lunch theorem proves that there is no one "best" classifier.
- Empirical observation:

There is no one best classifier author, either.

- **If you make your architecture open and pluggable, then good filter people will write code for you, and you can take the credit!***

*So, in the interests of honesty...

...I need to give **very strong acknowledgments and HUGE thanks to:**

Fidelis Assis, Shalendra Chhabra, Jaakko Hyvätti, Barry Jaspan, Jesus Freke, Ville Laurikari, Raul Miller, Paolo Panizza, Christian Siefkes, and the hundreds of code-readers, bug-whackers, and other wise contributors.

Summary

- Speed matters.
- Don’t throw away information.
- Use tuples. Markovian models are powerful.
- Keep stopwords. Keep hapaxes.
- Be symmetrical.
- Let the computer do the heavy lifting.
- Benchmark! Benchmark! Benchmark!
- Be open. Open source, open mind.
- People want to help. Help them to help.
- ... AND ACKNOWLEDGE THEM!

Future Ideas

- Higher speed and accuracy (of course)
- Anomaly / Error / Intrusion Detection System
- Operating on recognized speech
- Embedded “spam-pliance” version
- Information control in restricted environments

Questions?

CRM114 is licensed under the GPL Version 2. It is free for all to download, modify and use.

Complete source code, runnable binaries, and full documentation are available at:

http://crm114.sourceforge.net

Thank you very much.

Are there any questions?