

vb Spam supplement

CONTENTS

S1 NEWS & EVENTS

S1 FEATURE
OSBF-Lua

NEWS & EVENTS

EVENTS

The 2007 Spam Conference will take place on 30 March 2007 at MIT, Cambridge, MA, USA. The title for this year's conference is 'Spam, phishing and other cybercrimes'. See <http://spamconference.org/>.

The Authentication Summit 2007 will be held 18–19 April 2007 in Boston, MA, USA. The two-day intensive program will focus on online authentication, identity and reputation, highlighting best practices in email, web and domain authentication. For full details see <http://www.aotalliance.org/>.

The EU Spam Symposium takes place 24–25 May 2007 in Vienna, Austria. See <http://www.spamsymposium.eu/>.

Inbox 2007 will be held 31 May to 1 June 2007 in San Jose, CA, USA. For more details see <http://www.inboxevent.com/>.

The 10th general meeting of the Messaging Anti-Abuse Working Group (MAAWG) will take place 5–7 June in Dublin, Ireland (members only) and a further meeting – open to both members and non-members – will be held 3–5 October in Washington D.C., USA. For details see <http://www.maawg.org/>.

CEAS 2007, the 4th Conference on Email and Anti-Spam, takes place 2–3 August 2007 in Mountain View, CA, USA. Full details including a call for papers (submission deadline 23 March 2007) can be found at <http://www.ceas.cc/>.

The Text Retrieval Conference (TREC) 2007 will be held 6–9 November 2007 at NIST in Gaithersburg, MD, USA. As in 2005 and 2006, TREC 2007 will include a spam track, the goal of which is to provide a standard evaluation of current and proposed spam filtering approaches. For more information see <http://plg.uwaterloo.ca/~gvcormack/spam>.

FEATURE

OSBF-Lua

Fidelis Assis

Empresa Brasileira de Telecomunicações - Embratel, Brazil

Last month, Gordon Cormack reported on the results of the TREC 2006 spam filter evaluation track (see VB, January 2007, p.S2). One of the top performers in this year's evaluation was OSBF-Lua. Here, its creator Fidelis Assis describes the technology behind it.

The importance of feature extraction and feature selection in token-based spam classifiers is well known. OSBF-Lua is a C module, for the Lua language, which implements a Bayesian classifier. It uses two techniques to address feature extraction and selection: orthogonal sparse bigrams (OSB) for feature extraction [1], and exponential differential document count (EDDC) for feature selection [2].

spamfilter.lua is an anti-spam filter written in Lua using the OSBF-Lua module. It makes special use of EDDC to implement a new and highly effective training method known as TONE-HR (train on or near error with header reinforcement). The combination of OSB, EDDC and especially TONE-HR, to enhance a classical Bayesian classifier, resulted in the best spam-filtering performance in the TREC 2006 spam filter evaluation track [3].

FEATURE EXTRACTION

The OSB technique is a development of and improvement over the sparse binary polynomial hash (SBPH) tokenization technique [4]. The SBPH technique generates a large number of 'features' from incoming email text, then uses statistics to determine the weight of each feature in terms of its spam vs non-spam (ham) predictive value.

SBPH works by sliding a five-token window over a sequence of tokens (e.g. words). For each position, SBPH generates all of the possible in-order combinations of the four left-hand tokens in the window, then appends the rightmost one to each combination to form a set of features.

OSB works in the same way, but produces a subset of SBPH features, made up only of those features that cannot be generated by any combination of the others. Table 1 shows the features generated by SBPH and OSB, when the two

Index	SBPH	OSB
1	<skip> <skip> <skip> <skip> tokens	
2	<skip> <skip> <skip> from tokens	<skip> <skip> <skip> from tokens
3	<skip> <skip> derived <skip> tokens	<skip> <skip> derived <skip> tokens
4	<skip> <skip> derived from tokens	
5	<skip> are <skip> <skip> tokens	<skip> are <skip> <skip> tokens
6	<skip> are <skip> from tokens	
7	<skip> are derived <skip> tokens	
8	<skip> are derived from tokens	
9	features <skip> <skip> <skip> tokens	features <skip> <skip> <skip> tokens
10	features <skip> <skip> from tokens	
11	features <skip> derived <skip> tokens	
12	features <skip> derived from tokens	
13	features are <skip> <skip> tokens	
14	features are <skip> from tokens	
15	features are derived <skip> tokens	
16	features are derived from tokens	

Table 1: Features generated by SBPH and OSB when applied to the sentence 'features are derived from tokens'.

techniques are applied to the sentence 'features are derived from tokens'.

Since all features produced by SBPH can be generated by a combination of those produced by OSB (with a token-on-token 'OR' operation, where the result is either <skip> if there's no token in the position, or token otherwise), OSB is believed to be equivalent in expressiveness to SBPH, which has been supported by experiments. The fact that fewer features are produced by OSB means that this technique is considerably speedier than SBPH, as well as having decreased memory and storage requirements.

The single-word feature, or unigram, at position 1 in Table 1 is not present in the OSB column, despite the fact that it cannot be generated by any combination of the other four OSB features. This is because experiments have shown very similar results whether the unigram is included or not, and so it seems that it is not necessary to include it.

Feature	Distance	Weight
from tokens	0	3125
derived <skip> tokens	1	256
are <skip> <skip> tokens	2	27
features <skip> <skip> <skip> tokens	3	4

Table 2: Features are weighted according to the distance between the tokens.

Intuitively, the sparser the feature, the lesser its significance. To reflect this, we weight them as shown in Table 2.

The weights are calculated using the formula $(5-d)^{(5-d)}$, which was found experimentally, where d is the distance between the tokens, represented by the number of skipped tokens.

FEATURE SELECTION

Exponential differential document count (EDDC), or confidence factor, is an intuitively and empirically derived technique for the automatic reduction of the influence of features with low class separation power.

The idea here is to decrease the importance of features that occur approximately equally in both ham and spam classes. This is achieved by using the normalized counts of the documents containing the feature, in each class, to calculate its confidence factor. The calculated factor is then used to adjust the estimated local probabilities of the feature, in Bayes formula, towards the 'don't care' value (0.5, for two classes).

Figure 1 helps to visualize the effect of the confidence factor, showing how it approaches 0 when the counts are closer in the spam and ham classes and, inversely, how it approaches 1 for features with very different counts in the two classes. The net effect is an automatic selection of the most useful features, because those with a low level of information about their class are practically discarded.

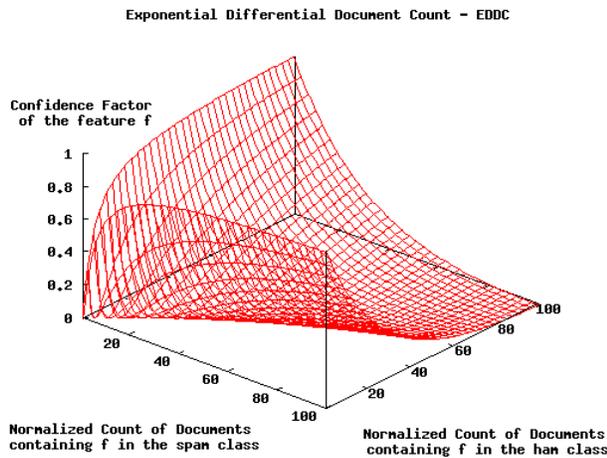


Figure 1: The confidence factor.

TRAINING METHODS

Statistic classifiers build their predicting models by learning from examples. A basic training method is to start with an empty model, classify each new sample and train it in the right class if the classification is wrong. This is known as train on error (TOE) [5]. An improvement to this method is to train also when the classification is right, but the score is near the boundary – that is, train on *or near* error (TONE). This method is also called thick threshold training [1, 5].

The advantage of TONE over TOE is that it accelerates the learning process by exposing the filter to additional hard-to-classify samples in the same training period. Pure TONE was the training method used by *spamfilter.lua* prior to TREC 2006.

TONE WITH HEADER REINFORCEMENT

TONE with header reinforcement, or TONE-HR, is a new training method that was developed for OSBF-Lua during the experiments for the TREC 2006 spam track. It can be seen as an extension to TONE that adds a mechanism similar to white/blacklisting, in the sense that it makes use of information present in the header of the message for the hard-to-classify and hard-to-learn cases. Unlike normal white/blacklisting, though, which is typically manual, header reinforcement (HR) is an entirely automatic process, from the detection of the cases where it applies, to the selection of the most interesting features in the header to be considered.

HR extends TONE in the following way: after a message is trained as in TONE, the new score is calculated and the training is repeated, this time using only the header of the message, while the following three conditions hold:

1. The new score remains near the boundary.
2. The absolute value of the variation of the score is less than a defined value.
3. The number of repetitions is less than the maximum allowed.

The first condition is used to detect when HR applies, and then, together with the second and third, to avoid over-training, which would result in poor score calibration. The limit values for these conditions were found experimentally and are documented in the `spamfilter_commands.lua` source code, which is available in the OSBF-Lua package.

The interesting aspect of this controlled repeated training using only the header, is that instead of just two ‘colours’ – black and white – we get many more gradations between those extremes, producing better calibrated scores and, as a result, an improved area under the ROC curve. Another nice characteristic is that it uses the normal training function already available in the filter, and it takes advantage of EDDC’s ability to select automatically, among the features present in the header, the most significant ones for classification.

Table 3 shows the evolution of OSBF from TREC 2005 to the present version, demonstrating the improvement due to TONE-HR. The measurements were made against the TREC 2005 full corpus.

Version	Training method	(1-ROCA)%
TREC 2005	TONE	0.019*
MIT Spam Conference 2006	TONE	0.016**
TREC 2006	TONE-HR	0.010

(*) Extra evaluation by Prof. Gordon Cormack.

(**) Better EDDC tuning.

Table 3: The evolution of OSBF from TREC 2005 to the present version.

THE ROC CURVE

The area under the ROC curve (AUC), or its complement (1-ROCA)%, is the main metric for ranking classifiers in TREC spam track [6]. While it is a good measurement of the overall performance, it is not enough to assess classifiers when the ROC curves cross each other.

For instance, a low ham misclassification percentage (hm%) [7], is more important than a low spam misclassification percentage (sm%) in spam filtering. An hm% value that is greater than 1% (to use a conservative value) is simply unacceptable. On the other hand, an sm% value greater than 10% is considered very poor for a spam filter. So, the area

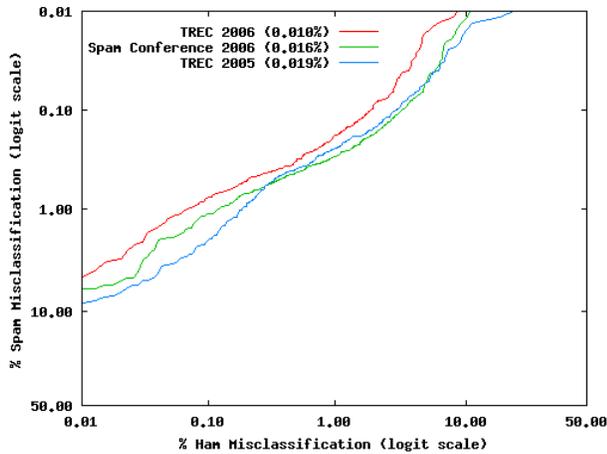


Figure 2: ROC curves for the three versions of OSBF listed in Table 3.

restricted to the acceptable operation region – for instance where $sm\% < 10\%$ and $hm\% < 1\%$ (or even a more restricted one considering the accuracy of present day spam filters) – would be more appropriate when the ROC curves intersect.

Figure 2 shows ROC curves for the three versions of OSBF listed in Table 3. The TREC 2006 curve exhibits the best (1-ROCA)% value and is not intersected by any other, so it is clearly the best of the three classifiers.

Since the other two curves intersect, the better (1-ROCA)% value of the version presented at the MIT Spam Conference 2006 is not enough to tell whether it is the better of the two. However, a visual inspection of these two curves shows that the MIT 2006 version dominates the TREC 2005 version during most of the region of the graph where $hm\% < 1\%$, and confirms that it is the second best overall.

CONCLUSIONS

Training methods play a very important role in the accuracy of adaptive anti-spam filters, side by side with techniques for feature extraction and feature selection for token-based filters, and the two deserve the same attention.

We introduced a training method for statistic anti-spam filters, TONE-HR, and achieved experimental results that demonstrate its significant contribution to the overall accuracy of OSBF-Lua.

OSBF-Lua is free software, under GPL, and can be downloaded from <http://osbf-lua.luaforge.net>. The spam filter *spamfilter.lua* is part of the OSBF-Lua package. For a general-purpose text classifier based on OSBF-Lua, see

Christian Siefkes' *Moonfilter*, at <http://www.siefkes.net/software/moonfilter>.

ACKNOWLEDGEMENTS

My thanks to William Yerazunis for creating the CRM114 project [8], where I found an exciting environment that helped me to develop OSB and EDDC.

A special thank you goes to Christian Siefkes, for his invaluable suggestions and contributions throughout this project.

REFERENCES

- [1] Siefkes, C.; Assis, F.; Chhabra, S.; Yerazunis, W. Combining Winnow and Orthogonal Sparse Bigrams for Incremental Spam Filtering. In European Conference on Machine Learning (ECML) / European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD). September 2004. <http://www.siefkes.net/ie/winnow-spam.pdf>.
- [2] Assis, F.; Yerazunis, W.; Siefkes, C.; Chhabra, S. Exponential Differential Document Count: A Feature Selection Factor for Improving Bayesian Filters Accuracy. In 2006 Spam Conference, Cambridge, MA. <http://osbf-lua.luaforge.net/papers/osbf-eddc.pdf>.
- [3] Cormack, G. The TREC 2006 Spam Filter Evaluation Track. Virus Bulletin. January 2007. <http://www.virusbtn.com/sba/2007/01/sb200701-trec>.
- [4] Yerazunis, W. S. Sparse binary polynomial hashing and the CRM114 discriminator. In 2003 Spam Conference, Cambridge, MA.
- [5] Yerazunis, W. S. CRM114 Revealed – Or How I learned To Stop Worrying and Trust My Automatic Monitoring Systems; this is the complete CRM114 manual available for free download at <http://crm114.sourceforge.net>.
- [6] Cormack, G. The TREC 2005 Spam Filter Evaluation Track. Virus Bulletin. January 2006. <http://www.virusbtn.com/sba/2006/01/sb200701-trec>.
- [7] Cormack, G. and Lynam, T. 2005. TREC 2005 spam track overview. <http://plg.uwaterloo.ca/~gvcormac/trecspamtrack05/trecspam05paper.pdf>.
- [8] Yerazunis, W. S. CRM114 Project. <http://crm114.sourceforge.net>.