Spam Filtering using a Markov Random Field Model with Variable Weighting Schemas

Shalendra Chhabra
UC Riverside
Riverside, California, USA
schhabra@cs.ucr.edu

William S. Yerazunis
MERL
Cambridge, Massachusetts, USA
wsy@merl.com

Christian Siefkes
GKVI*/ FU Berlin
Berlin, Germany
christian@siefkes.net

Abstract

In this paper we present a Markov Random Field model based approach to filter spam. Our approach examines the importance of the neighborhood relationship (MRF cliques) among words in an email message for the purpose of spam classification. We propose and test several different theoretical bases for weighting schemes among corresponding neighborhood windows. Our results demonstrate that unexpected side effects depending on the neighborhood window size may have larger accuracy impact than the neighborhood relationship effects of the Markov Random Field.

1 Introduction and Related Work

Spam filtering problem can be seen as a particular instance of the Text Categorization problem, in which only two classes are possible: *spam* and *legitimate email or ham*. In this paper, we present spam filtering based on the MRF Model with different weighting schemes of feature vectors for variable neighborhood of words. We present theoretical justification for our approach and conclude with results.

Recently Sparse Binary Polynomial Hash (SBPH), a generalization of the Bayesian method [9] and Markovian discrimination [10] have been reported for spam filtering. The classifier model in [10] uses empirically derived ad-hoc superincreasing weights. We develop more on [10], correlate it with MRFs, choose variable neighborhood windows for features using Hammersley-Clifford theorem [4] and present different weighting schemes for the corresponding neighborhood window. We tested these weighting schemes in CRM114 Discriminator Framework [3]. Our results reflect the effect of neighborhood relationship among features and provide evidence that this model is superior to existing Bayesian models used for spam filtering.

*The work of this author is supported by the German Research Society (DFG grant no. GRK 316).

2 Markov Random Fields

Let $\mathbf{F} = \{F_1, F_2 \ldots, F_m\}$ be a family of random variables defined on the discrete set of sites **S**, in which each random variable F_i takes a value f_i in the discrete label set **L**. The family **F** is called a random field. The notation $F_i = f_i$ denotes the event that F_i takes the value f_i and the notation $(F_1 = f_1, \ldots, F_m = f_m)$ denotes the joint event. A joint event (abbreviated as **F** = *f* where $f = \{f_1, \ldots f_m\}$) is a *configuration* of **F**, corresponding to a realization of the field. For the label set **L**, the probability that random variable F_i takes the value f_i is denoted by $P(F_i = f_i)$, and abbreviated as $P(f_i)$. The joint probability is denoted by $P(F = f) = P(F_1 = f_1, \ldots F_m = f_m)$ but abbreviated as *P(f)*. A site in the context of spam classification refers to the relative position of the word in a sequence and a label maps to word values. **F** is said to be a MRF on **S** with respect to a neighborhood *N* iff the following conditions hold:

1. $P(f) > 0, \forall f \in F \qquad (positivity)$
2. $P(f_i|f_{S-\{i\}}) = P(f_i|f_{N_i}) \qquad (Markovianity)$

where $\mathbf{S} - \{i\}$ is the set difference, $f_{S-\{i\}}$ denotes the set of labels at the sites in $\mathbf{S} - \{i\}$ and $f_{N_i} = \{f_{i'}|i' \in N_i\}$ stands for the set of labels at the sites neighboring i. When the positivity condition is satisfied, the joint probability of any random field is uniquely determined by its local conditional probabilities [2]. The Markovianity depicts the local characteristics of **F**. Only neighboring labels have direct interactions with each other. It is always possible to select sufficiently large N_i so that the Markovianity holds. Any **F** is a MRF with respect to such a neighborhood system.

3 Markov Random Fields and CRM114

We have implemented our scheme in CRM114 Discriminator Framework[1][3]. Like other binary document classi-

[1]The current version of CRM114 [3] is similar in spirit to MRF with lot of tweaks and hacks.

fiers, the CRM114 Discriminator associates a binary class value $S \in \{spam, nonspam\}$ with any given document $\omega = (\omega_1, \ldots, \omega_n)$. As a word context sensitive classifier CRM114 does not treat the input document ω as a bag of independent words, but rather considers all relations between neighboring words to matter, for neighborhoods with variable window size (for example: up to 3, 4, 5, 6 words etc.).

We now derive a *possible* MRF model based on this neighborhood structure, thereby casting the classification problem as a partial Bayesian inference problem. Our MRF model consists of a probability measure P defined on a set of configurations Ω. The elements $\omega \in \Omega$ represent all possible documents of interest, with the i-th component ω_i representing the i-th word or token. A random class function C is defined over Ω, $C : \Omega \rightarrow \{spam, nonspam\}$, such that C indicates the class of the document, and whose law is given by P.

In this framework the document classification problem can be treated as the problem of computing the probability $P(C(w) = s|\omega)$, or more precisely for MAP estimation (i.e. maximum a posteriori). The optimal class s^* is chosen as $s^* = \arg\max_{s \in S} P(C(w) = s|\omega)$.

Let us define a k-neighborhood consisting of k consecutive token positions in a document. The cliques are defined to be all possible subsets of a neighborhood. Thus if $\omega = (\omega_1, \ldots, \omega_n)$ is a document, then the first 3-neighborhood is $\{1, 2, 3\}$, and the associated cliques are $\{1\}$, $\{2\}$, $\{3\}$, $\{1, 2\}$, $\{1, 3\}$, $\{2, 3\}$. We assume that the measure P is an exponential form Markov Random Field conditioned on $C(w)$. This postulate is natural in view of the characterization by Hammersley and Clifford [4], in terms of conditional distributions on cliques. The functional form of P is therefore fixed as

$$P(\omega|C(\omega) = s) = Z_s^{-1} \exp\Big(\sum_i V_i^s(\omega_i) + \sum_{ij} V_{ij}^s(\omega_i, \omega_j) + \cdots + \sum_{i_1,\ldots,i_k} V_{i_1,\ldots,i_k}^s(\omega_{i_1}, \ldots, \omega_{i_k})\Big),$$

where Z_s is the appropriate normalizing constant which guarantees that $\sum_\omega P(\omega|C(\omega)) = 1$ when summed over all possible documents ω *(emails)*. By the Hammersley and Clifford [4] characterization of MRFs, the functions V are nonzero if and only if the indices form a clique.

We can identify the conditional MRF with a specific CRM114 instance by assigning the required V functions from the local probability formulas. For example,

$$V_{ij}^s(\omega_i, \omega_j) = \log \Pi_{ij}(\omega, s),$$

where $\Pi_{ij}(\omega, s) = (\textit{local probability for } (\omega_i, \omega_j), \textit{given } C(\omega) = s)$. $\Pi_{ij}(\omega, s)$ cannot be interpreted directly as conditional probabilities, however an easy product form solution is obtained i.e.

$$P(\omega|C(\omega) = spam) = Z_{spam}^{-1} \prod_{cliques\ c} \Pi_c(\omega, spam)$$

which is quite different from a naive Bayesian model. In the special case of neighborhoods with $k = 1$, this reduces to a naive Bayesian model. With this solution, Bayes' rule can be applied to obtain the class probability, given a document:

$$P(C(\omega) = spam|\omega) = \frac{P(\omega|C(\omega) = spam)P(spam)}{P(\omega)}.$$

In the Bayesian framework, the two unknowns on the right are $P(spam)$ (the prior) and $P(\omega)$. Normally, $P(\omega)$ is ignored, since it doesn't influence the MAP estimate, but it can also be expanded in the form

$$P(\omega) = Z_{spam}^{-1} \prod_{cliques\ c} \Pi_c(\omega, spam)P(spam) + Z_{nonspam}^{-1} \prod_{cliques\ c} \Pi_c(\omega, nonspam)(1 - P(spam))$$

While these Z^{-1} terms are unknown a possible value can be approximated by setting bounds based on the neighborhood structure of the clique.

4 Features Vectors in the Neighborhood

An incoming document *(email)* has to be broken into features to generate feature vectors. These feature vectors can be assigned weights so that the learning algorithms can compensate for the inter-word dependence. This is done by re-defining the learnable features to be both single tokens and groupings of sequential tokens, and by varying the length of the grouping window. By forcing the shorter groupings of sequential tokens to have smaller weightings, the inter-word dependence of natural language can be compensated that a Naive Bayesian Model normally ignores. The larger groupings have greater clique potentials and the smaller groupings have smaller clique potentials. In our scheme(s) feature vectors are assigned weights in a superincreasing fashion. The basic idea of superincreasing weights is to have a non linear classifier that is not bound by the limits of the Perceptron theorem [5]. This superincreasing classifier can cut the feature hyperspace along a curved (and possibly disconnected) surface, in contrast to a linear Bayesian classifier that is limited to a flat hyperplane. We now propose some weighting schemes with superincreasing weights.

Let *n-sequence* denote a feature containing n sequential nonzero tokens, not separated by placeholders; *n-term* denotes a feature containing n nonzero tokens, ignoring placeholders; e.g. "A B C" would be a 3-sequence and a 3-term, "A B ? D" would be a 3-term, but not a sequence. For each n-sequence, there are $Num(n) = 2^n - 2$ subterms (-2 because the empty feature (containing only placeholders) and the full n-sequence feature are both ignored). The number of subterms with k tokens is given by the binomial coefficient: $Num(n, k) = \binom{n}{k}$, for $0 < k < n$.

The weight $W(n)$ of a n-sequence should be larger than the weight of all subterms considered for this sequence i.e.

$$W(n) > \sum_{k=1}^{n-1} \left(\binom{n}{k} \times W(k) \right) \quad (1)$$

Minimum Weighting Sequences: The minimum weighting scheme for a superincreasing set of weights, can be evaluated as $W(n) = \sum_{k=1}^{n-1} \left(\binom{n}{k} \times W(k) \right) + 1$.

Exponential Weighting Sequences: For any given window length, the exponential weighting $W(k) = base^{k-1}$ can be evaluated as:

$$base^{n-1} > \sum_{k=1}^{n-1} \left(\binom{n}{k} \times base^{k-1} \right) \quad (2)$$

Applying the Binomial Theorem and setting a=1, b=$base$, we get: $2 \times base^n + 1 > (base + 1)^n$.

The resulting weighting schemas are shown in table 1.

n	MWS	ES
1	1	1
2	1, 3	1, 3
3	1, 3, 13	1, 5, 25
4	1, 3, 13, 75	1, 6, 36, 216
5	1, 3, 13, 75, 541	1, 7, 49, 343, 2401
6	1, 3, 13, 75, 541, 4683	1, 8, 64, 512, 4096, 32768

Table 1. Minimum & Exponential Weightings

5 Training and Prediction using CRM114

For evaluation, we used the same corpus and testing procedure as described in [10]. The 4147 messages (1397 spam, 2750 nonspam), were shuffled into ten different standard sequences. The last 500 messages of each standard sequence formed the *testing set* used for accuracy evaluation.

5.1 Models Tested

Four different weighting methodologies for differential evaluation of increasingly long matches were tested. These models correspond to increasingly accurate descriptions of known situations in the Markov Field Model.

The first model tested was Sparse Binary Polynomial Hashing (SBPH), which uses a constant weighting of 1.0 for all matches, irrespective of the length. With a window length of 1, SBPH is identical to the common Naive Bayesian model without discarding any features as "too uncommon" or "too ambivalent". Testing showed that best results occurred when the maximum window length was five tokens.

The second model tested was the exponential superincreasing model (ESM), which uses an empirically-derived formula that yields weights of 1, 4, 16, 64, 256, and 1024 for matches of one, two, three, four, five, and six words, respectively.

The third model tested was the Minimum Weighting System (MWS) model. This model uses the minimum weight increase necessary to assure that a single occurrence of a feature of length N words can override a single occurrence of all of its internal features (that is, all features of lengths $1, 2, \ldots, N-1$). This is a different notion than the superincreasing ESM model, and produces weights of 1, 3, 13, 75, 541 and 4683 as deduced above.

The fourth model tested uses a variable base to form an exponential series (ES), with a base chosen to assure that the values are always above the MWS threshold for any value of window length used. For our tests, we used a base of 8, yielding weights of 1, 8, 64, 512, 4096, and 32768. A summary of the term weighting length is shown in table 2.

Note that the ESM weight sequence is not larger than the MWS weight sequence for features of length 4 or longer. Thus, a long ESM-weighted feature may not be capable of overriding the weighting of it's internal subfeatures.

An example of the weighting model used for these tests is presented in the table 3. Here, the phrase "Do you feel lucky?" is broken into a series of subfeatures, and the respective weightings given to those subfeatures in each of SBPH, ESM, MWS, and ES are shown.

For each of these weighting schemes, these weights are used as multiplicative factors when calculating the local probability of each subfeature as it is evaluated in an otherwise-conventional Bayesian Chain Rule evaluation. The local probability of each class in CRM114 is given by

$$P = 0.5 + (((f_c * w) - (f'_c * w)) / (m * ((f_{totalhits} * w) + n))) \quad (3)$$

where f_c is the number of feature hits in this class, f'_c is the number of feature hits in the other class, $f_{totalhits}$ is the number of the total hits and w is the weight. In our implementation, $m = 16$ and $n = 1$. These experimentally determined constants generate probabilities close enough to 0.5 so as to avoid numerical underflow errors even over thousands of repeated applications of the Bayesian Chain Rule.

Model	Weighing Sequence
SBPH	1, 1, 1, 1, 1, 1
ESM	1, 4, 16, 64, 256, 1024
MWS	1, 3, 13, 75, 541, 4683
ES	1, 8, 64, 512, 4096, 32768

Table 2. A Summary of Tested Models with their Weighting Sequences

Text	SBPH	ESM	MWS	ES
Do	1	1	1	1
Do you	1	4	3	8
Do <*skip*>feel	1	4	3	8
Do you feel	1	16	13	64
Do <*skip*><*skip*>lucky?	1	4	3	8
Do you <*skip*>lucky?	1	16	13	64
Do <*skip*>feel lucky?	1	16	13	64
Do you feel lucky?	1	64	75	512

Table 3. Example Subphrases and Relative Weights with the Models Tested

5.2 Test Results and Discussion

In our tests we varied the window length parameter and the term weighting length table. All four models with a window length of 1 are exactly equivalent to each other and to a pure Bayesian model as postulated by Graham [1]. Each of these advanced models is also more accurate than a pure Bayesian model in every window length > 1.

The results are shown in the table 4. Figure 1 shows that even though ESM has a theoretical weakness due to coefficients less than the MWS for window lengths of four or greater, it has the best accuracy for all tested setups.

Size	1	2	3	4	5	6
SBPH	97.98	98.56	98.6	98.62	98.68	98.48
ESM	97.98	98.46	98.66	98.76	98.88	98.28
MWS	97.98	98.44	98.72	98.76	98.80	98.26
ES	97.98	98.48	98.58	98.32	98.80	98.16

Table 4. Accuracy (%) per 5000 test messages With Varying Window Sizes

6 Conclusion and Future Work

We have derived a generalized form of weighting schemas for classifiers with superincreasing weights. The weighting sequences define a set of clique potentials, where the neighborhood of a single word is given by the words surrounding it. For a window of size 2, "pairwise only dependence" is reflected [2].

Determining a generalized optimal window size may be the subject of future work. An interesting direction of future research is the combination of Sparse Binary Polynomial Hashing (SBPH) and feature weighting with other learning algorithms. Recently we have obtained significant improvements by combining SBPH with a variant of the *Winnow* algorithm [6] using TIES [8] [7].

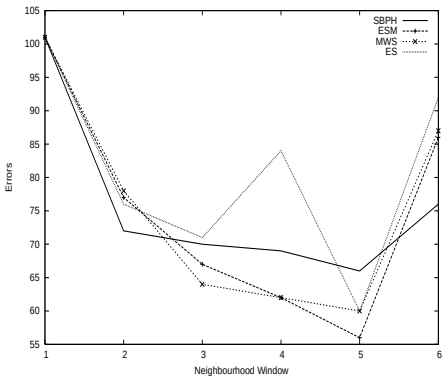

Figure 1. Errors in the Tested Models with Variable Neighborhood Windows

Acknowledgments

The authors are grateful to Laird Arnault Breyer, University of Lancaster, U.K. for fruitful discussions.

References

[1] A Plan for Spam. http://www.paulgraham.com/spam.html.
[2] J. Besag. Spatial Interaction and the Statistical Analysis of Lattice Systems. In *Journal of the Royal Statistical Society, Series B*, volume 36, pages 192–236, 1974.
[3] CRM114 Discriminator - The Controllable Regex Mutilator. http://crm114.sourceforge.net/.
[4] J. M. Hammersley and P. Clifford. Markov Field on Finite Graphs and Lattices. In *Unpublished*, 1971.
[5] Minsky and Papert. Perceptrons. 1969.
[6] Nick Littlestone. Learning Quickly When Irrelevant Attributes Abound: A New Linear-threshold Algorithm. *Machine Learning*, 2:285–318, 1988.
[7] C. Siefkes, F. Assis, S. Chhabra, and W. S. Yerazunis. Combining Winnow and Orthogonal Sparse Bigrams for Incremental Spam Filtering. In *Proceedings of ECML/PKDD 2004*, LNCS. Springer Verlag, 2004.
[8] TIES - Trainable Incremental Extraction System. http://www.inf.fu-berlin.de/inst/ag-db/software/ties/.
[9] W. S. Yerazunis. Sparse Binary Polynomial Hashing and the CRM114 Discriminator. In *MIT Spam Conference*, 2003.
[10] W. S. Yerazunis. The Spam Filtering Accuracy Plateau at 99.9 Percent Accuracy and How to Get Past It. In *MIT Spam Conference*, 2004.