Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

事前処理に k-means 法を利用したスパムフィルタの開発

北村祐貴† 狩野均†

近年、インターネット上のスパムメールによる被害が深刻な問題になっている。そのため、スパムメールと正規メールを精度よく分類するためのスパムフィルタが多数提案されている。本論文では、分類の前処理として k-means 法によるクラスタリングを行うことにより分類精度を向上させる手法を提案する。前処理後の分類方法としては、通常のベイジアンフィルタまたは SVM フィルタを用いる。まず、学習に使うメール集合に対して k-means 法を適用し、その後クラスタごとにどのような特徴が表れているかを分析する。その結果に基づいてクラスタごとにフィルタの調整を行うことで分類精度の向上を達成した。TREC Public Corpus を用いた評価実験から、本手法の有効性を確認することができた。

Development of Spam Filter using k-Means Clustering for Pre-Process

Yuuki Kitamura† and Hitoshi Kanoh†

In the recent years, the damage due to spam email has become a serious problem on the internet. Many spam filters have been proposed for classifying spam emails and regular emails with sufficient accuracy have been done. In this paper, we propose the method to improve the accuracy of classification by using a pre-clustering with k-means. A usual Bayesian filter or a usual SVM filter is used as a classification method after the pre-clustering. First, the k-means is applied to the mail set used for learning, and it is analyzed what feature appears for every cluster. Improvement in classification accuracy was achieved by adjusting a filter for every cluster based on the result of the analysis. The experiments using TREC Public Corpus proved that the proposed method is effective as a spam filter.

1. はじめに

近年、迷惑メール(スパム)の増加が大きな問題になっている[1],[2]。スパムとは受信者が望んでいないメールの総称で、受信者の望まない広告メールやチェーンメール、実際にコンピュータに被害をもたらすウィルスメールなど、様々な種類が存在している。大量のスパムは、受信者にとって必要なメール(正規メール)をスパムに埋もれさせて見つけ難くしてしまう。また、メールサーバーに大きな負荷がかかることで、不具合が発生したりする。これらのことから、スパムの除去は企業にとって予算を組んで専用の対策が必要になるほどに重要な問題となっている。

今日では、受信したメールをスパムと正規メールに自動的に分類するスパムフィルタの研究が盛んに行われている。中でも、過去に受信したメールのヘッダやアドレス、本文などからスパムと正規メールの特徴を学習し、その結果を利用して分類を行う学習型のフィルタが注目を集めている[1]-[11]。受信したメールがスパムであるか正規メールであるかという判断は使用者によって違うために他の種類のフィルタでは微妙な調整が難しいが、学習型のフィルタは使用者が過去に受信したメールから学習するため、個人に対応した分類を行うことができる。また、時間が経って送られてくるスパムの傾向が変わったとしても、最近のメールを学習することで変化にも柔軟に対応することができるため、性能を維持することができる[5]。

このような優れた特徴を持つ学習型フィルタは、これまでにも多くの手法が提案されている。統計的手法であるベイズ理論に着目し、メールに現れる単語の出現確率からスパムである確率を計算することで分類を行うベイジアンフィルタ[1]-[4]や、メール内の単語の出現頻度などを使ってメールをベクトル化し、2 クラス分類の手法である Support Vector Machine[8],[9]を使って分類を行う SVM フィルタ[10],[11]が有名である。これらの手法は、比較的分類精度の高い優秀な手法であるが、学習メール全体からすべてのメールに適用可能なルールを見つけ出すものであるため、一部のメールに必要な細かい分類が上手くいかないと考えられる。

そこで本研究では、従来手法によるスパムの分類に加えて、事前処理に k-means 法[12],[13]によるクラスタリングを行うことで分類精度の向上を目指す。本処理での分類には、一般的なフィルタであるベイジアンフィルタと SVM フィルタを使用する。まず、学習に使うメール集合に対して k-means 法を適用し、クラスタごとにどのような特徴が表れているかを分析する。その結果を元にしてクラスタごとにフィルタの調整を行い、分類精度の向上を目指す。

2 章では対象問題と従来研究を説明し、3 章では提案手法である k-means 法を利用したフィルタの説明を行う。4 章では、約 90000 データからなる TREC Public Corpus[14]を用いて、フィルタの性能実験とクラスタリング実験、従来手法との比較実験を行い、提案手法の有効性を確認する。

† 筑波大学
University of Tsukuba

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

2. 研究分野の概要

2.1 対象問題

(1) スパムの定義[2]

スパムの定義はいくつか提案されているが，本稿では「受信者が望んでいないメール」をスパムとする．同じメールでも受信者によってスパムかどうかの判断が異なる場合があるため，1 種類のフィルタで判断基準の違う個人に適した分類を行うのは非常に難しい．そのため，過去に受信したメールを使うことで個人に適応したフィルタを構成する，学習型のスパムフィルタが注目を集めている．

(2) フィルタの判定ミス[1]

フィルタの判定ミスは，正規メールをスパムと判定するミスと，それとは反対にスパムを正規メールと判定するミスの 2 通りの場合が考えられる．スパムフィルタの評価では，前者を誤検出(False positive)，後者を誤判定(False negative)と呼び，この 2 つを区別して考える．一般的に，誤判定に比べて誤検出の被害は甚大である．これは誤判定は分類の手間が増えるだけで済むが，誤検出は重要なメールに目を通せないという事態になるためである．そのため，いかに誤検出を抑えながら精度の高い分類を行うかという所に，スパムフィルタリングの難しさがある．

2.2 従来研究

2.2.1 ベイジアンフィルタを用いた手法[1][2]

ベイジアンフィルタは，統計的手法の 1 つであるベイズ理論を利用した学習型のスパムフィルタである．ベイズ理論は，過去に起きた事象の発生頻度(確率)から未来の出来事の発生頻度(確率)を予測するというものである．ベイズモデルはデータの数が多ければ多いほど，より確実な推測を引き出せるという特徴を持った自己修正型モデルである．学習に使用するデータの変化に応じて結果が変わることが，スパムフィルタにおける個人への柔軟な対応という形で有効に働いている．ベイジアンフィルタの動作は，事前学習とスパム判定の 2 つの部分に分けることができる．以下にそれぞれのアルゴリズムを示す．

(1) 事前学習

Step.1 コーパスの作成

収集した学習用メールデータをスパム集合と，正規メール集合に分類する．

Step.2 テキストをトークンへ分割

次に，アルファベット，数字，ダッシュ，アポストロフィをトークンの構成要素として，カンマ，ピリオド，スペースでテキストを分割してトークンを作成する．

Step.3 ハッシュテーブルの作成

各トークンがそれぞれの集合に現れた回数を数える．その後，トークンから出現回数を得ることのできるハッシュテーブルをそれぞれの集合に対して作成する．

Step.4 スパム確率のハッシュテーブルの作成

そのトークンが含まれるメールがスパムである確率を式 1 によって計算し，スパム確率を得る第三のハッシュテーブルを作成する．また，全体に 5 回以上現れないトークンに関しては計算から外している．

$$S = \frac{bad/nbad}{2\times good/ngood + bad/nbad} \tag{1}$$

S：あるトークンが含ま れているときに、その メールがスパムである 確率
$good$：あるトークンがスパ ムコーパスの中で現れ た回数　　$ngood$：正規メールの数
bad：あるトークンがスパ ムコーパスの中で現れ た回数　　$nbad$：スパムの数

ただし，2×good/ngood,bad/nbad が 1.0 を超えた場合その値を 1.0 として計算を行うとし，片方にしか現れない単語のスパム確率をそれぞれ 0.01，0.99 と設定した．式において *good* に 2 がかかっているが，これは誤検出回避の為のバイアスである．

‘(2) スパム判定

Step.1 新着メールの受信

判定を行うメールを取得する．

Step.2 テキストをトークンへ分割

事前学習のときと同様にテキストを分割してトークンを作成する．

Step.3 スパム確率の計算

新着メールのトークンで最も特徴的な 15 トークンを抽出する. ここで特徴的であるとはスパム確率が 0.5 から遠く離れていることとする．また，メール中のトークンで今まで一度も出てきていないもののスパム確率は 0.4 とした．これはスパムに含まれる単語が偏っていることによる．次に，抽出した 15 トークンのスパム確率からそのメールがスパムである確率を以下の式に基づいて計算する．

$$M = \frac{\prod_{i=1}^{15} x_i}{\prod_{i=1}^{15} x_i + \prod_{i=1}^{15} (1 - x_i)} \tag{2}$$

M：そのメールがスパム である確率
x_i：i番目に特徴的なトーク ンが含まれているメー ルがスパムである確率

Step.4 結合確率による判定

得られた M がしきい値である 0.9 以上のメールをスパムと判定する．

2.2.2 SVM フィルタを用いた手法[7],[10]

SVM フィルタは，教師あり学習を用いる識別手法の 1 つである SVM(Support Vector

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

Machine)を利用したスパムフィルタである．SVM は，学習データを用いて 2 クラスのパターン識別機を構成することで分類を行う．超平面上における識別境界を，境界に最も近い学習データとの距離(マージン)が最大になるように設定することで，高い分類精度を持つ．SVM フィルタの動作は，事前学習とスパム判定の 2 つの部分に分けることができる．以下にそれぞれのアルゴリズムを示す．

(1)事前学習

Step.1 教師ラベルを作成

最初に，メールが正規メールであるかスパムであるかを表す教師ラベルを作成する．

メールが正規メールであるならば 1，スパムであるならば-1 をそのラベルとする．

Step.2 テキストをトークンへ分割

次に，アルファベット，数字，ダッシュ，アポストロフィをトークンの構成要素として，カンマ，ピリオド，スペースでテキストを分割してトークンを作成する．

Step.3 ハッシュテーブルの作成

各トークンが現れた回数を数え，トークンから出現回数を得ることのできるハッシュテーブルを作成する．

Step.4 メールデータをベクトル化

その中から出現回数の多い順に 500 個のトークンを選び，すべてのメールに対し，それらのトークンがメール中に現れる回数を要素とするベクトルを作成する．ベクトルのノルムが 1 となるように正規化を行い，メールのデータベクトルとする．

(例)集合内で「you」が最も出現回数が多いとき，x_1にはベクトル化したいメール内で you が表れた回数を入れる．

$$x = \frac{(z_1, \cdots, z_{500})}{|(z_1, \cdots, z_{500})|} \tag{3}$$

z_i: i番目に出現頻度の多い　トークンの出現回数

Step.5 識別器を構成[8],[9]

式 4 は SVM の識別関数を表している．この式中のパラメータ w と b を，式 5 の条件を満たしながら調整することで識別器を構成する．

パラメータ w と b はラグランジュ乗数λを利用することで式 6,7 のように表すことができる．式 8 の更新式に従って最適なλを求めることで，識別関数 $f(x)$を求める．

$$f(x) = sign(w^t x + b) \tag{4}$$

$sign\ (x)$: 符号関数
x：入力ベクトル
w, b : パラメータ

$$g(x_i) = w^t x_i + b \begin{cases} \geq 1 & if\ x_i \in X_1 \\ \leq -1 & if\ x_i \in X_2 \end{cases} \tag{5}$$

$x_i\ (i = 1,2,\ldots,n)$: 入力ベクトル

$$w = \sum_{i=1}^{n} \lambda_i y_i x_i \tag{6}$$

$$b = y_s - w^t x_s \tag{7}$$

$$\lambda_i(t+1) = \lambda_i(t) + \eta(1 - \sum_{j=1}^{n} \lambda_j y_i y_j x_i^t x_j) \tag{8}$$

$x_i (i = 1,2,\ldots,n)$: 入力ベクトル
$y_i (i = 1,2,\ldots,n)$: 入力ベクトルの教師信号
$\lambda_i (\lambda_i \geq 0, i = 1,2,\ldots,n)$: ラグランジュ乗数
x_s: サポートベクトル
η : 学習係数

(2)スパム判定

Step.1 新着メールの受信

判定を行うメールを取得する．

Step.2 テキストをトークンへ分割

事前学習のときと同様にテキストを分割してトークンを作成する．

Step.3 メールデータをベクトル化

事前学習と同様にベクトル化を行う．

Step.4 識別関数による判定

事前学習で求めた識別関数にベクトルを代入し，g(x)が出力する値が 0 より大きければ正規メール，そうでないならばスパムと判定する．

2.2.3 従来手法の問題点

従来手法のベイジアンフィルタは，誤検出を抑えるための工夫として至る所でバイアスが掛けられており, そのせいで少し判別精度が落ちてしまっている. 反対に, SVM フィルタはバイアスがかかっていない分判別精度は高いが，バイアスが掛かっていないために誤検出が多くなってしまっている．

3. 提案手法

3.1 提案フィルタの概要

本研究では，事前処理にクラスタリングを利用したスパムフィルタを提案する．こ

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

れは学習データ全体からスパムを判別するルールを見つけるよりも，性質の比較的似ている集合に対して調整されたフィルタを使った方が細かい分類ができるという考えに基づいている．学習データに対しクラスタリングを行い，クラスタごとに調整されたフィルタを新着メールに対して適用することで，分類精度の高いフィルタを目指す提案フィルタは事前学習とスパム判定の二つの部分によって構成されている．

3.2 事前学習

Step1 学習メールをベクトル化

SVM フィルタと同じ方法を使い，メールデータのベクトル化を行う．

Step2.データベクトルをクラスタリング

作成したデータベクトル集合に対して，分割型クラスタリング手法の一つである *k*-means 法[12],[13]を適用する．データベクトル間の類似度はベクトル同士の余弦で表すとする．*k*-means 法によるクラスタリングは初期状態に強く影響されることが知られているため，本稿では 10 回 *k*-means 法を適用し，それぞれのベクトルが所属しているクラスタの重心ベクトルとの類似度の合計が，一番大きかったものをクラスタリング結果として使用した．

Step3.クラスタごとに調整されたフィルタの学習

得られたクラスタに所属するメールを，精度良く判別できるように調整したフィルタの学習を行う．ここで使うフィルタはベイジアンフィルタ[1],[7]と SVM フィルタ[10],[7]とする．

3.3 スパム判定

Step1.新着メールの所属クラスタを判別

まず，学習データから作った辞書を使用して新着メールからデータベクトルを作成する．次に，クラスタの重心ベクトルとの類似度を計算し，一番類似度の高い重心ベクトルを持つクラスタを，新着メールの所属クラスタに決定する．

Step2.クラスタごとに調整したフィルタを適用

所属クラスタのフィルタで新着メールの判定を行う．

Step3.結果を出力

フィルタの判定結果を提案フィルタの判定として出力する．

4. 実験

実験に用いるデータセットは，約 90000 通のメールデータを含む”2005 TREC Public Spam Corpus”[14]から，実験内容に応じて英文メールを抽出して作成した．

4.1 フィルタの性能実験

ベイジアンフィルタと SVM フィルタの性能を確認するために実験を行った．学習データ数による性能の変化を確認するために,学習データ数を 1000 刻みで変えて実験を行った．評価に使うメールは学習に使用しなかったメールの中から，ランダムに 10000 通を選んで使用した．実験をそれぞれ 5 回行い，結果の平均を表 1 に示す．表 1 を見ると，ベイジアンフィルタ，SVM フィルタ両方の性能が学習メール数の増加と共に良くなっている．また，学習データ数 2000 程度で正解数の変化が少なくなっていることから，これらのフィルタが十分な性能を発揮するためには，最低でも 1000 個以上の学習データが必要ということがわかる．ベイジアンフィルタの誤検出数は学習データ数の変化にあまり影響を受けていないが, SVM フィルタの値は学習データ数の増加に共に小さくなっていく．学習データ内のスパムの割合による性能の変化を確認するために，学習データ数 5000 で，含まれるスパムの割合を 20%刻みで変えて実験を行った．評価に使うメールは学習に使用しなかったメールの中から，ランダムに 10000 通を選んで使用した．実験を 5 回行った結果の平均値を表 2 に示す．表 2 を見ると，スパム率がどちらかに偏っていた方がフィルタの正解率が高くなっていることがわかる．SVM フィルタでは誤検出と誤判定，共にその傾向が見て取れるが，ベイジアンフ

表 1 学習データ数による性能の変化

学習データ数		1000	2000	3000	4000	5000
ベイジアンフィルタ	正解数	8797	9106	9088	9153	9175
	誤検出	5.4	7.8	2.6	4.6	2.8
	誤判定	1197	886	908	842	821
SVM フィルタ	正解数	9618	9692	9728	9775	9765
	誤検出	229	159	152	117	114
	誤判定	152	148	119	107	120

表 2 スパム率を変えたときのフィルタの性能の変化

スパム率		20%	40%	60%	80%
ベイジアンフィルタ	正解数	9375	9182	9256	9479
	誤検出	2.2	4.2	9.2	47
	誤判定	622	813	734	473
SVM フィルタ	正解数	9811	9795	9767	9813
	誤検出	102	102	117	99
	誤判定	86	102	115	86

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

ィルタの場合はスパム率の増加と共に誤検出が大きく増加している．これらの結果から，SVM フィルタにとってスパム率が偏っている方が性能が良くなるが，ベイジアンフィルタの場合はスパム率が高くなると性能が大きく下がってしまうということがわかった．十分に性能を発揮したときの両フィルタの性能を比べると，正解数では SVM フィルタが大きく上回っているが，誤検出数を比較するとベイジアンフィルタの方がかなり少ない．スパムフィルタには，誤検出の少なさが求められるので，スパムフィルタとしての性能は，ベイジアンフィルタの方が優れているといえる．

4.2 クラスタリング実験

4.2.1 クラスタ分割数の決定

クラスタ分割数を変えながら k-means 法によるクラスタリングを行い，分割数を変えたときにどのような特徴を持ったクラスタに分割されていくかを分析する．実験には，ランダムに選んだ 5000 通のメールデータを使用した．分割したクラスタ内の正規メール数とスパム数を表 3 に示す．また，クラスタ分割数を増やした際にメールの所属がどのクラスタに移動するかを図 1 に表した．クラスタ分割数を増やしていった場合分割数 3 で，0 番のクラスタのようなスパムと ham が混在していて後々まで分割されないクラスタと，1 番のクラスタのような ham だけで構成されたクラスタが生まれる．これらのクラスタは，分割数を増やしても後々まで崩れずに残る．これ以降の分割の様子を見てみると，スパムの割合が高く少ないが無視するには多い程度の ham で構成されたクラスタが生まれるようになる．細かすぎるクラスタリングは一般性を失ってしまう可能性があるため，大まかで比較的特徴を持ったクラスタに分割できる分割数 3 をクラスタ分割数とする．これ以降，特徴を持ったクラスタの視点で分析を行うため，分割数 3 のときのクラスタを以下のような名称で呼ぶ．

Ham クラスタ:所属メールが ham だけで構成された 1 番クラスタのようなクラスタ．
混在 A：一番数が多くて分割されやすい 2 番クラスタのようなクラスタ．
混在 B：一度分割されると中々分割されない，0 番クラスタのようなクラスタ．

表 3 クラスタ分割数を変えたときのクラスタ内のメールの分布

		クラスタ番号									
		0		1		2		3		4	
クラスタ分割数		ham	spam	ham	spam	ham	spam	ham	spam	ham	Spam
	2	744	485	1807	1964						
	3	607	458	1200	0	744	1991				
	4	594	431	1196	0	24	698	737	1320		
	5	591	419	1196	0	11	173	14	532	739	1325

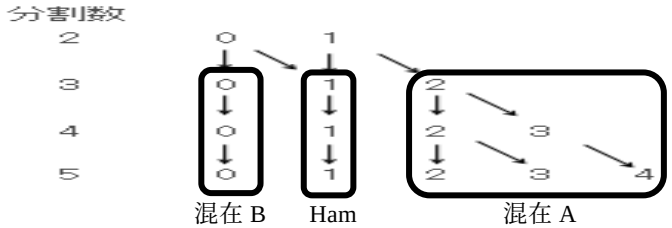

図 1 分割数を変えたときのクラスタ分割の様子

4.2.2 ベイジアンフィルタ

それぞれのクラスタに属しているメールが，ベイジアンフィルタによってどのように判定されているかを確認する．学習には，ランダムに選んだ 5000 通のメールデータを使い，評価には学習に使わなかった中から 10000 通を選んで使用した．これ以降に行う実験はこのデータセットを使用して行うとする．クラスタ分割数は 3 とし，学習データすべてを学習に使った場合と，クラスタ内のデータのみを学習に使った場合のフィルタの性能を，各クラスタごとに整理して表 4 に示し，学習データ内の ham と spam の数をクラスタごとに整理して表 5 に示す．また，学習データすべてを学習に使った場合のフィルタの出力値の分布を，ham と spam に分けてクラスタごとにまとめたものが表 6 である表 4 よりわかる通り，ham クラスタに所属しているメールの判定はどちらの学習方法でもすべて正解している．学習データの違いにより最も変化が大きいのは混在 A である．クラスタ内のデータのみで誤検出と正解率が増加するこの傾向は，表 5 で確認できる通り，学習データのスパム率が高くなったためと考えられる．また，誤検出に注目すると，すべて混在 A の中で起こり，他の 2 つのクラスタでは起こっていない．表 6 を見ると，混在 A では ham の分布にちらばりがあり，0.1 以上の出力値を持つメールが存在しているが，他の 2 つのクラスタではすべての出力値が 0.1 以下に収まっている．このことから，誤検出の起こりやすい混在 A のしきい値が 0.9 となるのは妥当であるが，混在 B のしきい値を下げることで誤検出を抑えたまま正解率を上げることができると考えられる．

表 4 クラスタごとのベイジアンフィルタの判定結果

	すべてのデータ				クラスタ内のデータ			
	正解数	FP	FN	正解率	正解数	FP	FN	正解率
ham	2390	0	0	100	2390	0	0	100
混在 A	4709	3	658	87.69	4806	9	555	89.50
混在 B	2100	0	140	93.75	2092	0	148	93.39
合計	9199	3	798	91.99	9288	9	703	92.88

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

表 5 クラスタ内のメールの分布

	ham	spam	総数	spam 率
ham	1200	0	1200	0
混在 A	744	1991	2735	72.80
混在 B	607	458	1065	43.00
全体	2551	2449	5000	48.98

表 6 ham とスパムのベイジアンフィルタの出力値の分布

ham				spam			
区間	混在 B	ham	混在 A	区間	混在 B	ham	混在 A
0.0-0.1	1246	2390	1410	0.0-0.1	128	0	541
0.1-0.2	0	0	0	0.1-0.2	4	0	22
0.2-0.3	0	0	0	0.2-0.3	0	0	17
0.3-0.4	0	0	0	0.3-0.4	1	0	9
0.4-0.5	0	0	0	0.4-0.5	3	0	8
0.5-0.6	0	0	0	0.5-0.6	1	0	13
0.6-0.7	0	0	1	0.6-0.7	0	0	9
0.7-0.8	0	0	0	0.7-0.8	1	0	18
0.8-0.9	0	0	0	0.8-0.9	2	0	21
0.9-1.0	0	0	3	0.9-1.0	854	0	3298
合計	1246	2390	1414	合計	994	0	3956

表 7 クラスタごとの SVM フィルタの判定結果

	すべてのデータ				クラスタ内のデータ			
	正解数	誤検出	誤判定	正解率	正解数	誤検出	誤判定	正解率
ham	2388	2	0	99.91	2390	0	0	100
混在 A	5188	90	92	96.61	5194	80	96	96.72
混在 B	2183	32	25	97.46	2183	35	22	97.46
合計	9759	124	117	97.59	9767	115	118	97.67

4.2.3 SVM フィルタ

それぞれのクラスタに属しているメールが，ベイジアンフィルタによってどのように判定されているかを確認する．学習には，ランダムに選んだ 5000 通のメールデータを使い，評価には学習に使わなかった中から 10000 通を選んで使用した．クラスタ分割数は 3 とし，学習データすべてを学習に使った場合と，クラスタ内のデータのみを学習に使った場合のフィルタの性能を，各クラスタごとに整理したものを表 7 に示す．両方の場合で学習をおこなったフィルタの出力値を，ham と spam に分けて 0.3 刻みの度数分布表にしたものが表 8 である学習データをクラスタ内のみに変えた場合，どの

表 8 学習データを変えたときの ham とスパムの SVM フィルタの出力値の分布

すべてのデータ						クラスタ内のデータ					
	ham	混在 A		混在 B			ham	混在 A		混在 B	
区間	ham	ham	spam	ham	spam	区間	ham	ham	spam	ham	spam
-3	0	4	2986	4	634	-3	0	2	2597	0	20
-2.7	0	1	171	0	76	-2.7	0	1	281	0	28
-2.4	0	2	149	0	67	-2.4	0	4	204	1	74
-2.1	0	0	113	2	50	-2.1	0	2	185	0	114
-1.8	0	5	114	1	27	-1.8	0	3	153	0	166
-1.5	0	9	91	1	45	-1.5	0	4	120	1	232
-1.2	0	9	68	1	26	-1.2	0	6	106	0	124
-0.9	0	3	72	0	14	-0.9	0	11	94	5	103
-0.6	1	13	55	3	20	-0.6	0	13	63	8	63
-0.3	0	22	23	14	8	-0.3	0	11	28	11	30
0	1	22	22	10	2	0	0	23	30	9	18
0.3	0	19	19	10	5	0.3	1	33	27	26	10
0.6	2	28	16	8	3	0.6	2	33	17	41	5
0.9	2	39	11	12	4	0.9	12	44	12	59	3
1.2	4	95	11	15	4	1.2	1496	98	6	70	2
1.5	9	58	8	11	0	1.5	879	114	11	83	1
1.8	16	54	3	15	1	1.8	0	121	7	76	1
2.1	11	85	2	23	3	2.1	0	146	3	76	0
2.4	15	98	2	23	1	2.4	0	140	3	78	1
2.7	24	80	7	32	2	2.7	0	136	4	75	0
3	22	130	2	24	1	3	0	88	3	88	0
次の級	2283	638	11	1037	1	次の級	0	381	3	539	0

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

クラスタでも誤検出の数と正解率が少し良くなっていることが表 7 からわかる．表 8 を見ると，クラスタ内のみを学習に使った場合，全部を学習に使った場合に比べて分布の範囲が狭い．これにより，性能が向上したものと考えられる．クラスタ分割数を更に増やすことで性能の改善を行える可能性がある．

4.3 フィルタの調整

4.3.1 クラスタごとのフィルタ調整

これまでの実験結果に基づいて，各クラスタに使用するフィルタの調整を行う．単に分類精度を高めるのではなく，誤検出をできるだけ抑えることで有効性の高いスパムフィルタとすることを目的にして調整を行う．

(1)ham

表 4 の実験結果を見ると，ham 集合に含まれているメールの判定ミスは 0 である．そのため，誤検出数，分類精度の観点からも，所属クラスタが ham 集合と判定されたメールの判定は，通常のベイジアンフィルタで行う．

(2)混在 A

表 4，表 7 の実験結果から，このクラスタは 3 つのクラスタの中で一番大きいが，どのフィルタでも一番正解率が低く誤検出が多いことから分類の難しいメールが集まっていると言える．4.2.2 で述べたが，表 5 からわかる通りこのクラスタのスパム率が高いのが，分類精度が悪くなっている原因と考えられる．そこで，クラスタ内の学習データのスパム率を下げるために，誤検出が発生し難くて分類しやすい正規メールの集合である ham クラスタを加えて学習を行い，判定を行う．さらに，ham クラスタと混在 B で学習を行った場合について行い，誤検出が少なかったベイジアンフィルタに関しては，しきい値を 0.1 に変えたものも実験を行った．これらの結果を比較しやすいように整理して表 9 に示す. 表 9 を見ると, 誤検出が小さく正解率が高くなるのは，ham クラスタと混在 A を学習に使用したベイジアンフィルタであることがわかる．このフィルタのしきい値を 0.9 にしたときは通常の学習にすべてのデータを使用したときと同程度の誤検出数に抑えたまま，分類精度を上げることができる．また，しきい値を 0.1 に変えると，誤検出の増加を最小限に抑えながら大きく性能を向上させることができる．混在 A と ham クラスタを使って学習を行い，しきい値を 0.1 と 0.9 にした場合の二つのベイジアンフィルタを，混在 A のフィルタとする．

(3)混在 B

表 4, 表 7 よりこのクラスタに対するフィルタの性能は, どちらのフィルタでも ham クラスタに次いで 2 番目に高い．表 5 を見るとクラスタ内の spam 率は低いが，精度を保ったまま学習数を増やすために ham クラスタを学習データに加えて実験を行う．この結果は表 10 に示す．また，このクラスタでは誤検出の発生が起きにくいことがわかっているため，誤検出が少ないフィルタに関しては，ベイジアンフィルタのしきい値を 0.9 から 0.1 に下げた場合の実験も行った．表 10 より，誤検出が低くて正解率が高かったのは，混在 B と ham クラスタを学習に使ったフィルタと，しきい値を 0.1 に下げたフィルタである．この二つは正解率に大きな差がないので，誤検出を抑えることを重視してしきい値 0.1 のベイジアンフィルタを混在 B のフィルタとする．

表 9 混在 A に対して調整したフィルタの性能

フィルタの種類	ベイジアンフィルタ						SVM フィルタ	
学習データ	A+ham		B+ham	すべて		クラスタ内	すべて	クラスタ内
しきい値	0.9	0.1	0.9	0.9	0.1	0.9	0	0
spam 率	45%	45%	16%	43%	43%	74%	43%	74%
誤検出	3	6	64	3	4	9	90	80
誤判定	480	356	1358	658	542	555	92	96
正解数	4887	5008	3948	4709	4824	4806	5188	5194
正解率	91.00%	93.26%	73.52%	87.69%	89.83%	89.50%	96.61%	96.72%

表 10 混在 B に対して調整したフィルタの性能

フィルタの種類	ベイジアンフィルタ						SVM フィルタ	
学習データ	A+ham	B+ham		すべて		クラスタ内	すべて	クラスタ内
しきい値	0.9	0.9	0.1	0.9	0.1	0.9	0	0
spam 率	45%	16%	16%	43%	43%	74%	43%	74%
誤検出	2	1	1	0	0	0	32	35
誤判定	758	125	121	140	128	148	25	22
正解数	1480	2114	2118	2100	2112	2092	2183	2183
正解率	66.07%	94.38%	94.55%	93.75%	94.29%	93.39%	97.46%	97.46%

4.3.2 調整したフィルタの性能

フィルタを調整した結果を表 11 に示す．このフィルタを使って実験を行い，その結果と他のフィルタとの比較結果を表 12 に示す．表 12 より，提案フィルタはベイジアンフィルタと同程度の誤検出数に抑えながら，約 2%の性能向上を達成した．また，しきい値を変えることで，誤検出の発生を抑えながら，さらに 1%の性能向上を見込める．SVM フィルタと比較すると正解率は見劣りするが，SVM フィルタは誤検出数が大きいため，提案フィルタの方がスパムフィルタとして有用であるといえる．これ

Vol.2009-MPS-76 No.12
Vol.2009-BIO-19 No.12
2009/12/17

表 11 各クラスタに対して調整したフィルタ

	フィルタの種類	学習データ	しきい値
ham	ベイジアンフィルタ	すべて	0.1
混在 A	ベイジアンフィルタ	混在 A＋ham	0.1 or 0.9
混在 B	ベイジアンフィルタ	すべて	0.1

表 12 調整した提案フィルタと他のフィルタとの比較

フィルタの種類	提案フィルタ		ベイジアンフィルタ			SVM フィルタ	
学習データ	調整	調整	すべて	クラスタ内	すべて	すべて	すべて
しきい値	調整	0.1	0.9	0.9	0.1	0,0	0.0
誤検出	3	6	3	9	4	124	115
誤判定	608	484	798	703	669	106	118
正解率	93.89	95.1	91.99	92.88	93.27	97.59	97.67

らの結果から，事前処理にクラスタリングを利用する本手法の有効性を確認できた．

5. おわりに

本研究では，事前処理に *k*-means 法を利用したスパムフィルタの開発を行った．TREC Public Corpus から作成したデータセットを用いて，*k*-means 法によるクラスタリング実験を行い，分割数の検討と各クラスタについての考察を行った．予備実験の結果，クラスタ分割数 3 が適していることがわかった．

次に，ベイジアンフィルタ，SVM フィルタの性能実験を行い，それぞれのクラスタに注目して分析を行った．その結果，正解率では SVM フィルタの方が優れているが同時に誤検出も多いため，スパムフィルタとしてはベイジアンフィルタの方が有用であることがわかった．

この結果に基づいてフィルタの調整を行い，従来手法との比較実験を行ったところ，正解率では SVM フィルタに届かなかったが，誤検出をベイジアンフィルタと同程度に抑えたまま約 2%正解率を向上させることができた．さらに，しきい値を 0.1 に引き下げることで誤検出の増加を最小限に抑えたまま，通常のベイジアンフィルタに比べて約 3%正解率を向上させることができた．

本研究では主にスパムと ham の割合が同程度であるデータセットを使って評価を行ったため，スパム率が高い状況でも提案フィルタが有効に働くかは検討の余地がある．ベイジアンフィルタはスパム率が高いデータに対しては誤検出が大幅に増えるため，スパム率の増加に対する影響が少ない SVM フィルタを上手く組み合わせることで高い性能が期待できる．今後はスパム率が高いデータに対して実験を行い，提案フィルタの有効性を確認したい．

参考文献

1) P.Graham,”A PLAN FOR SPAM”,
http://www.paulgraham.com/spam.html
2) 田端：SPAM メールフィルタリング:ベイジアンフィルタの解説，情報の科学と技術, Vol.56, No.10(20061001) pp．464-468
3) P.Graham,”Better Bayesian filtering”,
http://www.paulgraham.com/better.html
4) 佐々木：文書分類を用いたスパムメール判定手法，情報処理学会研究報告, Vol.2004, No.93(20040916) pp．75-82
5) 鳴海：複数のクラシファイアを用いた状況変化に対応可能なオンラインスパムフィルタリングシステム，電子情報通信学会技術研究報告 Vol.106, No.605(20070308) pp．1-6
6) 山本：利用者の特徴を考慮したメール分類機構の組み合わせ法の評価，情報処理学会研究報告, Vol.2007, No.16(20070301) pp．189-194
7) 新美:遺伝的プログラミングによるテキスト分類アルゴリズムの組み合わせ,第 21 回ファジィシステムシンポジウム論文集,pp147-162,2005
8) 津田:サポートベクターマシンとは何か,電子通信学会誌,83(6),pp460-466,2000
9) 栗田多喜男:サポートベクターマシン入門
http://www.neurosci.aist.go.jp/~kurita/lecture/svm/svm.html
10) 米倉:Support Vector Machine を用いた電子メールの自動分類,情報処理学会研究報告,No.083-004,pp.19-26,2003
11) Drucker,H:Support Vector Machines for Spam Categorization,IEEE Trans.Nerural Networks,Vol.10,No.5,PP.1048-1054,1999
12) 神嶌:データマイニング分野のクラスタリング手法(1)，人工知能学会誌，vol18, No.1, pp. 59-65, 2003
13) 神島:データマイニング分野のクラスタリング手法(2)，人工知能学会誌, Vol.18, No.2,pp．170-176,2003
14) 2005 TREC Public Spam Corpus
http://plg.uwaterloo.ca/~gvcormac/treccorpus/