クラスタリングを利用したベイジアン spam メールフィルターの改良

2001MT003 新井 雅典
指導教員　河野 浩之 教授

1　はじめに

David Mertz が示すこれまでの spam メールフィルター [1] の識別精度からベイジアンフィルター [2] に注目する．ベイジアンフィルターには無作為な spam メールの学習によって識別精度が低下する問題がある．本研究ではクラスタリングを利用した改善策を提案し，実装し，評価していく．

2　ベイジアンフィルターの問題点

識別精度が低下する理由に次の事がいえる．類似した spam メール群 A で高い spam 単語確率を示す単語 X も無作為な spam メール群ではその確率が低下する．図 1 で示すように単語 X の割合が低下するからである．spam 単語確率の低下が spam メール確率を低下させて識別精度が低下する．

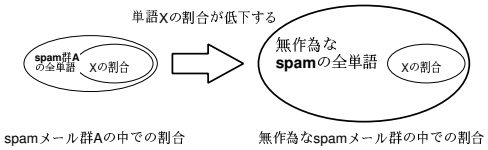

図 1　単語の占める割合の変化

3　クラスタリングを利用した改善策

本研究では図 2 のように予め類似した spam メール群にクラスタリングし，各クラスタ毎に spam メール確率を求める改善策を提案する．

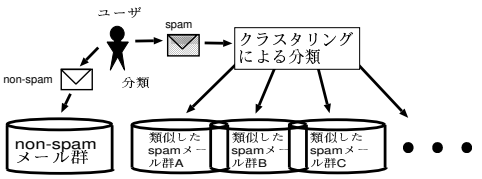

図 2　学習用 spam メール群のクラスタリング

クラスタリングは被クラスタリングメールの各クラスタへの所属率として定義するクラスタメール確率をもとに行う．クラスタメール確率は被クラスタリングメール中の単語の各クラスタへの出現率として定義するクラスタ単語確率の複合確率で計算する．後のクラスタリングのアルゴリズム中で各々の確率の詳しい計算式を示す．

ベイジアンフィルターが spam メールを識別できる為には 40 通〜60 通の学習が必要である．しかし各クラスタがこのような spam メール数を持っているとは限らないのでクラスタリングの条件であるクラスタメール確率の閾値をクラスタ中の spam メール数によって適当な値にしなければならない．次に示すクラスタリングのアルゴリズムで設定したクラスタメール確率の閾値は限られた学習用 spam メール群から求めた仮の閾値である．

クラスタリングのアルゴリズム

入力：「最初のクラスタとなる spam メール」,「学習用 spam メール群」,「学習用 non-spam メール群」

出力： 作られた全クラスタ

step1: 作られるクラスタを CL_j(j は 1〜作られるクラスタ数) とし,「最初のクラスタとなる spam メール」を CL_1 として登録

step2: 「学習用 spam メール群」の各 spam メールを $spam_k$(k は 1〜全学習用 spam メール数) とする

step3: クラスタリングされていない $spam_k$ を選ぶ．全 $spam_k$ がクラスタリングされていたら，全クラスタを出力して終了

step4: $spam_k$ の各 CL_j へのクラスタメール確率を $PM(spam_k, CL_j)$ とし，$PM(spam_k, CL_j)$ が未計算の CL_j を選ぶ．全 CL_j への $PM(spam_k, CL_j)$ が計算済みであったら，各 CL_j に以下の条件で $spam_k$ をクラスタリングして step3 に戻る
(CL_j中の $spam$ メール数 $<$ 6 通) の場合は $PM(spam_k, CL_j) >= 0.75$ なら CL_jへ $spam_k$ をクラスタリング
(18 通 $>$ CL_j中の $spam$ メール数 $>=$ 6 通) の場合は $PM(spam_k, CL_j) >= 0.80$ なら CL_j へ $spam_k$をクラスタリング
(50 通 $>$ CL_j中の $spam$ メール数 $>=$ 18 通) の場合は $PM(spam_k, CL_j) >= 0.85$ なら CL_j へ $spam_k$をクラスタリング
(CL_j中の $spam$ メール数 $>=$ 50 通) の場合は $PM(spam_k, CL_j) >= 0.90$ なら CL_jへ $spam_k$ をクラスタリング
どのクラスタの閾値も超えない場合は $spam_k$ を新しいクラスタとして登録する

step5: $spam_k$ 中の単語を抽出して W_i(i は 1〜全単語種数) とする

step6: 各 W_i の CL_j へのクラスタ単語確率を $PW(W_i, CL_j)$ とし，$PW(W_i, CL_j)$ が未計算の

W_i を選ぶ．全 W_i の $PW(W_i, CL_j)$ が計算済みなら，その上位 15 個で計算式 (1) のように $PM(spam_k, CL_j)$ を計算し，step4 に戻る

$$SC = P(W_1, CL_j) * P(W_2, CL_j) * \cdots * P(W_{15}, CL_j))$$
$$NSC = (1 - P(W_1, CL_j)) * (1 - P(W_2, CL_j)) * \cdots * (1 - P(W_{15}, CL_j)))$$
$$PM(spam_k, CL_j) = \frac{SC}{SC + NSC} \quad (1)$$

step7: 「学習用 non-spam メール群」の non-spam メール数を $nm(G)$，CL_j 中の spam メール数を $nm(CL_j)$，「学習用 non-spam メール群」中の W_i の出現回数を $nw(W_i, G)$，$spam_k$ 中の W_i の出現回数を $nw(W_i, spam_k)$，CL_j 中の W_i の出現回数を $nw(W_i, CL_j)$ とし，数式 (2) ように $PW(W_i, CL_j)$ を計算し，step6 に戻る

$$MCL = \frac{nw(W_i, CL_j) + nw(W_i, spam_k)}{nm(CL_j) + 1}$$
$$MG = \frac{nw(W_i, G)}{nm(G)}$$
$$PW(W_i, CL_j) = \frac{MCL}{MG + MCL} \quad (2)$$

以上のアルゴリズムにより類似した spam メール群のクラスタが複数できる．この各クラスタ毎に受信メールの spam メール確率を計算し，その最上位の値を最終的な spam メール確率として以下のように識別する．

最終的な spam メール確率 >= 0.9 の場合
spam メールと識別
最終的な spam メール確率 < 0.9 の場合
non-spam メールと識別

最終的な spam メール確率の閾値は，Paul Graham 方式の 0.9 を使用した．

4 識別精度の実験結果

図 3 と図 4 のグラフは既存のベイジアンフィルターとクラスタリングを利用したベイジアンフィルターに 2500 通の non-spam メールと 10 通〜8000 通の無作為な spam メールを変化させて学習させた場合の正削除率と誤識別率のグラフである．正削除率とは spam メールを正確に削除できる識別精度，誤識別率とは non-spam メールを誤って削除してしまう識別精度である．

縦軸は正削除率と誤識別率のパーセンテージを示し，横軸の一番上は学習させた spam メール数，二番目は正削除率の値，三番目は誤識別率の値を示す．テストに使用したのは，無作為な spam メールと non-spam メール 100 通ずつである．

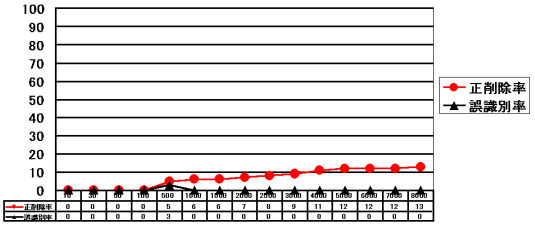

図 3　ベイジアンフィルターの識別精度

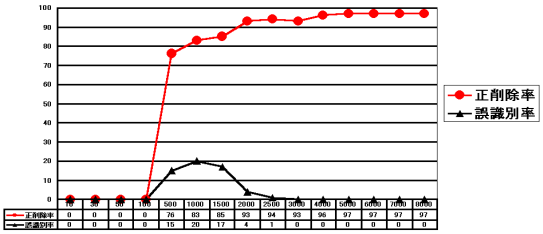

図 4　クラスタリングを利用したベイジアンフィルターの識別精度

各々のフィルターの特徴を以下に示す．

- ベイジアンフィルターでは 500 通の spam メールの学習で 3% の誤識別率が出てしまう程度だが，クラスタリングを利用したベイジアンフィルターでは 500 通〜2500 通の学習段階で最高 20% の高い誤識別率を出してしまう．しかし 3000 通以上の学習をこなせば誤識別率を 0% にすることができる．
- ベイジアンフィルターでは 8000 通の spam メールの学習をさせても有効な正削除率を示さないが，クラスタリングフィルターでは 4000 通を超える学習をさせれば 96% の高い正削除率を得ることができ，5000 通以上の学習では 97% に達する正削除率を得ることができる．

5 今後の課題

膨大な学習時間が必要となり，現在の計算機のスペックでは有効性がない．今後は有効性のあるオーダーのアルゴリズムを考え直す必要がある．

参考文献

[1] David Mertz：
『spam のより分け手法』，(accessed 2004.9.28)
http://www-6.ibm.com/jp/developerworks/linux/021129/j_l-spamf.html

[2] Daisuke IKEGAMI：
『ベイジアンフィルタについて』, (accessed 2004.9.1)
http://www.tom.comm.waseda.ac.jp/ ike/column/0006.html