69 SVMとベイズ理論によるメールフィルタを用いたスパムフィルタの構築

新美礼彦* 猪股宏史 宮本政輝 小西修
(公立はこだて未来大学)†

1 はじめに

近年、インターネットの普及にともない、メールサービスの利用が一般的になってきた。それにともない、スパムメールが問題になってきている。スパムメールとは、不特定多数に送られる迷惑メールのことで、一方的な広告メール、チェーンメール、架空請求メール、コンピュータウィルスによるメールなどが含まれる。スパムメールが問題になるのは、大量のスパムメールによりほかのメールが埋もれてしまうだけでなく、大量のメールがネットワーク上を流れることによりネットワークトラフィックの増加が起きるからである。そのため、ほかのインタネットサービスにも影響をおよぼす可能性がある。スパムメール対策のため、スパムメールを含む大量のメールから必要なメールだけを自動的に取り出す仕組みが必要とされている。

基本的にメールの内容はテキスト形式で記述されているので、スパムメールとそれ以外のメールに分類するという作業は、テキスト分類作業であるといえる。そのため、メール分類作業にテキスト分類で用いられる様々なアルゴリズムを適用することができる。とくに、スパムメールとそれ以外のメール(正当メール)をそれぞれ正例と負例と捕らえると2クラスへの分類問題と考えられる。

本論文では、テキスト分類アルゴリズムとして、テキスト分類で良く用いられているベイズ理論とSVM(Support Vector Machine)を取り上げ、それらによるフィルタを用いて、スパムメールとそれ以外のメールを分類するシステムを構築した。

2 メール分類

スパムメールに対する代表的なメールフィルタとして、以下のものがある。

1. 基本的なテキストフィルタ
2. ホワイトリストによるフィルタ
3. ブラックリストによるフィルタ

1は、今までに受け取ったことがあるメールを元に、簡単な文字列によるルール設定を作成し、そのルールに基づきメールを分類する方法である。たとえば、「Subjectヘッダに"未承諾広告※"を含んでいたらスパムメールである」などのルールを作成し、メールを分類する。一般的にこのルールを手作業で登録する必要があり、すでに受け取ったことのあるスパムメールからしかルールを作成できない、ルールを作成するのに時間がかかるなどの問題点がある。

2は、受信を許可するメールアドレスを記述しておき、それ以外のアドレスからのメールを受信しない方法である。受信者が受信許可するメールアドレスを登録する以外に、送信者がアドレスを登録するシステムもある。登録されていないメールアドレスからのメールは、受信者リストへの登録を呼びかけるメールを送信者に送り、応答のあったメールアドレスを自動的に受信者リストに登録する方法である。受信者リストをつくるのにコストがかかるという問題のほかに、正当なメッセージをフィルタリングしてしまいスパムメールと誤検知してしまう可能性が高いという問題がある。

3は、受信を許可しないサーバまたは、メールアドレス)を記述しておき、それ以外のメールのみ受信する方法である。2とは逆に、許可しないメールアドレスのリストを作成する方法である。一般的に許可するメールアドレスは個人ごとに異なる可能性が高いが、スパムメールのアドレス、もしくはスパムメールを配信しているサーバは共通していることが多いため、リストを共有することができる。この方法では、正当なメールを見逃してしまう可能性は低くなるが、スパムメールを見逃してしまいフィルタが効率よく動作しなくなる可能性が高い。

これらのフィルタはスパムメール、正当メールの特徴を手作業で抽出する方法である。これに対して、メールの特徴を自動的に抽出する方法が考えられる。メール情報はテキスト形式で記述されているので、メール分類はテキスト分類の一つと捕らえることができる。スパムメールとそれ以外のメール(正当メール)をそれぞれ負例と正例と捕らえると2クラスに分ける分類問題と考えられる。そのため、テキストの自動分類アルゴリズムをメール分類に利用することができる。

テキストの自動分類アルゴリズムは、すでにいくつか提案されている。[1, 2, 3] これらの成果をスパムフィルタの構築にも利用することは充分考えられる。

3 ベイジアン・スパムフィルタ

ベイジアン・スパムフィルタは、ベイズ理論を元にしたスパムフィルタである。[4] ベイズ理論では、ある事象の原因となるすべての事象の確率とその原因の元である事象が起こる条件付き確率をもとに、ある事象が起きたときにある原因が起きた確率を求めることができる。メールで使われている文字列(トークン)の出現確率から

hiimi@fun.ac.jp
北海道函館市亀田中野町116番地2

スパムメールであるかどうかの確率をベイズ理論により求め、フィルタリングするフィルタである。トークンとして、単語(またはその語幹)、n文字の連続する文字列などが用いられる。

あるトークン(w)が含まれているとき、そのメールがスパムメールである確率(スパム確率:$p(w)$)を、以下の式で定義する。

$$p(w) = \frac{b/n_{bad}}{\alpha g/n_{good} + b/n_{bad}} \quad (1)$$

ここで

$p(w)$: あるトークン(w)が含まれているときのスパムメールである確率(スパム確率)

n_{bad}: スパムメール数

$b(w)$: スパムメール中で、あるトークン(w)が出現した回数

n_{good}: 正当メールでないメール数

$g(w)$: 正当メール中で、あるトークン(w)が出現した回数

α: 重み

とした。この定義では、正当メール数に重みをつけることによって、スパムメールの誤検知率を減らすようにしている。

また、複数のトークンを同時に含む場合にスパムメールである確率(複合確率)は、以下のように定義した。

$$P(w_1, w_2, \ldots w_n) = \frac{p(w_1) \times p(w_2) \times \cdots \times p(w_n)}{p(w_1) \times \cdots \times p(w_n) + (1 - p(w_1)) \cdots (1 - p(w_n))} \quad (2)$$

ここで、

$P(w_1, w_2, \ldots w_n)$: あるトークン($w_1, w_2 \ldots w_n$)が同時に含まれているときのスパムメールである確率(複合確率)

$p(w_1)$: あるトークン(w_1)が含まれているときのスパム確率

とした。

メールをスパムメールかどうか判定する手順は以下の通りである。手順は事前処理(フィルタの学習)と判定処理(フィルタリング)に別れている。

事前処理(フィルタの学習) スパムメール、正当メールを集める。すべてのメールをトークンに分解し、トークンごとのスパム確率を計算し、データベースに登録する。

判定処理(フィルタリング) 判定するメールをトークンに分割する。得られたトークンのスパム確率をデータベースに問い合わせる。この中から特徴的なトークンを抽出し、複合確率を求める。複合確率が設定した閾値以上の場合、このメールをスパムメールと判断し、閾値未満なら正当メールと判断する。

特徴的なトークンとして、判定処理に適したトークンを用いる。スパム確率が0.5からより離れた確率を持つトークンを使う。スパム確率が0.5とは、どちらのメールともいえないトークンである。

4 SVMによるスパムフィルタ

SVM(Support Vector Machine)は、ベクトルで表されるデータ集合を2つのクラスに分類するためのアルゴリズムである。[5] SVMによるスパムフィルタでは、SVMを用いてメールをスパムメールと正当メールに分類する。

SVMは、入力としてベクトルで表されたデータ集合を使う。メールをSVMによって分類するには、メールデータをベクトル化する必要がある。テキストのベクトル化は、ベイジアン・スパムフィルタのときと同様にトークンに分割し、出現したトークンに対応するトークンコードとその出現頻度を求めることにより行う。トークンコードを定義するために、事前にメールに現れるトークンをすべて抽出しておく。出現頻度は、出現回数を数えたものやTF-IDFによる定義などが考えられる。

SVMを用いたメールをスパムメールかどうか判定する手順は以下の通りである。手順は事前処理(フィルタの学習)と判定処理(フィルタリング)に別れている。

事前処理(フィルタの学習) スパムメール、正当メールを集める。すべてのメールをトークンに分解し、トークンごとの出現頻度を求める。出現したトークンにトークンコードを定義する。トークンコードと出現頻度をもとにベクトル集合を作成する。ベクトル集合と、スパムメールか正当メールかのラベルを使いSVMにより学習し、分類器(フィルタ)を生成する。

判定処理(フィルタリング) 判定するメールをトークンに分割し、トークンコードと出現頻度のベクトルを作成する。作成したベクトルをフィルタによりスパムメールか正当メールかを判定する。

5 単語の頻度を元にしたフィルタの特徴

単語の頻度を元にフィルタを構築すると、フィルタを構築するコストを下げることができるという利点がある。単語の頻度情報は元になるテキストがあれば、簡単に求めることができる。これにスパムメールかどうかの情報を与えるだけで、フィルタを構築することができる。また、ベイジアン・スパムフィルタ、SVMスパムフィルタともユーザごとにフィルタを学習させることができるため、ユーザごとのスパムメールと正当メールの特徴に合わせてフィルタを構築することができる。単語の頻度のみに注目しているので、既知のメールだけでなく、未知のメールに対しても性能が期待できる。

しかし、単語の頻度を元にしたフィルタには、以下の問題点がある。

1. メール本文が短いと、判定しにくい。

2. 正確な学習データがなければ、正確な判定ができない。
3. 正当なメールに現れやすい単語を多く含んだスパムメールを見逃してしまう。
4. 中立な単語とURLのみを含んだメールを見逃してしまう。

メール本文が、短かったり、空だったりする場合、フィルタに入力するトークンが確保できない可能性がある。一般的に、これらのメールは必要でないメールであることが多いので、本文が短いメールはすべてスパムメールと判断することは可能であるが、日常的に短いメールのやり取りを行っている場合、スパムメールであるのか、正当メールであるのか判断しにくい。

正確な学習データがなければ、正確な出現頻度を求めることができないため、フィルタの性能に影響を与える。さらに、正確な学習データは正確な出現頻度が求められるほど多量にあればあるほどよい。これは、スパムメールを大量に受け取ってから学習しないと性能が良くならないということになる。この問題に対しては、スパムメールを収集しているサイトを活用することで、ある程度の性能のフィルタをつくることができる。ある程度一般的なスパムメールをフィルタリングできるフィルタを構築しておいて、運用しながら、フィルタを個人ごとに強化していくのである。

正当なメールに現れやすい単語を多く含んだスパムメールとは、例えば、"TOEICの勉強に関するメーリングリストのメール"(正当メール)に対する"TOEICの受験教材の広告メール"(スパムメール)などである。高頻度で出てくる単語が正当メールとスパムメールで大きな差がない場合に問題になる。この問題に対しては、メール本文だけでなく、メールヘッダも含めて頻度を調べる、事前にブラックリストによるフィルタを適用するなどの対策が考えられる。

中立な単語とは、あいさつなどのスパムメールにも正当メールにも出てきやすい単語のことである。中立な単語のみからスパムメールかどうかを判断することは難しい。広告メールなどで、メール本文には、あいさつとURLのみを記述しておき、URLのリンク先に広告を置くことで、ユーザを広告に誘導するメールがある。悪質なものになると、リンク先でコンピュータウィルスに感染するようになっているものもある。この問題に対しては、URLプリフェッチ方式を導入することが考えられる。[6] URLプリフェッチ方式とは、リンク先の情報もフィルタで学習するという方式が考えられる。フィルタリングする前にリンク先情報を自動的に習得し、入力データとするという方式である。

6 スパム・フィルタの実装

ベイジアン・スパムフィルタとSVMスパムフィルタを実装し、性能を評価した。性能評価には、適合率と再現率を用いた。適合率と再現率は以下のように定義した。

$$rel = s/n \tag{3}$$

$$rep = s/c \tag{4}$$

ここで、

rel: 適合率

rep: 再現率

n: フィルタが正当メールと判定したメールの総数

c: 正当メールの総数

s: フィルタが正当メールと判定したメールで実際に正当メールだったメールの総数

とした。

適合率により、フィルタが正当メールであると判断したメールにおける実際の正当メールの割合を示す。再現率により、実際の正当メールにおけるフィルタが正当メールと判断したメールの割合を示す。

6.1 ベイジアン・スパムフィルタの実装

ベイジアン・スパムフィルタによるメールフィルタを実装し、性能評価を行った。ベイジアン・スパムフィルタとしてbsfilter[7]を用いた。英語のトークンは、アルファベット、数字、アポストロフィ、ドルマークを構成要素と見なして、それ以外を区切り文字とした。日本語のトークンは、bigramを用い、連続する漢字2文字、カタカナをトークンとした。正当メール、スパムメールを日本語、英語とも150通ずつ用意し、交差検定法にて性能評価を行った。実験結果をTable 1に示す。

Table 1 ベイジアン・スパムフィルタによる分類性能

対象	適合率(%)	再現率(%)
日本語のみ	96.71	98.00
英語のみ	73.89	100
日本語、英語	82.40	98.33
+追加処理あり	98.66	98.33

全体的に、高い再現率を得られた。英語のみの適合率が低いのは、良い英語正当メール、スパムメールを用意できなかったためだと考えられる。日本語、英語を同時に対象とするフィルタでは、適合率が82.40%という結果が得られた。この結果に対し、以下の追加処理を行った結果、適合率を98.66%に上げることができた。

- メール本文が空のものは無条件でスパムメールと判断する
- メール本文にURLがあるがそのリンク先が切れているものを無条件でスパムメールと判断する

- リンクの切れていないのはURLプリフェッチ方式を適用する

この時、スパムメールであるのに正常メールであると分類したメールを調べた。これらのメールは正当メール中に良く似たメールが含まれていることがわかった。似たような出現頻度の正当メールとスパムメールが含まれていたので、うまくフィルタを学習できなかったと考えられる。

6.2 SVMスパムフィルタの実装

SVMスパムフィルタによるメールフィルタを実装し、性能評価を行った。SVMの実装として、SVM^{light}を用いてフィルタを構築した。英語のトークンは、TreeTagger[9]を用いて語幹を抽出して用いた。日本語のトークンは、Chasen[10]を用いて語彙を抽出して用いた。実験には、日本語スパムメール175通、日本語正当メール188通、英語スパムメール261通、英語正当メール300通の合計921通を用いて、フィルタの学習を行った。学習後のフィルタの性能をTable 2に示す。

Table 2 SVMフィルタによる分類性能

対象	適合率(%)	再現率(%)
日本語のみ	98.00	98.00
英語のみ	100	98.04
日本語、英語	97.59	90.00

実験結果から、日本語のみ、英語のみの場合、高い再現率と適合率が得られた。日本のメールや英語のメールのみのメールに対して、高性能のスパムフィルタが構築可能であるといえる。しかし、日本語と英語の両方を含んだメール集合に対しては、再現率が低くなる結果が得られた。日本語のトークンと英語のトークンからなる長いベクトルを入力として取り込むため、冗長な情報によりフィルタを構築することになるからだと考えられる。このため、日本語と英語の双方に対応したシステムを構築する場合、日本語と英語を含んだベクトルを入力に用いるより、入力メールの言語を判断して、日本語なら日本語用のメールフィルタを、英語なら英語用のメールフィルタを用いるようにした方が、効率がいいと考えられる。メールの言語を判断するには、新たにフィルタを作成しなくても、メールヘッダのContent-Typeを調べることにより、判断することが可能な場合が多いので、言語判定についての計算コストは無視できる。

7 おわりに

本論文では、スパムメールのフィルタリングに関する問題をテキスト分類問題として捕らえ、テキスト分類アルゴリズムを用いることによりフィルタを構築することを試みた。テキスト分類アルゴリズムとして、テキスト分類で良く用いられているベイズ理論とSVM(Support Vector Machine)を取り上げ、それらによるフィルタを用いて、スパムメールとそれ以外のメールを分類するシステムを構築した。実験により構築したスパムフィルタの性能を評価し、高い再現率と適合率を示すことを確かめた。これにより、ベイジアンフィルタやSVMフィルタはスパムフィルタとして有効であるといえる。また、ベイジアン・スパムフィルタにURLプリフェッチ方式を組み込み、適合率を高めることができた。このことから、URLプリフェッチ方式を組み込むことでスパムフィルタの性能を向上することができるといえる。

参考文献

[1] 市村 由美、長谷川 隆明、渡部 勇、佐藤 光弘: テキストマイニング - 事例紹介, 人工知能学会誌 Vol.16 No.2,pp.192–200 (2001).

[2] 那須川 哲哉、河野 浩之、有村 博樹:テキストマイニング基盤技術, 人工知能学会誌 Vol.16,No.2,pp.201–211 (2001).

[3] 永田 昌明、平 博順: テキスト分類 - 学習理論の「見本市」, 情報処理 Vol.42 No.1,pp.32–37 (2001).

[4] Paul Graham: A Plan for Spam, http://www.paulgraham.com/spam.html

[5] 平 博順、春野 雅彦: Support Vector Machineによるテキスト分類における属性選択, 情報処理学会誌, Vol.41, No.4,pp.1113-1123 (2000).

[6] 安東 孝二、河 正浩、安 在根、康 秀勳、北野 利治: SPAMメール対策における新方式の提案, マルチメディア, 分散, 協調とモバイル (DICOMO2003) シンポジウム (2003).

[7] nabeken: bsfilter / bayesian spam filter/ ベイジアン・スパムフィルタ, http://www.h2.dion.ne.jp/ nabeken/bsfilter/

[8] Thorsten Joachims: SVM - Light Support Vector Machine, http://svmlight.joachims.org/

[9] IMS Textcorpora and Lexicon Group: TreeTagger, http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[10] 松本 裕治、北内 啓、山下 達雄、平野 善隆、松田 寛、浅原 正幸:日本語形態素解析システム『茶筌』version 2.0 使用説明書 第二版 (1999).