3F6-2

知識検索サイトにおける不適切な投稿の分類
Classification of Spam Posts on Knowledge Searching Website

小林大祐[*1] 松村真宏[*2] 木戸冬子[*3] 石塚満[*1]
Daisuke Kobayashi Naohiro Matsumura Fuyuko Kido Mitsuru Ishizuka

[*1] 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology, The University of Tokyo

[*2] 大阪大学大学院経済学研究科
Graduate School of Economics, Osaka University

[*3] ヤフー株式会社
Yahoo Japan Corporation

It is important for knowledge searching website such that one user asks and another user answers to reduce spam question and answer for improving quality. Spam posts are filtered by human now, so this makes cost for filtering high and increasing spam posts become problem. This paper tries to classify such spam posts automatically with refining learning data to improve recall for filtering out spam posts.

1. はじめに

近年、Web の急速な普及によって膨大な量の知識が Web 上に蓄積されるようになってきた。このような知識を利用したサイトの一つとして、知識検索サイトがあり、ユーザは質問や回答を投稿することができる。しかし、全てのユーザが想定された適切な行動をとるとは限らないため、不適切な投稿の排除は欠かせない。

そこで、本研究では現在人手で行っている不適切な投稿の削除を機械学習による不適切投稿の発見を半自動化するために、教師つき負例と教師なし正例からなる学習コーパスからの SVM (Support Vector Machine)学習器の作成に取り組んだ。

2 章では Web 上の不適切な行動の自動推定に関する関連研究を、3 章では知識検索サイトについて述べる。4 章で本研究の手法について述べ、5 章で実験とその考察を行い、6 章でまとめる。

2. 関連研究

Web 上の不適切な投稿を自動的に分類する研究は、昔からさまざまな分野で行われてきた。有名なものとしては e-mail のスパムフィルタが挙げられる[1]。また、有害な Web サイトをブロックするための研究も行われてきた[Grailheres 04]。また、blog のエントリを対象として、spam かどうかを判定する研究にも取り組まれている [Kolari 06]。しかし、本研究で対象としている不特定多数のユーザが文書を書き込むサイトに対してはあまり研究が行われていないのが現状である。

また、一部の確かな正例・負例と，大部分の教師なしコーパスからなるコーパスから，より精度の高いコーパスを精錬して分類精度を高めようという Semi-supervised(半教師つき学習)と呼ばれる手法を用いた研究についても今までいくつか行われてきた．[Nigam 00]ではテキスト分類の際に EM アルゴリズムと呼ばれる手法を用いて少数の確かな学習データと大部分の教師なし学習データをコーパスとし，少数の確かな学習データのみから分類器を生成した時よりも高い分類精度を得ることに成功している．[鈴木 05]では EM アルゴリズムを用いた Naïve Bayes 分類器および SVM 分類器を作成し，ブログ上の評判情報が肯定的な表現であるか，あるいは否定的な表現であるかを分類している．これを用いた場合，EM アルゴリズムを用いた時の分類精度は EM アルゴリズムを用いない時よりも 0.5%～1.5%ほど向上している．

連絡先：小林大祐，東京大学大学院情報理工学系研究科，東京都文京区本郷 7-3-1，d-koba@mi.ci.i.u-tokyo.ac.jp

3. 知識検索サイト

知識検索サイトとは、あるユーザが投稿した質問に他のユーザが回答を寄せるサイトであり、お互いに知恵や知識を教え合うことを目的としている。過去に投稿された質問や回答は記録されており、後から検索することによって直接その質問と回答には関係ないユーザも情報を得ることもできる。このようなサイトとして、Yahoo! 知恵袋[1]や教えて！goo[2]などがある。

本研究では、Yahoo! 知恵袋の一月分の投稿データと、人手により削除された投稿データは、Yahoo！知恵袋のスタッフがチェックして不適切であると判定されたものである。今回は削除された投稿の約 85%を占めていた質問を対象とすることとした。

4. 手法

4.1 実験用コーパスの生成

本研究で用いるコーパスについて考察する。削除された質問は不適切な文であることがわかるため、これを負例として用いる。正例は、削除されていない質問を用いる。

本研究では、確実に不適切であると分かる文書と、適切・不適切が混ざった未知の文書があり、ここから精度の高いコーパスを得るにはどうしたらよいか、という問題に取り組む。そのため、コーパスに含まれる正例と負例を混ぜて、教師つき負例と教師なし正例からなる学習コーパスを作成する。

4.2 コーパスの生成

コーパスの生成手法として 3 種のアルゴリズムを提案し、5 章における実験にてそれぞれの比較を行う。

(1) スパム分離型の精錬手法

まず、未知コーパスから不適切な文書(スパム)を取り除いて

[1] http://chiebukuro.yahoo.co.jp/
[2] http://oshiete.goo.ne.jp/

いき、コーパスを精錬していく手法について述べる。そして、以下のような手順で精錬を行う。

1. コーパスから、教師つき負例データセット D_0、2組の未知データセットそれぞれ P_0^A と P_0^B を抽出する。
2. n は $n \geq 0$ の整数とする。D_{2n} と P_{2n}^A から文書を1:1にランダムに抽出して SVM 学習器を作成し、P_{2n}^B を分類する。これによってスパムだと判定されたもののうちある条件 Q を満たすものを S_{2n}^B とし、残りを N_{2n}^B とする。
3. $D_{2n+1} = D_{2n}$、$P_{2n+1}^A = P_{2n}^A$、$P_{2n+1}^B = N_{2n}^B$ とする。
4. D_{2n+1} と P_{2n+1}^B から文書を1:1にランダムに抽出して SVM 学習器を作成し、P_{2n+1}^A を分類する。これによってスパムだと判定されたもののうちある条件 Q を満たすものを S_{2n+1}^A とし、残りを N_{2n+1}^A とする。
5. $D_{2n+2} = D_{2n+1}$、$P_{2n+2}^A = N_{2n+1}^A$、$P_{2n+2}^B = P_{2n+1}^B$ とする。
6. 2.～5.を、S として取り除かれる投稿文が収まるまで繰り返す。

この時、P_0^B や P_0^A などから取り除かれた文の集合を $S = S_0^B + S_1^A + S_2^B + S_3^A + \cdots$ と定義する。ある条件 Q としては、SVM の出力がある一定値を越える、スコアの大きい M 件を取り除く、などの方法によって決めることができる。また、任意の n において $D_n = D_0$ となる。

(2) 負例追加型の精錬手法

この手法は、1)のスパム分離型の手法と似ている。これと異なる点は、スパムだと判定されて取り除かれていた S_{2n}^B などを、単純に取り除く代わりに D_0 に追加していくところである。よって、1)のアルゴリズムに以下の改良点を加えたものが負例追加型の精錬手法となる。

3. $D_{2n+1} = D_{2n} + S_{2n}^B$、$P_{2n+1}^A = P_{2n}^A$、$P_{2n+1}^B = N_{2n}^B$ とする。
5. $D_{2n+2} = D_{2n+1} + S_{2n}^A$、$P_{2n+2}^A = N_{2n+1}^A$、$P_{2n+2}^B = P_{2n+1}^B$ とする。

なお、任意の n において $D_n = D_0 + S$ となる。

(3) スパム･ハム分離型の精錬手法

不適切な文書のことはスパムと呼ばれるが、それとは逆に適切な文書のことはハムと呼ばれている。この手法は、未知コーパスからスパムのみを分離するのではなく、ハムも分離することによって、さらにコーパスの分類精度を上げようとするものである。この手法によるアルゴリズムは以下のようになる。

1. コーパスから、教師つき負例データセット D_0、2組の未知データセットそれぞれ P_0^A と P_0^B を抽出する。
2. n は $n \geq 0$ の整数とする。D_{2n} と P_{2n}^A から文書を1:1にランダムに抽出して作成された SVM 学習器と、S と H から作成された SVM 学習器を 2 つ用いて P_{2n}^B を分類する。この 2 つの SVM 学習器両方によってスパムだと判定されたもののうちある条件 Q を満たすものを S_{2n}^B とする。また、SVM 学習器によって出力されたもののうち、スパムでない符号で最も絶対値が大きいほうから S_{2n}^B の文書数 $|S_{2n}^B|$ と同じ件数だけをハム H_{2n}^B とし、同じく P_{2n}^B から分類する。残りを N_{2n}^B とする。
3. $D_{2n+1} = D_{2n}$、$P_{2n+1}^A = P_{2n}^A$、$P_{2n+1}^B = N_{2n}^B$ とする。
4. D_{2n+1} と P_{2n+1}^B から文書を1:1にランダムに抽出して作成された SVM 学習器と、S と H から作成された SVM 学習器 2 つを用いて P_{2n+1}^A を分類する。これによってスパムだと判定されたもののうちある条件 Q を満たすものを S_{2n+1}^A とする。2.と同様にハム H_{2n+1}^A も分離し、残りを N_{2n+1}^A とする。
5. $D_{2n+2} = D_{2n+1}$、$P_{2n+2}^A = N_{2n+1}^A$、$P_{2n+2}^B = P_{2n+1}^B$ とする。
6. 2.～5.を、S として取り除かれる投稿文が収まるまで繰り返す。

スパムとして取り除かれた文の集合を $S = S_0^B + S_1^A + S_2^B + S_3^A + \cdots$、ハムとして取り除かれた文の集合を $H = H_0^B + H_1^A + H_2^B + H_3^A + \cdots$ と定義する。アルゴリズムによる要請により、$|H_{2n}^B| = |S_{2n}^B|$, $|H_{2n}^B| = |S_{2n}^B|$ になるので、常に $|H| = |S|$ となる。また、任意の n において $D_n = D_0$ となる。

5. 実験

5.1 コーパス

本研究では, Yahoo! 知恵袋の一月分の投稿データと, 削除された投稿データの提供をいただいた. 一月分の投稿データにおける適切な質問投稿（215,288 投稿）と削除された質問投稿（33,852 投稿）をコーパスとした。なお、1 つの質問の長さは平均して 60～70 文字であった。

5.2 評価尺度

本研究では, 不適切な投稿(スパム)をいかにして取り除くかということが重要である. このため, 生成されたコーパスを分類尺度として不適切投稿の分類精度に着目した次の評価尺度を用いる.

$$\text{Spam} - \text{Recall} = SS/(SS + SN)$$
$$\text{Spam} - \text{Precision} = SS/(SS + NS)$$
$$\text{Spam} - \text{F'1} = \frac{2 * \text{Spam} - \text{Recall} * \text{Spam} - \text{Precision}}{\text{Spam} - \text{Recall} + \text{Spam} - \text{Precision}}$$
$$\text{Spam} - \text{Accuracy} = (SS + NN)/(SS + NS + SN + NN)$$

SS = スパム文のうち, スパムと分類された投稿文の数

SN = スパム文のうち, スパムと分類されなかった投稿文の数

NS = スパム文ではないもののうち, スパムと分類された投稿文の数

NN = スパム文ではないもののうち, スパムと分類された投稿文の数

Spam-Recall とは, スパム文をスパムだと判定できた割合であり, Spam-Precision はスパムだと判定された文書のうち, 実際にスパム文だった割合である.

5.3 素性の種類による分類精度の変化

まず素性の種類によって分類精度にどのような違いが出るかを見た. 学習データとしてスパム文 10,000 投稿とスパムでない文 20,000 投稿, テストデータとしてはヤフー株式会社の方にご協力いただいて確かにスパム/スパムではないと判別した 988 投稿(うちスパムが 500, スパムでないものが 488)を用いる.

表 1　素性の種類を変えたときの分類精度の変化

主義語	記号	文末	Spam-Precision	Spam-Recall	Spam-Fvalue	Spam-Accuracy	素性数
○			70.94%	88.52%	78.76%	76.42%	23,199
	○		62.76%	92.21%	74.69%	69.13%	1,248
○	○		72.80%	89.96%	80.48%	78.44%	24,347
		○	50.92%	95.90%	66.52%	52.33%	45
○		○	71.41%	89.55%	79.46%	77.13%	23,398
	○	○	65.18%	92.83%	76.59%	71.96%	1,293
○	○	○	73.13%	90.37%	80.84%	78.85%	24,646

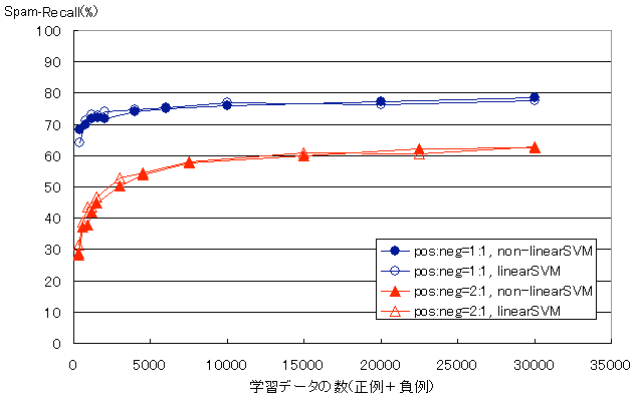

図 1　学習データ件数を変えたときの Spam-Recall の変化

特徴ベクトルは，第 2 章と同様に各素性の *tfidf* 値を用いた．(2.3.2 節参照)素性の種類は，単語単体で意味を持つと考えられる主義語(名詞，動詞，形容詞)，記号，そして文末表現と考えられる単語(助動詞，終助詞)である．これらの単語の N-Gram(N=1～5)を抽出して，出現回数が 3 回以上となるものを素性とした．形態素解析パーサとしては Chasen[26]を用いている．また，今回はストップワードとして半角のアルファベット 1 文字で構成される 52 の単語と，半角文字または全角文字による 1 文字の数字からなる単語 20 単語の計 72 単語を設定した．

素性の種類を様々に変えて分類した結果が表 1 である．これを見ると，やはり主義語が入っている場合は分類精度が高いことが分かる．次いで記号も素性として十分に機能していることが分かるが，文末表現に関してはわずかに精度が上昇するにとどまっている．これは，今回文末表現が助動詞，終助詞で構成されるものとして実験したが，実際には素性数が 45 であり，文末表現が助動詞，終助詞のみとは言えない．これは，「です」などの文末に用いられる表現が実際には動詞として解析されているケースも多いためである．ただし，この文末表現が入っていると多少は分類精度が上昇するため，今後の実験では主義語，記号，文末表現すべての品詞からなる N-Gram を素性とする．

5.4　学習データの件数による分類精度の変化

学習データ件数を変化させたときに分類精度がどうなるのかを見てみることとする。非線形 SVM のカーネル関数には 2 次の多項式カーネルを使用した．学習データの正例としてスパムでない投稿文，負例としてスパムである投稿文を用いている．テストデータの正例，負例は 3.4.3 節でも用いたテストデータ 988 投稿(うちスパムが 500，スパムでないものが 488)である．学習

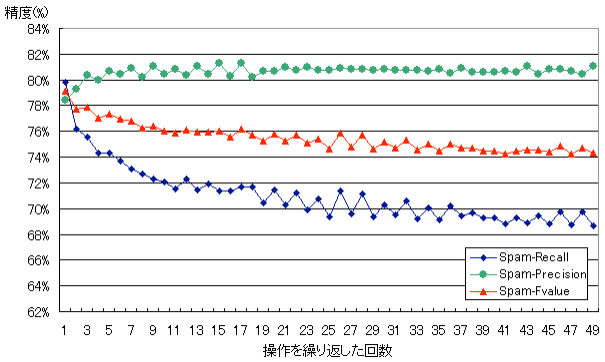

図 2　条件 Q を A)にした時のテストデータの分類精度

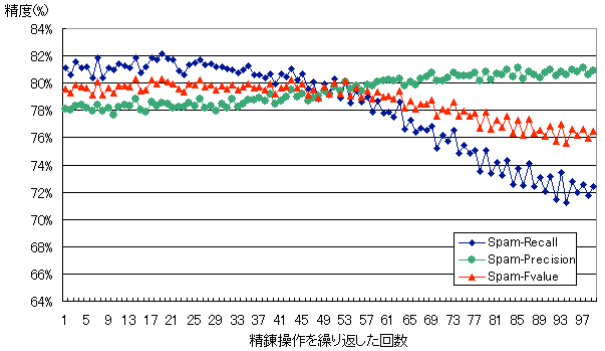

図 3　条件 Q を B)にした時のテストデータの分類精度

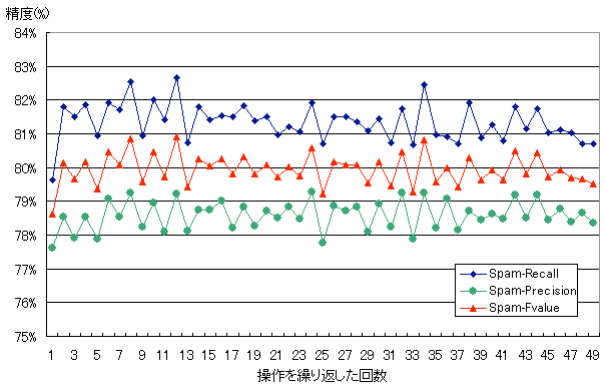

図 4　条件 Q を C)にした時のテストデータの分類精度

データの件数を変化させつつ，5 回学習と分類を行いその分類精度の平均をとった．

この結果が図 1 である。図 1 をみると、学習データの件数が 10,000 件ほどとなるところで精度の上昇が飽和していることが見て取れる。

5.5 条件 Q によるコーパス精錬の違い

コーパスの精錬に関する実験を行う. まずは, 4.2 節のスパム分離型を用いて、アルゴリズム中の条件 Q をどうするか調べる.

A) SVM によって出力が 0 より小さくなったものをすべて分離する.
B) SVM によって出力が 0 より小さくなったもののうち, 値が小さい上位 100 投稿について分離する. 出力が 0 より小さくなったものが 100 件に満たない場合はすべてを分離する.
C) SVM によって出力が-1 より小さくなったものをすべて分離する.

以上により実験を行う. D_0 はスパムである投稿文 5,000 投稿, P_0^A と P_0^B はそれぞれスパムである文書 8,000 投稿に加えてスパムではない文書 2,000 投稿で構成されている.

による結果は図 2, B)による結果は図 3, C)による結果は図 4 に載せた. これらのグラフの Spam-Recall, Spam-Precision, Spam-Fvalue は 5.2 節で述べたものである. コーパスの精錬を 5 回繰り返し, その平均をグラフに示した.

A)による結果である図 2 を見ると, 分類精度, 特に Spam-Recall が上がっておらず, コーパスの精錬手法としてはまったく適していないことが分かる.

B)の図 3 では、途中まで上がっていったテストデータの分類精度が下がっていってしまっているという問題が見られる。

C)による結果図 4 をみると、最初のみで収束してしまい, コーパス精錬の極大値に得たかどうか定かではない.

5.6 アルゴリズムの違いによる分類精度の変化

先ほどの条件 Q として、A)は精錬が行き過ぎであり、C)は収束してしまうので、精錬の様子がよく見られる B)を用い、4.2 節の 3 種のアルゴリズムを比較することとした。

スパム分離型のアルゴリズムを用いた時の分類精度の変化は図 3 のようになる。よって、残り 2 つのアルゴリズムに関するグラフを載せる。

図 5 は負例追加型のアルゴリズムによるグラフである。これを見ると, Spam-Recall の値はスパム分離型のアルゴリズムに比べてかなり高い値を示すことがわかる. しかし, コーパス精錬が進むにつれてこれらのテストデータの分類精度は下がっており, スパム分離型のアルゴリズムと同じ問題を抱えているということが分かる.

図 6 はスパム･ハム分離型のアルゴリズムによるグラフである。これによると, テストデータの分類精度も下がることなく収束している. よって, これらのアルゴリズムの中で一番優れているのはスパム･ハム分離型のアルゴリズムであることが分かる.

6. まとめ

今回の実験により、教師つき負例と教師なし正例からなるコーパスを精錬し、分類精度を高めることに成功した。本研究の提案手法は, 正例あるいは負例の片方のみに確かなコーパスが存在し, 残りが不確かなコーパスであることが関連研究で挙げた研究と異なる点である. 提案手法で 4%ほどの分類精度の向上が得られたことは, 本研究が効果的な手法であることを示していると考えられる。

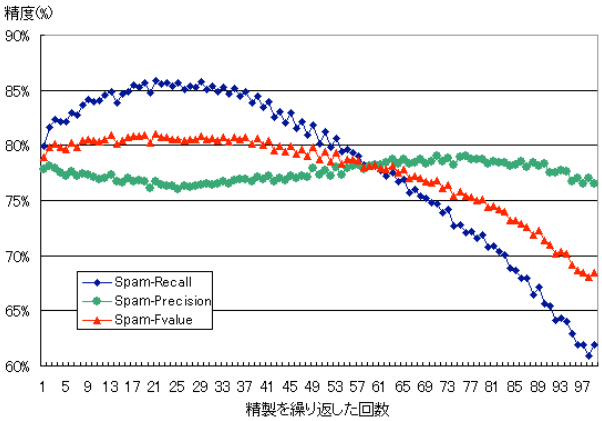

図 5 負例追加型アルゴリズムによるテストデータの分類精度

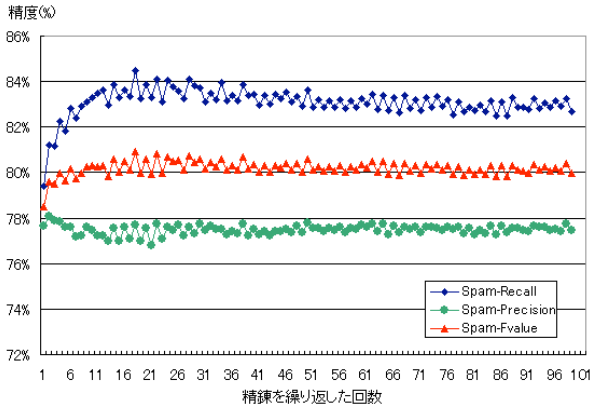

図 6 スパム･ハム分離型アルゴリズムによるテストデータの分類精度

謝辞

本研究では、Yahoo！知恵袋のデータを分析に利用させていただき、また、実験についてのご協力もいただきました。ご協力いただきましたヤフー株式会社様には記してお礼を申し上げます。

参考文献

[Drucker 99] H. Drucker, C. Wu and V. Vapnik, Support Vector Machines for Spam Categorization. IEEE Trans. On Neutral Networks, vol. 10, number 5, pp.1048-1054, 1999.

[Grailheres 04] B. Grailheres, S. Brunessaux, P. Leray, Combining Classifiers for harmful document filtering, RIAO’2004, Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval, 2004.

[Kolari 06] P. Kolari, A. Java and T. Finin, Characterizing the Splogosphere. Proceedings of the 3rd Annual Workshop on Weblogging Ecosystem: Aggregation, Analysis and Dynamics, 15th World Wide Web Conference, 2006.

[Nigam 00] K. Nigam, A. Mccallum, S. Thrun and T. Michell. Text Classification from Labeled and Unlabeled Documents using EM. Machine Learning, 39(2/3). pp. 103-134. 2000.

[鈴木 05] 鈴木 泰裕, 高村 大也, 奥村 学. “Semi-Supervised な学習手法による評価表現分類”, 言語処理学会 第 11 回年次大会, pp. 668--671 , March 2005.