自然言語処理 128−24
(1998. 11. 6)

Support Vector Machineによるテキスト分類

平 博順[†] 向内 隆文[†] 春野 雅彦[††]

[†] NTTコミュニケーション科学研究所
〒619-0237 京都府相楽郡精華町光台2-4
[††] ATR人間情報通信研究所
〒619-0288 京都府相楽郡精華町光台2-2
taira@cslab.kecl.ntt.co.jp

概要: 本稿では、Support Vector Machine (SVM) を用いたテキスト分類法を提案する。テキスト分類問題に対して学習手法を適用する場合、出現頻度の小さい単語まで考慮して学習を行なわいと、分類精度が落ちることが知られている。このため高い分類精度を実現するためには、高次元の単語ベクトルを用いなければならないが、過学習により分類精度が落ちてしまう危険性が生じる。SVMはKernel関数により非線形学習も可能であり、高次元の入力ベクトルを用いても過学習なしに最適解が得られる。SVMをテキスト分類に適用し、1. 異なる次元の単語ベクトル、2. 異なるKernel関数、3. 異なる目的関数、の3点について比較実験を行なった。その結果、SVMがテキスト分類問題に対して有効であることが確認された。

Text Categorization Using Support Vector Machines

Hirotoshi Taira[†] Takafumi Mukouchi[†] Masahiko Haruno[††]

[†] NTT Communication Science Laboratories
2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0237 Japan
[††] ATR Human Information Processing Research Laboratories
2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0288 Japan
taira@cslab.kecl.ntt.co.jp

Abstract: This paper describes a text categorization method that uses Support Vector Machines (SVMs). The accuracies of learning-based text categorization systems depend not only on frequent words but also on infrequent ones. However, high dimensionality of the data sometimes causes overfitting that harms the overall performance of the system. SVMs avoid the curse of dimensionality by using a quadratic optimization method. In addition, SVMs can also learn Non-linearity by introducing Kernel functions. We tested method from the following three perspectives: 1. word vectors of different dimensions, 2. different Kernel functions and 3. extended cost function. The results clearly show the effectiveness of SVMs for the text categorization task.

1 はじめに

大量の電子化されたテキスト情報の中から必要な情報を効率的に抽出することが重要な課題となっている。中でもテキストをあらかじめ与えられたジャンルに分類するテキスト分類の技術は重要である。例えばインターネット上の検索エンジンには、あらかじめ検索対象がツリー上に分類されているディレクトリ型と呼ばれるものがあり、日々新たなデータが登録されている。これらの分類は現在人手で行なわれており、繁雑な作業となっている。

テキスト分類問題に対して学習手法を適用する場合、出現頻度の小さい単語まで考慮して学習を行なわなければ分類精度が落ちることが知られている。高い分類精度を実現するためには、高次元の単語ベクトルを用いなければならない。しかし一方で高次元の単語ベクトルを用いると過学習により、結果的に分類精度が落ちてしまう問題が生じる [19]。

近年、機械学習の分野では Support Vector Machine や Adaboost [9, 13] などサンプルと分類境界の間隔 (margin) に基づく手法が提案されている。これらの手法は少数の事例のみに出現する情報も利用しつつ過学習を回避することが可能で、実際、Adaboost はテキスト分類 [18]、形態素解析 [10]、構文解析 [11] に適用されいずれも高い精度が報告されている。

本稿では、Support Vector Machine [1] を用いたテキスト分類手法について述べる。Support Vector Machine は従来の学習法に比べ、高次元のベクトルを用いても過学習せず最適解が得られるとされている。また Kernel 関数を用いることにより、非線形なベクトル空間に対する学習が可能であり、複雑な問題でも学習精度を上げることができる。

本稿の構成は以下の通りである。2 章で Support Vector Machine (SVM) とテキスト分類問題の概要を説明し、3 章で RWCP 毎日新聞コーパスを対象とし、1. 異なる次元の単語ベクトル、2. 異なる Kernel 関数、3. 異なる目的関数、の 3 つの場合の実験結果について述べる。最後に 4 章で結論と今後の課題についてまとめる。

2 Support Vector Machine とテキスト分類

2.1 Support Vector Machine

2.1.1 Optimal hyperplane

正例と負例の 2 つのクラスに属す訓練データのベクトルの集合を、

$$(\mathbf{x}_1, y_1), ..., (\mathbf{x}_l, y_l), \mathbf{x}_i \in \mathbf{R}^n, y_i \in \{-1, +1\}$$

とする。ここで、$\mathbf{x}_i$ はデータ i の特徴ベクトルで、y_i はデータ i が正例 (1) か負例 (-1) かを表すスカラである。テキスト分類問題では、テキストの特徴をテキスト中に出現する単語で代表させ、単語 w_i がテキスト中に出現する場合、$w_i = 1$、出現しない場合を、$w_i = 0$ としてテキストをベクトル $\mathbf{x}_i = (w_1, w_2, ..., w_n)$ で表す。テキストがあるカテゴリに含まれる場合を正例、含まれない場合を負例として、各カテゴリに対して、Support Vector Machine を構成する。

これらのデータを n 次元 Euclid 空間上の

$$(\mathbf{w} \cdot \mathbf{x}) + b = 0$$

なる hyperplane で分離する問題を考える。この際、近接する正例と負例のデータの間の間隔 (margin) が大きい方が、精度よくテストデータを分類できる。ここで separating hyperplane と呼ばれる 2 つの hyperplane を考える。

$$(\mathbf{w}_l \cdot \mathbf{x}_i) + b \geq 1 \quad if \quad y_i = 1 \qquad (1)$$

$$(\mathbf{w}_l \cdot \mathbf{x}_i) + b \leq -1 \quad if \quad y_i = -1 \qquad (2)$$

式 (1),(2) をまとめて書くと、

$$y_i[(\mathbf{w}_l \cdot \mathbf{x}_i) + b] \geq 1, \quad i = 1, ..., l.$$

となる。hyperplane から点 $\mathbf{x}$ までの距離 $d(\mathbf{w}, b; \mathbf{x})$ は、

$$d(\mathbf{w}, b; \mathbf{x}) = \frac{|\mathbf{w} \cdot \mathbf{x} + b|}{\|\mathbf{w}\|}$$

となり、2 つの separating hyperplane の間の margin は

$$\begin{aligned} & \min_{\mathbf{x}_i; y_i = 1} d(\mathbf{w}, b; \mathbf{x}_i) + \min_{\mathbf{x}_i; y_i = -1} d(\mathbf{w}, b; \mathbf{x}_i) \\ = & \min_{\mathbf{x}_i; y_i = 1} \frac{|\mathbf{w} \cdot \mathbf{x} + b|}{\|\mathbf{w}\|} + \min_{\mathbf{x}_i; y_i = -1} \frac{|\mathbf{w} \cdot \mathbf{x} + b|}{\|\mathbf{w}\|} \\ = & \frac{2}{\|\mathbf{w}\|} \end{aligned}$$

と計算される。このmarginを最大にするためには$\|\mathbf{w}\|$を最小化すればよい。Lagrangeの未定乗数法を用いて$\alpha_i \geq 0$のもとで、Lagrangianの$\mathbf{w}, b$に関する最小値を求めることになる。

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i [(\mathbf{w} \cdot \mathbf{x}_i) + b] - 1, \tag{3}$$

ここで、α_iはLagrange乗数である。この最小化問題は、双対問題(4)に置き換えることができる[2]。

$$\max_{\alpha} W(\alpha) = \max_{\alpha} \{\min_{\mathbf{w},b} L(\mathbf{w}, b, \alpha)\} \tag{4}$$

また、

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \tag{5}$$

$$\frac{\partial L}{\partial b} = 0 \tag{6}$$

より、

$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i \mathbf{x}_i y_i \tag{7}$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \tag{8}$$

が導かれる。ゆえに、式(4)は、式(7),(8)より、

$$\begin{aligned} & \max_{\alpha} W(\alpha) \\ = \; & \max_{\alpha} \{-\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) + \sum_{i=1}^{l} \alpha_i\} \end{aligned}$$

と書き直すことができ、その解は、$\alpha_i \geq 0$, $\sum_{i=1}^{l} \alpha_i y_i = 0$のもとで、

$$\bar{\alpha} = \arg\min_{\alpha} \{\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) - \sum_{i=1}^{l} \alpha_i\} \tag{9}$$

で与えられる。よって求める$\mathbf{w}, b$は、

$$\bar{\mathbf{w}} = \sum_{i=1}^{l} \bar{\alpha}_i \mathbf{x}_i y_i \tag{10}$$

$$\bar{b} = -\frac{1}{2} \bar{\mathbf{w}} \cdot (\mathbf{x}_p + \mathbf{x}_n) \tag{11}$$

で与えられる。ここで、$\mathbf{x}_p, \mathbf{x}_n$は、各々

$$\bar{\alpha_p}, \bar{\alpha_n} \geq 0, y_p = 1, y_n = -1 \tag{12}$$

を満たすベクトルである。特に$\alpha_i > 0$となるα_iを持つ$\mathbf{x}_i$をSupport Vectorと呼び[1]、Support Vectorを使うと、式(10)は

$$\bar{\mathbf{w}} = \sum_{\{i:\mathbf{x}_i \in SVs\}} \bar{\alpha}_i \mathbf{x}_i y_i \tag{13}$$

と書きかえることができる。

最終的にテストデータの正例、負例を判定する関数は、

$$f(\mathbf{x}) = sgn(\bar{\mathbf{w}} \cdot \mathbf{x} + \bar{b}) \tag{14}$$

で与えられる。ここで、

$$sgn(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

である。

2.1.2 hyperplaneでデータを分離できない場合

前述のhyperplaneで訓練データを完全に分離できない場合、新たな非負の変数$\xi_i (i = 1, ..., l)$を導入し、式(1), (2)のかわりに、

$$(\mathbf{w}_l \cdot \mathbf{x}_i) + b \geq 1 - \xi_i \quad if \;\; y_i = 1 \tag{15}$$

$$(\mathbf{w}_l \cdot \mathbf{x}_i) + b \leq -1 + \xi_i \quad if \;\; y_i = -1 \tag{16}$$

を考える。この場合、$\|\mathbf{w}\|^2$の最小化のかわりに、

$$\Phi = \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{l} \xi_i \tag{17}$$

の最小化を考える[3]。右辺第一項はmarginの大きさに関する項であり、第二項は、分離できなかった訓練データが二つのhyperplane, $\mathbf{w} \cdot \mathbf{x} + b = 1$あるいは$\mathbf{w} \cdot \mathbf{x} + b = -1$からどれだけ離れているかを示す誤差項である。$C$は第一項と第二項の重視の度合を決める正値のパラメータである。Cの値が大きいときは、相対的に訓練データのhyperplaneからの誤差が大きく評価されて、Cの値が小さいときは相対的にmarginの大きさが重視される。この(17)式についてLagrangianを考えると、

$$\begin{aligned} & L(\mathbf{w}, b, \xi, \alpha, \beta) \\ = \; & \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i \{y_i [(\mathbf{w} \cdot \mathbf{x}_i) + b] - 1 + \xi_i\} \\ & - \sum_{i=1}^{l} \beta_i \xi_i \end{aligned}$$

となる。ここで、α_i, β_i はLagrange乗数である。

この場合の最小化問題も、以下のような双対問題に置き換えることができる。

$$\max_{\alpha,\beta} W(\alpha,\beta) = \max_{\alpha,\beta}\{\min_{\mathbf{w},b,\xi} L(\mathbf{w},b,\xi,\alpha,\beta)\} \tag{18}$$

また、

$$\frac{\partial L}{\partial \mathbf{w}} = 0, \quad \frac{\partial L}{\partial b} = 0, \quad \frac{\partial L}{\partial \xi_i} = 0 \tag{19}$$

より、

$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i \mathbf{x}_i y_i, \quad \sum_{i=1}^{l} \alpha_i y_i = 0, \quad \alpha_i + \beta_i = C \tag{20}$$

が導かれる。ゆえに、双対問題の解は、

$$0 \le \alpha_i \le C, \quad i = 1, ..., l \tag{21}$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \tag{22}$$

の制約のもとで、

$$\bar{\alpha} = \arg\min_{\alpha}\{\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) - \sum_{i=1}^{l}\alpha_i\} \tag{23}$$

で与えられる。最終的にテストデータの正例、負例を判定する関数は、

$$f(\mathbf{x}) = sgn(\bar{\mathbf{w}} \cdot \mathbf{x} + \bar{b}) \tag{24}$$

で与えられる。

2.1.3 Kernel関数

特徴ベクトルを直接用いてhyperplaneを求めることが与えられた問題に対して適当ではない場合、特徴ベクトル$\mathbf{x}$を高次元空間に非線形的に射影し高次元空間上でhyperplaneを構成することができる [1]。この射影関数をKernel関数と呼ぶ。式(9)の$\mathbf{x}_i \cdot \mathbf{x}_j$をKernel関数$K(\mathbf{x}_i, \mathbf{x}_j)$で置き換えて、

$$\bar{\alpha} = \arg\min_{\alpha}\{\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{l}\alpha_i\} \tag{25}$$

とすると、射影を考えた式になる。$\mathbf{w}, b$は、式(10), (11)の代わりに、

$$\bar{\mathbf{w}} \cdot \mathbf{x} = \sum_{SVs} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) \tag{26}$$

$$\bar{b} = -\frac{1}{2}\sum_{SVs} \bar{\alpha}_i y_i \{K(\mathbf{x}_p \cdot \mathbf{x}) + K(\mathbf{x}_n \cdot \mathbf{x})\} \tag{27}$$

となる。

Kernel関数には、多項式関数、ガウス関数、シグモイド関数など多くの関数を使うことができる。

テストデータの正例、負例を判定する関数は、これまでと同様に

$$f(\mathbf{x}) = sgn(\sum_{SVs} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) + \bar{b}) \tag{28}$$

で与えられる。

2.2 新たな目的関数の導入

実際にSVMを用いたテキスト分類を行なうと、相対的に的中率が高く、網羅率が低い結果となる。テキスト分類問題では、正例が少なく、負例が多いため、SVMは正例か負例かの判定が微妙なテキストに対して負例と判定する傾向にある。網羅率を高めるために、式(17)のパラメータCを正例側の$C(C_p)$と負例側の$C(C_n)$に分けた(29)。ここで$C_p > C_n$とすれば正例とに判定しやすいhyperplaneを構成できると考えられる。

$$\begin{aligned}\Phi &= \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{l}\xi_i \\ &= \frac{1}{2}||\mathbf{w}||^2 + C_p\sum_{pos}\xi_i + C_n\sum_{neg}\xi_j \end{aligned} \tag{29}$$

なお、式(29)のように、Cを分離しても、導出上、他の制約式には影響を与えない。

3 実験結果

3.1 実験設定

実験には、RWCPテキストコーパス [4]を用いた。このコーパスは、1994年版の毎日新聞の約3万件の記事に、国際十進分類法に基づくUDCコード [5]を付与したものである。これらの記事の

表 1: カテゴリ毎のサンプル数

カテゴリ名	訓練データ	テストデータ
スポーツ	146	162
刑法	138	166
政府	129	148
教育	101	133
交通	97	118
軍事	96	132
国際関連	92	101
言語活動	92	67
演劇	90	91
作物	77	73

中から頻度の高い 10 種類の分類カテゴリ (スポーツ、刑法、政府、教育、交通、軍事、国際関連、言語活動、演劇、作物) をもつ訓練データ 1000 記事、テストデータ 1000 記事を選んだ。各カテゴリの訓練データ数、テストデータ数を表 1に示す。

これらの記事に対して NTT の形態素解析システム ALT-JAWS [17] により形態素解析を行ない、名詞と固有名詞を抜きだして特徴ベクトルとした。SVM に用いる Kernel 関数は 3.5 節以外は $d = 1$ の多項式関数を用いた。

3.2 評価方法

分類精度を評価するために、的中率、網羅率、F 値 [6] を用いた。各分類毎に、分類モデルと正解の正事例と負事例の数から、

- a: 正解が正事例で分類モデルも正事例と判断した数
- b: 正解が負事例で分類モデルも正事例と判断した数
- c: 正解が正事例で分類モデルも負事例と判断した数

を考えると、的中率 (P)、網羅率 (R) は、次のように定義される。

$$P = \frac{a}{a+b}$$

$$R = \frac{a}{a+c}$$

表 3: カテゴリ別学習精度

カテゴリ名	的中率	網羅率	F 値
スポーツ	0.984	0.765	0.861
刑法	0.884	0.554	0.669
政府	0.809	0.601	0.689
教育	0.880	0.330	0.480
交通	0.873	0.525	0.656
軍事	0.797	0.507	0.620
国際関連	0.528	0.544	0.536
言語活動	0.729	0.522	0.608
演劇	0.957	0.736	0.832
作物	0.964	0.739	0.837

また F 値は的中率、網羅率より、

$$F_\beta = \frac{1+\beta^2}{\frac{1}{P} + \beta^2 \frac{1}{R}}$$

で表される。ここで、β は重みづけパラメータで今回は $\beta = 1$ とした。

3.3 特徴ベクトル数の変化

特徴ベクトル数による影響を調べるため、使用する単語の数を頻度順に 500 から 3000 まで変えて実験を行なった。表 2に示すように単語の数を増やしていくにつれ、F 値が改善されていくことが分かる。10 カテゴリ中 7 カテゴリが単語数 3000 で、3 カテゴリが単語数 2500 で最も精度が高いことから、今回の実験の範囲では SVM が単語数を増やしても過学習を起こしていないことが分かる。この傾向からあらゆる出現単語を利用することも今後の方向として考えられる。

3.4 カテゴリの重複度と学習精度

ここでは精度の悪いカテゴリとカテゴリの重複度について考察する。表 3に 3000 次元の特徴ベクトルを用いた実験結果を示す。どのカテゴリにおいても的中率が高いことが分かる。この特長はデータ数を犠牲にしても正確な情報が必要な情報抽出技術に適している。網羅性についてはカテゴリによりばらつきが見られる。実験に用いたデー

表 2: 特徴ベクトル数による影響 (F 値)

カテゴリ名 \ 単語数	500	800	1000	1500	2000	2500	3000
スポーツ	0.788	0.845	0.846	0.835	0.849	**0.862**	0.861
刑法	0.581	0.601	0.613	0.647	0.669	0.669	**0.669**
政府	0.650	0.641	0.659	0.679	0.674	0.682	**0.689**
教育	0.445	0.464	0.452	0.459	0.478	**0.481**	0.480
交通	0.473	0.551	0.567	0.609	0.628	0.649	**0.656**
軍事	0.560	0.556	0.580	0.587	0.620	**0.623**	0.620
国際関連	0.489	0.527	0.529	0.509	0.519	0.526	**0.536**
言語活動	0.500	0.527	0.583	0.556	0.593	0.573	**0.608**
演劇	0.686	0.778	0.778	0.809	0.824	0.819	**0.832**
作物	0.772	0.800	0.784	0.852	0.825	0.832	**0.837**

表 4: カテゴリの重複

カテゴリ名	重複テキスト数	重複カテゴリ数
スポーツ	5	2
刑法	23	7
政府	10	3
教育	10	5
交通	8	4
軍事	20	4
国際関連	28	5
言語活動	11	4
演劇	2	1
作物	1	1

タは1つの記事に対して、複数のカテゴリ付与がされている。1つのカテゴリに含まれる記事の中で他のカテゴリも付与されている記事の数をを重複テキスト数、付与された他カテゴリ数を重複カテゴリ数として表4に示した。網羅率が0.70を越えているスポーツ、演劇、作物では重複テキスト数が5以下と小さくなっているのに対し、網羅率が0.60を下回っている6つのカテゴリについては交通を除いて、重複テキスト数が10以上となっている。重複テキストの多いカテゴリは、他のカテゴリと出現単語が似通っているために、正例と判定されるテキストが少なくなり、網羅率が下がっているものと考えられる。

表 5: Kernel 関数の次元による影響 (F 値)

カテゴリ名	$d=1$	$d=2$	$d=3$
スポーツ	0.861	**0.865**	**0.865**
刑法	0.669	0.669	0.669
政府	0.689	0.695	**0.700**
教育	0.480	**0.486**	0.477
交通	**0.656**	**0.656**	0.648
軍事	**0.620**	0.613	0.610
国際関連	**0.536**	0.527	0.535
言語活動	**0.608**	0.584	0.584
演劇	0.832	**0.837**	0.830
作物	0.837	0.837	0.837

3.5 Kernel 関数の次元と学習精度

次に Kernel 関数を変えることによる影響をみる実験を行なった。Kernel 関数には多項式関数 $K(\mathbf{x}_i, \mathbf{x}) = ((\mathbf{x}_i \cdot \mathbf{x}) + 1)^d$ を用い次元 d を1から3まで変えて実験した。n 次元を考えることは、n 個の単語の組まで利用して学習することに相当する。実験結果を表5に示す。1次元から2次元に変えた時にいくつかのカテゴリで精度が良くなっているが、3次元にしてもほとんど変化していない。今回の実験の範囲では、テキスト中に出現する単語の組合せに関しては、2つの単語の

表 6: C の分離による影響

カテゴリ名	C 非分離 F 値	C 分離 F 値
スポーツ	0.861	**0.865**
刑法	0.669	**0.690**
政府	0.689	**0.704**
教育	0.480	**0.500**
交通	0.656	**0.670**
軍事	0.620	**0.645**
国際関連	0.536	**0.541**
言語活動	0.608	**0.650**
演劇	0.832	**0.853**
作物	0.837	0.837

組程度の情報で十分であると言える。組合せの情報が学習に有効に働くのは出現頻度の低い単語の場合であると考えられる。3.3 節で示したようにSVM の特徴ベクトルの次元を増やしても過学習が起こらないことから、ベクトルの次元を高くしてKernel 関数の次元を上げるとより高い精度が得られる可能性がある。

3.6 パラメータ C の分離

式 (29) のようにパラメータ C を分離した効果を見るため、$C_p = 30$ と $C_n = 8$ とした場合の実験結果を $C = \infty$(非分離) の場合を比較したものを表 6に示す。各カテゴリとも C を分離した方が分類精度が高く、C の分離がSVM の性能を上げるのに有効であることが分かる。

4 結論

Support Vector Machine をテキスト分類問題に適用し、1. 異なる次元の単語ベクトル、2. 異なるKernel 関数、3. 異なる目的関数の 3 点に関する実験を行った。その結果、1. 単語の次元を増やしても過学習は起こらず分類精度が向上する、2. 今回の実験の範囲では $d = 2$ の多項式で精度が向上することがある、3. 目的関数を変更することで F 値が向上する、が明らかとなった。

今回の実験で用いたテキストには複数のラベルが付与されているにも関わらず、良好な結果が得られた。同じデータに線形学習アルゴリズム WINNOW [12] を適用した結果 [7] と比較すると、的中率が 1 割以上高く、網羅率も同程度であり、テキスト分類問題における SVM の有効性が示された。

今回の実験では、SVM の基本的な振舞いを調べるため、単語の出現有無 $(1, 0)$ だけで特徴ベクトルを構成した。最近、Reuter のデータ [16] に対して相互情報量、TF-IDF で特徴ベクトルを構成したSVM が高い分類精度を達成している [14, 15]。今回得られたSVM の基本的性質に基づきそれらの特徴ベクトルに関しても実験を行う予定である。また、単語の組合せ情報は低頻度の単語に関して有効であると考えられることから、3000 以上の高い次元の特徴ベクトルを用いて異なるKernel 関数の機能を比較することも今後の課題である。

謝辞

毎日新聞 94 年版の使用に関して、記事データの研究利用許諾を頂いた毎日新聞社に感謝致します。

参考文献

[1] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.

[2] R. Fletcher. *Practical Methods of Optimization*. John Wiley and Sons,Inc., 2nd edition, 1987.

[3] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20: pp. 273 – 297, 1995.

[4] 豊浦潤, 徳永健伸, 井佐原均, 岡隆一. RWC における分類コード付きテキストデータベースの開発. 情報処理学会研究報告 NLC96-13. IEICE, 1996.

[5] 情報科学技術協会. 国際十進分類法. 丸善,1994.

[6] B. M. Sundheim. Overview of the Fourth Message Understanding Evaluation and Conference. *Proceedings of Fourth Message Understanding Conference*, pp. 3 – 29, 1992.

[7] 山崎毅, イド・ダガン. 誤り駆動型学習とシソーラスを用いた文書自動分類. 情報処理学会研究報告 NL120-14,pp. 89 – 96,1997.

[8] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Proceedings and Management*,24(5), pp. 513 – 523, 1988.

[9] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and System Sciences*, 55, 1, pp. 119–139, 1997.

[10] M. Haruno and Y. Matsumoto. Mistake-driven Mixture of Hierarchical Tag Context Trees, *Proc. 35th Annual Meeting of the Association for Computational Linguistics*, 1997.

[11] M. Haruno., S. Shirai and Y. Ooyama. Using Decision Trees to Construct a Practical Parser, *Proc. 36th Annual Meeting of Association for Computational Linguistics*, 1998.

[12] N. Littlestone. Learning Quickly When Irrelevant Attributes Abound: A New Linear-threshold Algorithm, *Machine Learning*, 2, pp. 285–318,1988.

[13] R. Schaipre, Y. Freund, P. Bartlett and W. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods, *Proc. 14th International Conference on Machine Learning*, 1997.

[14] T. Joachims. Text Categorization with Support Vector Machines, *Europian Conference on Machine Learning(ECML)*, 1998.

[15] S. Dumais, J. Platt, D. Heckerman and M. Sahami. Inductive Learning Algorithms and Representations for Text Categorizaiton, *7th International Conference on Information and Knowledge Management*, to appear, 1988.

[16] http://www.research.att.com/~lewis /reuters21578.html.

[17] S. Ikehara, S. Shirai, A.Yokoo and H.Nakaiwa. Toward a MT system without pre-editing - effects of new methods in ALT-J/E., *Third Machine Translation Summit*, pp. 101-106, 1991.

[18] R. E. Schapire, Y. E. Singer. BoosTexter: A System For Multiclass Multi-label Text Categorization, http://www.research.att.com/~schapire /boost.html

[19] K. Fukunaga. *Introduction to Statisitical Pattern Recognition.* Academic Press, San Diego, 1990.