自 然 言 語 処 理 146－6
（２００１．１１．２０）

Support Vector Machine の多値分類問題への適用法について

山田 寛康, 松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
{hiroya-y, matsu}@is.aist-nara.ac.jp

本研究では, 日本語固有表現抽出タスクを題材に, 機械学習アルゴリズム Support Vector Machine(SVM) を多値分類問題に適用する手法を提案し, 代表的な従来手法である one vs. rest 法, 及び pairwise 法 との比較を行なう. 二値分類器である SVM を固有表現抽出タスクに適用するためには, 多値分類器に拡張する必要がある. しかし分類するクラス数に比例して計算コストが増加するため, 現実的な時間での学習, 及び分類が困難となる. 我々は, 多値分類問題を, 比較的分類が容易な二値分類へ分割し, 二分木を構築する手法を応用し, 効率的な学習, 及び分類ができるよう, SVM の多値分類器への拡張を行う. 固有表現抽出実験では, 従来法である pairwise 法, 及び one vs. rest 法と比べ, ほぼ同等な抽出精度を維持し, 抽出時間を削減できることを確認した.

キーワード:情報検索, 情報抽出, 固有表現抽出, サポートベクター学習, 多値分類問題

Applying Support Vector Machine to Multi-Class Classification Problems

Yamada Hiroyasu, Matsumoto Yuji
Graduate School of Information Science, Nara Institute Science and Technology
{hiroya-y, matsu}@is.aist-nara.ac.jp

This paper proposes a method for multi-class classification with Support Vector Machines(SVM) and evaluates its effectiveness using Japanese named entity extraction task. Multi-class problems with more than two classes have typically been solved by combining independently produced binary classifiers, such as pairwise and one vs. rest method. However, these methods require large computational cost with increasing the number of classes. We propose a method to reduce multi-class classification to binary using a method called as tree-structured model for efficient learning and classifying. Results of our extraction experiments suggest that the method is comparable to the one vs. rest and pairwise methods, and it can reduce the extraction time.

Keywords : Information Retrieval, Information Extraction, Named Entity, Support Vector Learning, Multi-Class Classification

1 はじめに

近年, 統計的自然言語処理の様々なタスクで, 機械学習を使用した手法が数多く研究されている. Support Vector Machine(SVM) は, 汎化誤差が素性空間の次元に依存しないことが理論的に証明されており, 実験的にも, Chunk 同定問題 [7], 日本語係受け解析 [4], 日本語未知語品詞推定 [5], 日本語固有表現抽出 [9], 文書分類 [12, 8, 3] など, 高次元素性空間における学習を必要とするタスクにおいて, 高精度の結果が報告されている. また SVM は Kernel 関数の適用により, 非線形分離が可能であり, 特に多項式 Kernel 関数の適用により, 素性の組合せを考慮した学習が可能となる. Chunk 同定問題, 日本語係受け解析, 日本語未知語品詞推定, 日本語固有表現抽出, 及び文書分類では, 多項式 Kernel 関数の適用した, 素性の組合せを考慮した学習が, 精度に貢献することが報告されている.

SVM は Kernel 関数を適用することで, 高精度な分類規則の学習が可能である一方で, これまでの学習アルゴリズムと比較し, 学習, 及び分類に多く時間を必要とする. また SVM は二値分類器であるため, 複数クラスの分類を必要とする自然言語処理の様々なタスクでは, 学習, 及び分類に要する時間が, 分類すべきクラス数に比例して膨大になる. このため, 実時間での学習, 及び分類を行なうためには, 計算コストの問題が軽視できない.

本稿では, 日本語固有表現抽出を題材に, 二値分類器である SVM を多値分類へ拡張する手法を提案し, 従来の代表的な手法である one vs. rest 法, 及び pairwise 法との比較を, 学習及び分類の効率という観点から行う. 以下次節では IREX 日本語固有表現抽出の説明と, SVM の適用法について説明する. 3 節で SVM を多値分類に拡張する従来手法と新たに提案する手法について述べる. 4 節で実験結果とその考察行い, 5 節でまとめと今後の課題について述べる.

2 日本語固有表現抽出

IREX 日本語固有表現抽出タスク [2] では表 1 に示す 8 種類の固有表現を定義し, それぞれの固有表現は入れ子にはならないとしている.

固有表現抽出は, 入力文中の単語列が固有表現か否かを識別する Chunk 同定問題と見なすことができ, Chunk 同定問題では 一つ以上の要素列からなる Chunk を

単語	固有表現タグ
エリツィン	B-PERSON
大統領	O
は	O
四	B-DATE
日	I-DATE
,	O
日	B-LOCATION
米	B-LOCATION
両国	O

図 1: IOB2 タグ

IOB1, IOB2, IOE1, 及び IOE2 という 4 種類のタグを使用して表記する手法が提案されている [1]. これとは別に, 内元らは複数の単語列からなる日本語固有表現のために Start/End (SE) というタグを使用した表記法を提案している [11]. 本研究ではこの中で実験的によい結果が得られた IOB2 タグを採用した [9].

図 1 は “エリツィン大統領は四日, 日米両国 ···” という文中で, エリツィン:人名 (PERSON), 四日:日付 (DATE), 日:地名 (LOCATION), 米:地名 (LOCATION) という 4 つの固有表現に対して, IOB2 タグを使用した場合の例を示す. IOB2 は, 固有表現の開始に位置する単語に B というタグを付与し, 開始以外の固有表現の単語には I というタグを付与する. 固有表現以外の単語は O というタグを付与する. 以降本稿では, 図 1 に示すような, B-DATE のような表記を**固有表現タグ**と呼ぶ. 固有表現タグを用いることで, 固有表現抽出規則の学習は, 入力文中の各単語を 17 種類 (B-ARTIFACT, I-ARTIFACT, B-DATE, I-DATE, ···, B-TIME, I-TIME, O) の固有表現タグに分ける分類規則の学習として扱うことが可能となる.

2.1 日本語固有表現抽出規則の学習

固有表現抽出規則の学習は, 入力された文の各単語に対し, 固有表現タグに分類する規則を学習することである. また固有表現抽出は, 未知の文に対して, 各単語の固有表現タグを推定することである. そのため, 先ず文を形態素解析し, 単語列に分割する. 固有表現タグの学習, 及び固有表現タグの推定を行なう単位は単語単位であり, 一つの事例は一単語に対応する. このとき, 文頭から順に固有表現タグを推定する方法を**右向き解析**と呼び, 逆に文末から順に推定する方法を**左向き解析**と呼

表 1: IREX で使用する固有表現の種類と頻度分布

固有表現の種類		例	出現率 (%)	(単語数)
ARTIFACT	固有物名	ノーベル文学賞	0.70	(2,042)
DATE	日付表現	五月五日	2.38	(6,936)
LOCATION	地名	日本, 韓国	2.69	(7,854)
MONEY	金額表現	２０００万ドル	0.28	(831)
ORGANIZATION	組織名	社会党	2.35	(6,859)
PERCENT	割合表現	二〇%, 三割	0.33	(975)
PERSON	人名	村山富市	2.04	(5,949)
TIME	時間表現	午前五時	0.41	(1,186)
O	非固有表現	–	88.81	(259,071)
合計				291,703

ぶ. 右向き解析と左向き解析とでは, 学習及び抽出時に使用する素性が異なる. 本研究では, 実験的に高精度の結果が得られた左向き解析を使用した [9].

固有表現抽出規則の学習

左向き解析を行なう場合, 学習時では, 文末から i 番目の単語に関する素性は $i-2$ から $i+2$ 番目までの各単語の, 単語自身, 品詞 (大分類と細分類の両方), 及び文字種を使用する. また複数の単語からなる固有表現を考慮するために, $i-2$ と $i-1$ 番目の固有表現タグ (学習時は既知) も素性として使用する. これらの素性を要素とするベクトル $\mathbf{x}$ と, i 番目の固有表現タグを, 分類すべきクラス y とすれば, $(\mathbf{x}, y)$ という組が一つの事例となる.

固有表現抽出

文末から i 番目の固有表現タグの推定には, 学習時と同様 $i-2$ から $i+2$ 番目までの各単語の, 単語自身, 品詞, 及び文字種を素性として使用する. 未知の文に対する固有表現抽出では, 解析の最初では, 固有表現タグは未知である. このため $i-2$ と $i-1$ 番目の固有表現タグは, 各位置で推定した結果をそのまま使用する. このように各単語の固有表現タグの推定を決定的に行い, 固有表現を抽出する.

図 2 は, 入力文 “大統領は五日午前 $\cdots$” に対して, 左向き解析の場合に使用する素性の例を示す.

入力文が訓練データの場合, 各単語に対する固有表現タグは既知であるため, 文末から i 番目に位置する単語 “五” に関する事例の素性は, $i-1$, 及び $i-2$ 番目の固有表現タグを含め, 枠内の要素をすべて使用する. この素性を要素とするベクトルと, 分類するクラス B-DATE の組が一つの事例となる.

同様の文をテストデータとした場合, 左向き解析では, 文末から順に各単語の固有表現タグを推定していく. “五” の素性は学習時と同様に枠内の要素すべてを使用する. テストデータでは, 最初, 固有表現タグは未知であるため, $i-2$ と $i-1$ 番目の固有表現タグは, i 番目の単語の推定を行う以前に, それぞれの位置で SVM によって推定した結果をそのまま使用する. また推定した “五” の固有表現タグは, 以後の “は” と “大統領” の固有表現タグ推定に素性として使用する.

3 SVM による多値分類

SVM を多値分類に拡張する手法について説明する.

pairwise 法

pairwise 法は k 個のクラスから任意の 2 つのクラスに関する二値分類器を ${}_k\mathrm{C}_2$ 個構築する方法である. 今, クラス s とクラス t を分類する二値分類器 $f_{st}(\mathbf{x})$ を考える. $f_{st}(\mathbf{x})$ は, $f_{st}(\mathbf{x}) \geq 0$ のとき, 事例 $\mathbf{x}$ を クラス s と判定し, 反対に $f_{st}(\mathbf{x}) < 0$ のときクラス t と判定する. クラス c の投票数 V_c を, ${}_k\mathrm{C}_2$ 個ある二値分類器のうちクラス c と判定した分類器の個数とする. pairwise 法での最終的な分類クラスは, 投票数 V_c が最も多いクラスに決定される. 図 3 に pairwsie 法の例を示す. A, B, 及び C の 3 クラスの分類を行なうために, 任意の 2 つのクラスを分類する二値分類器, $f_{AB}(\mathbf{x})$, $f_{AC}(\mathbf{x})$, 及び $f_{BC}(\mathbf{x})$ を構築する.

位置	$i+2$	$i+1$	i	$i-1$	$i-2$
入力文	大統領	は	五	日	午前
品詞	名詞	助詞	名詞	名詞	名詞
文字種	漢字	平仮名	漢字	漢字	漢字
固有表現タグ	O	O	B-DATE	I-DATE	B-TIME

図 2: 使用する素性

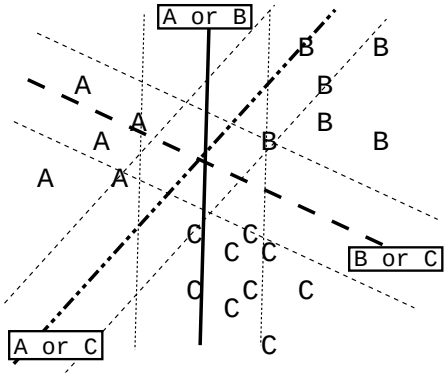

図 3: pairwise 法

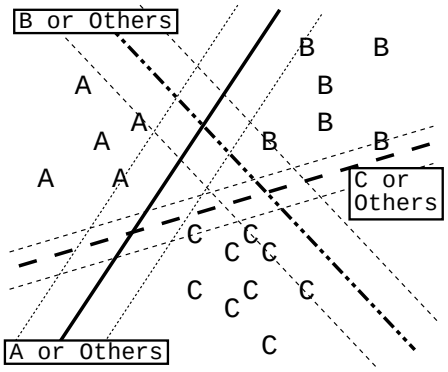

図 4: one vs. rest 法

one vs. rest 法

one vs. rest 法は, k 個の各クラスに対して, あるクラスか, それ以外かという二値分類器 $f_c(\mathbf{x})$ を, k 個構築する手法である. SVM を用いた場合, 未知事例に対して分類を行う場合, 未知事例 $\mathbf{x}'$ に対し, $f_c(\mathbf{x}')$ の値が最大となる分離器に対応するクラスに決定される. 図 4 に one vs. rest 法の例を示す. A, B, 及び C の 3 クラスを分類するために, $f_A(\mathbf{x})$, $f_B(\mathbf{x})$, 及び $f_C(\mathbf{x})$ の 3 つの二値分類器を構築する.

Sequential Binary Tree (SBT) 法

Schwenker らは, 手書き数字文字認識における多値分類問題において, 二分木の各ノードが, 二値分類器である木構造の多値分類器を構築する手法を提案している [6]. 彼らは, 分類する複数のクラスに対し, K-mean (k=2) クラスタリングを使用して二分木を構築し, 各ノードでの二値分類に SVM を使用している. これにより, ルートノードでは, 比較的分類が容易な二値分類が行われ, また, ルートノードから葉にいくにつれ, 各二値分類で使用する学習事例が減少する. これにより非常に効率のよい学習, 及び分類が可能である.

表 1 に示すように, 固有表現抽出では, 9 割近い事例が一つのクラス (非固有表現である O) に属する偏った頻度分布である. このような状況では, pairwise, 及び one vs .rest 法は 訓練データすべてを扱う学習と同等な規模の学習を, 17(pairwise 法では 16) 回行なう必要があり, 非効率である.

そこで Schwenker らの手法を応用し, 分類すべき 17 種類の固有表現タグに順序を定義し, 17 個のノードをもった二分木を構築をする. ここで順序は, 訓練事例中に出現した各固有表現タグの頻度の降順と定義し, 二分木を構築するコストを軽減する. この二分木の多値分類器を Sequential Binary Tree(SBT) 法と呼ぶことにする (図 5 に構築した二分木の例を示す). SBT 法では, 各ノードで正例として使用した学習事例は, 以降の子ノードの学習に使用しない. 従って, ルートノードにおいて, O タグかそれ以外かという二値分類を学習すれば, 次以降の子ノードでは O 以外の事例だけで学習が行われる. その結果, 学習データすべてを扱う規模の学習は, 一度だけに軽減できる.

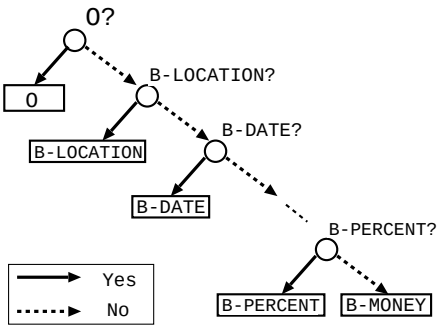

図 5: Sequential Binary Tree 法

また未知の事例に対し分類を行う場合, paiwise, 及び one vs. rest 法では, 構築した二値分類器すべてに対し分類結果を求める必要がある. しかし SBT 法ではルートノードから順に各ノードで構築した SVM で分類を行い, 葉に到達した場合 (正例と判定された場合), そこで分類クラスが決定する. 従って, すべての分類器に分類結果を求める頻度が減少し, 効率的な分類が期待できる.

SVM の分離平面は Support Vector と呼ばれる分離に寄与する訓練事例の重み付き線形和で構成される. 各ノードにおける学習で 正例の Support Vector となった訓練事例は, 他のクラスの学習にも重要な事例であると考え, 以降の子ノードの学習において, 負例として使用する別の手法を試みる (SBT' 法と呼ぶ). これにより one vs. rest とほぼ等価な学習が期待でき, かつ学習に必要な時間を削減できる.

4 実験

データ

実験には CRL (郵政省通信総合研究所) 固有表現データを使用した. CRL 固有表現データは, 毎日新聞 95 年度版 1,174 記事, 約 11,000 文に対して IREX で定義された固有表現がタグ付けされている. このデータ中の固有表現の総数は 19,262 個であった. 形態素解析器は茶筌 [10] を使用し, 評価は CRL 固有表現データを 5 等分に分割し, 訓練 4, テスト 1 の比率で交差検定を行ない, それらの総合の F 値 ($\beta = 1$) を使用した. また実験で使用する Kernel 関数は, 日本語固有表現抽出で高精度な結果が得られた 2 次の多項式関数 $K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^2$ を採用した [9].

実験結果と考察

結果を表 2 に示す. 表 2 で, 学習, 及び抽出時間は, paiwise 法での平均時間を 1 とした場合の比率を表す[1].

表 2 より, one vs. rest 法が最良で, F 値で 83.7 という高精度の抽出結果を得た. また SBT' 法 は SBT 及び, pairwise 法に比べ良い精度を得ることができた.

最高の抽出精度であった one vs. rest 法は, 学習に要した時間では 4 つの手法の中で最長であった. これは訓練事例すべての使用する規模の学習が, 分類するクラス数回行われるためである. pairwise 法と SBT 法を比較すると, 事例の数が減少し, 効率的な学習を行なう SBT 法のほうが学習時間が長かった. この原因について調査した. pairwsie 法で最も学習データが多い二値分類は, O タグと B-LOCATION についてであった. この学習には約一時間半を要した. また学習モデルの複雑さを示す指標の一つに Leave-one-out(LOO) bound と呼ばれるものがあり, SVM では LOO bound を, 訓練データ総数に対する Support Vector となった訓練事例数の比で計算できる. O タグと B-LOCATION の分類における LOO bound は 0.0124 であった. これに対して SBT 法では, 最も学習データが多いルートノードにおける O タグか否かという二値分類学習では, 約 19 時間を要した. これは全体の学習時間の 9 割に近い時間で, またこのときの LOO bound も 0.0642 と pairwise 法にくらべ非常に高い値となった. これは O か否かという分類が, 本質的に難しい分類問題であることを示唆している. SVM の学習時間は, 学習データ数だけでなく, 分類問題の本質的な難しさにも強く依存する. SBT 法は ルートノードから順に学習事例を減少させ, 多値分類学習を効率化している. しかし日本語固有表現抽出では, ルートノードで行う固有表現か否かという分類自身が難しい分類問題であるため, 結果的に pairwise 法よりも多くの学習時間が必要となった.

抽出時間は, 従来手法である pairwise 法, 及び one vs. rest 法は構築したすべての二値分類器に分類結果を求める必要があるため, ほぼ同じ時間を必要とした. これ対し SBT, 及び SBT' 法は, 各ノードで正例と判定されると, その時点で分類クラスが決定され, 以降子ノードでの分類を必要としない. これにより従来手法と比

[1] 学習に使用した計算機の CPU は PentiumIII 600MHz で, 一回の交差検定は平均約 8800 文 (240,000 事例) の学習を行い, pairwise 法では約 20 時間であった. 固有表現抽出では, PentiumIII 933MHz の計算機を使用し, pairwise 法では約 1880 文を解析するのに, 約 40 分であった

表 2: 実験結果

	$F_{\beta=1}$ 値			
	pairwise	one vs rest	SBT	SBT'
ARTIFACT	47.5	48.4	45.4	46.5
DATE	91.9	92.6	91.7	92.3
LOCATION	82.3	83.2	82.7	82.8
MONEY	94.0	94.0	92.0	92.6
ORGANIZATION	78.9	79.3	77.6	78.5
PERCENT	94.3	94.6	91.6	93.7
PERSON	86.1	86.6	85.1	86.3
TIME	89.0	89.5	86.9	87.9
総合	83.0	83.7	82.4	83.1
学習時間	1	2.63	1.02	1.24
抽出時間	1	1.13	0.61	0.61

べ, 抽出速度を約 2/3 に短縮することができた.

5 まとめと今後の課題

本研究では, 日本語固有表現抽出タスクを題材に, 二値分類器である SVM を多値分類に拡張する手法を提案し, 代表的な従来手法である pairwise 法, 及び one vs. rest 法との比較を行なった. 日本語固有表現抽出タスクのように, 分類するクラスの頻度が偏った分布である場合, 従来法では学習, 及び分類において非常に効率が悪い. これらの学習, 及び分類の効率を改善するために, クラスの頻度の降順で順序を定義し, 二分木を構築する Sequential Binary Tree 法を提案した. 固有表現抽出実験では, 従来法と比較し, ほぼ同等な精度が得られ, かつ抽出時間を削減できることが確認できた.

今後は, Schwenker らが提案した, クラスタリング手法を用いた二分木構築法と比較を行うことが挙げられる. また, 今回提案した多値分類手法を, 文書分類などのタスクに適用し, 有効性を検証する予定である.

参考文献

[1] Erik F. Tjong Kim Sang and Jorn Veenstra. Representing text chunks. In *In Proceedings of EACL'99*, pp. 173–179, 1999.

[2] IREX 実行委員会. IREX ワークショップ予稿集, 1999.

[3] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pp. 137–142, 1998.

[4] Kudo Taku, Yuji Matsumoto. Japanese Dependency Analysis Based on Support Vector Machines. In *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 18–25, 2000.

[5] Tetsuji Nakagawa, Taku Kudoh, and Yuji Matsumoto. Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines. In *Proceedings of 6th Natural Language Processing Pacific Rim Symposium*, 2001. (to appear).

[6] Friedhelm Schwenker and Günther Palm. Tree-Structured Support Vector Machines for Multi-class Pattern Recognition. In *Multiple Classifier Systems*, pp. 409–417, July 2001.

[7] Taku Kudoh and Yuji Matsumoto. Use of Support Vector Learning for Chunk Identification. In *Computational Natural Language Learning (CoNLL-2000)*, pp. 142–144, 2000.

[8] Yiming Yang , Xin Liu. A Re-examination of Text Categorization Methods. In *SIGIR '99 *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval, University of California, Berkeley*, pp. 42–49, 9 Aug 1999.

[9] 山田 寛康, 工藤 拓, 松本 裕治. Support Vector Machine を用いた日本語固有表現抽出. 情報処理学会研究会報告, No. NL-142-17, pp. 121–128, 2001.

[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, 12 1999.

[11] 内元 清貴 , 馬 青 , 村田 真樹 , 小作 浩美 , 内山 将夫 , 井佐原 均. 最大エントロピーモデルと書き換え規則に基づく固有表現抽出. 自然言語処理, 第 7 巻, pp. 63–90, 2000.

[12] 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. 情報処理学会論文誌, 第 41 巻, pp. 1113–1123, 4 2000.