情報処理学会論文誌 Vol.58 No.12 2046–2059 (Dec. 2017)

同一カテゴリ内での二値判別を許容する符号表に基づくECOC多値判別法

鈴木 玲央奈[1,a)] 山下 遥[2,b)] 後藤 正幸[1,c)]

受付日 2017年3月23日, 採録日 2017年9月5日

概要：本研究では，多くの適用事例を有する多値判別手法の中でも，二値判別器を組合せによって多値判別を行うECOC法に着目する．ECOC法は，各行にカテゴリ，各列に二値判別器構成を表した符号表により判別器を構成し，二値判別器の出力結果から新規データのカテゴリを推定するものである．本研究では，より複雑な多値分類問題における分類精度向上のために，同一カテゴリ内での二値判別を許容する符号表を提案する．また，同一カテゴリ内での二値判別を行わない従来の考え方に基づく符号表と組み合わせた手法も提案する．さらに，所属カテゴリを推定する手順において，提案する符号表に適した多数決方式によるカテゴリ判別方法を提案する．実ベンチマークデータを用いた実験により，提案手法の有効性を示す．

キーワード：ECOC法，多値判別問題，符号表，復号，テキスト分類

A Study of ECOC Multi-category Classification Approach Based on Code Table Considering Binary Classification in Same Category

Leona Suzuki[1,a)] Haruka Yamashita[2,b)] Masayuki Goto[1,c)]

Received: March 23, 2017, Accepted: September 5, 2017

Abstract: We focus on the ECOC approach which combines binary classifiers to the multi-category classification. The ECOC approach consists of two steps: designing code table for making classifiers and decoding for predicting category of a new input data. In the step of designing code table, it constructs a code table whose row represents each category and column represents configuration of each binary classifier. In the decoding step, the category of a new input data is predicted by integrating outputs of all binary classifiers. In this study, we improve both steps of the ECOC method, i.e., code table designing and decoding considering the classification accuracy. For the coding step, we propose the code table which allows us to classify in same category in order to grasp complex property, and also propose a method combined with the code table without classifying in the same category. For the decoding step, we propose a new method based on majority rule which is suitable for proposed code table. Through the simulation experiment with data set of news paper articles, we show the effectiveness of the proposed method.

Keywords: ECOC approach, multi-category classification, coding and decoding, text classification

1. 研究背景・目的

カテゴリ数が3以上の分類を扱う多値判別問題は，広い適用範囲を持つ重要な課題である．多値判別問題の解法は，単一の判別器で多値判別を行うアプローチ，複数の二値判別器を組み合わせるアプローチ [1], [2], [3] の2つに大別される．

前者のアプローチでは，単一の多値判別器を学習する必要があるため，一般に複雑な問題を解く必要があり，長い計算時間を要する．一方で，二値判別器はより単純な問題として扱うことができるため，多値判別器と比べて短い時

[1] 早稲田大学
Waseda University, Shinjuku, Tokyo 169–8555, Japan
[2] 上智大学
Sophia University, Chiyoda, Tokyo 102–0081, Japan
[a)] davinci.1031@suou.waseda.jp
[b)] h-yamashita-1g8@sophia.ac.jp
[c)] masagoto@waseda.jp

間での学習が可能である．このアプローチは，独立に学習して得られた二値判別器を用いて多値判別器を構成するため，同時に並行して学習を行った複数の二値判別器を統合して，多値判別を行うことが可能となる．

以上より，本研究では，二値判別器を組み合わせるアプローチに着目し，またその中でも分類精度が高いと知られている Error-Correcting Output Coding 多値判別法 [1]（以下，ECOC 法）を扱う．

ECOC 法は，符号表生成と復号の 2 つのステップから構成される．符号表生成のステップにおいては，各行にカテゴリ，各列に二値判別器の構成を表した符号表と呼ばれる数値表を生成する．判別を意味する復号のステップでは，新規データに対する複数の二値判別器の出力を統合したベクトルを用いて，新規データの所属カテゴリを推定する．符号表生成に対するアプローチには，適応的に符号表を生成する方法 [4], [5], [6] と学習前に符号表の全構成を与える方法（非適応的な方法）[7], [8] の 2 通りが存在する．前者では，判別器の学習結果を反映して適応的に符号表を生成していくため，高い分類性能が期待できる反面，個々の判別器の学習に対して並列処理ができず，学習において計算時間がかかってしまうという問題がある．一方の非適応的な手法は，各二値分類器の学習前に符号構成表が与えられるため，それぞれの二値分類器を並列計算できるという大きなメリットがある．よって並列計算が可能な手法の場合，R 個の判別器の学習にかかる演算時間が並列処理によって 1/R 程度で済むため，多くの判別器を統合して判定する ECOC 法の 1 つの強力なメリットと考えられる．そこで本研究では，並列処理を行うことのできる非適応的なアプローチのみに着目する．

本研究では，符号表生成と復号の双方のステップに対して，新たな ECOC 多値分類法を提案する．これにより，分類精度の向上を期待することができる．まず，符号表生成では，データに対して適応的にサブクラスを形成しつつ，得られたサブクラスを基にし，符号表を判別器の学習を行う前に生成する．これにより，データに対して適応的であり，なおかつ二値判別器の学習に対して並列処理が可能である，適応的な生成法と非適応的な生成法の双方の長所を有する生成アルゴリズムを実現する．その際，各行が 1 つの学習データに対応する符号表へと変換を行う．この変換は同一カテゴリ内での二値判別を許容し，上記の同一カテゴリ内で二値判別を行わないという制約を解消する．これにより，分類が容易である部分データを判別する精度の高い二値判別器を多数生成し，全体としての多値判別の精度を向上させることが期待できる．さらに，復号のステップに着目し，1 つ 1 つの学習データに符号語を対応させる符号表に適した推定方法を提案する．上記 2 つの提案の有効性についてベンチマークデータを用いた多値判別問題に適用し，検証を行う．

2. 準備

2.1 ECOC 法

本研究では，入力 $\boldsymbol{x}$ に対し，その所属カテゴリ c_k（$1 \leq k \leq K$）を推定する問題を扱う（$K \geq 3$）．ECOC 法は，符号理論で用いられる誤り訂正技術を多値判別問題に応用した手法であり，カテゴリが未知の新規データに対し，複数の二値判別器を組み合わせることにより，所属カテゴリを推定する．複数の二値判別器の構成は符号表と呼ばれる {1,0} の二値を要素とする数値表により表現される．いま，二値判別器の個数を R とすると，符号表 $\boldsymbol{W}$ は $K \times R$ 行列で与えられる．ここに，符号表 $\boldsymbol{W}$ の各列ベクトルは二値判別器の構成を表現しており，要素が 1 のカテゴリ集合と要素が 0 のカテゴリ集合を二値判別するための R 個の判別器が学習される．そのため，0 と 1 が反転した列ベクトルは，等価な二値判別器を意味する．また，符号表 $\boldsymbol{W}$ の k 行目の行ベクトルをカテゴリ c_k の符号語と呼び $\boldsymbol{w}_k$ で表す．新規データの所属カテゴリを推定する際には，新規データに対する二値判別器の出力結果ベクトルと各カテゴリに与えられている符号語の比較により，分類を行う．

符号表の中には {1,0} の二値で表される二元符号表のほかに，判別に用いないカテゴリを許容した三元符号表がある．ここでは，判別に用いないカテゴリを $*$ で表す．$\boldsymbol{w}_k$ の r 番目の値を w_k^r とすると，w_k^r が $*$ の場合には r 番目の判別器を学習する際に，カテゴリ c_k の学習データは除外される．そのため，三元符号表は各二値判別器で用いる学習データ数を減少させ，二元符号表に対して学習計算量の低減を可能とする．

2.2 従来の符号表生成法

符号表の生成法については，これまで数多く提案されている．適応的な符号表生成と非適応的な符号表生成の 2 つの観点から，従来の符号表生成法について述べる．

2.2.1 適応的な符号表生成法

近年，多くの適応的な符号表生成手法が提案されている．これらの手法は，符号語間のハミング距離は考慮せずに，データの構造に着目し，適応的に符号表の生成を行う．Crammer ら [9] は，誤り率の最小化を行いつつ，判別器数の少ない符号表を生成する手法を提案している．また，適応的にカテゴリにおける木構造を学習する手法も存在する．Pujol ら [4] は，相互情報量 [10] を用いた分類基準を最大化しつつ，木構造で表される階層的な符号表を生成する Discriminant ECOC を提案している．木構造を用いた手法は，このほかにも多く提案されている [11], [12], [13], [14]．さらには，サブクラスと呼ばれるカテゴリをデータの特徴により分割した部分集合を活用する手法 [6], [15], [16] も存在する．これらの手法は，サブクラスへの分割を行うことで，各二値判別器における判別を容易にし，符号表全体と

表 1 一対他法 ($K = 5$)
Table 1 Code table of the 1-vs-the rest method ($K = 5$).

		判別器				
		1	2	3	4	5
カテゴリ	c_1	1	0	0	0	0
	c_2	0	1	0	0	0
	c_3	0	0	1	0	0
	c_4	0	0	0	1	0
	c_5	0	0	0	0	1

表 2 一対一法（$K = 5$）
Table 2 Code table of the 1-vs-1 method ($K = 5$).

		判別器									
		1	2	3	4	5	6	7	8	9	10
カテゴリ	c_1	1	1	1	1	∗	∗	∗	∗	∗	∗
	c_2	0	∗	∗	∗	1	1	1	∗	∗	∗
	c_3	∗	0	∗	∗	0	∗	∗	1	1	∗
	c_4	∗	∗	0	∗	∗	0	∗	0	∗	1
	c_5	∗	∗	∗	0	∗	∗	0	∗	0	0

しての分類精度の向上を図っている．このほかにも，様々な適応的な符号表生成の手法が提案されている [17], [18]．しかしながら，これらの適応的な判別器の学習を行いながら符号を構成していく手法であるため，各判別器の学習に対して並列処理ができず，学習における計算時間が並列処理が可能な場合に比べて増大するといった問題点がある．

2.2.2 非適応的な符号表生成法

本研究では，各二値判別器の学習に対して並列処理が可能である非適応的な符号表生成による ECOC 法に着目する．以下では，代表的な非適応的な符号表生成法について述べる．

一対他法による ECOC 法

1 つのカテゴリとそれ以外を二値判別する判別器をカテゴリ数分用意する符号表構成であり，$R = K$ となる．**表 1** は，$K = 5$ の場合の一体他法の例であり，判別器数がカテゴリ数と一致していることが分かる．この方法は非常にシンプルな方法であるものの，有効な手法であると知られている [19]．

一対一法による ECOC 法

一対一法とは，考えられるすべてのカテゴリの組合せに対して二値判別器を用意する判別器構成であり，$R = {}_K C_2$ となる．**表 2** は，$K = 5$ の場合の一対一法の例であり，$R = {}_5C_2 = 10$ である．一対一法では，判別器を学習するためのデータ数が抑えられるため，計算時間が非常に小さくなる一方で，分類精度も低くなるといった特徴がある．

ランダム符号を用いた ECOC 法

カテゴリのカテゴリ集合への分割を，ある一定のルールに従ってランダムに行うことにより符号表を生成する方法が存在する [7]．この方法により生成されたランダム符号は，ランダム基準に基づいて生成されるため，高い性能を得る保証はない．そのため，これらの手法はベンチマークの手法として用いられるか，繰り返し処理により符号表を生成する手法の初期値の符号表として用いられることが多い [20]．

ランダム符号には 2 つのタイプが存在する．Dense random 符号と Sparse random 符号である．Dense random 符号では，$\{0, 1\}$ を各要素にランダムに割り当てた符号表を多く生成する．また，生成される符号表の判別器数は，$\lceil 10 \log_2 K \rceil$ である．ここで，$\lceil \cdot \rceil$ は天井関数であり，$\lceil x \rceil$ は x 以上の最小の整数である．そして，得られた複数の符号表の中から，符号語間の最小ハミング距離が最も大きいものを選択し，それを用いて ECOC 法を行う．

Sparse random 符号は，Dense random 符号と同様の手順により生成される．各要素は，$\{0, 1, *\}$ であり，$*$ は 0.5 の確率で，0 または 1 は 0.25 の確率でランダムに割り当てられる．また，判別器数は $\lceil 15 \log_2 K \rceil$ である．

BCH 符号を用いた ECOC 法

BCH 符号とは，符号理論の分野で知られている効率的な符号である．BCH 符号は 2 つの要因によって一意に定まる．それは，誤り訂正能力と符号長である．ここで，符号長，情報記号列長，誤り訂正能力をそれぞれ υ，κ，τ とする．BCH 符号のこれらのパラメータにより (υ, κ, τ) と表される．BCH 符号により ECOC 法を行う際には，υ と $2\tau + 1$ がそれぞれ判別器数 R と最小ハミング距離に対応する．そのため，BCH 符号を用いた ECOC 法では，最小ハミング距離を自由に設定することが可能である．上記で述べたとおり，ECOC 法において，カテゴリ間でのハミング距離が大きいと識別が容易である．そのため，最小ハミング距離を自由に設定することが可能である BCH 符号は，ECOC 法に適した符号である．

表 3 は $K = 8$ の場合の $(15, 5, 3)$ BCH 符号の例である．

Reed Muller 符号を用いた ECOC 法

Reed Muller 符号（RM 符号）[21] とは，符号理論で用いられる符号の 1 つであり，ECOC 法との相性が良いことが知られている．RM 符号を用いるとカテゴリ数が 2 の冪乗の場合は，符号表の各列の 0 と 1 の数が $K/2$ ずつになる．そのため，各カテゴリの学習データ数が等しい場合，カテゴリ集合間で学習データ数が等しくなる．加えて，「各カテゴリの符号語間のハミング距離が大きい」，「各判別器間のハミング距離がすべて等しい」という特徴もある．これらの 3 つの特徴から，カテゴリ数が 2 の冪乗の場合には，RM 符号による ECOC 法により分類精度の高い判別器構成が可能である．

表 4 は，$K = 8$ の場合の，RM 符号である．すべての列において 0 と 1 の数が 4 つずつとなっており，各カテゴリの学習データ数が等しい場合，カテゴリ集合間で学習データ数が等しくなることが分かる．

表 3 (15, 5, 3) BCH 符号（$K = 8$）
Table 3 Code table of the (15, 5, 3) BCH code ($K = 8$).

		判別器														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
カテゴリ	c_1	1	0	1	0	0	1	1	0	1	1	1	0	0	0	0
	c_2	0	1	0	1	0	0	1	1	0	1	1	1	0	0	0
	c_3	0	0	1	0	1	0	0	1	1	0	1	1	1	0	0
	c_4	0	0	0	1	0	1	0	0	1	1	0	1	1	1	0
	c_5	0	0	0	0	1	0	1	0	0	1	1	0	1	1	1
	c_6	1	1	1	1	0	1	0	1	1	0	0	1	0	0	0
	c_7	1	0	0	0	1	1	1	1	0	1	0	1	1	0	0
	c_8	1	0	1	1	0	0	1	0	0	0	1	1	1	1	0

表 4 RM 符号における符号表（$K = 8$）
Table 4 Code table of the RM code ($K = 8$).

		判別器						
		1	2	3	4	5	6	7
カテゴリ	c_1	0	0	1	0	1	1	0
	c_2	1	0	0	0	0	1	1
	c_3	0	1	0	0	1	0	1
	c_4	1	1	1	0	0	0	0
	c_5	0	0	1	1	0	0	1
	c_6	1	0	0	1	1	0	0
	c_7	0	1	0	1	0	1	0
	c_8	1	1	1	1	1	1	1

RM 符号を用いた手法は上記のように，カテゴリ数が 2 の冪乗の場合は，カテゴリ集合間で学習データ数を等しくすることができるというメリットがある．一方でカテゴリ数が 2 の冪乗でない場合，K 以上で最小の 2 の冪乗行となる RM 符号から，K 行を選択して適用する必要がある．そのため，たとえ各カテゴリの学習データ数が等しい場合でも，カテゴリ集合間で学習データ数がアンバランスになるという問題点がある．

Exhaustive 符号を用いた ECOC 法

Exhaustive 符号とは [1]，Dietterich らによって考案された符号表であり，$2^{K-1} - 1$ 個の考えられるすべての 2 群分類に対する判別器を用いる判別器構成となっている．すなわち，上記で述べた 3 つの二元符号表（一対他法，Dense random 符号，BCH 符号）を内包している符号表である．Exhaustive 符号の構成法は次のように与えられる．

(1) $\boldsymbol{w}_1$ はすべて 1 で構成する．
(2) $k = 2, 3, \ldots, K$ において $\boldsymbol{w}_k$ は 2^{K-k} 個の連続する 0 と 1 を交互に $R = 2^{K-1} - 1$ 個になるまで並べて構成する．

表 5 は，$K = 4$ のときの Exhaustive 符号の例であり，$2^{4-1} - 1 = 7$ 個の判別器から構成されている．Exhaustive 符号は考えられるすべての 2 群分類に対する判別器を用いるため K が大きいと，判別器数は非常に多くなる．それにより，高い分類精度が期待できる．一方で，判別器の学習に膨大な時間を要してしまう．またすべての二値判別器を用いることが必ずしも新規入力データの分類精度の面から最適とはいえず，より少ない二値判別器で分類した方が精度が高くなるという場合があるという実験結果も示されている．

表 5 Exhaustive 符号における符号表（$K = 4$）
Table 5 Code table of the Exhaustive code ($K = 4$).

		判別器						
		1	2	3	4	5	6	7
カテゴリ	c_1	1	1	1	1	1	1	1
	c_2	0	0	0	0	1	1	1
	c_3	0	0	1	1	0	0	1
	c_4	0	1	0	1	0	1	0

2.3 従来の復号における分類基準

新規入力データに対する分類基準はいくつか存在するが [22], [23] ここでは主な 2 つについて述べる．以下では判別器の出力値が 1 つのカテゴリのみを選ぶ判定を硬判定，カテゴリに属する確率である判定を軟判定と呼ぶ．

従来の復号法における分類基準として，類似度最大に基づく分類基準が存在する．入力 $\boldsymbol{x}$ に対する $r(1 \leq r \leq R)$ 番目の二値判別器の出力を $g_r(\boldsymbol{x})$ としたとき，類似度最大に基づく分類基準では，符号語 $\boldsymbol{w}_k$ と $\boldsymbol{g} = (g_1(\boldsymbol{x}), ..., g_R(\boldsymbol{x}))^T$ の類似度 $S(\boldsymbol{w}_k, \boldsymbol{g})$ が最大となるカテゴリ $c_{\hat{k}}$ の番号 $\hat{k}$ を式 (1) により導出し，分類する．

$$\hat{k} = \arg\max_k S(\boldsymbol{w}_k, \boldsymbol{g}) \tag{1}$$

本研究では，二値判別器として判別精度が高く，出力値が軟判定となる RVM [24] を用いる．軟判定を用いた際の新規入力に対する類似度 S は以下の式 (2) により算出される．

$$S(\boldsymbol{w}_k, \boldsymbol{g}) = \prod_{r=1}^{R} g_r^{w_k^r} (1 - g_r)^{1 - w_k^r} \tag{2}$$

また，類似度 S の代わりにハミング距離を用いて，ハミング距離が最小のカテゴリへと分類する，最小距離復号と

呼ばれる手法も存在する．類似度最大に基づく分類基準，最小距離復号のどちらの手法も，二値判別器の出力例から，最も類似性の高いまたは最も距離が小さい符号語を 1 つ選ぶ操作となっている．

3. 提案手法

3.1 提案手法の着眼点

提案手法では，符号表生成と分類基準の双方に着目し，新たな ECOC 法のアプローチを提案する．まず，符号表生成に関しては，サブクラスを用いた符号表生成を考える．サブクラスとは，カテゴリをデータの特徴により分割した部分集合であり，たとえば，新聞記事における「スポーツ」カテゴリには，「プロ野球」,「J リーグ」,「オリンピック」などの異なる話題を持ったサブクラスが含まれると考えられる．

これらのサブクラスを考慮した手法は 2.2.1 項で述べたように，文献 [6], [15], [16] など，数多く提案されている．しかしながら，これらの手法は適応的な手法であり，得られた判別器の結果を符号表の構成に反映させるアプローチでとなっている．したがって，判別器を学習し得られた判別器を評価することで，次に学習すべき判別器を明らかにするため，実際にデータを学習させながら符号表が構成されていくことになる．よって，複数の二値判別器の学習を同時に並列処理することができないため，学習における計算時間に対して効率性に問題が存在する．

さらに，これらのサブクラスを考慮した適応的な手法は，同一カテゴリ内に性質が異なる複数のサブクラスが存在する場合や，異なるカテゴリに属するサブクラスどうしの特徴が類似している場合について，複雑なデータ構造を二値判別器の学習に反映させることは難しい．たとえば，「スポーツ」カテゴリの中でも，「オリンピック」の話題を持ったサブクラスは「国際」カテゴリにおける「オリンピック」関連のサブクラスと類似した特徴を持っているものと思われる．このような場合に，従来の符号表生成法では，異なるカテゴリ間でのサブクラスの特徴の類似性を考慮することができず，多値判別の精度向上に貢献するような二値判別器の構成が困難となる可能性がある．

以上のように，従来のサブクラスを用いた符号表生成法では，「各二値判別器の学習を並列に行うことができない」,「異なるカテゴリ間でのサブクラスの特徴の類似性を考慮できない」という 2 つの問題点がある．そこで，提案する 3 つの手法（以下，提案手法 1，2，3）ではデータに対して適応的にサブクラスを形成しつつ，得られたサブクラスの特徴を基にサブクラス間の特徴の類似性を考慮した符号表を判別器の学習を行う前に生成する．このようなアプローチにより，データを学習する以前から符号構成表を与えられ，同時に別々のマシン上で並列学習をすることが可能となる．すなわち，R 個の判別器をすべて並列処理学習する場合，$1/R$ 程度で済むため，多くの判別器を統合して判定する ECOC 法においては，1 つの強力なメリットとなることが期待される．これにより，異なるカテゴリ間でのサブクラスの類似性を考慮しつつ，なおかつ二値判別器の学習に対して並列処理が可能である生成アルゴリズムを提案することができる．

表 6 サブクラスを用いた符号表
Table 6 Code table using the subclasses.

		判別器				
c_k	$c_{k,j}$	1	2	3	$\cdots$	R
c_1	$c_{1,1}$	1	0	0	$\cdots$	$*$
	$c_{1,2}$	$*$	$*$	0	$\cdots$	0
	$c_{1,3}$	$*$	1	$*$	$\cdots$	0
c_2	$c_{2,1}$	0	0	1	$\cdots$	1
	$c_{2,2}$	0	$*$	1	$\cdots$	$*$
	$c_{2,3}$	1	1	$*$	$\cdots$	1
c_3	$c_{3,1}$	1	0	0	$\cdots$	$*$
	$c_{3,2}$	1	$*$	0	$\cdots$	0
	$c_{3,3}$	$*$	1	$*$	$\cdots$	0

ここでカテゴリ c_k から分割された j（$1 \leq j \leq C$）番目のサブクラスを $c_{k,j}$ とすると，サブクラスを用いた符号表は**表 6** のようになる．このように，各サブクラスに 1 つの符号語を対応させることで，同一カテゴリ内においても，サブクラスごとに異なるカテゴリ集合と見なして二値判別を行うための判別器を構成することが可能となる．

このように，提案する符号表は，各カテゴリが複数の符号語を持つ．しかしながら，従来の符号表では，各カテゴリは単一の符号語を持ち，分類基準に関してもこれを前提とした分類基準が適用されている．そこで本研究では，各カテゴリが複数の符号語を持つ符号表に適した分類基準についても考えることにする．

3.2 符号表生成アルゴリズム

この節では，提案する符号表生成アルゴリズムについて述べる．符号表生成アルゴリズムは，1) ランダム性のあるサブクラス生成，2) 二値判別器構成の決定（0，1，$*$ の割当て）で構成され，1) と 2) の繰り返しにより符号表を生成する．まず，サブクラスの生成では，各カテゴリごとにあらかじめ決められた数のサブクラスを構成するための代表ベクトルをランダムに選択し，代表ベクトルとの距離による学習データのクラスタリングにより，サブクラスを生成する．これにより，データの特徴からカテゴリをサブクラスへと分割することが可能となる．

次に，2 つ目の手順の二値判別器構成の作成について述べる．本研究では，個々の二値判別器の精度を向上させるために，同一カテゴリ内で二値判別を許容する符号表の生成を行う．しかし，同一カテゴリ内で二値判別を許容する符号表のみを用いると，カテゴリ間での識別の劣化が懸念

表 7 同一カテゴリ内での二値判別を許容しない符号表
Table 7 Code table not allowing us to classify in same category.

判別器					
c_k	$c_{k,j}$	1	2	3	···
c_1	$c_{1,1}$	1	1	∗	···
	$c_{1,2}$	1	∗	1	···
	$c_{1,3}$	∗	1	1	···
c_2	$c_{2,1}$	0	0	∗	···
	$c_{2,2}$	0	∗	0	···
	$c_{2,3}$	∗	0	0	···
			⋮		

表 8 同一カテゴリ内での二値判別を許容する符号表
Table 8 Code table allowing us to classify in same category.

判別器					
c'	$c_{k,j}$	1	2	3	···
c'_1	$c_{1,1}$	1	1	∗	···
	$c_{2,1}$	1	∗	1	···
	$c_{2,2}$	∗	1	1	···
c'_2	$c_{1,2}$	0	0	∗	···
	$c_{1,3}$	0	∗	0	···
	$c_{2,3}$	∗	0	0	···
			⋮		

される．そこで，同一カテゴリ内で二値判別を許容する符号表と（従来の考え方に基づく）同一カテゴリ内で二値判別を許容しない符号表との 2 つを作成し，その 2 つを適度にマージすることで分類精度の向上を図る．

3.2.1 同一カテゴリ内で二値判別を許容しない符号表の生成

まず，同一カテゴリ内で二値判別を許容しない判別器構成の決定では，∗ を用いてサブクラス間で符号語に差異が生じるように判別器構成を表現する．

表 7 は，サブクラスを用いた同一カテゴリ内での二値判別を許容しない符号表の例である．c_1 に着目すると，各サブクラスごとで符号語に差異はあるものの，1 または ∗ が割り当てられており，0，1 の割付けによる二値判別は行われていない．一方で，c_1 と c_2 を比較すると，0，1 の割当てにより二値判別が行われていることが分かる．ここで述べた二値判別器構成の決定手順は，3.2.4 項で述べるアルゴリズムの Step2 にあたる．

3.2.2 同一カテゴリ内で二値判別を許容する符号表の生成

同一カテゴリ内で二値判別を許容する判別器構成の決定では，ランダム生成で得られるサブクラスを，サブクラス間の距離によりグルーピングをし，サブクラスグループを生成する．さらに生成されたサブクラスグループをそれぞれ別々のカテゴリと見なし，従来のカテゴリ集合間の二値判別による判別器構成と同様の割当てを行い，符号表を生成する．これにより，異なるカテゴリに属するサブクラスであっても，サブクラスどうしの特徴が類似したものは同一のグループ，同一カテゴリ内でも特徴が異なるものは違うグループと見なした符号表を生成することが可能となる．

表 8 は，同一カテゴリ内での二値判別を許容する符号表の例である．以下では，c' をサブクラスグループとする．ここで，c'_1 に着目すると c'_1 には，$\{c_{1,1},c_{2,1},c_{2,2}\}$ が含まれており，異なるカテゴリのサブクラスが同一のグループに属している．また，$\{c_{1,1}\}$ と $\{c_{1,2},c_{1,3}\}$ は同一カテゴリでありながら，異なるグループに属している．このようなグループを生成することにより，$\{c_{1,1}\}$ と $\{c_{1,2},c_{1,3}\}$ の間で二値判別が行われている．このように，サブクラスグループを用いることで，サブクラス間の類似度を考慮した同一カテゴリ内での二値判別を許容する符号表の生成が可能となる．ここで述べた二値判別器構成の決定手順は，3.2.4 項で述べるアルゴリズムの Step3 にあたる．

3.2.3 符号表生成の繰り返しと結合

3.2.1 項と 3.2.2 項の手順により，カテゴリ情報とサブクラスの特徴を考慮した 2 種類の符号表生成が可能となる．しかし，生成される符号表の判別器数が少ない場合には，分類精度の低下が懸念される．そのため，2 種類の符号表の作成を，判別器数が所望の大きさとなるまで繰り返す．これにより様々なサブクラスに対する二値判別器を十分な数だけ持つ 1 つの符号表の生成が可能となる．

しかし，各繰り返しで生成されるサブクラスは異なるため，同じ行番号であっても同一のサブクラスとは限らず，繰り返しによって得られた複数の符号表を単純に連接させることができない．そこで提案手法では，行数が総学習データ数となる符号表に変換を行うことでこれを解決する．具体的には，各学習データの符号語はその学習データの属するサブクラスの符号語と同一のものにする．これにより，学習データごとに符号語の追加を行うことが可能となる．またサブクラスグループを用いて符号表を生成する際には，あらかじめ K 行の符号表を準備しそれに基づき生成する．この際，準備する符号表は従来手法と同一のものでよく，様々な特徴を持つ符号表を本提案手法に容易に適用することが可能である．ここで述べた繰り返しと結合による符号表生成は，3.2.4 項で述べるアルゴリズムの Step4 にあたる．

3.2.4 提案符号表生成アルゴリズム

以下では全学習データ数を N，あらかじめ準備する K 行の符号表を $\boldsymbol{H}$ とする．

Step0 基となる符号表 $\boldsymbol{H}$ の構成

Step1 サブクラスのランダム生成

Step1-1 サブクラスの代表ベクトルの選択

各カテゴリ c_k において，ランダムに C 個の学習デー

タを選択し，それを各サブクラスの代表ベクトルとする．

Step1-2 サブクラスの生成

各カテゴリ c_k において，サブクラス $c_{k,j}$ の代表ベクトルと学習データとの距離を計算し，代表ベクトルとの距離が小さい学習データをサブクラスへと分割する．

Step2 同一カテゴリ内で二値判別しない割当て

各カテゴリ c_k において，ランダムに 1 つのサブクラスを選択し，そのサブクラスは判別に用いないことにする．つまり符号語要素を $*$ とする．その他のサブクラスはあらかじめ準備した $\boldsymbol{H}$ と同様の割り付けとする．上記の操作を，すべてのサブクラスが 1 回ずつ選択されるまで行うことにより，符号表 $\boldsymbol{H}_1$ を得る．

Step3 同一カテゴリ内で二値判別を許容する割当て

Step3-1 サブクラスグループの代表ベクトルの選択

各カテゴリ c_k において，Step2 で生成されたサブクラスの中から 1 つ選択し，選択されたサブクラスの代表ベクトルを各サブクラスグループの代表ベクトルとする．

Step3-2 サブクラスのグルーピング

サブクラスグループの代表ベクトルと各サブクラスの代表ベクトルとの距離を計算し，距離の近いグループへとサブクラスを分割する．

Step3-3 サブクラスグループに基づく符号表の生成

生成された K 個のサブクラスグループをあらかじめ準備した $\boldsymbol{H}$ の K 個のカテゴリと見なし，Step2 と同様に割り付けを行うことにより，符号表 $\boldsymbol{H}_2$ を得る．

Step4 繰り返しと符号表の結合

Step1 から Step3 を M 回繰り返す．各繰り返しで生成される 2 つの符号表 $\boldsymbol{H}_1$，$\boldsymbol{H}_2$ は，各行を学習データとした N 行の符号表へと変換される．各学習データの符号語は属するサブクラスの符号語とする．変換後の 2 つの符号表を並べて 1 つにしたものを $\boldsymbol{H}'$ とし，M 個の $\boldsymbol{H}'$ を並べて 1 つの符号表とする． □

3.3 l 多数決による分類基準

上記のとおり，提案手法で生成される符号表は，各カテゴリが複数の符号語を持つ．すなわち，新規データに対する判別器の出力と比較される各カテゴリのテンプレートが複数存在する．そのため，類似度が最も高い符号語のみを考慮して分類を行うよりも，複数の符号語の類似度を総合して分類を行うことにより，複雑な構造を持つデータに対して頑健な分類が可能になると考えられる．一方，従来の類似度最大に基づく分類基準は，各カテゴリに対応する符号語が 1 つしか存在しないことを前提とした分類基準であり，複数の符号語を持つ符号表に対する分類基準として適していない．

そこで，提案する分類基準では，新規データに対する判別器出力と全符号語との類似度を従来と同様に計算し，その中から類似度の高い上位 l 個の符号語の属するカテゴリの多数決によってカテゴリを推定する．

4. 評価実験

本章では，提案手法の有効性を確認するために，新聞記事データを用いた自動文書分類の実験を行う．はじめに，9 カテゴリの日本語の新聞記事データにおいて，正解率と学習における計算時間の 2 つの指標から提案手法の有効性の評価を行う．次に，同様の新聞記事データに対して，カテゴリ数を変化させた場合の正解率の変化について検証を行う．さらに，提案手法のランダム性が結果にどのような影響を与えるかを検証した．また，英語の新聞記事データを用いて，データの特徴を変化させた際に提案手法がどのような挙動を示すかを検証する．最後に新聞記事以外のベンチマークデータに適用し，様々な観点から提案手法の性能について検証を行う．

4.1 9 カテゴリの新聞記事データの分類実験

提案した符号表生成アルゴリズムと分類基準の有効性を検証するために新聞記事を用いた分類実験を行った．実験データは 2010 年の毎日新聞記事から 9 カテゴリ（社説，国際，経済，家庭，文化，読書，芸能，スポーツ，社会）を使用した．学習データは各カテゴリ 200 件とし，テストデータは各カテゴリ 100 件とした．評価指標はテストデータに対する正解率，判別器の学習における計算時間とし，それぞれ 5 回の実験の平均値を用いた．二値判別器には RVM [24] を用い，カーネル関数には線形カーネルを適用した．また，学習時間に関しては，各判別器の学習を並列処理した場合についても示す．また，比較手法として非適用的な手法に着目し，ここでは，ランダム符号を用いた方法（Dense random 符号と Sprase random 符号），Exhaustive 符号を用いた方法，一対他に基づく方法を用いて実験を行う．これら手法を用いた実験結果と，提案手法 1 として Step2 で生成された同一カテゴリ内で二値判別を許容しない符号表のみを用いた手法，提案手法 2 として Step3 で生成された同一カテゴリ内で二値判別を許容する符号表のみを用いた手法，提案手法 1 と提案手法 2 の両方を用いた手法を提案手法 3 を用いた実験の結果を比較することにする．ただし，本実験で扱われていない非適応的な従来手法に関しては，Exhaustive 符号の正解率の方が高いことが先行研究 [25] より示されているため，今回の実験では割愛した．また，各提案手法のあらかじめ準備する符号表 $\boldsymbol{H}$ は，一対他法を用いた．各手法における二値判別器数は**表 9** に示す．提案手法 1，2 の繰り返し数 $M = 10$，提案手法 3 の $M = 5$，提案手法 1，2，3 のサブクラス数 $C = 3$，多数決数 $l = 3$ としたときの正解率と判別器の学習における計算

表 9 判別器数
Table 9 The number of the binary classifiers.

符号表	判別器数	
	二元	三元
Dense random 符号	32	-
Sparse random 符号	-	48
Exhaustive 符号	255	-
一対他法	9	-
提案手法 1	-	$27 \times M$
提案手法 2	-	$27 \times M$
提案手法 3	-	$54 \times M$

表 10 正解率と計算時間
Table 10 Result of accuracy rate and computational time.

符号表	正解率	計算時間（秒）	
		直列	並列
Dense random 符号	0.760	7,366	308
Sparse random 符号	0.760	1,858	147
Exhaustive 符号	0.763	58,822	330
一対他法	0.723	1,259	230
提案手法 1	0.767	11,844	111
提案手法 2	0.776	15,711	113
提案手法 3	0.775	13,888	111

時間の結果を**表 10** に示す．

表 10 より，正解率は提案手法 2 と提案手法 3 が高くなり，学習の計算時間は通常の直列処理の場合は一対他が最も短く，並列処理の場合は提案手法が短くなった．

また，Exhaustive 符号に基づく方法と 3 つの提案手法を比較すると，判別器数はほぼ同程度である一方で，計算時間，正解率ともに提案手法の方が優れた値となった．計算時間に関しては，提案手法は三元符号表であるため，学習に使用しないデータの分だけ，計算時間が低減されたと考えられる．また，正解率に関しては，提案手法 2，提案手法 3 は一対他に基づく手法でありながら，高い正解率となった．

また，ランダム符号と提案手法によって得られた結果の精度を比較すると，Dense ramdom 符号と Sparse ramdom 符号のどちらよりも，提案手法の正解率が高いことが分かった．また，計算時間で比較すると並列の場合は提案手法の計算時間が低くなった．まず提案手法が Dense ramdom 符号と比較して計算時間が低減している原因として，提案手法は三元の符号表であることが考えられる．次に，提案手法と Sparse ramdom 符号の計算時間に着目しよう．Sparse ramdom 符号を構成する各二値判別器において，学習に使用しないデータ数はランダムに決まるため，学習に用いない学習データ数は各二値判別器によって異なる．よって学習に使用するデータ数が多い二値判別器が 1 つでも存在する場合に，並列処理の際にその二値判別器の学習が大きなボトルネックとなる．一方で，提案手法は，すべての二値判別器で学習に使用しない学習データ数が等しくなるように二値判別器構成が決定されている．そのため，各二値判別器の学習で大きなボトルネックは発生しない．以上により，提案手法が Sparse ramdom 符号よりも計算時間が小さくなったものと考えられる．

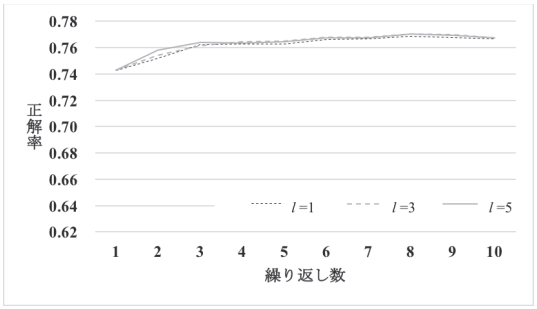

図 1 提案手法 1 の正解率
Fig. 1 Accuracy rate of proposal method 1.

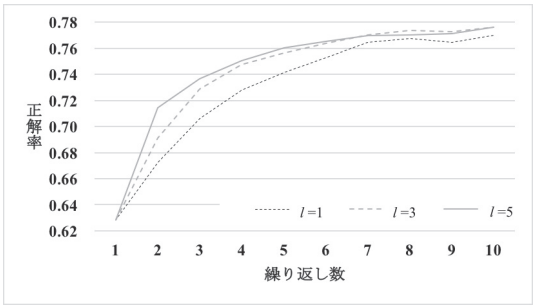

図 2 提案手法 2 の正解率
Fig. 2 Accuracy rate of proposal method 2.

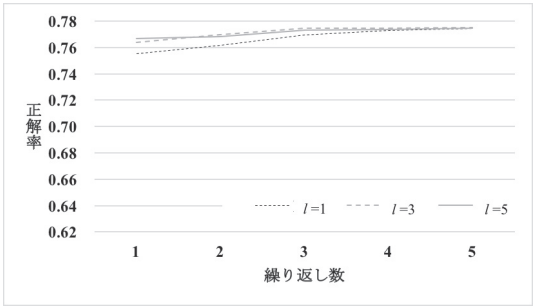

図 3 提案手法 3 の正解率
Fig. 3 Accuracy rate of proposal method 3.

次に，多数決分類基準と繰り返しによる効果を確認するために，繰り返し数 M と多数決数 l を変化させた際の提案手法 1，2，3 の正解率の変化について示す．$l = 1, 3, 5$ と変化させ，M は，提案手法 1，2 と提案手法 3 の判別器数をそろえるために，提案手法 1，2 では，$M = 1$〜10，提案手法 3 では $M = 1$〜5 とした．**図 1** が提案手法 1，**図 2** が提案手法 2，**図 3** が提案手法 3 の結果である．

まず，繰り返し数 M の変化が正解率に与える影響に着目する．提案手法 1 と提案手法 2 を比較すると，繰り返し数が大きいとき，正解率の最大値は提案手法 2 の方が高いものの，繰り返し数が少ないときには，提案手法 1 の正

解率が大きく上回っている．これは，提案手法 2 では，サブクラスグループを用いて，異なるカテゴリに属するサブクラスも同じカテゴリと見なした符号表生成を行うため，繰り返し数が少ないときには，異なるカテゴリ間で類似した符号語が多く存在することが理由だと考えられる．一方で，繰り返し数の増加にともない，提案手法 1 では，比較的類似した符号表が追加されるのに対して，提案手法 2 では，様々なサブクラスグループに対して符号表が生成されるため，符号表の多様性が向上し分類精度が大きく向上したと考えられる．

また，提案手法 3 の結果に着目すると，小さい繰り返し数ですでに Exhaustive 符号と同程度の正解率となり，繰り返し数が 5 のときは，提案手法 2 と同様の高い正解率となった．このことから，提案手法 3 は，繰り返し数が小さいときにも比較的高い正解率となり，繰り返し数を大きくすることによってさらに高い正解率となる提案手法 1，2 のそれぞれの長所を有した手法といえる．

次に，多数決数 l の変化が与える影響について考察する．各図より，繰り返し数が小さいときに多数決数による正解率の差が見られる．すなわち，判別器数が少ないときに多数決による分類基準が有効に機能すると考えられる．また，提案手法 1 では，多数決数によって正解率に大きな差が見られない．提案手法 1 では，同一カテゴリ内では二値判別を行わないため，同じカテゴリに属している学習データに，類似した符号語が付与されたものと考えられる．よって，上位 l 個に含まれる符号語の属するカテゴリの多くが同一のカテゴリに限定され，多数決による効果が小さくなったと考えられる．提案手法 1 と 2 を組み合わせている提案手法 3 においても，同様のことがいえる．

4.2 カテゴリ数の変化に対する評価

次に，上述した毎日新聞 2010 年の新聞記事データに対して，カテゴリ数を 3（経済，芸能，スポーツ），5（国際，経済，芸能，スポーツ，社会），7（社説，国際，経済，家庭，芸能，スポーツ，社会）と変化させた際の，各提案手法と Exhaustive 符号の正解率の変化について検証を行った．各カテゴリの学習データ数，テストデータ数は 9 カテゴリのときと同様に，それぞれ 200 件，100 件とした．多数決数 $l = 3$ とした．その他の実験条件は，4.1 節の 9 カテゴリの場合と同様である．**表 11**，**表 12**，**表 13** はそれぞれカテゴリ数 3，5，7 のデータセットに対する結果であり，提案手法 1，2 は繰り返し数を 1，2，5，10 とし，提案手法 3 は提案手法 1，2 と判別器数が同数になるように繰り返し数を 1，3，5 とした．表中の () 内の値は，提案手法 3 の繰り返し数である．

表 11〜表 13 より，提案手法 2 が，すべてのカテゴリ数において最も高い正解率となった．また，カテゴリ数 7 の場合には，Exhasutive 符号の正解率が提案手法 2 と同じ値であり，提案手法 1，3 よりも高くなった．一方，4.1 節の実験結果より，カテゴリ数 9 の場合には，提案手法の正解率が Exhaustive 符号よりも高い正解率となった．すなわち，Exhaustive 符号はカテゴリ数 7 の場合に高い正解率となった．これは Exhasutive 符号の判別器数は $2^{K-1}-1$ であるため，カテゴリ数が小さいときには，判別器が少なくなり，分類精度が低く，カテゴリ数が大きいときには，判別器数が膨大になるが，類似した判別器数が多く存在し，判別器の数ほど高い分類精度とならないことが原因と考えられる．すなわち，判別器数と分類精度の関係はカテゴリ数によって大きく変化する．一方で，カテゴリ数 7 の場合には，判別器数の数が分類精度の面で過不足のない状態であるため，カテゴリ数 7 の場合には，Exhasutive 符号の正解率が提案手法 2 と同じ値であり，提案手法 1，3 よりも高くなったと考えられる．実際，文献 [1] では，カテゴリ数が 7 以下の場合には Exhasutive 符号を用いているが，カテゴリ数が 8 以上の場合には，Exhaustive 符号から判別器の選択をしている．

次にカテゴリ数と繰り返し数に着目すると，カテゴリ数が小さいほど，繰り返し数の効果が小さい．これはカテゴリ数が小さいときには，分類が容易であるため多くの二値判別器を用いる必要がないからであると考えられる．

表 11 カテゴリ数 3 のデータセットの結果
Table 11 Accuracy rate ($K = 3$).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.898	-	-	-
提案手法 1	0.891	0.893	0.900	0.902
提案手法 2	0.834	0.847	0.907	0.909
提案手法 3	-	0.893	0.907	0.906

表 12 カテゴリ数 5 のデータセットの結果
Table 12 Accuracy rate ($K = 5$).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.800	-	-	-
提案手法 1	0.782	0.787	0.799	0.800
提案手法 2	0.674	0.726	0.797	0.805
提案手法 3	-	0.786	0.794	0.798

表 13 カテゴリ数 7 のデータセットの結果
Table 13 Accuracy rate ($K = 7$).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.761	-	-	-
提案手法 1	0.733	0.744	0.747	0.752
提案手法 2	0.630	0.686	0.744	0.761
提案手法 3	-	0.735	0.753	0.755

表 14 提案手法 1 のランダム性の検証
Table 14 Verification of randomness for proposal method 1.

実験	繰り返し数			
	1	2	6	10
1	0.733	0.745	0.747	0.752
2	0.739	0.744	0.753	0.756
3	0.735	0.749	0.753	0.755
4	0.739	0.753	0.754	0.757
5	0.730	0.737	0.753	0.756
6	0.733	0.741	0.753	0.753
7	0.735	0.739	0.756	0.755
8	0.741	0.745	0.755	0.755
9	0.734	0.742	0.749	0.755
10	0.729	0.733	0.748	0.752
平均	0.735	0.743	0.752	0.755
最大値	0.741	0.753	0.756	0.757
最小値	0.729	0.733	0.747	0.752
標準偏差	0.004	0.006	0.003	0.002

表 15 提案手法 2 のランダム性の検証
Table 15 Verification of randomness for proposal method 2.

実験	繰り返し数			
	1	2	6	10
1	0.629	0.686	0.744	0.761
2	0.619	0.693	0.753	0.765
3	0.634	0.692	0.755	0.759
4	0.629	0.680	0.747	0.757
5	0.649	0.691	0.756	0.756
6	0.634	0.692	0.757	0.763
7	0.629	0.691	0.759	0.762
8	0.654	0.689	0.755	0.766
9	0.621	0.680	0.757	0.763
10	0.638	0.690	0.749	0.762
平均	0.634	0.688	0.753	0.761
最大値	0.654	0.693	0.759	0.766
最小値	0.619	0.680	0.744	0.756
標準偏差	0.011	0.005	0.005	0.003

4.3 ランダム性の与える影響の評価

サブクラスの生成とサブクラスグループの生成におけるランダム性が実験結果に与える影響を評価するために，実験を複数回行った．上記の毎日新聞のカテゴリ数 7 のデータセットに対して，学習データとテストデータを変えずに提案手法を 10 回適用した．そのときの提案手法 1，2，3 の正解率が以下の**表 14**，**表 15**，**表 16** になる．

各提案手法の正解率のばらつきを比較すると，提案手法 2 のばらつきが最も大きくなった．これは，サブクラスの生成とサブクラスグループの生成の両方においてランダム性のある生成を行っていることが理由として考えられる．また，提案手法 2 は繰り返し数が小さいときには，ばらつきが大きいが，繰り返し数を増やすことで，他の 2 手法と同程度になる．また，正解率の平均値で比較すると提案手法 2 の値が最も高い．このことから，十分な繰り返しを行える状況下では，提案手法 2 を用いることで，高い正解率を安定して得ることが可能になると考えられる．

表 16 提案手法 3 ランダム性の検証
Table 16 Verification of randomness for proposal method 3.

実験	繰り返し数[*1]		
	1	3	5
1	0.735	0.753	0.755
2	0.739	0.758	0.758
3	0.741	0.758	0.761
4	0.741	0.754	0.760
5	0.739	0.754	0.757
6	0.737	0.756	0.756
7	0.742	0.763	0.767
8	0.745	0.757	0.764
9	0.742	0.753	0.758
10	0.730	0.747	0.757
平均	0.739	0.755	0.759
最大値	0.745	0.763	0.767
最小値	0.730	0.747	0.755
標準偏差	0.004	0.004	0.003

4.4 サブカテゴリ数の変化が与える影響の評価

ベンチマークデータセットの中にはあらかじめサブカテゴリが付与されたデータが存在する．サブカテゴリとは，カテゴリを分割した部分集合のことであるが，データの特徴から分割されたサブクラスとは異なり，あらかじめデータにラベルとして与えられたものである．ここでは，20 newsgroups [*2] と呼ばれるあらかじめサブカテゴリが付与された英語の新聞記事データに対して，提案手法を適用し，サブカテゴリ数の変化が正解率に与える影響について検証を行う．20 newsgroups では，4 つのトップカテゴリがあり，そのそれぞれにサブカテゴリが付与されている．本実験では，以下の**表 17** に示したトップカテゴリとサブカテゴリを用いた．

各トップカテゴリの学習データ数を 300，テストデータ数を 120 とし，各トップカテゴリ内のサブカテゴリ数を 2，3，4 と変化させた．各サブカテゴリ数における使用サブカテゴリは，表 17 の上位に記載されているものを用いた（サブカテゴリ数 2 の場合，comp カテゴリにおいては graphics と os．ms-windows.misc を用いた）．また，サブカテゴリ間で学習データ数とテストデータ数は同数となるように抽出した．その他の実験条件については，毎日新聞の実験条件と同様である（サブクラス数 $C = 3$，多数決数 $l = 3$）．Exhaustive 符号と各提案手法を比較する．サブカテゴリ数 2，3，4 の結果はそれぞれ**表 18**，**表 19**，**表 20** である．

[*1] 提案手法 3 の繰り返し数 1，3，5 のときの判別器数は提案手法 1，2 の繰り返し数 2，6，10 のときの判別器数とそれぞれ等しい．
[*2] http://qwone.com/~jason/20Newsgroups/

表 17 20 newsgroups のカテゴリ詳細
Table 17 Detail of 20 newsgroups' categories.

トップカテゴリ	サブカテゴリ	トップカテゴリ	サブカテゴリ
comp	graphics os.ms-windows.misc sys.ibm.pc.hardware windows.x	sci	cypt electronics med space
rec	autos motorcycles sport.baseball sport.hockey	talk	politics.guns politics.mideast politics.misc religion.misc

表 21 ベンチマークデータセット
Table 21 Benchmark dataset.

データセット	カテゴリ数	特徴量数	各カテゴリの学習データ数	各カテゴリのテストデータ数
Optidigits	10	64	100	200
Pendigits	10	16	150	300

表 18 サブカテゴリ数 2 の結果
Table 18 Accuracy rate (the number of sub-categories is 2).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.662	-	-	-
提案手法 1	0.663	0.675	0.676	0.678
提案手法 2	0.556	0.628	0.663	0.677
提案手法 3	-	0.670	0.683	0.681

表 19 サブカテゴリ数 3 の結果
Table 19 Accuracy rate (the number of sub-categories is 3).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.699	-	-	-
提案手法 1	0.712	0.723	0.731	0.728
提案手法 2	0.594	0.666	0.714	0.720
提案手法 3	-	0.704	0.724	0.735

表 20 サブカテゴリ数 4 の結果
Table 20 Accuracy rate (the number of sub-categories is 4).

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.668	-	-	-
提案手法 1	0.667	0.681	0.693	0.687
提案手法 2	0.538	0.610	0.664	0.669
提案手法 3	-	0.663	0.691	0.692

上記の結果より，すべてのサブカテゴリ数において，提案手法の正解率が Exhaustive 符号を上回った．特に，提案手法のサブクラス数と等しいサブカテゴリ数 3 のときに，その差は大きくなった．一方で，サブクラス数とサブカテゴリ数が等しくない場合（サブカテゴリ数 2，4 の場合）でも，提案手法は Exhaustive 符号よりも高い正解率となっている．提案手法の各繰り返しで生成されるサブクラス数は 1 つの値に固定されているものの，それを繰り返すことにより，各学習データに単一の符号語を割り当てることができている．よって，サブカテゴリ数に大きな影響を受けず，提案手法は安定して高い正解率になったと考えられる．

表 22 Optidigits の正解率
Table 22 Accuracy rate for Optidigits.

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.930	-	-	-
提案手法 1	0.938	0.939	0.940	0.941
提案手法 2	0.874	0.909	0.946	0.949
提案手法 3	-	0.940	0.944	0.945

また，提案手法 1 と提案手法 2 の正解率を比較すると，提案手法 1 の正解率が高くなり，毎日新聞を用いて行った実験とは異なる結果となった．これは，20newsgroup では，異なるカテゴリ間での類似したサブクラスが多く存在しないためだと考えられる．

4.5 新聞記事データ以外での評価

新聞記事データ以外へのベンチマークデータセットに適用し，様々な観点から提案手法の評価を行った．以下の**表 21** が用いたデータセットである．各データセットに対する正解率は以下の**表 22**，**表 23** になる．

表 22，表 23 の結果より，新聞記事データ以外のデータに対しても提案手法が有効であることが確認できる．また，各提案手法を比較すると，繰り返し数が大きいときには提案手法 2，繰り返し数が小さいときには提案手法 1 の正解率が高くなり，新聞記事データと場合と同様の傾向が見られる．また，提案手法 3 が繰り返し数に対して安定し

表 23 Pendigits の正解率
Table 23 Accuracy rate for Pendigits.

符号表	繰り返し数			
	1	2(1)	6(3)	10(5)
Exhaustive 符号	0.837	-	-	-
提案手法 1	0.886	0.886	0.891	0.892
提案手法 2	0.845	0.892	0.909	0.915
提案手法 3	-	0.895	0.900	0.902

て高い精度である点も同様の傾向である．以上から，新聞記事データ以外の場合においても，各提案手法が同じ特性を有するもの考えられる．

5. 考察

本節では，本研究の提案手法の性質とその波及効果などについて全体的な視点から，考察を行う．

5.1 多数決分類基準について

多数決分類基準は，符号表によって学習データを符号語空間上へと写像し，新規データの符号語が得られた際には，最も近い k 個の符号語に関する多数決により分類を行っている．すなわち，符号語空間上で kNN 法により分類を行っていることと同義である．そのため，kNN 法の特性と同様に多数決を行う際に，各カテゴリごとにテンプレートが少数であると，分類精度が劣化すると考えられる．

符号表によって定められる各学習データの写像先に正確に写像が行えるかどうかは，二値判別器の分類精度に依存する．また，二値判別器の分類精度は，どのようなカテゴリ集合を二値判別するかを決定する判別器構成および，二値判別器の性能の両方によって定まる．つまり比較的判別が容易なカテゴリ集合間の識別であれば，符号表によってあらかじめ定められた写像先への正確な写像が可能となり，各二値判別器が非線形識別関数である場合には，非線形な写像が可能となる．そのため，符号表は分析者によって任意のものを生成することができるが，二値判別器の性能を考慮して生成しなければ，たとえ多数決分類基準を用いたとしても，分類精度の向上は難しいと考えられる．

また，本研究では符号語空間上で多数決，すなわち kNN 法の適用を行ったが，符号語を判別器数次元のベクトルとして考えれば，その他の多値判別手法の適用も考えられる．その際には，計算時間の問題や符号語空間との相性などにより，適切な多値判別手法を考える必要がある．

5.2 繰り返し数について

繰り返し数は，正解率と計算時間に大きく影響をする値である．繰り返し数が大きい場合には，正解率が高い一方で，計算時間が大きい．繰り返し数が小さい場合には正解率が低い一方で，計算時間は小さく抑えられる．そのため，分析者が繰り返し数を決定する際には，与えられた計算環境と計算時間の中でできる限り繰り返し数を増やすことが望ましい．

5.3 サブクラス数の決定について

サブクラスは，各カテゴリに含まれる潜在的なトピックの集合として位置付けられ，問題によって多くのサブクラスを持つ場合と比較的少ないサブクラスを持つ場合が存在するものと思われるが，その数を事前に観測することはできない．そこで本研究では，事前実験において，サブクラスを変化させ，安定した性能を有するサブクラス数 $C = 3$ に固定してすべての実験を行った．ただし，これは全カテゴリに対してサブクラス数は同じという設定の下での結果であるということに注意が必要である．

ここで，問題の構造の変化（サブカテゴリ数の変化）による正解率への影響を示した表 18～表 20 に着目しよう．一般に，サブカテゴリ数と同じ数のサブクラス数を設定することができれば，分類精度が最大になると考えられる．実際にその傾向を表 18～表 20 から読み取ることができる．よって，サブクラスの数を正しく判断することができるならば，提案手法による正解率の高い判別が実現するものと思われる．しかしながら，真の構造を事前に学習することは不可能である．ただし，表 18～表 20 より，真のサブカテゴリ数 3 に対してサブカテゴリ数が 2 や 4 と多少外れた場合であっても，従来手法である Exhaustive 符号よりは提案手法の方が分類精度が高いことが分かる．よって，サブクラス数が最適な数から多少ずれてしまった場合にも，従来法より良い精度の判別が可能となっており，ここに提案手法の有効性を指摘することができる．

以上より，サブクラスの数を決定する際には，データの構造，性質を考慮しながら決定することにより，従来手法では達成することのできない判別精度を実現しうることが示唆される．

5.4 計算時間について

提案手法は，一対他法よりも通常の直列処理による計算量は増えているものの，Exhaustive 符号と比較すると学習の計算時間は現実的な量に抑えることが可能となった．これは，提案手法は「3 元の符号表を用いたこと」，「繰り返し数により判別器数の調整が可能であること」の 2 つが理由であると考えられる．また，本実験では学習データ数は最大で 1,800 件（9 カテゴリ×200 件）であったが，学習データ数がより大きい場合には提案手法は学習における計算時間の面でより有効であると考えられる．

5.5 各提案手法の使い分け

各提案手法の使い分けについて，正解率の面から考察を行う．まず，全体の傾向として，繰り返し数が少ない場合

には提案手法 1，繰り返し数が大きい場合には提案手法 2 の精度が高く，提案手法 3 は繰り返し数に対して安定した精度を有していることが分かる．

次に，与えられた計算環境によっては，すべての二値判別器を並列に学習することができなくなる．そのような場合，符号表の学習の繰り返しに対して，計算時間が膨大となると考えられる．そのため，計算環境が乏しく，分析にかけられる時間が少ないため，符号表の学習の繰り返しが十分に行えない場合には提案手法 1，計算環境が整っている，分析にかけられる時間が多いなど十分に符号表の学習の繰り返しが行える場合には提案手法 2 を用いるのが良いと考えられる．また，提案手法 3 は繰り返し数に対して比較的安定した精度を保ち，また実験条件によっては他の 2 手法よりも高い正解率となった．提案手法 3 は他の 2 手法を組み合わせているため，安定性が増していると考えられる．このことから，データの特徴が明確でない場合や繰り返し数に対する正解率が定かでなく，提案手法 1 と 2 のどちらを用いたほうが有効かの検討がつかない場合は提案手法 3 を用いることが好ましいと考えられる．

5.6 適用可能なデータ

本研究では，新聞記事データを用いた文書分類を行った．しかしながら，ECOC 法は判別器構成を得る手法であるため，様々なデータに適用可能である．提案手法は，サブクラスを用いることにより，各カテゴリのデータすべてを用いていた判別問題をカテゴリ内の一部のデータを用いる部分問題へと分割しており，カテゴリ情報のみを用いる場合に比べて，問題を単純化することができている．そのことから，カテゴリ数や学習データ数が多く，すべての学習データを判別に用いると同一カテゴリ集合内で性質の異なるデータが混在するようなデータに対して，提案手法はより有効であると考えられる．

6. まとめ

本研究では，多値判別問題を対象とし，カテゴリ内に存在する潜在的な部分集合であるサブクラスに着目し，同一カテゴリ内での二値判別を許容した符号表生成とそれにともなう多数決分類基準の提案を行った．提案手法で作成される符号表は，十分な判別機数を確保しつつ，各判別器に用いられる学習データ数を低減することで，現実的な計算量内で高い分類精度を得ることが可能となった．

実際の新聞記事の分類問題に適用し，正解率，計算量の面での有効性を示した．カテゴリ数やサブカテゴリを変えた実験により様々な実験条件に対して，提案手法が様々な条件に対しても，正解率の面で有効であることを示した．また，正解率の観点から各提案手法の使い分けも示唆され，実務への適応が容易であることも示した．

本研究で提案した符号生成法では，ランダムに学習データを選択することでサブクラスを構成しており，平均的な意味で分類精度の高い符号構成法を与えていると解釈することができる．また，サブクラス数を変化させた場合に対する提案手法 1，2，3 の精度については，対象問題によって適切なサブクラス数も異なると考えられる．また，これら 3 つの手法を正しく評価できるだけの実験およびその考察をすることは，本論文においては議論が大変困難である．そのため，本論文ではそのような状況下でどのように使い分けを考えればよいのかについての定性的な考察をするにとどまっている．これに対してランダム性をなくしたうえで，各繰り返しでの多様性を確保するサブクラスやサブクラスグループの生成方法を検討すること，サブクラス数と提案手法との正解率の関係について明らかにすることは今後の課題である．また，本研究ではあらかじめ準備する符号表 $\boldsymbol{H}$ として一対他法を用いたが，その他の従来の符号表を用いることも可能であり，その有効性の検証についても今後の課題とする．

謝辞 本研究にあたり，多くのご助言をいただいた湘南工科大学の三川健太先生，後藤研究室の方々に深く感謝いたします．本研究の一部は科学研究費（26282090，26560167）の助成を受けたものである．

参考文献

[1] Dietterich, T.G. and Bakiri, G.: Solving Multiclass Learning Problems Via Error-correcting Output Codes, *Journal of Artificial Intelligence Research*, Vol.2, pp.263–286 (1995).
[2] 池田思朗：2 クラス判別器の組み合せによる多クラス判別統計モデルとパラメータ推定，統計数理，Vol.58, No.2, pp.157–166 (2010).
[3] Huang, T.-K., Weng, R.C. and Lin, C.-J.: Generalized Bradley-Terry Models and Multi-class Probability Estimates, *Journal of Machine Learning Research*, Vol.7, No.Jan, pp.85–115 (2006).
[4] Pujol, O., Radeva, P. and Vitria, J.: Discriminant ECOC: A Heuristic Method for Application Dependent Design of Error Correcting Output Codes, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.28, No.6, pp.1007–1012 (2006).
[5] Zhong, G. and Cheriet, M.: Adaptive Error-Correcting Output Codes, *IJCAI*, Citeseer (2013).
[6] Zhang, X.-L.: Heuristic Ternary Error-correcting Output Codes Via Weight Optimization and Layered Clustering-based Approach, *IEEE Trans. Cybernetics*, Vol.45, No.2, pp.289–301 (2015).
[7] Allwein, E.L., Schapire, R.E. and Singer, Y.: Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers, *Journal of Machine Learning Research*, Vol.1, No.Dec, pp.113–141 (2000).
[8] 大山賀己，竹之内高志，石井 信：ECOC 復号法に基づく階層的多値判別法，電子情報通信学会技術研究報告．NC，ニューロコンピューティング，Vol.107, No.542, pp.337–342 (2008).
[9] Crammer, K. and Singer, Y.: On the Learnability and Design of Output Codes for Multiclass Problems, *Machine Learning*, Vol.47, No.2-3, pp.201–233 (2002).

[10] Cover, T.M. and Thomas, J.A.: *Elements of Information Theory*, John Wiley & Sons (2012).
[11] Escalera, S. and Pujol, O.: ECOC-ONE: A Novel Coding and Decoding Strategy, *18th International Conference on Pattern Recognition, ICPR 2006*, Vol.3, pp.578–581, IEEE (2006).
[12] Escalera, S., Pujol, O. and Radeva, P.: Boosted Landmarks of Contextual Descriptors and Forest-ECOC: A Novel Framework to Detect and Classify Objects in Cluttered Scenes, *Pattern Recognition Letters*, Vol.28, No.13, pp.1759–1768 (2007).
[13] Pujol, O., Escalera, S. and Radeva, P.: An Incremental Node Embedding Technique for Error Correcting Output Codes, *Pattern Recognition*, Vol.41, No.2, pp.713–725 (2008).
[14] Xue, A., Wang, X., Song, Y. and Lei, L.: Discriminant Error Correcting Output Codes Based on Spectral Clustering, *Pattern Analysis and Applications*, pp.1–19 (2015).
[15] Escalera, S., Tax, D.M., Pujol, O., Radeva, P. and Duin, R.P.: Subclass Problem-dependent Design for Error-correcting Output Codes, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.30, No.6, pp.1041–1054 (2008).
[16] Bouzas, D., Arvanitopoulos, N. and Tefas, A.: Optimizing Subclass Discriminant Error Correcting Output Codes Using Particle Swarm Optimization, *The 2010 International Joint Conference on Neural Networks* (*IJCNN*), pp.1–7, IEEE (2010).
[17] Escalera, S., Pujol, O. and Radeva, P.: Recoding Error-correcting Output Codes, *International Workshop on Multiple Classifier Systems*, pp.11–21, Springer (2009).
[18] Escalera, S., Masip, D., Puertas, E., Radeva, P. and Pujol, O.: Online Error Correcting Output Codes, *Pattern Recognition Letters*, Vol.32, No.3, pp.458–467 (2011).
[19] Rifkin, R. and Klautau, A.: In Defense of One-vs-all Classification, *Journal of Machine Learning Research*, Vol.5, No.Jan, pp.101–141 (2004).
[20] Chmielnicki, W.: Creating Effective Error Correcting Output Codes for Multiclass Classification, *International Conference on Hybrid Artificial Intelligence Systems*, pp.502–514, Springer (2015).
[21] 荻原大陸，三川健太，後藤正幸：Reed Muller 符号を用いた階層的 ECOC 法による多値文書分類，第 36 回情報理論とその応用シンポジウム（SITA2013），No.5.3.1 (2013).
[22] Escalera, S., Pujol, O. and Radeva, P.: On the Decoding Process in Ternary Error-correcting Output Codes, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.32, No.1, pp.120–134 (2010).
[23] Smith, R.S. and Windeatt, T.: Decoding Rules for Error Correcting Output Code Ensembles, *International Workshop on Multiple Classifier Systems*, pp.53–63, Springer (2005).
[24] Tipping, M.E.: Sparse Bayesian Learning and the Relevance Vector Machine, *Journal of Machine Learning Research*, Vol.1, No.June, pp.211–244 (2001).
[25] Suzuki, L., Mikawa, K. and Goto, M.: Multi-valued Classification of Text Data Based on an ECOC Approach Using a Ternary Orthogonal Table, *Industrial Engineering & Management Systems*, Vol.16, No.2, pp.155–164 (2017).

鈴木 玲央奈

1992 年生．2015 年早稲田大学創造理工学部経営システム工学科卒業．2017 年同大学大学院修士課程修了．文書分類をはじめとする，様々な分類問題の研究に興味を持つ．

山下 遥

1987 年生．2010 年東京理科大学理工学部経営工学科卒業．2012 年慶應義塾大学大学院修士課程修了．2015 年同大学大学院博士課程修了．博士（工学）．2015 年早稲田大学創造理工学部助手，2017 年上智大学理工学部助教．品質管理，統計学，情報工学を融合させた新たなデータ解析方法に関する研究に従事．応用統計学会，日本経営工学会，日本品質管理学会等，各会員．

後藤 正幸 （正会員）

1969 年生．1994 年武蔵工業大学大学院修士課程修了．1997 年早稲田大学理工学部助手．2000 年早稲田大学大学院理工学研究科博士課程修了．博士（工学）．2000 年東京大学大学院工学系研究科助手．2002 年武蔵工業大学環境情報学部助教授．2008 年早稲田大学創造理工学部経営システム工学科准教授．2011 年同大教授．情報数理応用とデータサイエンスの研究に従事．著書に，『入門パターン認識と機械学習』，コロナ社 (2014)，『ビジネス統計～統計基礎とエクセル分析』，オデッセイコミュニケーションズ (2015) 等．IEEE，電子情報通信学会，人工知能学会，日本経営工学会，日本オペレーションズ・リサーチ学会，経営情報学会等，各会員．