A Multi-labeled Classification based on Error-correcting Output Coding

Tetsuya Yamashita Takashi Takenouchi Shin Ishii

Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara 630-0192, Japan
{tetsuya-y, ttakashi, ishii}@is.naist.jp

Abstract

We propose a new framework to deal with multi-labeled classification problems based on error-correcting output coding (ECOC). In multi-labeled classification problems, it is required to assign multiple classes to a single input. In this study, we show naive multi-label approach can be improved based on probabilistic modeling of misclassification process in the ECOC method. We examine performance of the proposed method with synthetic datasets and show that the proposed method accurately predicts multiple labels of a new input, relatively to the existing method.

1 Introduction

In a multi-labeled classification problem, a single input is assigned to multiple classes or categories. A typical example of multi-labeled problems is the text categorization problem on the World Wide Web, where each text may belong to some of multiple categories. For such a problem, there are two conventional approaches. One is the binary classification approach in which each text is classified into multiple classes by integrating individual results from binary classifiers. The other approach simultaneously deals with multiple classes by considering multinomial models; Ueda and Saito [5] proposed Parametric Mixture Model (PMM) in which the multinomial distribution is extended to represent the dependence of multiple classes. In this study, we take the former approach.

In the context of multiclass classification problems, the error-correcting output coding (ECOC) method was formerly proposed by Dietterich and Bakiri [2]. This method decomposes the original multi-class classification problem into multiple binary classification problems whose each output is {+1, -1}. In the framework of ECOC, a k class problem is decomposed into l binary classification problems and each class label is represented by a code word which is a row vector of a code matrix $W \in \{+1, -1\}^{k \times l}$. An example code matrix in a four-class problem would be

$$W = \begin{pmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}.$$

Then, each binary classifier is trained using binary labels associated with the corresponding column vector of W. We can predict the class label of a new input using the outputs of binary classifiers, where the simplest method is the Hamming decoding. For outputs of binary classifiers, the closest code word in W with respect to the Hamming distance is used as the predicted class label of the input. Allwein et al. [1] proposed a more flexible framework that allows W to include 0 which signifies "do not care" classes in the corresponding binary classifier. Moreover, the Hamming distance was extended to general loss functions.

In this study, we apply the framework of ECOC to multi-labeled problems by formulating a probabilistic model that represents the relationship between a code word and a set of outputs from classifiers. We in particular develop the method based on the information transmission theory. The model regards a misclassification of each binary classifier as a bit inversion in the code word due to a noisy channel. An addition of parity bits which leads to redundant representation of multiple labels and adaptive identification of the noisy channel based on the probabilistic model enables an accurate prediction of multiple labels.

In section 2, basic setting and a probabilistic model of the noisy channel are formulated. In section 3, we examine the performance of the proposed method, in comparison to the simple multi-labeled classification method, using synthetic datasets.

2 Method

In this section, we describe the probabilistic model of the bit inversion, which represents the misclassification error, in the multi-label classification, and its decoding method.

2.1 Notation

Let $\boldsymbol{x}$ be an input vector and $\boldsymbol{y} \in \{1,-1\}^b$ be a vector of labels associated with b categories. Typically, $\boldsymbol{x}$ is an element of the *Bag-of-Words* [3]. We assume that a dataset $\{\boldsymbol{x}^i, \boldsymbol{y}^i\}_{i=1}^N$ is given. Let $y_j (1 \leq j \leq b)$ be a component of $\boldsymbol{y}$, defined by

$$y_j = \begin{cases} 1 & \boldsymbol{x} \text{ belongs to class } j \\ -1 & \text{otherwise.} \end{cases} \quad (1)$$

We consider parity bits $\boldsymbol{z} \in \{1,-1\}^c$ associated with $\boldsymbol{y}$, whose j-th component is denoted as z_j. Note that the parity $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{y}) : \{1,-1\}^b \to \{1,-1\}^c$ is designed in *a priori* manner.

Using the dataset and the corresponding set of parity bits, $\{\boldsymbol{z}^1, ..., \boldsymbol{z}^N\}$, we can train $(b+c)$ binary classifiers. We denote outputs of binary classifiers associated with the code word $\boldsymbol{y}$ as $\tilde{\boldsymbol{y}}$, and those augmented by the parity bits $\boldsymbol{z}$ as $\tilde{\boldsymbol{z}}$, both of which are assumed to be disturbed by a noisy transmission channel.

2.2 Probabilistic model and decoding method

The probabilistic model of $(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}})$ given $\boldsymbol{y}$ is defined as

$$\begin{aligned} p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}) &= p(\tilde{\boldsymbol{y}}|\boldsymbol{y}) p(\tilde{\boldsymbol{z}}|\boldsymbol{y}) \\ &= \exp(\beta \tilde{\boldsymbol{y}}^{\mathrm{t}} \boldsymbol{y} - b\varphi(\beta)) \exp(\beta \tilde{\boldsymbol{z}}^{\mathrm{t}} \boldsymbol{z} - c\varphi(\beta)) \\ &= \exp(\beta(\tilde{\boldsymbol{y}}^{\mathrm{t}} \boldsymbol{y} + \tilde{\boldsymbol{z}}^{\mathrm{t}} \boldsymbol{z}) - (b+c)\varphi(\beta))), \quad (2) \end{aligned}$$

where t denotes a transpose, $\varphi(\beta) = \ln(e^{\beta} + e^{-\beta})$ is a normailzation constant, and β is a positive constant that represents the noise level of the noisy channel. We assume that the label prior $p(\boldsymbol{y})$ is the uniform distribution, then $p(\boldsymbol{y}|\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}})$ is calculated as follows.

$$\begin{aligned} p(\boldsymbol{y}|\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}) &= \frac{p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}) P(\boldsymbol{y})}{\sum_{\boldsymbol{y}} p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}) P(\boldsymbol{y})} \\ &\propto (\tilde{\boldsymbol{y}}^{\mathrm{t}} \boldsymbol{y} + \tilde{\boldsymbol{z}}^{\mathrm{t}} \boldsymbol{z}). \quad (3) \end{aligned}$$

This is equivalent to the Hamming distance between $(\tilde{\boldsymbol{y}}^{\mathrm{t}}, \tilde{\boldsymbol{z}}^{\mathrm{t}})$ and $(\boldsymbol{y}, \boldsymbol{z})$, then an estimate $\hat{\boldsymbol{y}}$ of the original label $\boldsymbol{y}$ can be decoded as

$$\hat{\boldsymbol{y}} = \mathrm{argmax}_{\boldsymbol{y}} p(\boldsymbol{y}|\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}). \quad (4)$$

This decoding method is equivalent to the Hamming decoding with parity bits and reduces to the naive multi-labeled decoding method when the parity bits $\boldsymbol{z}$ are omitted.

2.3 Estimation of confidence

In model (2), β was set at a positive constant, implying all classifiers are assumed to have the same reliability (confidence). This assumption is not natural, however, because performances of classifiers are different from each other according to the difficulty in corresponding classification problems or the underlying geometrical structure of dataset. Takenouchi and Ishii [4] introduced the confidence of classifiers into multiclass classification problems. Following this existing study, we introduce the confidence for each classifier, and then the probabilistic model is extended as

$$\begin{aligned} p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}; \boldsymbol{\beta}, \boldsymbol{\gamma}) &= p(\tilde{\boldsymbol{y}}|\boldsymbol{y}; \boldsymbol{\beta}) p(\tilde{\boldsymbol{z}}|\boldsymbol{y}; \boldsymbol{\gamma}) \\ &= \exp\Big(\sum_{j=1}^{b} (\beta_j \tilde{y_j} y_j - \varphi(\beta_j))\Big) \\ &\quad \times \exp\Big(\sum_{k=1}^{c} (\gamma_k \tilde{z_k} z_k - \varphi(\gamma_k))\Big) \quad (5) \end{aligned}$$

where $\boldsymbol{\beta} = (\beta_1, ..., \beta_b)$ and $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_c)$ are parameter vectors representing the confidence of each dimension of the noisy channel. Parameters $\boldsymbol{\beta}, \boldsymbol{\gamma}$ can be estimated as to maximize the log-likelihood:

$$L(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \sum_{i=1}^{N} \log\ p(\tilde{\boldsymbol{y}}^i, \tilde{\boldsymbol{z}}^i|\boldsymbol{y}^i; \boldsymbol{\beta}, \boldsymbol{\gamma}). \quad (6)$$

Stationary conditions of equation (6) become

$$\begin{aligned} \frac{\partial L}{\partial \beta_j} &= \sum_{i=1}^{N} (\tilde{y}_j^i y_j^i - \frac{\exp(\beta_j) - \exp(-\beta_j)}{\exp(\beta_j) + \exp(-\beta_j)}) \\ &= 0, \quad (7) \\ \frac{\partial L}{\partial \gamma_k} &= \sum_{i=1}^{N} (\tilde{z}_k^i z_k^i - \frac{\exp(\gamma_k) - \exp(-\gamma_k)}{\exp(\gamma_k) + \exp(-\gamma_k)}) \\ &= 0. \quad (8) \end{aligned}$$

Those equations can be analytically solved as

$$\hat{\beta_j} = \frac{1}{2} \ln \frac{1 - C_{y_j}}{C_{y_j}}, \hat{\gamma_k} = \frac{1}{2} \ln \frac{1 - C_{z_k}}{C_{z_k}}, \quad (9)$$

where $C_{y_j} = \frac{1}{N} \sum_{i=1}^{N} \frac{1 - (\tilde{y_j}^i y_j^i)}{2}$ is the error rate of the classifier associated with the code $\boldsymbol{y}_j$ and $C_z =$

$\frac{1}{N}\sum_{i=1}^{N}\frac{1-(\tilde{z_k}^i z_k^i)}{2}$ is that with the parity bits $\boldsymbol{z}_k$. After identifying the characteristics of the noisy channel, the label $\hat{\boldsymbol{y}}$ can be decoded as a maximum *a posteriori* (MAP) estimate:

$$\hat{\boldsymbol{y}} = \mathrm{argmax}_{\boldsymbol{y}} p(\boldsymbol{y}|\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}; \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}). \tag{10}$$

Note that the posterior probability in equation (10) is rewritten as

$$\begin{aligned} p(\boldsymbol{y}|\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}; \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}) &= \frac{p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}; \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})p(\boldsymbol{y})}{\sum_{\boldsymbol{y}} p(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}|\boldsymbol{y}; \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}})p(\boldsymbol{y})} \\ &\propto \Big(\sum_{j=1}^{b}\beta_j \tilde{y_j} y_j + \sum_{k=1}^{c}\gamma_k \tilde{z_k} z_k\Big) \end{aligned} \tag{11}$$

Namely, this decoding is a weighted version of the Hamming decoding, but the weight is estimated by the maximum likelihood estimation from multi-labeled classification results of the training data. This approach is natural, because our decoding is based on the weighted Hamming distance and its weight is determined based on the classification performance of each binary classifier for the training dataset.

2.4 Difficulty of decoding process

In the previous subsection, we applied the MAP decoding (10) in which the number of candidates to be searched becomes exponential, 2^b. If the number b of categories is large, this decoding process becomes hard. To avoid this problem, one practically good way is to restrict the multiplicity being small such as $\{\boldsymbol{y}|\sum_{i=1}^{b}\frac{y_i+1}{2} \leqq 3\}$.

This restriction seems plausible, in practical for example, in the text categorization problem, because major part of texts are likely assigned each to a few classes. Using this assumption, the number of code candidates to be searched is reduced to ${}_aC_3$, which makes the MAP decoding feasible.

3 Experiment result

In this section, we examine the performance of the proposed method by comparing with the naive classification based method, using synthetic datasets. The parity bits were randomly generated and the MAP decoding was employed in the proposed method. For binary classifiers, we employed the simplest linear discriminant analysis. A typical example of dataset consisting of 90 input data with 3 categories, is shown in Figure 1.

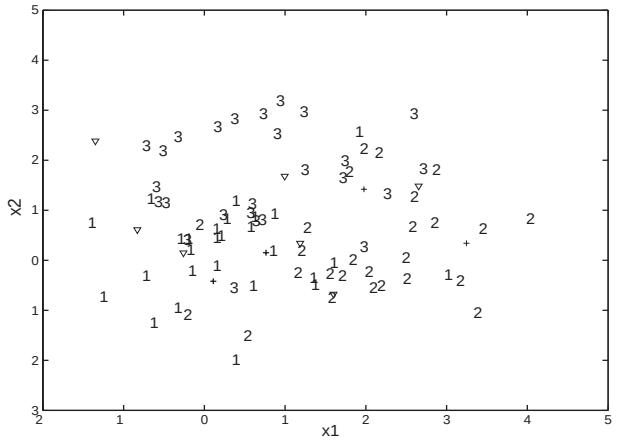

Figure 1: An example dataset. Each numerical character indicates the class to which an input vector is assigned. A symbol $\bigtriangledown$ indicates an input assigned to two categories and + means that the corresponding input does not belong to any category.

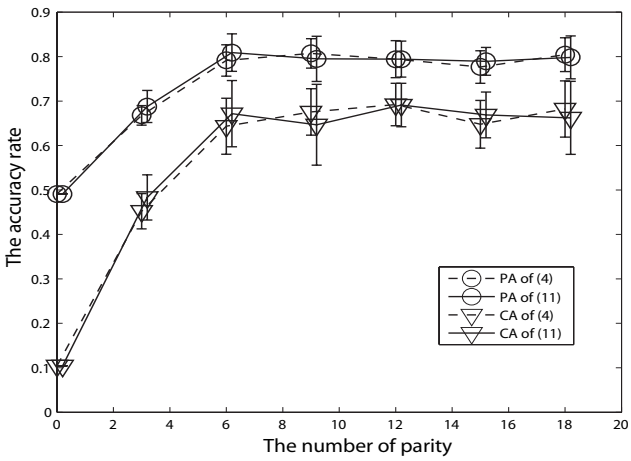

Figure 2: PA and CA of the two decoding methods, the Hamming decoding with parity bits(equation(4)) and the weighted Hamming decoding with parity bits(equation(11)), against the length of parity bits.

For performance evaluation, we used two kinds of quantities: complete accuracy rate (CA) and partial accuracy rate (PA). CA is the rate of events in which the vector (three-dimensional, in this case) of decoded label completely coincides with that of the original label, and PA is the rate with which the element of the decoded label is consistent with that of the original label.

Figure 2 shows averaged CA and PA of the proposed method for 10 trials. The error bar is the standard deviation in the 10 trials. In this figure, we compared the Hamming decoding with parity bits (equation(4)) and the weighted Hamming decoding with parity bits(equation(11)). For comparison, we also applied a naive multi-labeled classification method, that is, a set of individual binary classifiers assign the label for the corresponding category. CA and PA of the naive method were 0.1111 and 0.5056, respectively. From the comparison, we can see (1) the proposed methods outperformed the naive method; and, (2) when the parity length was $c \geq 6$, the performance by our methods was consistently good.

4 Conclusion

We proposed a novel method for multi-labeled classification problems based on the framework of ECOC. This regard the classification errors of classifies as the noise of the channel. Using the parity associated with the label, we can improve the classification accuracy in comparison to the naive classification based method. As a future work, we plan to apply the proposed method for real datasets. For this application, the decoding method discussed in subsection 2.4 will be improved. However, the design of parity bits on in our future work.

References

[1] E. L. Allwein, R. E. Schapire, Y. Singer. "Reducing Multiclass: A unifying Approach for Margin Classifiers," *Jounal of Machine Learning Research*, **1**, 113-141, 2001.

[2] T. G. Dietterich, G. Bakiri. "Solving multiclass learning problems via error-correcting output codes, "*Journal of Artifical Intelligence Research*, **2**, 263-286, 1995.

[3] S. T. Dumais, J. Platt, D. Heckerman, M. Sahami. "Inductive learning algoritms and representations for text categorization, "*ACM-CIKM'98*, 1998.

[4] T. Takenouchi, S. Ishii. "Multiclass classification by an extention of ECOC decoding, "*2006 Workshop on Information-Based Induction Sciences*, 2006 (in Japanese).

[5] N. Ueda, K. Saito. "Parametric Mixture Models for Multi-Labeled Text," *Advances in Neural Information Processing Systems 15*, 721-728, 2002.