Construction of Error Correcting Output Codes for Robust Deep Neural Networks Based on Label Grouping Scheme

Hwiyoung Youn, Soonhee Kwon, Hyunhee Lee, Jiho Kim, Songnam Hong, Dong-Joon Shin

Department of Electronic Engineering, Hanyang University, Seoul, Korea
yhyh82@hanyang.ac.kr, tnsgml1991@hanyang.ac.kr, fly4hyun@hanyang.ac.kr, jihokim@hanyang.ac.kr, snhong@hanyang.ac.kr, djshin@hanyang.ac.kr

2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC) | 978-1-6654-0582-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/IC-NIDC54101.2021.9660486

Abstract: Error-Correcting Output Codes (ECOCs) have been proposed to construct multi-class classifiers using simple binary classifiers. Recently, the principle of ECOCs has been employed for improving the robustness of deep classifiers. In this paper, a novel ECOC framework is developed by presenting a novel label grouping and code-construction method. The proposed label grouping is based on linear discriminant analysis (LDA) similarity. Via simulations, it is demonstrated that deep classifiers trained with the proposed ECOC yield better classification performance on pure data and better adversarial robustness than the state-of-the-art deep neural classifiers using ECOCs.

Keywords: Adversarial robustness, Classification, Error -correcting output codes, Label grouping, Linear discriminant analysis.

1 Introduction

Error-correcting output codes (ECOCs) are flexible and effective methods that combine trained binary classifiers to conduct multi-class classifications. The first ECOC [1] shows high multi-class classification accuracy by using simple but powerful binary classifiers such as support vector machines (SVMs), where a codebook is designed via two design criteria as row separation and column separation. The row separation means that the codewords should be well-separated to each other in Hamming distance and the column separation means that each bit-position function should be uncorrelated with the functions to be learned for the other bit positions [1]. In [2], an ECOC codebook design is proposed by using the Hadamard matrix in order to ensure large Hamming distance. Accordingly, it is ensured that the Hamming distance of each class label is $n/2$. In [3], novel ECOC construction criteria are proposed in order to enhance the performance and robustness of multi-class classifier, which contain not only the row separation and column separation, but also the balanced columns and data distribution. Then, an optimal codework to satisfy such design criteria via integer programming.

Focusing on deep neural networks (DNNs), a novel ECOC construction framework, named multi-hot target encoding (MUTE), is constructed [4] where a similarity between two classes is measured by the generative model such as auto-encoders [5]. Note that MUTE identifies such similar classes, from which target codes having larger Hamming distances can be assigned between them.

In this paper, a novel ECOC is developed by proposing a novel label grouping scheme based on linear discriminant analysis (LDA) inter-class similarity. To perform the grouping of classes, the so-called similarity between classes should be measured. In the proposed ECOC, the similarity measure based on the linear discriminant analysis (LDA) [6], is adopted. Via simulations, the superiority of the proposed ECOC is demonstrated. For fair comparison, the proposed ECOC scheme is set to have the same number of output encoding length as MUTE and conventional one-hot encoding. By CIFAR-10 [7], it is verified that DNN trained by the proposed ECOC shows better classification performance on pure data and it also shows better adversarial robustness than the deep neural networks trained by one-hot encoding or MUTE. In order to compare the robustness, salt and pepper noise, negative images, fast gradient sign method (FGSM) [8] are considered, which is usually used for generating adversarial examples. For deep neural network structure, ResNet [9] and ResNeXt [10] are considered which are widely used CNN architectures.

2 A Novel Label Grouping Scheme Based on LDA Inter-Class Similarity

In this section, a novel label grouping scheme is proposed. Also, an encoding scheme is proposed such that for the classes in the same group ECOC codewords are assigned to have large Hamming distances among them. Although there are many other methods to measure inter-class similarity [11, 12], it is experimentally confirmed that the linear discriminant analysis (LDA) method is one of the most effective ones for label grouping. Also, by designing ECOCs based on the proposed group sets, it shows better performance against both the noise and adversarial attack over the one-hot encoding and other ECOC methods.

The proposed novel label grouping scheme is based on the LDA-based inter-class similarity matrix as follows. Let the whole training set be $\mathcal{T} = \{(X_1, y_1), \dots (X_m, y_m)\}$ where (X_i, y_i) are the input data and the corresponding class label, respectively. And Let $\mathcal{T}_i = \{(X, y) | y = i \; and \; (X, y) \in \mathcal{T}\}$ be the subset of training examples corresponding to class i. In [6], they find a basis set $\mathcal{B}$ for the subset of training examples corresponding to the class i is obtained, which maximizes the sum of the inter-class covariances. From the projected samples in each class $0 \le i \le k-1$ (k is the number of classes), they compute the average per-class vector $v_i = \frac{1}{\#\mathcal{T}_i} \sum_{(X,y) \in \mathcal{T}_i} LX$, where L is the

978-1-6654-0582-9/21/$31.00 ©2021 IEEE

projection matrix and $\#\mathcal{T}_i$ represents the number of training samples from class i. Then a $k \times k$ matrix A representing the inter-class the similarities is constructed by [6]:

$$A_{i,j} = \begin{cases} \dfrac{s(v_i,\ v_j)}{\left\{\sum_{j=0}^{k-1} s(v_i,\ v_j)\right\} - s(v_i,\ v_i)} & ,\ j \neq i \\ 0 & ,\ j = i \end{cases} \quad (1)$$

where:

$$s(v_i, v_j) = \frac{1}{1+e^{d(v_i,v_j)}}$$

$$d(v_i, v_j) = 1 - \frac{\langle v_i,\ v_j \rangle}{\|v_i\|_2 \|v_j\|_2}.$$

Note that $\langle \cdot\ ,\ \cdot \rangle$ represents the usual dot product and $s(\cdot\ ,\ \cdot)$ measures a similarity by using the cosine of the angle between vectors. For the CIFAR-10 training set, we construct the corresponding matrix A as a measure of inter-class similarity. In this case, if $A_{i,j} > A_{i,k}$, it can be said that the DNN misclassifies the class i by the class k more than by the class j.

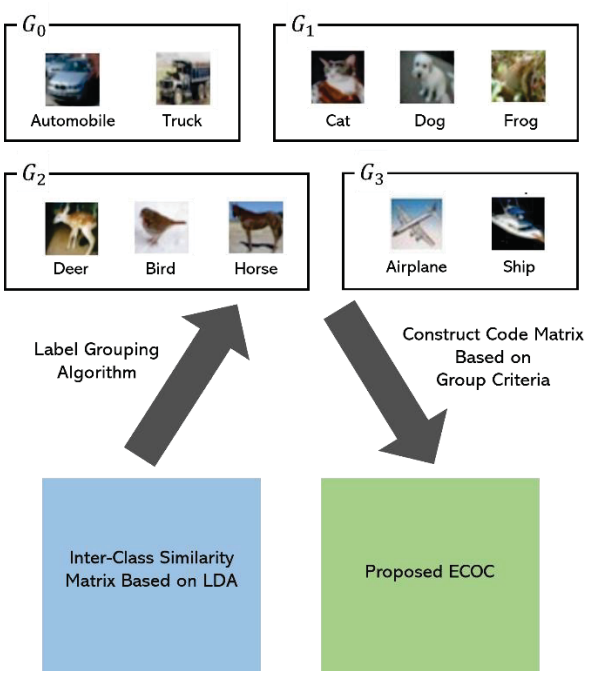

Figure 1. The proposed ECOC encoding.

Figure 1 illustrates the core idea of the proposed ECOC encoding by showing the process of grouping similar classes for CIFAR-10 and the overview of code matrix design which is discussed in Section 3. The detailed label grouping algorithm is given as Algorithms 1 and 2.

Algorithm 1 : Label Grouping Algorithm

Input : A ▷ $k \times k$ matrix with the values $A_{i,j}$
▷ $A_{i,j}$ is computed by equation (1)
Output : $G_0, \ldots, G_k$ ▷ Generate Group Sets
Initialization :
$U = \{C_0,\ C_1,\ \ldots\ C_{N-1}\}$ ▷ C_i denotes the Class i
$k = 0$

for $i = 0 \rightarrow N-1$ **do**
  $[j,\ val] = max\{A_{i,o}, \ldots, A_{i,N-1}\}$
  if $C_i \in U\ and\ val > \tau$ **then**
    Update Class Category $(G_0, \ldots, G_k, C_i, C_j, U)$

for $i = 0 \rightarrow N-1$ **do**
  if $C_i \in U$ **then**
    $[j,\ val] = max\{A_{io}, \ldots, A_{ij}\}$
    Update Class Category $(G_0, \ldots, G_k, C_i, C_j, U)$

Algorithm 2 : Update Class Category$(G_0, \ldots G_k, C_i, C_j, U)$

$G_k = \{C_i\} \cup \{C_j\}$
$temp = 0$
for $r = 0 \rightarrow k-1$ **do**
▷ The process of checking whether there is a generated group that C_j already belongs to
  if $C_j \in G_r$ **then**
    $G_r = G_r \cup G_k$
    $temp = 1$
    if $temp = 0$ **then**
      $k = k + 1$
    $U = U \setminus G_k$

Algorithm 1 yields group sets as output when the similar matrix A obtained by (1) is input into it. Algorithm 2 defined as 'Update Class Category' is a block used in the process of Algorithm 1. The function $[j, val] = \max\{A_{i,o}, \ldots, A_{i,N-1}\}$ in Algorithm 1 defines that val is the maximum value among $A_{i,o}, \ldots, A_{i,N-1}$ and j is the second subscript of the maximum value. τ in Algorithm 1 is a threshold value, which is set to 0.14 for the simulation using the CIFAR-10 datasets.

As shown in Figure 1, the group sets generated by the proposed algorithm are almost similar to the categories from the human's point of view in the real world. In the next section, we will briefly discuss objective function that is criterion for code matrix design by using this label group sets.

3 Code Matrix Design and Training/Testing Methods

3.1 Heuristic Code Matrix Design

In this section, an ECOC code matrix design scheme is proposed. In section 2, group sets are generated such that each group contains similar classes. Then, it is intuitively good to make ECOC codewords to have Hamming distance between classes in the same group larger than the Hamming distance between the classes in different groups.

For one-hot encoding, a large error in a single logit could alter the classification result. Whereas in multi-hot encoding like ECOC, single logit alone would not alter the classification result, because there are still other logits which support correct classification. Note that similar images in the same group are very likely to be misclassified [13], so the above code matrix design criterion is quite persuasive.

Let $H_{i,j}$ denote the Hamming distance between C_i and C_j which are codewords corresponding to the class i and the class j respectively. And define the function $G(\cdot)$ that input is $i \in \{0, \dots k-1\}$ (k is the number of classes) and output is the number of the group to which C_i belongs to. For example, if $C_i \in G_n$, then $G(i) = n$. To obtain optimal code matrix based on proposed label grouping scheme, we propose constraint as,

$$H_{i,j} > H_{i,l}, \ \forall i \qquad (2)$$

subject to :

$$G(i) = G(j) \text{ and } G(i) \neq G(l)$$

where $i, j, l \in \{0, \dots k-1\}$ (k is the number of classes) that each is one of the elements of the label set. To find a code matrix that satisfies proposed criterion corresponding to inequality (2), we use a heuristic search method. First, assign any one of the codewords in the codeword sets to C_0. Second, we derive a table of the number of codewords according to the Hamming distance by applying combination calculation. For example, in 4-hot bit 10 code set, the number of codewords that satisfies $H_{o,j} = 4$ is $C(4,2) \cdot C(6,2) = 90$ and the number of codewords that satisfies $H_{o,j} = 6$ is $C(4,1) \cdot C(6,3) = 320$. Third, by considering the variables obtained in the second step and label grouping algorithm, a codeword with an appropriate Hamming distance is assigned to C_i where i satisfies $G(0) = G(i)$. By repeatedly applying this method and inequality (2) which is proposed criterion, the codeword corresponding to the other class is gradually filled. By taking this approach we can improve the computational efficiency of search algorithm. While our current algorithm for the third step is more heuristic in nature, we are currently developing a more sophisticated algorithm with aims to extend and improve our method for future works.

3.2 Training / Testing Method

For CIFAR-10 dataset, the code matrix is constructed such that each codeword has 10 bits with 4-hot as far MUTE [4]. Then, the proposed encoding can be used with no change to the neural network architecture. Since the proposed encoding has the same output length as one-hot encoding, both DNN models have the same size. However, unlike one-hot encoding architecture, the proposed encoding architecture adopts a sigmoid layer instead of a softmax layer.

DNN is trained by back-propagating the binary cross-entropy loss function at each bit. For testing the trained DNN, we calculate the correlation value between the output of the sigmoid layer (final layer) and each codeword. Then the label corresponding to the codeword showing the highest correlation is the classification result.

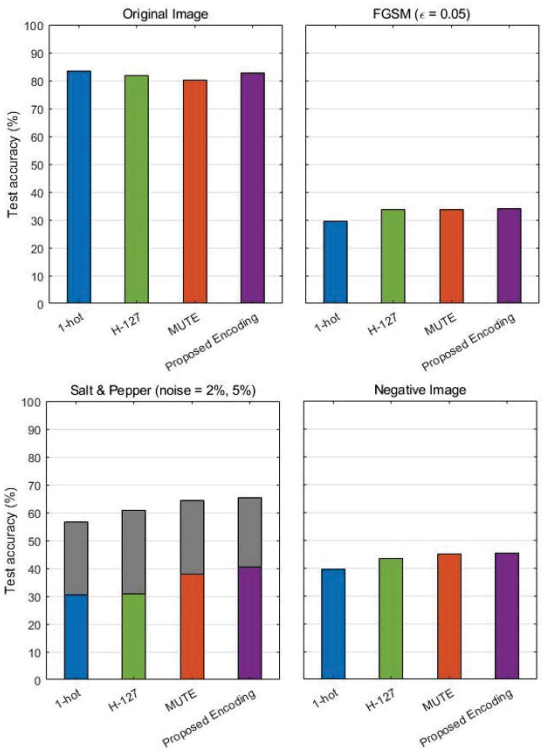

Figure 2. Test Accuracy in ResNet-20.

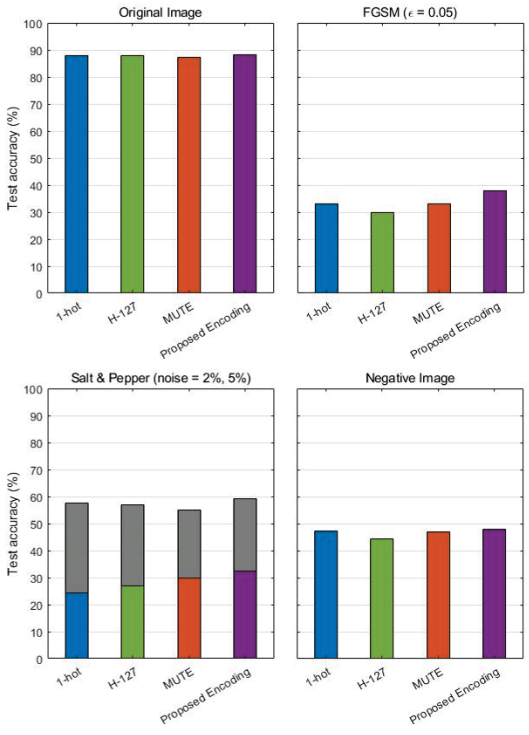

Figure 3. Test Accuracy in ResNeXt-29.

4 Simulation Results and Discussions

The CIFAR-10 datasets [7] are used for the simulation. Also, ResNet [9] and ResNeXt [10] are used which are widely used CNN architectures. ResNet has the depth of 20 and ResNeXt has depth of 29 and cardinality of 8. The standard train/test split for CIFAR-10 datasets [7] is used.

In addition to the original test set, noisy and adversarial versions of the original test sets are generated to evaluate the robustness of trained models. Salt & Pepper noise of density for 2% and 5% are generated. Negative images are generated, which have the same spatial structure as the original images but are in a diagonally opposite color space. Adversarial images are generated by using the fast gradient sign method (FGSM) [8] with epsilon = 0.05, 0.1, 0.15, 0.2, 0.25, 0.3. ResNet and ResNeXt are trained with one-hot encoding, state-of-the-art Hadamard target encoding having Length 127, MUTE with 4-hot-bit 10 code of length 10 and the proposed encoding. The proposed encoding has the same 4 hot bits and the same code length as such MUTE (for example, C_3 = [1 0 0 1 0 0 0 1 0 1]). The CIFAR-10 images are used for training with 50 epochs from random initialization. The same set of hyper parameters are used across the different target encodings. The batch size is 128 and CNNs are optimized using SGD with 09. momentum, 0.0001 weight decay, and 0.1 initial learning rate.

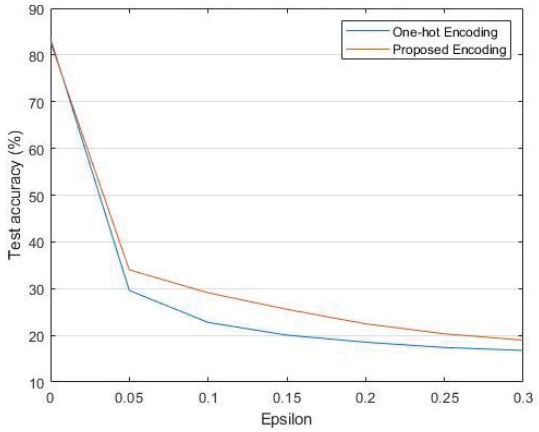

Figure 4. Test Accuracy vs Epsilon (ResNet-20).

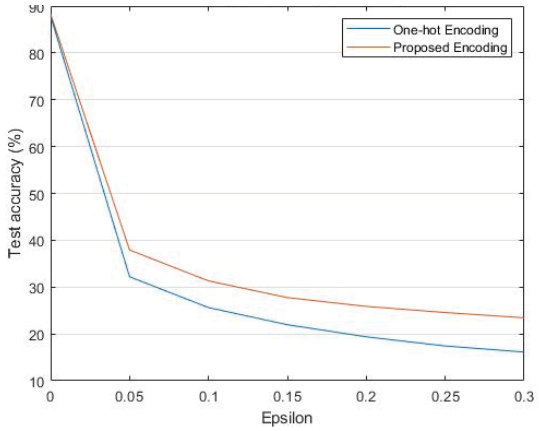

Figure 5. Test Accuracy vs Epsilon (ResNeXt-29).

Figure 2 and 3 show the test accuracy of ResNet and ResNeXt, respectively with various encodings trained on the original images of CIFAR-10 dataset and tested on the original, noisy, adversarial versions of CIFAR-10 test dataset. Note that the dark gray part in the bar graph of Salt & Pepper shows the results when increasing density from 2% to 5%. The proposed encoding shows better average performance than one-hot encoding and Hadamard target encoding. The proposed encoding with ResNet improves one-hot average accuracy by 6.3% and MUTE by 1.5%. The proposed encoding with ResNeXt also improves one-hot average accuracy by 4% and MUTE by 3.4%. In particular, the proposed method outperforms both CNN archiectures for FGSM adversarial examples and outperforms ResNeXt for Salt & Pepper noisy datasets. Figures 4 and 5 shows the test accuracy of ResNet and ResNeXt, respectively, by changing the epsilon value of FGSM. It is shown that the average test accuracy was 3.2% and 5.2% higher than that of one-hot encoding, respectively.

5 Conclusion and Future Works

In this paper, a label grouping method is proposed, which provides a simple and persuasive criterion for code matrix design. By using this scheme, a new code design method is proposed, which shows better robustness than the conventional one-hot encoding and the other ECOC encodings.

While extracting similarity distribution from linear discriminant analysis (LDA) is effective, the better way to learn inter-class similarities for constructing ECOC still remains open. However, the idea of grouping labels like a superclass or semantic category is meaningful. Also, it is a good future work to investigate how to design good ECOC code matrix based on label grouping.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2014057).

References

[1] T. G. Dietterich and G. Bakiri, “Solving Multiclass Learning Problems via Error-Correcting Output Codes,” Journal of Artificial Intelligence Research 2, 1995, pp. 263-286.

[2] S. Yang, P. Luo, C. C. Loy, K. W. Shum, and X. Tang, “Deep Representation Learning with Target Coding,” in Proceedings of the 29th AAAI Conference on Artificial Intelligence, 2015, pp. 3848-3854

[3] S. Gupta and S. Amin, “Integer Programming-based Error-Correcting Output Code Design for Robust Classification,” arXiv: 2011.00144, 2020.

[4] M. S. Jaiswal, B. Kang, J. Lee, and M. Cho, “MUTE: Inter-class Ambiguity Driven Multi-hot Target Encoding for Deep Neural Network Design,” in Proceedings of the IEEE/CVF CVPR Workshops, 2020, pp. 754-755.

[5] I. Hwang, J. Lee, F. Liu, and M. Cho. “SimEx: Express Prediction of Inter-dataset Similarity by a Fleet of Autoencoders.” arXiv preprint arXiv:2001.04893, 2020.

[6] P. S. Negi, D. Chan, and M. Mahoor, "Leveraging Class Similarity to Improve Deep Neural Network Robustness," arXiv: 1812.09744, 2018.
[7] Alex Krizhevsky and Geoffrey Hinton, "Learning Multiple Layers of Features from Tiny Images" Technical report, Citesser, 2009.
[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples" arXiv preprint arXiv:1412.6572, 2014.
[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE conference on CVPR, 2016, pp. 770-778.
[10] S. Xie, R. Girshick, P. Doll´ar, Z. Tu, and K. He. "Aggregated Residual Transformations for Deep Neural Networks." In Proceedings of the IEEE conference on CVPR, 2017, pp. 1492-1500.
[11] G. R. Gare and J. M. Galeotti. "Exploiting Class Similiarity for Machine Learning with Confidence Labels and Projective Loss Function" arXiv preprint arXiv:2103.13607, 2021.
[12] G. Charpiat, N. Girard, L. Felardos, Y. Tarabalka, "Input Similarity from the Neural Network Perspective" in Proceedings of the NeurIPS, 2019, 5342-5351.
[13] K. Arino and Y. Kikuta, "ClassSim: Similarity between Classes Defined by Misclassification Ratios of Trained Classifiers" arXiv:1082.01267, 2018.