MUTE: Data-Similarity Driven Multi-hot Target Encoding for Neural Network Design

Mayoore S. Jaiswal[1] **Bumsoo Kang**[2] **Jinho Lee**[3] **Minsik Cho**[1]

Abstract

Target encoding is an effective technique to deliver better performance for machine learning methods, but, existing approaches require significant increase in the learning capacity, thus demand higher computation power and more training data. In this paper, we present a novel and efficient target encoding scheme, MUTE to improve both generalizability and robustness of a target model by understanding the inter-class characteristics of a target dataset. By extracting the confusion-level between the target classes in a dataset, MUTE strategically optimizes the Hamming distances among target encoding. Such optimized target encoding offers higher classification strength for neural network models with negligible computation overhead and without increasing the model size.

1 Introduction

Scalable artificial intelligent systems require a methodology for efficient neural network design that can generalize well, learn semantics of the training dataset, and resist adversarial attacks. However, existing methods have been shown to learn dataset bias (Torralba et al., 2011; Tommasi et al., 2017; Li et al., 2014), and fail to deliver sufficient generalization capability. Poor generalization makes models unpredictable, causes potential ethical issues, and misguides neural network design (Zhang et al., 2017; Gebru et al., 2018). To tackle the generalization problems, target encoding has been studied for both conventional machine learning and deep neural network architectures and proven to be highly effective (Akata et al., 2016; Frome et al., 2013; Hsu et al., 2009; Cisse et al., 2013). Yet, many prior works in target encoding require a long encoding sequence (which increases the model size) and fail to tailor the encoding for a given task or dataset. Furthermore, they do not investigate the effects of different target encodings against noisy data and adversarial attacks.

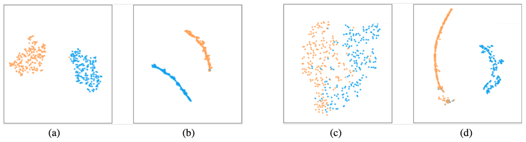

Figure 1. TSNE (Maaten & Hinton, 2008) visualizations of a ConvNet (refer Section 3 for architecture) with one-hot encoding trained on MNIST (a, c) and a ConvNet with MUTE trained on MNIST (b, d). Classes well-separated in features space in models trained with one-hot encodings (a), remain well separated in models trained with MUTE (b). However, class not-well-separated in features space in models trained with one-hot encodings (c), are well-separated in feature space when trained with MUTE.

In this work, we propose MUTE, a systematic approach to make deep learning models generalize better by optimizing the target encoding (Akata et al., 2016; Frome et al., 2013; Hsu et al., 2009; Cisse et al., 2013). Unlike the conventional one-hot method where the Hamming distance between labels is fixed at 2, MUTE generates a multi-hot encoding by exploiting the expression power of a given output encoding length. MUTE strategically extracts the *similarity* between pairs of classes from a dataset, and leverages that information to obtain a multi-hot encoding such that semantically closer classes are forced to be further apart in the label space in terms of the Hamming distance. MUTE ensures that Stochastic Gradient Decent (SGD) algorithm extracts distinctive features between two easy-to-confuse classes, which in turn reduces the chance of mis-prediction under noisy and noiseless conditions. Figure 1 illustrates the high-level idea in MUTE, that is it increases the distance between classes in feature space.

2 Data-Similarity Driven Multi-hot Target Encoding

We propose a new target encoding system, MUTE, where multiple output bits are activated. For an $N-$class classification problem with MUTE, the output layer has $N-$bits, out of which K bits are 1s as chosen, where $K > 1$. Each output bit has a bounded non-linear activation function and trained such that the binary cross entropy loss between the prediction and target label is minimized.

[1] IBM, Austin, TX, USA [2] KAIST, Daejeon, South Korea [3] Yonsei University, Seoul, South Korea. Correspondence to: Mayoore S. Jaiswal <mayoore.s.jaiswal@ibm.com>.

Proceedings of the 3^{rd} MLSys Conference, Austin, TX, USA, 2020. Copyright 2020 by the author(s).

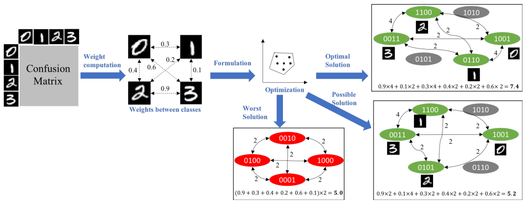

Figure 2. Flow chart of multi-hot encoding generation in MUTE.

Generating Weights For Target Codes. The objective of this step is to quantitatively identify similar classes and assign weights to the level of similarity. A confusion matrix (CM) representing inter-class similarities is a $N \times N$ matrix for a $N-$class dataset with the elements being the confusion metric between a pair of classes as shown in Figure 2 (top-left). We use the method proposed in (Anonymous, 0000) to generate CM where the confusion-level between classes of a dataset can be determined by reconstructing data for each class using an autoencoder trained using class C_i and determining the reconstruction error for every other class C_j in the dataset. Once CM is obtained for a given dataset, MUTE can convert the confusion matrix into weights in the following method: the weights are obtained by subtracting the diagonal of CM (self-error values) from CM, finding the minimum error of the upper and lower triangles of CM, thresholding larger errors to eliminate dissimilar classes, and scaling the non-zero values of CM.

Generating Target Codes. With weights obtained from previous step, the next step in MUTE is to generate a set of multi-bit target encoding. Our goal is two-fold: **a**) to generate encoding that maximize Hamming distances among all encoding and **b**) to assign a pair of encoding that has larger Hamming distance to a pair of more similar classes. In this light, we generate encodings that maximize the total minimum Hamming distance and the Hamming distances between classes based on inter-class similarities.

Figure 2 shows an example of possible outcomes from the optimization step. With one-hot encoding (bottom-center), there exists one trivial solution and the weighted sum of the Hamming distances is the lowest among the solutions illustrated. Figure 2 also shows the difference between the optimal and a possible solution in terms of the Hamming distance and the generated encoding: the optimal solution picks a set of encoding and assigns them to the four classes such that more similar classes (i.e., 2 vs. 3 with weight 0.9) are assigned to two codes with the larger Hamming distance.

Training Method. The MUTE could be used with any CNN model with no changes to the architecture. The softmax classification layer is replaced with a sigmoid layer. In this method, the number of neurons in the CNN and the computational complexity is the same as using one-hot encoding.

Inferencing Method. An input image is forward propagated through the trained model. The output of the sigmoid layer is thresholded by setting all but the $top - K$ activated bits to 0, where K is the number of bits used in the MUTE. Then, the Euclidean distance between the thresholded output and each of MUTE encodings are computed. The label corresponding to the closest MUTE code is the classification result.

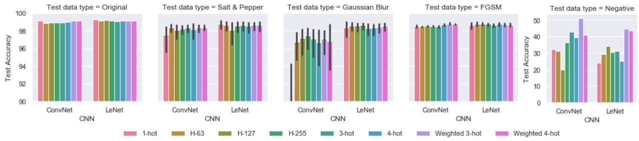

Figure 3. MUTE with LeNet and ConvNet architectures trained on MNIST data improves the one-hot average test accuracy by 2.8% and 7.1% respectively. Whereas Hadamard target encoding improves one-hot performance only by 1.7% and 3.5%.

3 Results & Discussion

MNIST (LeCun et al., 1998) dataset was trained and tested with two CNNs: LeNet (LeCun et al., 1998) and ConvNet. ConvNet has 2 convolutional layers with 5×5 kernel size, followed by a fully connected layer with 50 neurons and a final sigmoid layer. CNN architectures were trained with original images in the training dataset for 200 epochs. The trained models were tested with original images, and noisy and adversarial versions of the original images in the test set. Noisy images were negative images (Hosseini et al., 2017), Gaussian blurred images with $\sigma = 1$ and 2 and Salt & Pepper noise at 2% and 5%. Adversarial images were created using the Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2014) with $\epsilon = 0.05, 0.1$ and 0.2. In our experiments, we compared against the conventional one-hot encoding and Hadamard target encoding methods. We also conducted experiments with weighted and unweighted MUTE for different number of hot bits.

Figure 3 shows the test accuracy of LeNet and ConvNet architectures with different target encodings trained on original images in the MNIST training dataset and tested on original, noisy and adversarial versions of the MNIST test dataset. The barplots illustrate the central tendency for different test datasets and uncertainty (error bars) for test images impacted by varying amounts of Gaussian blur, Salt & Pepper noise and FGSM. The proposed MUTE method has better average test performance than one-hot encoding or Hadamard target encoding (H-63, H-127, and H-255). The best test accuracy on original images reported by (Yang et al., 2015) using Hadamard Codes on MNIST is 85.47% using direct classification with H-255. The proposed MUTE method improves this result by 13.61 percentage points on average.

References

Akata, Z., Perronnin, F., Harchaoui, Z., and Schmid, C. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7): 1425–1438, 2016.

Anonymous. Task-agnostic early prediction of inter-dataset similarity by using monomath autoencoders. In *Anonymous*, pp. 0. Anonymous, 0000.

Cisse, M. M., Usunier, N., Artieres, T., and Gallinari, P. Robust bloom filters for large multilabel classification tasks. In *Advances in Neural Information Processing Systems*, pp. 1851–1859, 2013.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pp. 2121–2129, 2013.

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H. M., III, H. D., and Crawford, K. Datasheets for datasets. *CoRR*, 2018.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Hosseini, H., Xiao, B., Jaiswal, M., and Poovendran, R. On the limitation of convolutional neural networks in recognizing negative images. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 352–358. IEEE, 2017.

Hsu, D. J., Kakade, S. M., Langford, J., and Zhang, T. Multi-label prediction via compressed sensing. In *Advances in neural information processing systems*, pp. 772–780, 2009.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, Y., Hou, X., Koch, C., Rehg, J. M., and Yuille, A. L. The secrets of salient object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 280–287, 2014.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov): 2579–2605, 2008.

Tommasi, T., Patricia, N., Caputo, B., and Tuytelaars, T. A deeper look at dataset bias. In *Domain Adaptation in Computer Vision Applications*, pp. 37–55. Springer, 2017.

Torralba, A., Efros, A. A., et al. Unbiased look at dataset bias. In *CVPR*, volume 1, pp. 7. Citeseer, 2011.

Yang, S., Luo, P., Loy, C. C., Shum, K. W., and Tang, X. Deep representation learning with target coding. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. 2017. URL https://arxiv.org/abs/1611.03530.