EE-AE: AN EXCLUSIVITY ENHANCED UNSUPERVISED FEATURE LEARNING APPROACH

Jingcai Guo, Song Guo

Department of Computing, The Hong Kong Polytechnic University, Hong Kong
cscjguo@comp.polyu.edu.hk, song.guo@polyu.edu.hk

ABSTRACT

Unsupervised learning is becoming more and more important recently. As one of its key components, the autoencoder (AE) aims to learn a latent feature representation of data which is more robust and discriminative. However, most AE based methods only focus on the reconstruction within the encoder-decoder phase, which ignores the inherent relation of data, i.e., statistical and geometrical dependence, and easily causes overfitting. In order to deal with this issue, we propose an Exclusivity Enhanced (EE) unsupervised feature learning approach to improve the conventional AE. To the best of our knowledge, our research is the first to utilize such exclusivity concept to cooperate with feature extraction within AE. Moreover, in this paper we also make some improvements to the stacked AE structure especially for the connection of different layers from decoders, this could be regarded as a weight initialization trial. The experimental results show that our proposed approach can achieve remarkable performance compared with other related methods.

Index Terms— Unsupervised learning, Exclusivity, Feature learning, Autoencoder.

1. INTRODUCTION AND RELATED WORK

With the development of deep learning techniques, more and more applications adopt the deep neural networks (DNNs) to handle multiple tasks and obtain successive state-of-the-art performances [1, 2]. However, most of such DNNs are categorized into the supervised learning [3] with high demands for labeled data. On the other hand, it is much easier to collect the unlabeled data, then how to utilize it falls into unsupervised learning [4]. To this issues, the natural idea is to combine them. We apply the DNNs with the fashion of unsupervised learning, and aim to obtain more robust and discriminative feature representation without the rich resource of labeled data.

As one key component of unsupervised learning, the autoencoder (AE) is a widely used and promising method aims
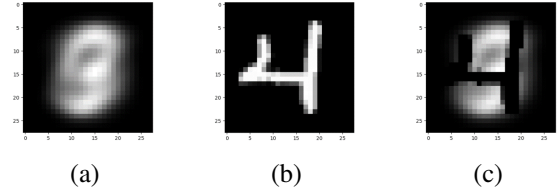
Fig. 1. Exclusive Feature Illustration: (a) is the mean feature representation, (b) is one specific example, i.e., digit 4, (c) is the feature representation which exclude this example.

to learn a latent feature representation of data [5]. It is flexible to implementation and can be stacked easily to form a deep structure, which endows a model the ability to perform complex information extraction and representation [6]. The AE is with an encoder-decoder structure. Raw data example x is first projected to a latent feature space by the encoder, which is more compressed, sparser, or any specific nature as we need:

$$h = a_e\left(w_e \cdot x + b_e\right),\tag{1}$$

where w_e is the weight of encoder, b_e is a bias and a_e is a nonlinear activation function used to achieve nonlinearity so that the network can better model complex problem. Next, the latent feature representation h is projected back to the original feature space by the decoder, in which a reconstructed data example $\hat{x}$ is obtained:

$$\hat{x} = a_d\left(w_d \cdot h + b_d\right),\tag{2}$$

where w_d is the weight of decoder, b_d is a bias and a_d is a nonlinear activation function. The loss of $\hat{x}$ and x is calculated as:

$$L_a = \|x - \hat{x}\|_2^2.\tag{3}$$

The AE learns to minimze Eq. (3) and forces h to retain the most powerful expression of data.

However, if the AE is endowed with too much capacity, the encoder and decoder may perform among themselves and eventually converge to an identity function [7]. In recent years, some related regularization approaches have been proposed to address this issue, such as Denoising AE [6], Sparse AE [8], Graph AE [9], Winner-take-all AE [10], Similarity-aware AE [11], etc. Despite the efforts made, however, most AE based methods only focus on the reconstruction and may

ignore some inherent relation of data, i.e., statistical and geometrical dependence. We argue it an obstacle to learning a more robust and discriminative feature representation. Let us use the handwritten digits as an illustration. In Fig. 1, start from (a), we first obtain the global mean feature representation of whole-set. Then we randomly choose an example, i.e., a digit 4, as shown in (b). Next, we exclude (b) out from the global feature representation and obtain (c). We can observe that the obtained feature representation (c) is radically different from the chosen digit 4. Actually, in an ideal state, if we flatten these two representations, we may obtain two orthogonal vectors. Although the ideal state is not fully available, this exclusivity intuitively exists. Taking the exclusivity into account, we may deal with each example in the case of heterogeneous and homologous respectively, and discover more latent information from the raw data.

This insight inspires us to propose a novel regularization technique to boost the unsupervised feature learning within AE. In this paper, we propose a novel AE based unsupervised feature learning approach called Exclusivity Enhanced Autoencoder (EE-AE), we introduce two exclusivity constraints that can better deal with the statistical and geometrical dependence of data. Moreover, we also make some improvements to the stacked AE structure especially for the connection of different layers on the decoder part, and this makes full utilization of the layer difference and cooperation within AE. Extensive experiments are carried out on several benchmark datasets which show that the proposed EE-AE outperforms other existing representative methods.

2. EXCLUSIVITY ENHANCED AUTOENCODER

2.1. Exclusivity Concept

Data has its own inherent relation, i.e., statistical and geometrical dependence. In unsupervised learning, we don't take labels into account, so how to extract useful information from data itself counts. Suppose we have several examples in total: $C = \{x_1, x_2, \cdots x_n\}$, where C is the global-set includes n examples. Let x_j be a random example, then we exclude x_j out from C, the rest of global-set is $C' = \{x_1, x_2, \cdots x_n\} / \{x_j\}$. Then we want to compare C' with x_j, normally we need to compare $n-1$ pairs. However, the one to one comparison is not very necessary in this case. So we consider to compare x_j with the global of C', here we replace C' with the mean feature representation $\bar{C}'$. The similarity S between x_j and $\bar{C}'$ is excepted to have a lower bound γ:

$$\lim S\left(x_j, \bar{C}'\right) \to \gamma, \tag{4}$$

where S is a similarity function and $\gamma \to 0$ under cosine measurement. On the other hand, the rest of global-set C' still contains a certain number of examples that belong to the same class of x_j. So we can exclude m examples out from C' and obtains: $D = \{x_1^m, x_2^m, \cdots x_m^m\} \subseteq C'$, where D contains

$m\, (m \ll n)$ examples that are most similar with x_j. We also replace D with the mean feature representation $\bar{D}$, the similarity S between x_j and $\bar{D}$ is excepted to have a upper bound δ:

$$\lim S\left(x_j, \bar{D}\right) \to \delta, \tag{5}$$

where $\delta \to 1$ under cosine measurement.

Statistically speaking, the variation between x_j and $\bar{C}'$ should be large enough, and the variation between x_j and $\bar{D}$ should be small enough. Geometrically speaking, x_j and $\bar{C}'$ should be far enough, and x_j and $\bar{D}$ should be close enough. We use $\bar{C}'$ to define the heterogeneous case (i.e., different classes) to example x_j, and use $\bar{D}$ to define the homologous case (i.e., same class). Although the ideal state is not fully available, i.e., C' contains a certain number of examples whose class is the same as x_j, the impact is limited in the global perspective, the exclusivity intuitively exists.

2.2. Exclusivity Enhanced Autoencoder

The conventional AE focuses on the reconstruction from the encoder-decoder phase. We consider n examples dataset, similar with Eq. (3), the objective can be written as:

$$L_a = \sum_{i=1}^{n} \|x_i - \hat{x}_i\|_2^2. \tag{6}$$

This objective ignores some inherent relation of data and easily causes overfitting. Inspired by the concept of exclusivity, we find the heterogeneous case $C'^{(i)}$ and the homologous case $D^{\overline{(i)}}$ for each example x_i in the original feature space. Then, we focus on the latent feature space and force h_i to apart from $f_e(C'^{(i)})$ and close to $f_e(D^{\overline{(i)}})$, respectively. Where $f_e(\cdot)$ is the encoder that projects element to latent feature space.

To this end, we first introduce an auxiliary function $\Omega(\cdot)$ that activates vector ε on dimension-wise:

$$\Omega\left(\varepsilon\right) : \left\{ \begin{array}{ll} \varepsilon_i & \text{if } \varepsilon_i \geq 0 \\ 0 & \text{if } \varepsilon_i < 0 \end{array} \right. , \tag{7}$$

where we activate each dimension ε_i in vector ε. Then, in order to force the latent feature representation of data to apart from their heterogeneous case, we consider a new constraint $L_h^{(1)}$ to maximize the variation between examples and their heterogeneous case:

$$L_h^{(1)} = \sum_{i=1}^{n} \left[\frac{\Omega\left[f_e(C'^{(i)}) - h_i\right] \cdot h_i}{\left\|\Omega\left[f_e(C'^{(i)}) - h_i\right]\right\| \|h_i\|} \right] = \sum_{i=1}^{n} \left[\frac{\Omega\left[f_e(C'^{(i)}) - f_e(x_i)\right] \cdot f_e(x_i)}{\left\|\Omega\left[f_e(C'^{(i)}) - f_e(x_i)\right]\right\| \|f_e(x_i)\|} \right]. \tag{8}$$

We can further minimize the variation between examples and their homologous case by considering another constraint $L_h^{(2)}$:

$$L_h^{(2)} = \sum_{i=1}^{n} \left[\frac{\Omega\left[f_e(D^{\overline{(i)}}) - h_i\right] \cdot h_i}{\left\|\Omega\left[f_e(D^{\overline{(i)}}) - h_i\right]\right\| \|h_i\|} \right] = \sum_{i=1}^{n} \left[\frac{\Omega\left[f_e(D^{\overline{(i)}}) - f_e(x_i)\right] \cdot f_e(x_i)}{\left\|\Omega\left[f_e(D^{\overline{(i)}}) - f_e(x_i)\right]\right\| \|f_e(x_i)\|} \right]. \tag{9}$$

For these two functions, the $L_h^{(1)}$ has a lower bound that close to 0, and the $L_h^{(2)}$ has an upper bound that close to 1. In

order to jointly optimize these two terms with conventional AE objective, we subtract $L_h^{(2)}$ from 1, and combine these two terms:

$$L_h = L_h^{(1)} + \left(1 - L_h^{(2)}\right). \tag{10}$$

Finally, the unified objective of our Exclusivity Enhanced Autoencoder (EE-AE) can be described as:

$$\begin{aligned} L &= L_a + \lambda L_h \\ &= L_a + \lambda \left[L_h^{(1)} + \left(1 - L_h^{(2)}\right) \right], \end{aligned} \tag{11}$$

where we jointly minimize this objective during training. L_a is the conventional AE reconstruction loss, $L_h^{(1)}$ is the exclusivity constraint that maximizes the variation between examples and their heterogeneous case, $(1 - L_h^{(2)})$ is the exclusivity constraint that minimizes the variation between examples and their homologous case. λ is an hyper-parameter controls the balance between the conventional reconstruction loss and added constraints.

2.3. Stacked Autoencoders Structure

The conventional stacked autoencoders (AEs) adopt the stacked structure to every single AE [6, 12]. Each time, the latent feature representation h of the current AE is fed to the input of next AE. Such a process is repeated with certain steps to realize the stacked structure.

Recently, Yosinski et al. [13] experimentally quantified the generality versus specificity of each layer in DNNs, and shown that the feature representation of an example normally go through a **specific** $\rightarrow$ **general** $\rightarrow$ **specific** three level phases from the input to output layers, the deeper the more general. Inspired by this analysis, we propose to fully consider the connection of different layers from decoders. Specifically, we not only stack the encoder but also the decoder. Then in the last stacked structure network, we further fine-tune the whole network to achieve better cooperation of different AE layers (Fig. 2). This can be regarded as a weight initialization trial. we constrain the weight variance ratio of layers as: $1 - \eta \le \frac{\|W_F\|_{p.1}}{\|W_F'\|_{p.1}} \le 1 + \eta$, where W_F and W_F' are the original and fine-tuned weight respectively. $\|\cdot\|_{p.1}$ is the l_p norm on a vector and η is a hyper-parameter controls the weight variance ratio of layers.

3. EXPERIMENTS

We compare our proposed Exclusivity Enhanced Autoencoder (EE-AE) with several state-of-the-art methods on two widely used datasets COIL100 [14] and MNIST [15]. The experimental results and analysis are discussed in detail for each dataset respectively.
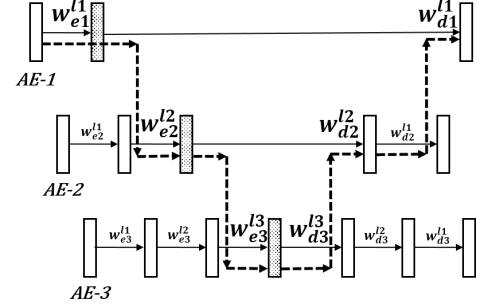

Fig. 2. Modified Stacked Autoencoders Structure: A case for 3 AEs (AE-1 to 3). Solid line stands for each single AE and the dotted line is the stacked structure. W_{ei}^{li} and W_{di}^{li} is the weight for encoder and decoder, respectively.

3.1. Results on COIL100

Experiment Setup: The COIL100 contains 7200 color images of 100 objects. Images of the objects are taken at pose intervals of 5 degrees, corresponding to 72 poses per object. We convert these images to grayscale images and resize them to 32x32 pixels. We randomly select 10 images for each object to form the training set and the rest images are the testing set. For the training set, we also consider their horizontal mirror feature. We use our proposed EE-AE for unsupervised feature learning. The nearest neighbor classifier whose inputs are the learned feature representations from EE-AE is applied to achieve the recognition. These processes are repeated 10 times and we report the average recognition results. We also split the dataset to 20, 40, 60, 80 object subsets and compare the recognition accuracy on these subsets and whole-set respectively. We compare our proposed approach with 9 existing representative methods (Table 1). All competitors and our EE-AE are under the same settings and tuned to achieve their best performance. For all AE based methods, we also adopt the same network architecture and report the results on the learned feature representations of the last stacked hidden embedding layer.

Table 1. Accuracy (%) for COIL100

Method	20	40	60	80	100	Average
PCA [16]	90.41	89.27	87.26	85.80	84.17	87.37
KPCA [17]	90.12	88.06	85.75	83.97	82.52	86.08
NPE [18]	91.76	89.59	87.27	85.94	85.03	87.91
SCC [19]	91.49	89.39	85.91	83.91	82.91	86.72
SDNMF [20]	89.91	87.88	82.94	81.44	78.60	84.15
Denoise-AE [6]	91.75	90.52	88.58	86.35	85.17	88.47
Sparse-AE [8]	92.26	91.17	88.59	86.63	85.55	88.83
Graph-AE [9]	93.37	91.33	89.11	86.67	85.96	89.28
SSA-AE [11]	96.97	94.04	91.87	90.42	88.78	92.41
EE-AE (ours)	**98.39**	**95.20**	**93.09**	**92.58**	**90.90**	**94.03**

3519

Results & Analysis: The results are shown in Table 1. We can see that our proposed EE-AE outperforms all competitors with great advantage on each subset contains 20, 40, 60 and 80 objects, and whole-set (100 objects) respectively. What's more, we can also have the observation that all the AE based methods generally obtain better performance than traditional methods, which highlights the merit of AE for unsupervised feature learning. Then we further evaluate the sensitivity of the hyper-parameters: (1) The λ controls the balance between the conventional reconstruction loss and exclusivity constraints in Eq. (11). (2) The number (m) of most similar examples when finding the homologous case ($\bar{D}$) for each example; (3) The number of stacked AEs (denote as s for simplicity). The evaluation is conducted on the 20-object subset of COIL100. When evaluating the sensitivity of one hyperparameter, we fix others to their best points. Fig. 3 shows the results. We can see that as λ becomes larger, the performance of our model becomes better and then decreases, the optimal λ is around 7. As to m, as it becomes larger, the performance of our model becomes better and finally tends to be relatively stable, and the optimal m is around 6. As to s, the optimal number is 3.
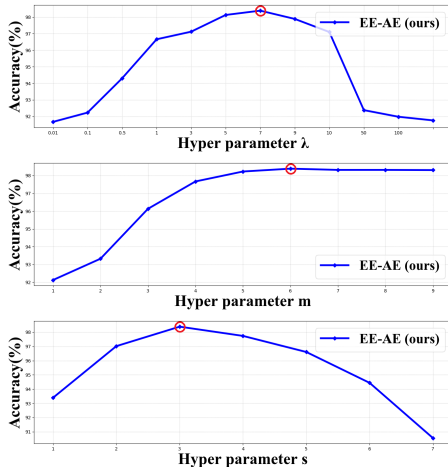

Fig. 3. Sensitivity of Hyper-parameters

3.2. Results on MNIST

Experiment Setup: The MNIST [15] contains 60,000 training images and 10,000 testing images. In order to better evaluate the learning ability of our proposed EE-AE, we split the MNIST into two scenarios: (1) 1-set: The whole-set. (2) $\frac{1}{6}$-set: Only contains 10000 (out of 60000) training images and 10,000 testing images. We train EE-AE and classifier under these two scenarios respectively. Our proposed EE-AE is used for unsupervised feature learning. With the learned feature representations, we train an SVM classifier to achieve the recognition. As to the SVM, we adopt 1-versus-1 setting [21,22] for the multi-class classification. The purpose of

Table 2. Accuracy (%) for MNIST

Method		1-set	$\frac{1}{6}$-set
SAE [6]		98.60	90.37
SDAE [6]		98.72	91.04
EE-AE (ours)	$\eta = 0.0$	98.43	93.56
	$\eta = 0.2$	98.59	94.33
	$\eta = 0.4$	98.67	94.89
	$\eta = 0.6$	98.67	**95.23**
	$\eta = 0.8$	**98.81**	94.61
	$\eta = 1.0$	98.73	94.57

our experiments on the MNIST is to quantitatively evaluate the effectiveness of our model.

Results & Analysis: The results are shown in Table 2. When evaluating on 1-set scenario, our model obtains a comparable performance with these competitors. However, when evaluating on $\frac{1}{6}$-set, our model outperforms these competitors with great advantages. This shows that our proposed EE-AE has significant better learning ability to discover and extract useful information from data, even the data volume is relatively small. This results in a better generalization ability.

3.3. Environment and Implementation

We use Pytorch to implement our experiment. As to AEs, we apply convolution, batch normalization, ReLU and maxpooling layers to form the encoder. The convolution layers have 3 settings: 16 channels, 3x3 kernel, stride=1 and padding=1. 32 channels, 3x3 kernel, stride=1 and padding=0; 64 channels, 3x3 kernel, stride=1 and padding=0. For all the maxpooling layers, we set a 2x2 kernel. As to the decoder, we apply de-convolutional layers which are exactly the reverse of convolution to obtain up-exampled feature map. Also, we use batch normalization, ReLU and maxunpooling layers to cooperate with the network. We append an embedding layer on the top of the hidden codes with 128-dimensional features. The stochastic gradient descent (SGD) is applied for the optimization.

4. CONCLUSION AND FUTURE WORK

In this paper, we propose a novel autoencoder based unsupervised feature learning approach called Exclusivity Enhanced Autoencoder (EE-AE). We utilize the exclusivity constraints to cooperate with feature extraction within AE. Our model can better deal with the statistical and geometrical dependence of data and results in a more robust and discriminative feature representation. Extensive experiments verified the effectiveness of our model. In the future, we plan to investigate more efficient techniques and apply the exclusivity to larger datasets.

5. REFERENCES

[1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436, 2015.

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *CVPR*, 2018, vol. 3, p. 6.

[3] David H Wolpert, "The status of supervised learning science circa 1994: the search for a consensus," in *The mathematics of generalization*, pp. 1–10. CRC Press, 2018.

[4] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe, "Unsupervised learning of depth and ego-motion from video," in *CVPR*, 2017, vol. 2, p. 7.

[5] Pierre Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*, 2012, pp. 37–49.

[6] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[7] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio, *Deep learning*, vol. 1, MIT press Cambridge, 2016.

[8] Jun Xu, Lei Xiang, Qingshan Liu, Hannah Gilmore, Jianzhong Wu, Jinghai Tang, and Anant Madabhushi, "Stacked sparse autoencoder (ssae) for nuclei detection on breast cancer histopathology images," *IEEE transactions on medical imaging*, vol. 35, no. 1, pp. 119–130, 2016.

[9] Wenchao Yu, Guangxiang Zeng, Ping Luo, Fuzhen Zhuang, Qing He, and Zhongzhi Shi, "Embedding with autoencoder regularization," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 208–223.

[10] Alireza Makhzani and Brendan Frey, "A winner-take-all method for training sparse convolutional autoencoders," in *NIPS Deep Learning Workshop*. Citeseer, 2014.

[11] Wenqing Chu and Deng Cai, "Stacked similarity-aware autoencoders," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. AAAI Press, 2017, pp. 1561–1567.

[12] Joao Maria, Joao Amaro, Gabriel Falcao, and Luís A Alexandre, "Stacked autoencoders using low-power accelerated architectures for object recognition in autonomous systems," *Neural Processing Letters*, vol. 43, no. 2, pp. 445–458, 2016.

[13] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, "How transferable are features in deep neural networks?," in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[14] Sameer A Nene, Shree K Nayar, Hiroshi Murase, et al., "Columbia object image library (coil-20)," 1996.

[15] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[16] Matthew Turk and Alex Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[17] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[18] Xiaofei He, Deng Cai, Shuicheng Yan, and Hong-Jiang Zhang, "Neighborhood preserving embedding," in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*. IEEE, 2005, vol. 2, pp. 1208–1213.

[19] Deng Cai, Hujun Bao, and Xiaofei He, "Sparse concept coding for visual analysis," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2905–2910.

[20] Wei Qian, Bin Hong, Deng Cai, Xiaofei He, Xuelong Li, et al., "Non-negative matrix factorization with sinkhorn distance.," in *IJCAI*, 2016, pp. 1960–1966.

[21] Kai-Bo Duan, Jagath C Rajapakse, and Minh N Nguyen, "One-versus-one and one-versus-all multiclass svm-rfe for gene selection in cancer classification," in *European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*. Springer, 2007, pp. 47–56.

[22] Jingcai Guo, "An improved incremental training approach for large scaled dataset based on support vector machine," in *Big Data Computing Applications and Technologies (BDCAT), 2016 IEEE/ACM 3rd International Conference on*. IEEE, 2016, pp. 149–157.