17th International Conference on Machine Vision Applications (MVA)
Fully Online, July 25–27, 2021.

Learning VAE with Categorical Labels for Generating Conditional Handwritten Characters

Keita Goto
Tokyo Institute of Technology
goto.k.al@m.titech.ac.jp

Nakamasa Inoue
Tokyo Institute of Technology
inoue@c.titech.ac.jp

Abstract

The variational autoencoder (VAE) has succeeded in learning disentangled latent representations from data without supervision. Well disentangled representations can express interpretable semantic value, which is useful for various tasks, including image generation. However, the conventional VAE model is not suitable for data generation with specific category labels because it is challenging to acquire categorical information as latent variables. Therefore, we propose a framework for learning label representations in a VAE by using supervised categorical labels associated with data. Through experiments, we show that this framework is useful for generating data belonging to a specific category. Furthermore, we found that our framework successfully disentangled latent factors from similar data of different classes.

1 Introduction

Frameworks for learning data representations have been actively researched in recent years. In particular, unsupervised learning of useful embedded features of speeches [1] and images [2] has led to performance improvements on various tasks.

Among them, variational autoencoders (VAEs) [3] have succeeded in disentangling data into a finite number of intuitively interpretable representations without any supervision information. Each of the disentangled representations depicts a semantic value, such as *angle*, *thickness*, *width*, and *height*, for example, for handwritten characters.

Similar to autoencoders, VAEs consists of an encoder that generates latent variables from data and a decoder that reconstructs the data from the variables. The VAE acquires a disentangled representation by assuming that the variables are generated from independent normal distributions. VAEs have been applied to data generation tasks such as speaker conversion [4] and image-to-image translation [5] since they can generate data from human interpretable factors.

Various methods for improving naive VAEs have been proposed. β-VAE [6] has succeeded in enhancing the disentanglement of latent variables by increasing the regularization weight. The original β-VAE has also been reported to generate lower quality images, but this can be improved using some learning techniques [7, 8].

While the VAE and β-VAE only consider continuous latent variables, the Conditional VAE (CVAE) [9] considers conditional distributions with labels associated with the data. The CVAE generates latent variables from data and labels corresponding to data and reconstructs data from latent variables and labels. The CVAE also predicts labels as needed, which allows for training and data generation without labels.

Although the VAE and β-VAE consider that latent variables follow one continuous distribution, the CVAE [9] allows learning labels associated with data as a discrete latent variable. CVAE predicts latent variables from data and labels and reconstructs data from latent variables and labels. It can also be trained without labels by predicting labels if necessary.

The CVAE performs conditional generation by using category labels externally. Joint-VAE [10], on the other hand, successfully obtains general categorical factors by introducing discrete latent variables and their generating distributions. However, the correspondence between discrete latent variables and the true category is undefined, and this is a problem when generating data for a particular category.

Therefore, we propose a learning framework for joint continuous and discrete latent variables with supervised categorical labels. In this case, we suppose that discrete latent variables follow a label-specific distribution. This framework allows for conditional generations with the desired category label via trained models. This framework also allows conditional generation with labels without any classifier since the encoder can predict the labels.

In this paper, we introduce the learning method for the Joint-VAE based model with supervised categorical labels. Through experiments we show that this method can be used to generate images conditioned by categories. We then examine the effect of supervised learning on class-confusing data.

2 Related work

2.1 Vanilla VAE

Let X be a data set consisting of data $\boldsymbol{x}$ sampled from distribution $p(\boldsymbol{x})$. Vanilla VAE [3] acquires latent variables $\boldsymbol{z}$ for $\boldsymbol{x}$ by maximizing the training objective:

$$\begin{aligned}\mathcal{L}_{\mathrm{VAE}}(\theta,\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] \\ - D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})),\end{aligned} \tag{1}$$

where $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ and $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ are an encoder and decoder of a VAE with learnable parameters θ and ϕ, respectively. The first term of the objective is reconstruction error between $\boldsymbol{x}$ and generated data from $\boldsymbol{z}$, and the second term is the Kullback–Leibler (KL) divergence between the prior and the encoder approximated by the posterior distribution. Factor disentanglement is achieved by assuming the distribution $p(\boldsymbol{z})$ to be the standard normal distribution $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, which is uncorrelated in each dimension.

To calculate this objective function, we need to use the "reparameterization trick" to sample z from q. The encoder represents the normal distribution by inferring the mean $\boldsymbol{\mu}(\boldsymbol{x})$ and standard deviation $\boldsymbol{\sigma}(\boldsymbol{x})$ instead of $\boldsymbol{z}$ directly from input data:

$$q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{x}), \mathrm{diag}(\boldsymbol{\sigma}(\boldsymbol{x}))). \tag{2}$$

Finally, the latent variables $\boldsymbol{z}$ are sampled by using inferred $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, and random variable $\epsilon \sim \mathcal{N}(0, 1)$:

$$z_i = \mu_i + \sigma_i \epsilon. \tag{3}$$

In the case where the encoder outputs $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, it is straightforward to compute the KL divergence between $\mathcal{N}(\boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}))$ and $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$:

$$\begin{aligned} &D_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}))\|\mathcal{N}(\boldsymbol{0}, \boldsymbol{I})) \\ &= \frac{1}{2}\sum_i \left(1 + \log \sigma_i^2 - \mu_i^2 - \sigma_i^2\right). \end{aligned} \tag{4}$$

2.2 Conditional VAE

The CVAE [9] focuses on conditional data generation using semi-supervised learning. In the CVAE, the encoder approximates the generative distribution of the latent variables when data and labels are given, and the decoder approximates the generative distribution of the data when latent variables and labels are given. If necessary, semi-supervised learning can be undertaken by inferring labels from the data.

Let y be a label associated with data $\boldsymbol{x}$, then the objective of CVAE is defined as

$$\begin{aligned} \mathcal{L}_{\mathrm{CVAE}}(\theta, \phi) = &\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}, y)}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}, y)] \\ &- D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, y)\|p(\boldsymbol{z})). \end{aligned} \tag{5}$$

The CVAE requires labels in both the encoder and decoder. Therefore, we need to predict labels of unlabeled data using a classifier.

2.3 β-VAE

β-VAE [6] is a modification of vanilla VAE that adds a hyperparameter β to penalize the KL divergence term:

$$\begin{aligned} \mathcal{L}_{\beta\text{-VAE}}(\theta, \phi) = &\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] \\ &- \beta D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})). \end{aligned} \tag{6}$$

A larger $\beta > 1$ improves the disentanglement of the latent factor, but the reconstruction error will increase. Some learning techniques have been developed to enhance disentanglement without increasing reconstruction error. Burgess et al. [7] proposed gradually increasing the upper bound of the KL divergence term. Let the bound be C_z, and the new objective is defined as

$$\begin{aligned} \mathcal{L}(\theta, \phi) = &\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})] \\ &- \beta|D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})) - C_z|, \end{aligned} \tag{7}$$

where γ is a large positive value that forces the KL divergence term to be near the bound C_z. By gradually increasing C_z from a small value, the constraint of the KL divergence term is weakened, and it is expected that the reconstruction error will converge.

2.4 Joint-VAE

Joint-VAE [10] considers the case where latent variables consist of continuous and discrete values. Discrete latent variables are useful for representing categories, such as the types of characters in handwriting. In Joint-VAE, discrete latent variables that follow a discrete distribution can be acquired without labels.

Assuming that continuous and discrete values are generated from different distributions, the objective function is defined as

$$\begin{aligned} \mathcal{L}_{\text{Joint-VAE}}(\theta, \phi) = &\mathbb{E}_{q_\phi(\boldsymbol{z}, \boldsymbol{c}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{c})] \\ &- \beta|D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})) - C_z| \\ &- \gamma|D_{\mathrm{KL}}(q_\phi(\boldsymbol{c}|\boldsymbol{x})\|p(\boldsymbol{c})) - C_c|, \end{aligned} \tag{8}$$

where $\boldsymbol{z}$ are the continuous latent variables, and $\boldsymbol{c}$ are the discrete latent variables with the same number of dimensions as the number of categories. The prior distribution $p(\boldsymbol{c})$ is assumed to be a discrete uniform distribution. In this case, the KL divergence term can be calculated as

$$D_{\mathrm{KL}}(q_\phi(\boldsymbol{c}|\boldsymbol{x})\|p(\boldsymbol{c})) = \sum_{i=0}^{n} c_i \log c_i + \log n \tag{9}$$

where n is the number of categories. To reparameterize discrete variables, the Gumbel-softmax reparameterization trick [11] is used.

3 Proposed method

Joint-VAE can acquire categorical latent variables, but the correspondence between discrete latent variables and true categorical labels is undefined. To generate data belonging to the desired category, it is more useful to have the discrete latent variables associated with labels.

Therefore, we propose a modification of Joint-VAE to learn categorical factors with categorical labels. Suppose the categorical label of $\boldsymbol{x}$ is y, then our objective is defined as

$$\mathcal{L}_{\text{Ours}}(\theta,\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z},\boldsymbol{c}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{c})] - \beta|D_{\text{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x})\|p(\boldsymbol{z})) - C_z| - \gamma|D_{\text{KL}}(q_\phi(\boldsymbol{c}|\boldsymbol{x})\|p(\boldsymbol{c}|y)) - C_c|. \quad (10)$$

The difference from the objective of Joint-VAE is that the prior distribution $p(\boldsymbol{c})$ for $\boldsymbol{c}$ is a categorical specific distribution $p(\boldsymbol{c}|y)$. Note that Joint-VAE considers general discrete variables, but we consider discrete variables to represent only one categorical meaning like a class of data.

In this form, the KL divergence term for the supervised categorical distribution can be computed by cross-entropy:

$$D_{\text{KL}}(q_\phi(\boldsymbol{c}|\boldsymbol{x})\|p(\boldsymbol{c},y)) = -\log c_y, \quad (11)$$

where $\boldsymbol{c}$ has the same dimension as the number of classes, and y is a category number.

4 Experiments

4.1 Architecture

We employ a simple encoder-decoder model for all experiments. The encoder consists of 3 convolution modules that have 32 filters, each 4x4 in shape. Convolved feature maps are flattened and inputted into linear transformation layers. The features compressed in 256 dimensions are linearly transformed into three latent vectors: mean, log variance, and categorical variable. As noted above, the mean and log variance vectors are used to sample z by using the "reparameterization trick." Categorical variables reparameterize into one-hot-like vectors that represent which category the input data belongs to.

The decoder is simply a reversed encoder structure. All convolution modules are replaced with transposed convolution modules. The decoder outputs images regularized by a sigmoid function.

The ReLU activation function and batch normalization follow after all convolution modules and linear modules of the encoder and decoder except the last module. A schematic diagram of this model is provided in Figure1.

4.2 Implementation details

We use the PyTorch[1] framework for all implementations of model training and evaluation. Details concerning the parameters for training are as follows. We set β and γ to 30, and the maximum value of C_z to 5, which gradually increases from 0 over 25k iterations. The maximum value of C_c is set to 5 for training without labels, but 0 for training with labels to provide a strong constraint. With a batch size of 64, we repeated training for 100 epochs. Adam [12] is used as the optimizer, with the learning rate set to 10^{-3} and the other parameters maintained at the PyTorch default values.

[1] https://pytorch.org

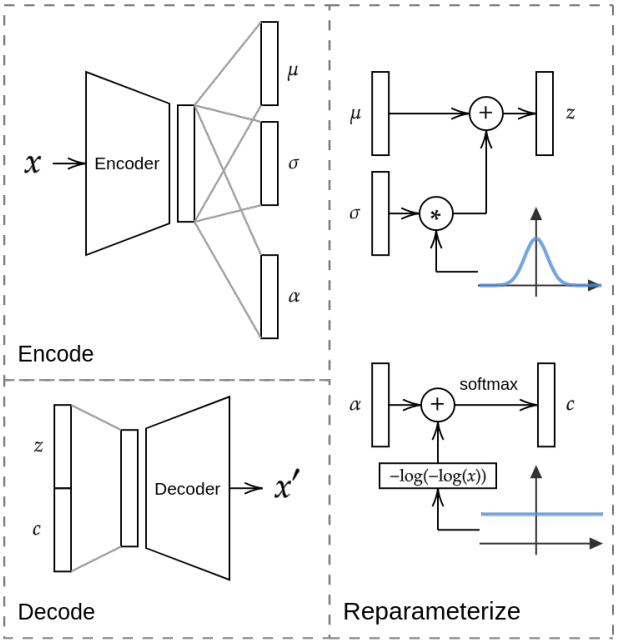


Figure 1. The architecture of the Joint-VAE based model. The encoder outputs $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, and $\boldsymbol{\alpha}$ and they are used for reparameterization. $\boldsymbol{z}$ are continuous latent variables and $\boldsymbol{c}$ are categorical latent variables, which are one-hot-like vectors.

4.3 Evaluation

We measured the reconstruction error for the test data and observed the images generated from the latent variables. We use the model trained on the MNIST dataset with and without categorical labels. Note that the Joint-VAE and our method differ only in terms of the prior distribution of categorical latent variables.

If reconstruction error is high, this suggests that variables with different semantics are entangled. This can also be observed in images generated by the decoders. So we ensure that the images generated with different conditions of categories represent different characters.

Additionally, to examine the effect of disentanglement of different characters that look similar, we experimented with a dataset of Japanese characters. The dataset[2] consists of 73 different characters, some of which are very similar. We observed the results of image generation with models trained on this dataset.

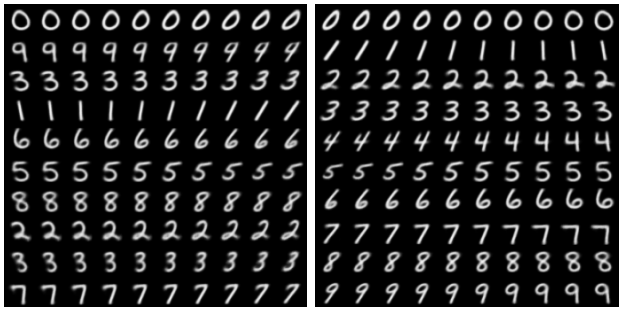
4.4 Results

The reconstruction error for each training method is shown in Table 1. We found that the reconstruction error is reduced by using our proposed method. This suggests that category labels have a positive effect on the disentanglement of variables.

[2] https://github.com/ndl-lab/hiragana_mojigazo

Table 1. Reconstruction error KL divergence of each training method. These values are calculated for the test data. Reconstruction errors are average mean squared errors of pixels.

Method	Rec. error	KL divergence loss
VAE [3]	0.0068	18.930
β-VAE [6]	0.0219	4.903
Joint-VAE [10]	0.0251	4.958
Ours	0.0243	4.986



(a) Unsupervised (Joint-VAE) (b) Supervised (Ours)

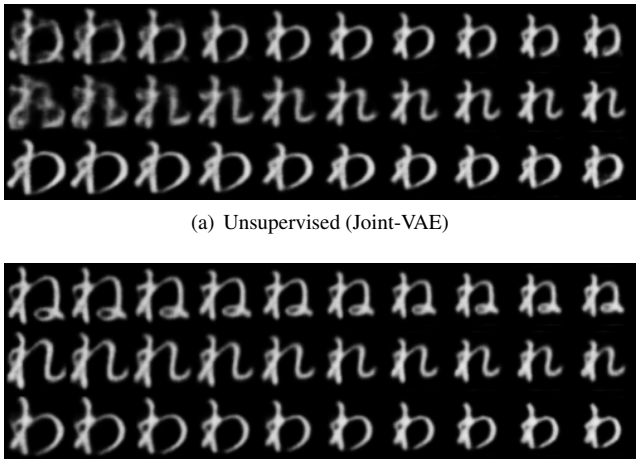
Figure 2. Images generated using VAE. The images in each row are generated by fixing a category $\boldsymbol{c}$. The images in each column have the same random contentious latent variables $\boldsymbol{z}$ and images in each row have same a categorical latent variable, which represents the class 0 to 9, from top to bottom.

Images generated from latent variables that are fixed to a specific category are shown in Figure 2. Looking at the same row of images, images generated by the Joint-VAE method have varied characters despite fixing the category. This means that the type of character depends on the continuous latent variables $\boldsymbol{z}$, not the categorical variables c. On the other hand, our method can generate unique types of characters independent of $\boldsymbol{z}$. Furthermore, in our method, the categorical latent variables correspond to one-hot representations of labels, which allows us to generate the desired character.

We show the results of image generation in Japanese characters in Figure 3. For comparison, we focus on specific discrete latent variables corresponding to three similar characters. It transpires that supervised learning can often classify characters well.

5 Conclusion

We have proposed a learning framework for Joint-VAE to learn with supervised labels. Through experiments it is shown that this method can be used to generate data that belong to any desired category. We also found that the framework can learn latent variables effectively even when similar data are included.



(a) Unsupervised (Joint-VAE)

(b) Supervised (Ours)

Figure 3. Japanese characters generated using VAE. For comparison, we focus on a specific discrete latent variables corresponding to similar characters.

It would be interesting for future research to explore how fixing the dimension of one or more variables impacts the outcomes.

Acknowledgment

This work was partially supported by the Japan Science and Technology Agency, ACT-X Grant JPMJAX1905.

References

[1] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. In *Proc. Interspeech 2019*, pages 3465–3469, 2019.

[2] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *The International Conference on Learning Representations (ICLR)*, 2018.

[3] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *The International Conference on Learning Representations (ICLR)*, 2014.

[4] Patrick Lumban Tobing, Yi-Chiao Wu, Tomoki Hayashi, Kazuhiro Kobayashi, and Tomoki Toda. Non-parallel voice conversion with cyclic variational autoencoder. *Interspeech 2019*, 2019.

[5] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems*, 2017.

[6] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *The International Conference on Learning Representations (ICLR)*, 2016.

[7] Christopher P. Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerch-

ner. Understanding disentangling in β-vae. In *NIPS Workshop on Learning Disentangled Representations*, 2017.

[8] Xiaodong Liu Jianfeng Gao Asli Celikyilmaz Lawrence Carin Hao Fu, Chunyuan Li. Cyclical annealing schedule: A simple approach to mitigating KL vanishing. In *NAACL*, 2019.

[9] Diederik P. Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in Neural Information Processing Systems*, 27:3581–3589, 2014.

[10] Emilien Dupont. Learning disentangled joint continuous and discrete representations. In *Advances in Neural Information Processing Systems*, pages 707–717, 2018.

[11] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *The International Conference on Learning Representations (ICLR)*, 2017.

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *The International Conference on Learning Representations (ICLR)*, 2015.