JADE: Joint Autoencoders for Dis-Entanglement

Ershad Banijamali[1*], **Amir-Hossein Karimi**[1*], **Alexander Wong**[2], **Ali Ghodsi**[3]
[1]School of Computer Science, University of Waterloo
[2]Systems Design Engineering, University of Waterloo
[3]Department of Statistics and Actuarial Science, University of Waterloo
{a6karimi, sbanijam, a28wong, aghodsib}@uwaterloo.ca

Abstract

The problem of feature disentanglement has been explored in the literature, for the purpose of image and video processing and text analysis. State-of-the-art methods for disentangling feature representations rely on the presence of many labeled samples. In this work, we present a novel method for disentangling factors of variation in data-scarce regimes. Specifically, we explore the application of feature disentangling for the problem of supervised classification in a setting where few labeled samples exist, and there are no unlabeled samples for use in unsupervised training. Instead, a similar datasets exists which shares at least one direction of variation with the sample-constrained datasets. We train our model end-to-end using the framework of variational autoencoders and are able to experimentally demonstrate that using an auxiliary dataset with similar variation factors contribute positively to classification performance, yielding competitive results with the state-of-the-art in unsupervised learning.

1 Introduction

In machine learning, samples in a dataset originate via complicated processes driven by a number of underlying factors. Individual factors lead to independent directions of variations in the observed samples, while the accumulation of factors give rise to the rich structure characteristic of these datasets. The underlying factors often interact in complicated and unpredictable ways, and appear tightly *entangled* in the raw data. Being able to tease apart the effect of underlying factors is a fundamental challenge in understanding these datasets.

For instance, a dataset containing images of natural scenery may be subject to variation in lighting conditions, camera elevation, and the appearance of the scene itself. Controlling and restraining variation at data acquisition time is difficult, and limits the number of acceptable samples in the dataset. On the other hand, capturing annotations for every direction of variation is time-consuming and infeasible. Therefore, designing methods that automatically learn to separate out underlying factors (known and unknown) is relevant for many applications in machine learning.

One area that has enjoyed tremendous success for separating factors of variation is supervised learning. The representations learned here aim to satisfy a specific task that is driven by the explicit labels in the dataset. Therefore, these representations are invariant to factors of variation that are uninformative for solving the task at hand. For example, when identifying individuals in a school yearbook, the identity of the person is paramount compared to their facial expression. Hence, a simple method that simply discards the irrelevant variation in expression will perform really well. Learning invariant representations, however, require many samples and comes at the cost of needing to train a new model for a closely related task that depends on an alternative direction of variation.

[*]equal contribution

31st Conference on Neural Information Processing Systems (NIPS 2017), Workshop on Learning Disentangled Representations: from Perception to Control, Long Beach, CA, USA.

It would seem reasonable then to desire a strategy that captures all directions of variation in a single model in a *disentangled* manner allowing one to infer all factors for a given sample in the absence of labels for each factor.

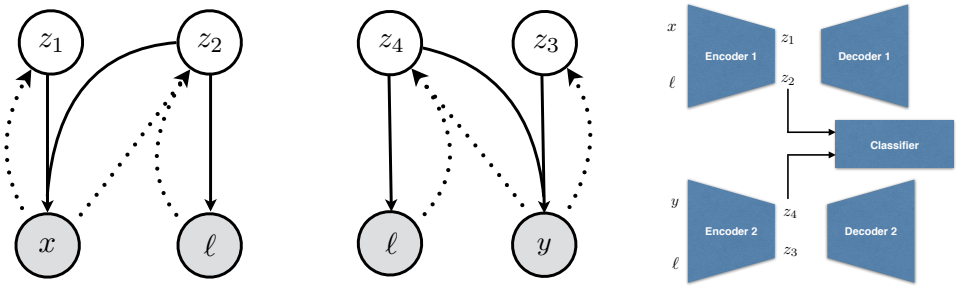
Current state-of-the-art strategies for disentangling factors of variation mostly fall victim to the challenges in deep learning and rely on the presence of abundant data samples. In [5], the authors were able to accurately separate out lighting, pose, and shape while sampling seemingly unlimitedly from an auxiliary generative model that creates samples with different variations. The results presented in [9, 7] also build upon datasets containing often hundreds of thousands of samples. Whereas [3, 11] use very few samples in their training process, these methods are semi-supervised and have access to unlabeled samples from the same dataset following the same statistical distribution.

In this work, we explore classification in a data-scarce scenario where not only are there few labeled samples available, there are also no unlabeled samples from which one could perform semi-supervised training. These situations commonly arise in medical imaging datasets, e.g., pancreatic cancer MRI images are scarce whereas breast cancer MRI images are abundant ([2] and references therein). In such a situation, we ask whether one can employ a secondary dataset, with many samples, similar content, but different style, to improve the performance of a benchmark classification model. What remains to be demonstrated is how to learn good intermediate representations that can be shared across tasks and use the disentanglement process of the secondary dataset to effectively disentangle the factors of variation in the primary dataset of interest. Essentially we are entangling together the feature disentangling of two similar datasets. This is the focus of the work below.

2 Model Description

In this work, we consider a situation where we are given a labeled dataset, X, with limited number of points. We denote the label variable by ℓ. We also have access to another dataset Y with a larger number of points that share the same categories as Y. However, the underlying distribution of the datasets are different. Let us denote the distribution for X and Y by $p(x)$ and $p(y)$, respectively. Suppose that our goal is to classify unseen data points that come from $p(x)$, i.e. to maximize $p(\ell|x)$. Building a classifier that simply uses X can lead to low accuracy and overfitting, due to its small size. Therefore we want to leverage the information of Y about the label variable and build a model that can classify the points from $p(x)$ with higher accuracy.

Our approach to address this problem is to disentangle the features in X and Y that contribute in predicting the label variable (i.e., content) from the features that contribute to the style of X and Y. Consider the graphical model in Fig. 1a. We assume there are two pairs of latent variables that describe each of x and y. Based on this figure, suppose that z_1 and z_2 generate samples in dataset X and z_3 and z_4 generated samples in dataset Y. If we assume that z_2 and z_4 are the latent variables that carry all the information about the label variable ℓ then $p(\ell|z_2) = p(\ell|z_4)$. Considering the same prior distributions over z_2 and z_4, i.e. $\mathcal{N}(0, I)$, we can guarantee the disentanglement of latent features by asserting that $p(z_2|\ell) = p(z_4|\ell)$. However, these posteriors are intractable. To approximate them we use the framework of variational inference where $p(z_2|\ell)$ and $p(z_4|\ell)$ are approximated by $q(z_2|x, \ell)$ and $q(z_4|y, \ell)$, respectively. Therefore, by matching these approximating distribution, we guarantee that only z_2 and z_4 carry information regarding the label ℓ (i.e., content) and therefore are disentangled from z_1 and z_3 respectively which represent style.



(a) (b)

Figure 1: (a) Graphical models of the method. (b) Network structure of the method

All the conditional distributions on the graphical models in Fig. 1a are parameterized by the neural networks depicted in Fig. 1b. The joint model here builds on earlier work in [10] where an autoencoder and a discriminator were trained in the framework of contractive discriminative analysis for semi-supervised learning. Here, we use the variational autoencoding [4] approach to jointly train two networks that simultaneously extract shared discriminative features present in the primary and secondary datasets. This architecture is reminiscent of Domain Separation Networks [1]. The proposed JADE model, however, focuses on a shared classifier for improved classification and joint disentanglement instead of a shared encoder and decoder.

The variational lower bound on the joint distribution of the observations is:

$$\begin{aligned}
\log p(x,\ell) \geq \mathcal{L}(x,\ell) = \mathbb{E}_{\substack{q(z_1|x) \\ q(z_2|x,\ell)}} \big[\log p(x|z_1,z_2)\big] + \mathbb{E}_{q(z_2|x,\ell)}\big[\log p(\ell|z_2)\big] \\
-\mathrm{KL}\big(q(z_1|x) \parallel p(z_1)\big) - \mathrm{KL}\big(q(z_2|x,\ell) \parallel p(z_2)\big)\big) \\
\log p(y,\ell) \geq \mathcal{L}(y,\ell) = \mathbb{E}_{\substack{q(z_3|y) \\ q(z_4|y\ell)}} \big[\log p(x|z_3,z_4)\big] + \mathbb{E}_{q(z_4|y,\ell)}\big[\log p(\ell|z_4)\big] \\
-\mathrm{KL}\big(q(z_3|x) \parallel p(z_3)\big) - \mathrm{KL}\big(q(z_4|x,\ell) \parallel p(z_4)\big)\big)
\end{aligned} \tag{1}$$

We would like to maximize the sum over the above lower bounds. The approximating distributions are from exponential family (Gaussian) and to match them we assume that for the samples that are from the same class in the two datasets, we want to minimize $\mathrm{KL}\big(q(z_2|x,\ell) \parallel q(z_4|y,\ell)\big)$. Given this condition, the overall objective of the model is:

$$\max_{\Theta} \mathcal{L}(x,\ell) + \mathcal{L}(y,\ell) - \mathrm{KL}\big(q(z_2|x,\ell) \parallel q(z_4|y,\ell)\big) \tag{2}$$

where Θ represents the entire parameter set of neural networks.

3 Experiments

Datasets: Our framework addresses the problem of performing supervised classification in data-scarce regimes where there exists a secondary dataset that has at least one direction of variation in common with the primary sample-constrained dataset. In our experiments we emulate this scenario with commonly used datasets such as MNIST [6] and SVHN [8]. Because MNIST is relatively easier to learn, even with very few samples, we select SVHN as the sample-constrained primary dataset that is difficult to learn, and use the entirety of MNIST as the secondary dataset. These datasets differ in appearance and style: whereas MNIST is gray-scale and comes in 28×28 pixel images, SVHN has three color channels and comes in 32×32 pixel images. However, both datasets represent the same content (i.e., digit values) across different styles. This similarity in content of both datasets is what makes MNIST a good secondary dataset to boost SVHN's supervised classification performance.

Model Comparison: To evaluate the performance of our framework, we first develop a benchmark for supervised classification of SVHN. Here, we choose a relatively powerful convolutional neural network (CNN) architecture combined with a multi-layer perceptron (MLP) as the supervised classification model. The CNN architecture comprises of 4 layers of 3×3 spatial convolutions ($\{64, 96, 64, 8\}$ filters respectively) followed by ReLU and interspersed with 3 layers of $2\times$ max-pooling. The MLP contains 3 blocks of 500-dimensional fully connected layers, followed by ReLU and Dropout ($p = 0.5$) layers [12]. A 10-dimensional bottleneck layer was placed in between the CNN and the MLP to encourage only important features from being retained. A final softmax layer is present at the end of the network for 10-way classification. The loss for this model is measured using categorical cross-entropy. This architecture is referred to as *single classifier* (i.e., benchmark).

A simple extension of above setup is a model that jointly trains SVHN and MNIST on a shared MLP classifiers using features extracted from separate CNN feature extractors, one per dataset. The CNN used for SVHN and the MLP follow the same architecture as the benchmark above. The CNN architecture for MNIST comprises of 3 layers of 3×3 spatial convolutions ($\{32, 32, 16\}$ filters respectively) followed by ReLU and interspersed with 3 layers of $2\times$ max-pooling. A 10-dimensional bottleneck layer was placed in between the CNN for MNIST and the shared MLP to capture the latent features of MNIST. Feature-extracted samples from both datasets are fed into the shared MLP in alternation and trained jointly. The loss of the system is the sum of the categorical cross-entropy losses for both datasets on the shared classifier. This setup is called *paired classifier*.

Table 1: Classification error rates for SVHN on limited data: 100 samples per each class. Error rates calculated using the entirety of SVHN's test set. Results of our experiments are averaged over 3 runs. We observe improved SVHN classification performance without sacrificing near state-of-the-art performance on MNIST.

Method	SVHN (1000 samples)	MNIST (45K samples)
VAE (M1+M2) [3]	36.02±0.10	-
Siddharth et al. [11]	28.71±2.38	-
Single Classifier (benchmark)	32.31±1.56	-
Paired Classifier	30.17±2.77	0.82±0.05
JADE (proposed)	29.08±0.92	0.72±0.03

Finally, the proposed model (outlined in Fig. 1b) extends upon the previous two methods by adding a decoder network to reconstruct the 10-dimensional latent representations from each of the CNN feature extractors. To encourage disentanglement of features in the latent space, and to perform factor separation in a way that the MLP classifier is only given content-related features (i.e., digit values), we increase the size of the latent spaces from 10 to 20 dimensions. However, only 10 of the latent dimensions resulting from each CNN are passed into the shared MLP, essentially keeping consistent with the previous method in terms of classifier capacity. All 20 latent dimensions are used to reconstruct the inputs via a decoder that identically mirrors the corresponding CNN ($2\times$ up-sampling layers used in place of $2\times$ max-pooling). Losses are defined in Section 2. Due to the autoencoding structure of this model, we refer to it as *JADE: Joint Autoencoders for Dis-Entanglement*.

Discussion: The results of our experiments have been presented in Table 1. Here we compare the results of the single classifier (i.e., benchmark model), paired classifier, and proposed model (JADE) alongside those from Kingma et al. [3] and Siddharth et al. [11]. It is worth pointing out that the former 3 models are trained only on 1000 labeled sample from SVHN, whereas the cited models use the remainder of the SVHN training dataset in an unsupervised fashion. We, on the other hand, use all of the MNIST dataset to train the paired classifier in JADE.

These results demonstrate that when dealing with sample-constrained regimes without unlabeled samples, one can use a similar dataset with at least one shared direction of variation to improve classification performance. This can be seen when comparing the performance of a single classifier (32.31±01.56) with that of a paired classifier (30.17±02.77). On top of this, we see that the JADE model which learns to jointly disentangle SVHN and MNIST features performs even better than the former methods, sitting at 29.08±00.92. This is in line with our hypothesis that only the directions of variation shared between MNIST and SVHN (i.e., content) will contribute positively to classification performance on SVHN, and other factors of variation should be disentangled.

We hypothesize that actively attempting to disentangle variation factors (i.e., in JADE) is better than allowing the network to attempt to discard uninformative factors (i.e., paired classifier) given the sample-constrained regime. To assert that the JADE setup is indeed disentangling variation factors, we conduct the following simple experiment: observe the variation in latent space values as different types of samples are passed into the network. In Fig. 2a, we have shown how latent activations change when the SVHN CNN is fed with 500 samples from the same class (i.e., same content but varying style). These activations are shown for the 20 latent parameters (of which only 10 are passed into the MLP classifier, and all used for reconstruction) across 10 classes of digits in MNIST. We observe that in this setup where content is fixed, the normalized variance of the latent variables that

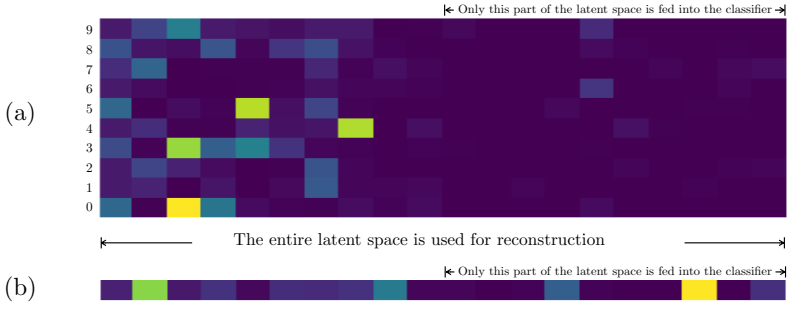


Figure 2: **(a)** variance normalized activations of latent space parameters, averaged over 500 random samples from each of 10 classes in SVHN; when content is fixed, the part of the latent space that feeds into the classifier exhibits weaker variance in activations compared to the part of the latent space that seemingly represents style over the 500 samples. **(b)** variance normalized activations of latent space parameters for 2500 random samples from SVHN spanning various style and content; all 20 latent space parameters fire for random splits of the data.

are fed into the MLP classifier is much lower than the variance of latent variables that are solely used for reconstruction. In Fig. 2b, we observe an interesting and complementary phenomena when we pass in 2500 randomly selected test samples into the SVHN CNN. Here, both the style and the content vary between input samples, and we observe that all 20 latent parameters are active given the varying input. These observations suggest that JADE is able to successfully disentanglement content and style in low-data SVHN using the help of MNIST as an auxiliary similar dataset.

4 Conclusion and Future Work

In this work, we explore the application of feature disentangling for the problem of supervised classification in a setting where few labeled samples exist, and there are no unlabeled samples for use in unsupervised training. Instead, a similar datasets exists which shares at least one direction of variation with the sample-constrained datasets. We train our model end-to-end using the framework of variational autoencoders and experimentally demonstrated that using a secondary dataset with similar content to SVHN leads to improvements in supervised classification performance.

Given the autoencoding structure of the proposed framework, a reasonable next step is to explore using an ensemble of auxiliary datasets, say one for content and another for style, to augment not only the classification power of the system, but also its reconstruction and generation ability. Currently, reconstruction quality is lacking as samples are being generated using the limited samples. Finally, an exciting extension of the JADE framework is cross-task or cross-modality data synthesis, e.g., learning a joint representation that captures high-level concepts for all modalities of the same object allows for bi-directional generation of missing modalities from the remaining modalities [13].

References

[1] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems*, pages 343–351, 2016.

[2] H. Greenspan, B. van Ginneken, and R. M. Summers. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5):1153–1159, 2016.

[3] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pages 3581–3589, 2014.

[4] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[5] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems*, pages 2539–2547, 2015.

[6] Y. LeCun, C. Cortes, and C. J. Burges. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

[7] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprechmann, and Y. LeCun. Disentangling factors of variation in deep representation using adversarial training. In *Advances in Neural Information Processing Systems*, pages 5040–5048, 2016.

[8] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 5, 2011.

[9] S. Reed, K. Sohn, Y. Zhang, and H. Lee. Learning to disentangle factors of variation with manifold interaction. In *International Conference on Machine Learning*, pages 1431–1439, 2014.

[10] S. Rifai, Y. Bengio, A. Courville, P. Vincent, and M. Mirza. Disentangling factors of variation for facial expression recognition. *Computer Vision–ECCV 2012*, pages 808–822, 2012.

[11] N. Siddharth, B. Paige, V. de Meent, A. Desmaison, F. Wood, N. D. Goodman, P. Kohli, P. H. Torr, et al. Learning disentangled representations with semi-supervised deep generative models. *arXiv preprint arXiv:1706.00400*, 2017.

[12] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958, 2014.

[13] M. Suzuki, K. Nakayama, and Y. Matsuo. Joint multimodal learning with deep generative models. *arXiv preprint arXiv:1611.01891*, 2016.