Memory Augmented Neural Network with Gaussian Embeddings for One-Shot Learning

Hanna Tseran[1] **Tatsuya Harada**[1,2]
[1]Graduate School of Information Science and Technology, University of Tokyo, Tokyo
[2]AIP, RIKEN, Tokyo
{hanna,harada}@mi.t.u-tokyo.ac.jp

Abstract

Memory augmented neural networks that use pointwise embeddings have been successfully applied to one-shot learning, however, Gaussian embeddings are more versatile and provide an opportunity to build models capturing latent structure. As a step towards combining both, we construct a memory augmented network with Gaussian embeddings. We provide results of one-shot classification on Omniglot and LFW-a datasets, and since the resulting model is generative, image reconstruction results on Omniglot. Additionally, we explain how to learn more classes in one-shot using memory augmented neural networks with a method that does not depend on the type of embeddings.

1 Introduction

One-shot learning refers to learning from a single example. The problem has been addressed in a number of recent studies [9, 5, 21, 7] with several generative models being introduced [15, 13]. We build on the idea of applying memory augmented neural networks to one-shot classification [5, 17]. Similarly, we use a meta-learning strategy that usually refers to learning at two time scales. In our case network acts as a slow learner and memory as a rapid one.

The main goal of our work is to introduce a memory augmented model with embeddings that can better capture latent representations and would allow for structure in the latent space. For this we use density estimations instead of pointwise embeddings. In order to get Gaussian embeddings we employ variational autoencoder[6], thus presented generative model can also be described as a memory augmented VAE. Inspired by [12], we use disentangled representations to separate style and content. We treat content distribution parameters as embeddings and store them in the memory. Additionally, we show how to parallelize training process to learn more classes in one-shot using memory augmented networks such as [5] and present results for 400 Omniglot and 80 LFW-a classes. To sum up, our contributions are twofold: 1. We introduce a structured generative model based on a memory augmented network that uses Gaussian embeddings instead of pointwise; 2. We show how to learn more classes in one-shot with the memory augmented neural networks.

2 Related work

One-shot learning is a problem of learning from a single example. It has received a lot of attention recently [9, 5, 21] and we build our work on the application of memory augmented neural networks to this problem [17, 5], mainly on [5]. Memory augmented networks were popularized by works such as Neural Turing Machines [2]. The memory module in them can be addressed using either content-based addressing, location-based addressing or a combination of both [17]. Our model uses only content-based addressing. In the memory we store more versatile Gaussian embeddings instead

Second workshop on Bayesian Deep Learning (NIPS 2017), Long Beach, CA, USA.

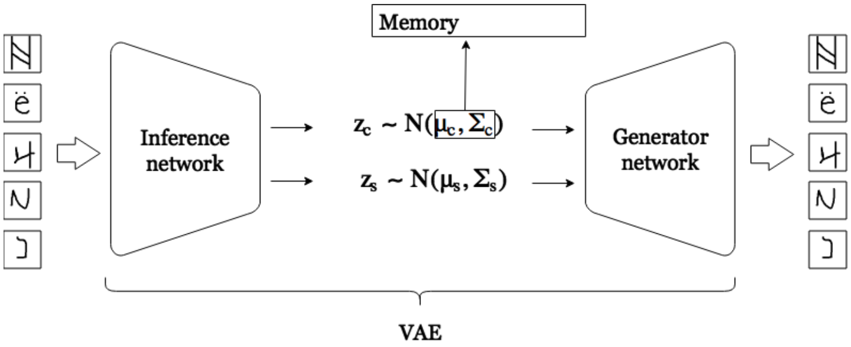

Figure 1: Model overview. We use a VAE with two independent latent variables z_c and z_s, where z_c is responsible for the content and z_s for the style. Only the distribution parameters for z_c are stored in the memory.

of pointwise embeddings. We use disentangled representations in an attempt to introduce simple structure in the latent space. Application of disentangled representations to one-shot learning has been studied before, e.g. in [3].

Meta-learning or learning to learn usually refers to learning at two time scales. Rapid learner learns the specific details while slow learner learns more general information [17]. Meta-learning has been applied to one-shot learning in several works [17, 21, 5]. In our case network acts as a slow learner and memory as a rapid one.

Additionally, our model is generative. Several generative models have already been introduced for one-shot learning, e.g. [15], including a recent memory augmented generative model [1] that operates in a different way from ours.

3 Method

3.1 Description

The model can be described as a memory augmented neural network with Gaussian embeddings or as a memory augmented VAE [6] with disentangled representations. Overview of the model is provided in the figure 1.

In the case of a VAE one latent variable is usually used. Recently a lot of work has been done in order to allow for more complicated structures in the latent space. Inspired by [12, 11] we use two independent latent variables z_c and z_s, where z_c is responsible for the content and z_s for the style. Latent variables are assumed to be independent and distributed according to the normal distribution with a diagonal covariance matrix.

$$z_c, z_s \sim p(z_c, z_s) = \mathcal{N}(0, I)\mathcal{N}(0, I)$$

$$z_c, z_s|x \sim q_\phi(z_c, z_s|x) = q_\phi(z_c|x)q_\phi(z_s|x) \quad x|z_c, z_s \sim p_\theta(x|z_c, z_s)$$

Here q_ϕ is the posterior outputted by an inference network with parameters ϕ, and p_θ is the likelihood outputted by a generator network with parameters θ.

Only z_c is used to classify samples. Its distribution parameters are treated as class embeddings and stored in the memory. When a new sample arrives, label of the most similar element in the memory is returned. If the answer is correct, stored parameters are updated, otherwise a new element is created. The memory module is not differentiable and uses only content-based addressing.

Symmetric KL divergence mapped to $[0, 1]$, where 1 corresponds to the most similar, is used as a similarity measure:

$$\text{similarity}(q_\phi(z_{c1}), q_\phi(z_{c2})) = exp\{-(D_{KL}(q_\phi(z_{c1}) \parallel q_\phi(z_{c2})) + D_{KL}(q_\phi(z_{c2}) \parallel q_\phi(z_{c1})))\} \quad (1)$$

Cluster update is done according to [16] that was proved to minimize KL divergence between a new distribution and the mixture.

$$\mu_{new} = w_1\mu_1 + w_2\mu_2$$

Model	5-way		20-way	
	1-shot	5-shot	1-shot	5-shot
Prototypical Networks [20]	98.8%	99.7%	96.0%	98.9%
Matching Network [21]	98.1%	98.9%	93.8%	98.5%
CNN with Memory Module [5]	98.4%	99.6%	95.0%	98.6%
Ours ($n_{train} = n_{test}$)	97.4%	98.9%	92.3%	98.4%
Ours ($n_{train} < n_{test}$)	-	-	92.6%	97.9%

Table 1: Omniglot results. Here $n_{train} = n_{test}$ stands for training and testing on the same number of classes with the batch size of 16. In the case of $n_{train} < n_{test}$ the model was trained for a 5-way task with the batch sizes of 16 and tested on 20 classes.

$$\begin{aligned}\Sigma_{new} &= w_1\Sigma_1 + w_2\Sigma_2 + w_1 w_2(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T = \\ &= [\text{we approximate the last matrix with a diagonal one}] = \\ &= w_1\Sigma_1 + w_2\Sigma_2 + w_1 w_2 diag(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T\end{aligned}$$

Here w_1, w_2 are mixture components weights that we approximate as $w_1 = \frac{k-1}{k}, w_2 = \frac{1}{k}$, where k is a shot number.

The memory module operation is based on [5], except the fact that we use only content based addressing. Similarly, the triplet loss [18] is used to enforce embeddings to be amenable to one-shot learning. Using similarity definition from the equation 1, we define the triplet loss in our case as:

$$\text{triplet loss} = \max(\text{similarity}(q, q_{neg}) + (1 - \text{similarity}(q, q_{pos})) + \alpha, \alpha) - \alpha,$$

where $\alpha > 0$ is a small constant, we used $\alpha = 0.1$.

Total model loss consists of the sum of the triplet loss and the VAE loss [6] that has a latent loss and a generative loss parts. Additionally, to force the content description to be stored in z_c we add a generative loss for restoring an image with z_c from a different sample of the same class.

$$\text{total loss} = c_1\text{triplet loss} + c_2\text{latent loss} + c_3(\text{generative loss}_{z_c 1} + \text{generative loss}_{z_c 2}),$$

where c_1, c_2, c_3 are loss weights.

3.2 Learning more classes in one-shot

One-shot learning is often formalized as an N-way K-shot learning, where N is the number of classes to learn and K is the number of times a sample from one class is presented. Results are usually reported for (5 to 20)-way learning for Omniglot and for 5-way for miniImageNet, e.g. [5, 21, 17, 7]. [21, 5] use the same number of classes is used for training and testing. However, when a network is augmented with memory and trained using mini-batch gradient descent with embeddings for all batches stored in the same memory similarly to [5], it is effectively trained to do one-shot learning for (number of batches × number of classes in a batch) classes because all of them must be distinguished between one another. This observation allows to parallelize the training process. It is model-independent as long as a model is augmented with external memory and embeddings for all batches are stored in the same memory.

4 Experiments

We tested our model on Omniglot, LFW-a and miniImageNet [21] datasets. Results on Omniglot were slightly lower than the state-of-the-art, while miniImageNet results were poor, suggesting that datasets with high interclass variability are not suitable for the method now, so they are not provided. Instead we tested our model on the LFW-a dataset resizing images to be the same size as in miniImangeNet.

4.1 Omniglot

Omniglot dataset was introduced in [9]. It consists of 1623 characters from 50 alphabets, each hand-drawn by 20 people. We preprocessed it in the same way as in [21], augmenting it with random rotations by multiples of 90 degrees. 1200 classes are used for training and remaining classes for evaluation. Results are presented in tables 1 and 2. Examples of the images reconstructed during 100-way test are given in the figure 2.

Model	100-way		400-way	
	1-shot	5-shot	1-shot	5-shot
Ours	82.2%	92.8%	71.4%	87.1%

Table 2: Omniglot results. The model was trained for a 5-way task with a batch size of 20 and for a 20-way task with a batch size of 20 and tested on 100 and 400 classes respectively.

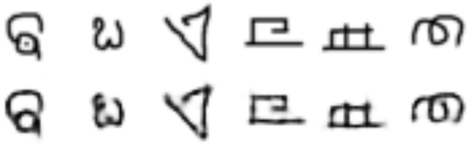

Figure 2: 100-way Omniglot test. First row: original images. Second row: reconstructed images.

Model	5-way		20-way		80-way	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Ours	64.4%	84.0%	39.9%	63.4%	22.7%	44.0%

Table 3: LFW-a results.The model was trained for a 5-way task with a batch size of 16 and tested on 5, 20 and 80 classes respectively.

4.2 LFW-a

Labeled Faces in the Wild-a (LFW-a) is a database of grayscale labeled face images consisting of more than 13,000 images of faces [22]. For the experiment were chosen only people who have 3 or more images, and all images were vertically flipped to create twice more samples and resized to 84×84 pixels to be the same size as images in miniImageNet. Images of 343 people were used for training and 80 for testing. Results are shown in the table 3.

5 Discussion

In general, structured knowledge representation and external memory can be seen as being vital to the construction of artificial intelligent agents [8]. We have described a generative model combining these properties that can be interpreted either as a memory augmented neural network that uses Gaussian embeddings instead of pointwise or as a memory augmented VAE with disentangled representations. Also, we have shown how to parallelize training of a memory augmented network.

Our method employs a simple structure in the latent space based on the disentanglement of content and style and shows competitive, though slightly lower than state-of-the-art, performance on one-shot learning tasks with datasets where such structure makes sense, such as Omniglot and LFW-a. Different ways of introducing structure in VAE latent space have been studied recently, e.g. [4, 10], and we suggest that a richer structure should allow our model to capture more complicated dependencies. Additionally, usage of distributions in place of pointwise embeddings opens other interesting applications for memory augmented networks, such as continual learning through experience replay [14, 19]. We leave them as directions for the future work.

6 Acknowledgments

Hanna Tseran is supported by the MEXT Scholarship, Japan.

References

[1] J. Bornschein, A. Mnih, D. Zoran, and D. J. Rezende. Variational memory addressing in generative models. In *Advances in Neural Information Processing Systems*, pages 3921–3930, 2017.

[2] A. Graves, G. Wayne, and I. Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[3] E. Hoogeboom. Few-shot classification by learning disentangled representations. 2017.

[4] M. Johnson, D. K. Duvenaud, A. Wiltschko, R. P. Adams, and S. R. Datta. Composing graphical models with neural networks for structured representations and fast inference. In *Advances in Neural Information Processing Systems*, pages 2946–2954, 2016.

[5] Ł. Kaiser, O. Nachum, A. Roy, and S. Bengio. Learning to remember rare events. *arXiv preprint arXiv:1703.03129*, 2017.

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

[7] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015.

[8] D. Kumaran, D. Hassabis, and J. L. McClelland. What learning systems do intelligent agents need? complementary learning systems theory updated. *Trends in Cognitive Sciences*, 20(7): 512–534, 2016.

[9] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[10] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther. Auxiliary deep generative models. *arXiv preprint arXiv:1602.05473*, 2016.

[11] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[12] 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.

[13] A. Mehrotra and A. Dukkipati. Generative adversarial residual pairwise networks for one shot learning. *arXiv preprint arXiv:1703.08033*, 2017.

[14] D. C. Mocanu, M. T. Vega, E. Eaton, P. Stone, and A. Liotta. Online contrastive divergence with generative replay: Experience replay without storing data. *arXiv preprint arXiv:1610.05555*, 2016.

[15] D. Rezende, I. Danihelka, K. Gregor, D. Wierstra, et al. One-shot generalization in deep generative models. In *International Conference on Machine Learning*, pages 1521–1529, 2016.

[16] A. R. Runnalls. Kullback-leibler approach to gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems*, 43(3), 2007.

[17] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850, 2016.

[18] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[19] H. Shin, J. K. Lee, J. Kim, and J. Kim. Continual learning with deep generative replay. *arXiv preprint arXiv:1705.08690*, 2017.

[20] J. Snell, K. Swersky, and R. S. Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

[21] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, pages 3630–3638, 2016.

[22] L. Wolf, T. Hassner, and Y. Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *IEEE transactions on pattern analysis and machine intelligence*, 33(10):1978–1990, 2011.