Unsupervised Meta-Learning via Latent Space Energy-based Model of Symbol Vector Coupling

Deqian Kong* **Bo Pang*** **Ying Nian Wu**
Department of Statistics
University of California, Los Angeles
{deqiankong, bopang}@ucla.edu ywu@stat.ucla.edu

Abstract

Meta-learning aims to learn a model from a stream of tasks such that the model is able to generalize across tasks and rapidly adapt to new tasks. We propose to learn an energy-based model (EBM) in the latent space of a top-down generative model such that the EBM in the low dimensional latent space is able to be learned efficiently and adapt to each task rapidly. Furthermore, the energy term couples a continuous latent vector and a symbolic one-hot label. Such coupling formulation allows the model to be learned in an unsupervised manner when the labels are unknown. Our model is learned unsupervisedly in the meta-training phase and evaluated semi-supervisedly in the meta-test phase. We evaluate our model on widely used benchmarks for few-shot meta-learning, Omniglot, and Mini-ImageNet. Our model achieves competitive or superior performance compared to previous state-of-the-art meta-learning models.

1 Introduction

Meta-learning, or *learning how to learn* enjoys rapid progress as machine learning is maturing. It was originally proposed in the [26] to interpret human learning from experience. Human beings are not only able to learn the concept[25] from the recurrent tasks, but also learn the bias to generalize the learned concept into new scenarios.

Most few-shot meta-learning algorithms are based on supervised learning [6, 7, 8, 23, 18]. Although each task requires a small to a modest amount of labeled data, the model needs to learn from a large number of tasks in the meta-training phrase. Hence learning these models still require a considerable amount of human effort to label data. A few recent works [17] have started investigating unsupervised meta-learning, where the meta-training phrase is done in an unsupervised manner.

Recent works [20, 29] have demonstrated that EBM is highly effective in modeling high-dimensional signals like images but expensive to train since it involves MCMC sampling in the high-dimensional data space. In our model, EBM is in a low dimensional latent space and thus MCMC is efficient and mixes well.

Furthermore, the latent space EBM couples a continuous latent vector and a discrete one-hot symbolic vector. The continuous latent vector allows for convenient gradient-based sampling such as Langevin dynamics, and thus the model can be learned by maximum likelihood or its approximate variants. The one-hot symbolic vector naturally connects the generative model to a discriminative model. Due to the coupling formulation, given the inferred continuous vector, the class label of an input example can be inferred from it via a standard softmax classifier (see Equation 3). A similar model has been studied in [22] for conditional text generation and text classification, but their model is designed to solve a single fixed task. In contrast, we leverage the expressiveness and efficiency of

*Equal contribution

5th Workshop on Meta-Learning at NeurIPS 2021, held virtually.

EBM in the latent space to adapt to novel tasks rapidly after learning from a stream of tasks. Since we learn a symbol-vector coupling EBM to achieve meta-learning, we call our model as Symbol-Vector Coupling Energy-Based Model for Meta-Learning (Meta-SVEBM).

The proposed Meta-SVEBM based on a top-down generative model can be learned in an unsupervised setting when no category labels are provided. The symbol-vector coupling, the generator network, and the inference network are learned jointly by maximizing the variational lower bound of the log-likelihood. The model can also be learned in a semi-supervised setting where the category labels are provided for a subset of training examples. We leverage the flexibility of Meta-SVEBM to achieve unsupervised or semi-supervised meta-learning.

During the meta-training phase, we sample a series of tasks from an unlabeled dataset. The inference network and the top-down generation network are considered as meta-parameters and shared across tasks. The EBM prior adapts to each task. Although the one-hot symbolic vector (i.e., symbol) is summed out, the sampling of the continuous vector (i.e., vector) is aware of the symbol due to the symbol-vector coupling. Thus, the model is able to learn the category structure of each task in the unsupervised meta-training. During the meta-test phase, we adapt the model to novel tasks where a small number of labels are available. The meta-parameters are assumed to be generalizable to new tasks and thus fixed. The small EBM is updated to adapt to each task through semi-supervised learning. To test the effectiveness of the proposed Meta-SVEBM, we evaluate our model on two standard few-shot learning benchmarks with various settings. Our model achieves competitive performance on Omniglot and outperforms prior state-of-the-arts models on the challenging Mini-ImageNet dataset.

Our main contributions are as follows.

- We propose to learn the latent space EBM with a symbol-vector coupling formulation such that meta-training can be done in an unsupervised manner, while meta-test can be done in a semi-supervised manner.
- We demonstrate that our model achieves competitive or superior performance on standard benchmarks in various settings, compared to prior state-of-the-arts models.

2 Unsupervised meta-learning via Meta-SVEBM

2.1 Problem statement

As in [6], meta-learning treats each few-shot classification task $\mathcal{T}_i$ with its associating dataset $\mathcal{D}_i$ as a training example and assume that they all come from the same task distribution as $\{\mathcal{T}_i\}_{i=1}^N \sim p(\mathcal{T})$. To be specific, a supervised K-way, S-shot, Q-query classification task $\mathcal{T}_i$ with corresponding dataset $\mathcal{D}_i = \{\mathcal{S}_i, \mathcal{Q}_i\}$ can be defined as to utilize the learned knowledge from the support set $\mathcal{S}_i = \{(\mathbf{x}_{ij}^s, \mathbf{y}_{ij}^s)\}_{j=1}^{KS}$ with S data and labels per class and K total classes to correctly predict the labels of query set with Q unlabelled data per class as $\mathcal{Q}_i = \{\mathbf{x}_{ij}^q\}_{j=1}^{KQ}$.

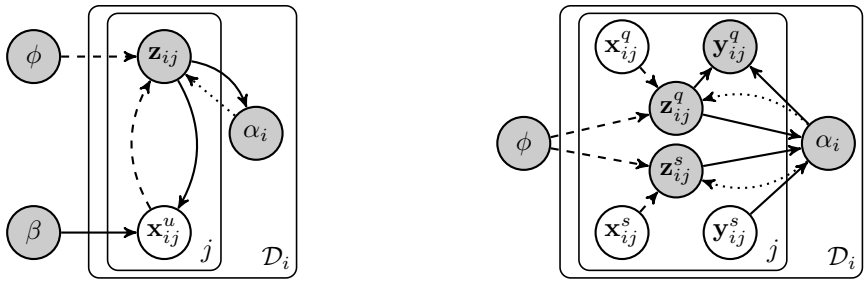
In the unsupervised meta-learning setting as suggested in [11], we only assume the task with unlabelled dataset $\mathcal{D}_i = \{\mathbf{x}_{ij}^u\}_{j=1}^U$ in the meta-training phase. The goal is to learn the meaningful meta-parameters in the meta-training stage so that they can be successfully adapted to solve a supervised M-way, S-shot classification task in the meta-test stage as mentioned above.

2.2 Model: SVEBM

We shall first describe the model in a general form. Let $\mathbf{x} \in \mathbb{R}^D$ be an observed example, $\mathbf{z} \in \mathbb{R}^d$ be the contious latent vector (*vector*) and $\mathbf{y} \in \{0, 1\}^K$ be the corresponding symbolic one-hot label (*symbol*) indicating its belonging in total K categories. With the assumption that $\mathbf{y}$ is conditionally independent of $\mathbf{x}$ given $\mathbf{z}$, our model is defined as

$$p_\theta(\mathbf{y}, \mathbf{z}, \mathbf{x}) = p_\alpha(\mathbf{y}, \mathbf{z}) p_\beta(\mathbf{x}|\mathbf{z}) \tag{1}$$

where $p_\alpha(\mathbf{y}, \mathbf{z})$ is the EBM prior model with parameters α. $p_\beta(\mathbf{x}|\mathbf{z})$ is the top-down generative model with parameters β and $\theta = (\alpha, \beta)$. With this definition, the label $\mathbf{y}$ can be sufficiently inferred from the contious vector $\mathbf{z}$ after the inference of $\mathbf{z}$ from the sample $\mathbf{x}$, i.e. $\mathbf{z}$ is the information bottleneck.



(a) Unsupervised meta-training (b) Semi-supervised meta-test

Figure 1: Graphical illustration of Meta-SVEBM. Gray circles denote the unobserved parameters and white circles denote the observed examples. Dashed lines indicates variational inference and dotted lines means the short-run MCMC procedure[20]. (a) Our model treats ϕ, β as meta-parameters and introduces task-specific prior α_i in the meta-training phase. (b) We transfer the learned ϕ and makes predictions over $\mathbf{y}^q_{ij}$ by semi-supervised updates of α_i.

The vector $\mathbf{z}$ and symbol $\mathbf{y}$ are coupled together in the latent space as the form of EBM prior model,

$$p_\alpha(\mathbf{y}, \mathbf{z}) = \frac{1}{Z_\alpha} \exp\left(\langle \mathbf{y}, f_\alpha(\mathbf{z}) \rangle\right) p_0(\mathbf{z}) \tag{2}$$

where $p_0(\mathbf{z})$ is the reference distribution as isotropic Gaussian, $f_\alpha(\mathbf{z}) \in \mathbb{R}^K$ is a small multi-layer perceptron and Z_α is the normalizing constant or partition function.

The negative energy term $\langle \mathbf{y}, f_\alpha(\mathbf{z}) \rangle$ couples the contious vector $\mathbf{z}$ and symbolic vector $\mathbf{y}$ as associative memory. The inference of symbolic vector $\mathbf{y}$ can be achieved from latent vector $\mathbf{z}$ using a softmax classifier,

$$p_\alpha(\mathbf{y}|\mathbf{z}) \propto \exp\left(\langle \mathbf{y}, f_\alpha(\mathbf{z}) \rangle\right). \tag{3}$$

Therefore, $f_\alpha(\mathbf{z})$ maps a latent vector in $\mathbb{R}^d$ to logit scores in $\mathbb{R}^K$.

The marginal distribution of latent variable $\mathbf{z}$ is computed by summation over $\mathbf{y}$,

$$p_\alpha(\mathbf{z}) = \frac{1}{Z_\alpha} \exp\left(F_\alpha(\mathbf{z})\right) p_0(\mathbf{z}) \tag{4}$$

where $F_\alpha(\mathbf{z})$ denotes marginal energy as the form of log-sum-exponential

$$F_\alpha(\mathbf{z}) = \log \sum_{\mathbf{y}} \exp\left(\langle \mathbf{y}, f_\alpha(\mathbf{z}) \rangle\right). \tag{5}$$

Similar to the top-down network in VAE, the above prior model $p_\alpha(\mathbf{y}, \mathbf{z})$ stands on a generative model $p_\beta(\mathbf{x}|\mathbf{z})$ with parameters β. To be specific, for each observed example $\mathbf{x}$ as an image or feature,

$$\mathbf{x} = g_\beta(\mathbf{z}) + \epsilon \tag{6}$$

where $\epsilon \sim N(0, \sigma^2 I_D)$ is random noise with assumed variance σ^2, and $\mathbf{x} \sim N(g_\beta(\mathbf{z}), \sigma^2 I_D)$.

Sampling the prior $p_\alpha(\mathbf{z})$ and the posterior $p_\theta(\mathbf{z}|\mathbf{x})$ can be both accomplished by Langevin dynamics[27]. The prior sampling is affordable and computationally efficient due to the low-dimensional latent space while the posterior sampling requires more expensive backpropagation through the whole top-down model. Hence we shall recruit another inference network $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$ and amortize the sampling procedure as in VAE. Our model SVEBM is illustrated in Figure 2.

2.3 Unsupervised meta-training

Since we only assume the unlabelled dataset in the meta-training phase, we are supposed to perform unsupervised learning of SVEBM as in [22] and we recruit another inference network $q_\phi(\mathbf{z}|\mathbf{x})$ to approximate the true posterior $p_\theta(\mathbf{z}|\mathbf{x})$. Following VAE, we tend to learn the inference model $q_\phi(\mathbf{z}|\mathbf{x})$, the generator model $p_\beta(\mathbf{x}|\mathbf{z})$ with additional prior model $p_\alpha(\mathbf{z})$ jointly.

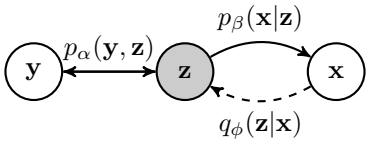


Figure 2: Graphical illustration of SVEBM. The white circle denotes the variable that can be observed and the grey circle is unobserved. Dashed lines denote variational inference. In the latent space, y and $\mathbf{z}$ are coupled by the prior model $p_\alpha(\mathbf{y},\mathbf{z})$ where y can be sufficiently inferred from $\mathbf{z}$. Given $\mathbf{z}$, $\mathbf{x}$ can be generated by the top-down model $p_\beta(\mathbf{x}|\mathbf{z})$ and the inference of $\mathbf{z}$ is accomplished by an inference network $q_\phi(\mathbf{z}|\mathbf{x})$.

Here we denote each randomly sampled unlabelled dataset as $\mathcal{D}_i = \{\mathbf{x}_j\}_{j=1}^U$ for simplicity and treat each dataset as an training example. We also adopt the set-dependent variational posterior [17] in the sense that the additional multi-head self-attention modules are added in the inference network to model the data dependency within each dataset such that the posterior of each instance is inferred conditioned on the given dataset as $q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)$.

The log-likelihood $p_\theta(\mathcal{D}_i)$ is lower bounded by evidence lower bound (ELBO).

$$\begin{aligned}\mathrm{ELBO}(\theta,\phi,\mathcal{D}_i) &= \sum_{j=1}^{U}\left[\log p_\theta(\mathbf{x}_j) - \mathbb{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)\|p_\theta(\mathbf{z}_j|\mathbf{x}_j)\right)\right] \\ &= \sum_{j=1}^{U}\left[\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}\left[\log p_\beta(\mathbf{x}_j|\mathbf{z}_j)\right] - \mathbb{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)\|p_{\alpha_i}(\mathbf{z}_j)\right)\right]\end{aligned} \tag{7}$$

where $\mathbb{D}_{\mathrm{KL}}$ denotes the Kullback-Leibler divergence.

For the task-specific prior parameter α_i, the learning gradient for a dataset $\mathcal{D}_i$ is

$$\nabla_{\alpha_i}\mathrm{ELBO} = \sum_{j=1}^{U}\left[\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}\left[\nabla_{\alpha_i}F_{\alpha_i}(\mathbf{z}_j)\right] - \mathbb{E}_{p_{\alpha_i}(\mathbf{z}_j)}\left[\nabla_{\alpha_i}F_{\alpha_i}(\mathbf{z}_j)\right]\right] \tag{8}$$

where $\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}$ is approximated by samples from the inference network with the reparametrization trick[15], while $\mathbb{E}_{p_{\alpha_i}(\mathbf{z}_j)}$ is approximated by short-run MCMC[20] from the prior. The short-run MCMC dynamics is initialized from the fixed distribution p_0 as $\mathbf{z}_j^0 \sim p_0(\mathbf{z}_j)$ and runs a fixed number of steps $t = 1,\ldots,T$.

$$\mathbf{z}_j^{t+1} = \mathbf{z}_j^t + s\nabla_{\mathbf{z}_j}\log p_{\alpha_i}(\mathbf{z}_j^t) + \sqrt{2s}\epsilon^t, \epsilon^t \sim N(0, I_d) \tag{9}$$

Let $\psi = (\phi,\beta)$ be the placeholder for the meta-parameters, i.e. the inference network ϕ and the generator network β. The learning gradient for ψ is

$$\begin{aligned}\nabla_\psi\mathrm{ELBO} = \sum_{j=1}^{U}\big[&\nabla_\psi\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}\left[\log p_\beta(\mathbf{x}_j|\mathbf{z}_j)\right] \\ &- \nabla_\psi\mathbb{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)\|p_0(\mathbf{z}_j)\right) + \nabla_\psi\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}\left[F_{\alpha_i}(\mathbf{z}_j)\right]\big]\end{aligned} \tag{10}$$

where $\mathbb{E}_{q_\phi(\mathbf{z}_j|\mathbf{x}_j,\mathcal{D}_i)}$ involved in the two terms is approximated by samples from the inference network with reparametrization trick and the $\mathbb{D}_{\mathrm{KL}}$ term is analytical tractable.

2.4 Semi-supervised meta-test

During the meta-test phase, we aim to learn a task-specific prior α_i according to the held-out few-shot classification task using the learned parameter ϕ in the meta-training phase. For a randomly sampled meta-test dataset $\mathcal{D}_i = \{\mathcal{S}_i,\mathcal{Q}_i\}$ where $\mathcal{S}_i = \{(\mathbf{x}_j^s,\mathbf{y}_j^s)\}_{j=1}^{KS}$ and $\mathcal{Q}_i = \{\mathbf{x}_j^q\}_{j=1}^{KQ}$, we can still perform unsupervised updates of task-specific parameters α_i as in meta-training phase using the query set.

$$\nabla_{\alpha_i}\mathrm{ELBO}(\mathcal{Q}_i) = \sum_{j=1}^{KQ}\left[\mathbb{E}_{q_\phi(\mathbf{z}_j^q|\mathbf{x}_j^q,\mathcal{Q}_i)}\left[\nabla_{\alpha_i}F_{\alpha_i}(\mathbf{z}_j^q)\right] - \mathbb{E}_{p_{\alpha_i}(\mathbf{z}_j^q)}\left[\nabla_{\alpha_i}F_{\alpha_i}(\mathbf{z}_j^q)\right]\right] \tag{11}$$

While for the support set with labels, the log-likelihood can be decomposed into

$$\log p_\theta(\mathcal{S}_i) = \sum_{j=1}^{KS} \left[\log p_\theta(\mathbf{x}_j^s) + \log p_\theta(\mathbf{y}_j^s|\mathbf{x}_j^s)\right] \tag{12}$$

The first term can be optimized as the unsupervised learning above and the second term can also be approximated by samples from the inference network,

$$\log p_\theta(\mathbf{y}_j^s|\mathbf{x}_j^s) = \log \mathbb{E}_{p_\theta(\mathbf{z}_j^s|\mathbf{x}_j^s)}\left[p_{\alpha_i}(\mathbf{y}_j^s|\mathbf{z}_j^s)\right] \approx \log \mathbb{E}_{q_\phi(\mathbf{z}_j^s|\mathbf{x}_j^s,\mathcal{S}_i)}\left[p_{\alpha_i}(\mathbf{y}_j^s|\mathbf{z}_j^s)\right]. \tag{13}$$

The learning gradients for task-specific prior model is computed accordingly. As in (3), $p_{\alpha_i}(\mathbf{y}_j^s|\mathbf{z}_j^s)$ is a softmax classifier.

$$\nabla_{\alpha_i} \log p_\theta(\mathbf{y}_j^s|\mathbf{x}_j^s) \approx \nabla_{\alpha_i} \log \mathbb{E}_{q_\phi(\mathbf{z}_j^s|\mathbf{x}_j^s,\mathcal{S}_i)}\left[p_{\alpha_i}(\mathbf{y}_i^s|\mathbf{z}_j^s)\right] \tag{14}$$

Algorithm 1: Unsupervised meta-training

Input : An unlabelled dataset $\mathcal{D}_u$, T training iterations, batch size B.
Output : Meta-parameter ϕ
for $t = 0 : T-1$ **do**
  Sample B tasks. Each task is $\{\mathbf{x}_j\}_{j=1}^U$.
  for $i = 0 : B-1$ **do**
    1. Sample $\mathbf{z}_j^- \sim p_{\alpha_i}(\mathbf{z}_j)$ using (9) and $\mathbf{z}_j^+ \sim q_\phi(\mathbf{z}_j|\mathbf{x}_j, \mathcal{D}_i)$ from the inference network.
    2. Compute task-specific parameters $\alpha_i^{t+1} = \alpha_i^t - \eta \nabla_{\alpha_i} \text{ELBO}$ using (8).
    3. Update meta-parameters using (10) $\psi^{t+1} = \psi^t - \eta' \nabla_\psi \text{ELBO}$, where $\psi = (\phi, \beta)$.

Algorithm 2: Meta-test for a held-out task

Input : A meta-test dataset $\mathcal{D} = \{\mathcal{S}, \mathcal{Q}\}$, $\mathcal{S} = \{(\mathbf{x}_j^s, \mathbf{y}_j^s)\}_{j=1}^{KS}$, $\mathcal{Q} = \{\mathbf{x}_j^q\}_{j=1}^{KQ}$, T iterations, meta-parameter ϕ.
Output : $\{\mathbf{y}_j^q\}_{j=1}^{KQ}$
for $t = 0 : T-1$ **do**
  1. Sample $\mathbf{z}_j^- \sim p_{\alpha_i}(\mathbf{z}_j)$ using (9) and $\mathbf{z}_j^+ \sim q_\phi(\mathbf{z}_j|\mathbf{x}_j, \mathcal{D}_i)$ from the inference network for each $\mathbf{x}_j^s$ and $\mathbf{x}_j^q$.
  2. Unsupervised updates of α with each $\mathbf{x}_j^s$ and $\mathbf{x}_j^q$: $\alpha^{t+1} = \alpha^t - \eta \nabla_\alpha \text{ELBO}$ using (11).
  3. Supervised updates of α with support set $\mathcal{S}$: $\alpha^{t+1} = \alpha^t - \eta' \nabla_\alpha \log p_\theta(\mathbf{y}_j^s|\mathbf{x}_j^s)$ using (14).
Predict the labels of query set: $\mathbf{y}_j^q = \arg\max p_\alpha(\mathbf{y}_j^q|\mathbf{z}_j^q)$.

3 Related work

Unsupervised meta-learning Unsupervised meta-learning tends to learn meaningful internal representations from a given unlabelled dataset during the meta-training phase and transfers the learned knowledge to solve a held-out tasks. There are two lines of research around unsupervised meta-learning. The first one learns to generate synthetic tasks from the unlabelled dataset. [11] learns to cluster the feature embeddings. [12] learns to augment and randomly sample the in-class datapoints. [1] augment the data through random selection. [13] leverages the generative models to group the in-class and out-of-class data pairs. Another line of research directly learns the multi-modality within each randomly sampled task without explicit task generation[17]. Our meta-SVEBM also follows this type of method. However, our method differs from [17] in that our task-specific prior is formulated as a much more flexible latent space EBM that is learned jointly with inference network and generative network, while Meta-GMVAE updates a Gaussian Mixture prior with on-the-fly Expectation Maximization (EM) algorithm, which makes the whole learning dynamics distinct.

Energy-based Model Energy-Based Model (EBM) captures dependencies between variables by assigning an energy scalar to each configuration of the variables, where observed examples are assigned with low energy. [10][20][29] have shown the expressiveness and effectiveness of EBM in the data space. [21] defines an EBM in the latent space as a correction of the non-informative uniform prior or isotropic Gaussian prior. Since the learning of EBM in the data space requires expensive MCMC sampling, this latent space EBM prior model can benefit from the much lower dimensional latent space sampling and well-mixed MCMC steps. [9] introduced joint EBM to reinterpret the discriminative classifier as a generative model and [30] further applied it on semi-supervised classification tasks.

4 Experiments

4.1 Experiment settings

We shall validate the effectiveness and expressiveness of meta-SVEBM on two commom benchmarks as Omniglot[16] and Mini-ImageNet[28]. (a) Omniglot consists of 1623 distinct categories of $28 \times 28 \times 1$ gray-scale hand-written characters, each of which has 20 instances. We split the dataset into three subsets with $1200/100/323$ classes for meta-training, validation and meta-test respectively. Following [11], each instance is further rotated by 90, 180, and 270 degrees so that we have total 1623×4 classes. We test 5-way and 20-way classification tasks on Omniglot where each task is given 1 or 5 labelled data to predict 15 queries. (b) Mini-ImageNet is composed of $60,000$ colour images of size $84 \times 84 \times 3$ with 100 classes, each having 600 examples. We use 64 classes for meta-training, 16 classes for validation and meta-test on the remaining 20 classes. We report S-shot 5-way 15-query classification results on Mini-ImageNet, where $S = 1, 5, 20, 50$.

4.2 Baselines

We compare our model with current state-of-the-art unsupervised meta-learning approaches. CACTUs[11] learns the clustering in the latent space and construct the meta tasks automatically. UMTRA[12] learns to generate meta-tasks using domain-specific augmentations. LASIUM[13] groups in-class pairs and out-of-class pairs sampled from the latent space into a meta-task. Meta-GMVAE[17] learns a VAE with task-specific Gaussian mixture prior and set-dependent variational posterior to solve meta-test tasks. We also compare our proposed method with supervised meta-learning algorithms as MAML[6] and ProtoNets[24].

A naive baseline might be training the raw images from scratch, while better performance could be obtained if training from the feature embeddings learned from the off-the-shelf unsupervised learning algorithms as ACAI[2], BiGAN[5], DeepCluster[3] and SimCLR[4].

4.3 Model architectures

We use neural networks to parameterize the three models as the inference network ϕ, the generative network β and the EBM prior model α.

In comparison to aforesaid baselines with fair heuristics, we adopt the same model choice as [17] for inference network and generative network. On Omniglot, we start from the raw image and learn from the scratch. The inference network comprises four stacked convolutional blocks (as conv4 in [13]), two multi-head self-attention layers and one additional linear layer to predict set-dependent mean and log-variance. Each convolutional block consists of one convolutional layer with 64 3×3 filters, batch normalization, ReLU activation and 2×2 max-pooling in sequence. The generative network is symmetric to the inference network with deconvolution operations. On Mini-ImageNet, we start from the feature embeddings learned from SimCLR. Therefore we can eliminate the feature extraction modules used before. The inference network has two multi-head self-attention layers and one linear projection layer to get the mean and log-variance and the generative network only has three linear layers with ReLU activation to project from the latent space to the feature embeddings.

As for the EBM prior model, we parametrize it as a simple multi-layer perceptron with three linear layers and leaky ReLU activation. In the meta-training phase, we add spectral normalization[19] for each linear layer to stabilize the long-term training procedure.

The three models are trained jointly in the meta-training phase with 9×10^4 iterations and batch size 4 by Adam optimizer[14] with learning rate 10^{-4} for inference and generative networks, 10^{-5} for the prior model. During the meta-test stage, only the prior model is updated for few-shot classification using Adam with learning rate 10^{-3} for both supervised and unsupervised updates, and we report the best classification accuracy with batch size 1 for 1-shot tasks and batch size 4 for the rest. All experiments have be done on a single Nvidia GeForce RTX 3080 GPU.

4.4 Experiment results

Table 1 shows the results on the Omniglot dataset. We find that the our method achieves competitive results comparing to the unsupervised approaches. Table 2 shows the experiments on Mini-ImageNet.

Meta-SVEBM outperforms all state-of-the-art baselines and even outperforms the supervised ProtoNets on 5-way 50-shot classification task with $\sim 1\%$ labels used. Although the use of feature embeddings extracted from SimCLR seems to be better representations, the comparison with CACTUs, UMTRA and Meta-GMVAE still demonstrate the expressiveness of the Meta-SVEBM.

Table 1: Summary of few-shot classification results (way, shot) on the Omniglot dataset. The accuracy is calculated over 1000 randomly sampled meta-test tasks. Bold number denotes the best performance.

Method	Feature Extractor	(5,1)	(5,5)	(20,1)	(20,5)
CACTUs-MAML[11]	BiGAN	58.18	78.66	35.56	58.62
CACTUs-ProtoNets[11]	BiGAN	54.74	71.69	33.40	50.62
CACTUs-MAML[11]	ACAI	68.84	87.78	48.09	73.36
CACTUs-ProtoNets[11]	ACAI	68.12	83.58	47.75	66.27
UMTRA-MAML[12]	N/A	83.80	95.43	74.25	92.12
LASIUM-N-VAE-MAML[13]	N/A	76.11	94.42	–	–
LASIUM-OC-VAE-ProtoNets[13]	N/A	73.22	85.05	–	–
LASIUM-RO-GAN-MAML[13]	N/A	83.26	95.29	–	–
LASIUM-RO-GAN-ProtoNets[13]	N/A	80.15	91.10	–	–
Meta-GMVAE[17]	N/A	**94.92**	97.09	**82.21**	90.06
Meta-SVEBM (Ours)	N/A	91.85	**97.21**	79.66	**92.21**
MAML (*supervised*)	N/A	94.46	98.83	84.60	96.29
ProtoNets (*supervised*)	N/A	98.35	99.58	95.31	98.81

Table 2: Summary of few-shot classification results (way, shot) on the Mini-ImageNet dataset. The accuracy is calculated over 1000 randomly sampled meta-test tasks. Bold number denotes the best performance.

Method	Feature Extractor	(5,1)	(5,5)	(5,20)	(5,50)
CACTUs-MAML[11]	BiGAN	36.24	51.28	61.33	66.91
CACTUs-ProtoNets[11]	BiGAN	36.62	50.16	59.56	63.27
CACTUs-MAML[11]	DeepCluster	39.90	53.97	63.84	69.64
CACTUs-ProtoNets[11]	DeepCluster	39.18	53.36	61.54	63.55
UMTRA-MAML[12]	N/A	39.93	50.73	61.11	67.15
LASIUM-N-GAN-MAML[13]	N/A	40.19	54.56	65.17	69.13
LASIUM-N-GAN-ProtoNets[13]	N/A	40.05	52.53	59.45	61.43
CACTUs-MAML[17]	SimCLR	40.39	52.35	61.09	64.89
UMTRA-MAML[17]	SimCLR	40.85	51.47	61.03	67.30
Meta-GMVAE[17]	SimCLR	42.82	55.73	63.14	68.26
Meta-SVEBM (Ours)	SimCLR	**43.38**	**58.03**	**67.07**	**72.28**
MAML (*supervised*)	N/A	46.81	62.12	71.03	75.54
ProtoNets (*supervised*)	N/A	46.56	62.29	70.07	72.04

5 Conclusion

Meta-learning is a key step for artificial intelligence to achieve the efficiency of human learning. In this work, drawing inspirations from a theory of human learning, Meta-SVEBM is top-down generative model with an EBM as its prior, and we jointly learn a variational inference network to infer the latent variables. Due to the low-dimensionality of the latent space and the expressiveness of the top-down generation network, a small multi-layer perceptron is able to capture the data regularities effectively in the latent space, and the EBM can rapidly adapt to the specifics of each task. Evaluation on standard benchmarks demonstrates that our model achieves competitive or superior performance compared to previous state-of-the-arts meta-learning models.

References

[1] A. Antoniou and A. Storkey. Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation. *arXiv preprint arXiv:1902.09884*, 2019.

[2] D. Berthelot, C. Raffel, A. Roy, and I. Goodfellow. Understanding and improving interpolation in autoencoders via an adversarial regularizer. *arXiv preprint arXiv:1807.07543*, 2018.

[3] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[4] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[5] J. Donahue, P. Krähenbühl, and T. Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[6] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[7] C. Finn, K. Xu, and S. Levine. Probabilistic model-agnostic meta-learning. *arXiv preprint arXiv:1806.02817*, 2018.

[8] S. Flennerhag, A. A. Rusu, R. Pascanu, F. Visin, H. Yin, and R. Hadsell. Meta-learning with warped gradient descent. *arXiv preprint arXiv:1909.00025*, 2019.

[9] W. Grathwohl, K.-C. Wang, J.-H. Jacobsen, D. Duvenaud, M. Norouzi, and K. Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

[10] T. Han, E. Nijkamp, L. Zhou, B. Pang, S.-C. Zhu, and Y. N. Wu. Joint training of variational auto-encoder and latent energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7978–7987, 2020.

[11] K. Hsu, S. Levine, and C. Finn. Unsupervised learning via meta-learning. *arXiv preprint arXiv:1810.02334*, 2018.

[12] S. Khodadadeh, L. Bölöni, and M. Shah. Unsupervised meta-learning for few-shot image classification. *arXiv preprint arXiv:1811.11819*, 2018.

[13] S. Khodadadeh, S. Zehtabian, S. Vahidian, W. Wang, B. Lin, and L. Bölöni. Unsupervised meta-learning through latent-space interpolation in generative models. *arXiv preprint arXiv:2006.10236*, 2020.

[14] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

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

[16] B. Lake, R. Salakhutdinov, J. Gross, and J. Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the annual meeting of the cognitive science society*, volume 33, 2011.

[17] D. B. Lee, D. Min, S. Lee, and S. J. Hwang. Meta-gmvae: Mixture of gaussian vae for unsupervised meta-learning. In *International Conference on Learning Representations*, 2020.

[18] Z. Li, F. Zhou, F. Chen, and H. Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

[19] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[20] E. Nijkamp, M. Hill, S.-C. Zhu, and Y. N. Wu. Learning non-convergent non-persistent short-run mcmc toward energy-based model. *arXiv preprint arXiv:1904.09770*, 2019.

[21] B. Pang, T. Han, E. Nijkamp, S.-C. Zhu, and Y. N. Wu. Learning latent space energy-based prior model. *arXiv preprint arXiv:2006.08205*, 2020.

[22] B. Pang and Y. N. Wu. Latent space energy-based model of symbol-vector coupling for text generation and classification. In *International Conference on Machine Learning*, pages 8359–8370. PMLR, 2021.

[23] S. Ravi and A. Beatson. Amortized bayesian meta-learning. In *International Conference on Learning Representations*, 2018.

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

[25] J. B. Tenenbaum. *A Bayesian framework for concept learning*. PhD thesis, Massachusetts Institute of Technology, 1999.

[26] S. Thrun and L. Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[27] G. E. Uhlenbeck and L. S. Ornstein. On the theory of the brownian motion. *Physical review*, 36(5):823, 1930.

[28] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29:3630–3638, 2016.

[29] J. Xie, Y. Lu, S.-C. Zhu, and Y. Wu. A theory of generative convnet. In *International Conference on Machine Learning*, pages 2635–2644. PMLR, 2016.

[30] S. Zhao, J.-H. Jacobsen, and W. Grathwohl. Joint energy-based models for semi-supervised classification. In *ICML Workshop on Uncertainty and Robustness in Deep Learning*, 2020.