UNDERSTANDING SELF-SUPERVISED LEARNING VIA INFORMATION BOTTLENECK PRINCIPLE

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-supervised learning alleviates the massive demands for annotations in deep learning, and recent advances are mainly dominated by contrastive learning. Existed contrastive learning methods narrows the distance between positive pair while neglect the redundancy shared by positive pairs. To address this issue, we introduce the information bottleneck principle and propose the Self-supervised Variational Information Bottleneck (SVIB) learning framework. Specifically, We apply Gaussian Mixture Model (GMM) as the parametric approximate posterior distribution to the real feature distribution and introduce the categorical latent variable. Features from different augmentations are forced to infer the other one and the latent variable together. Then, we propose variational information bottleneck as our objective, which is composed of two parts. The first is maximizing the mutual information between the inferred feature and the latent variable. The second is minimizing the mutual information between the other feature and the latent variable. Compare to previous works, SVIB provides the self-supervised learning field with a novel perspective from the variational information bottleneck, while also highlighting a long-neglected issue. Experiments show that SVIB outperforms current SOTA methods in multiple benchmarks.

1 INTRODUCTION

Self-supervised learning has raised widespread interest because of alleviating the massive demand for annotations in deep learning, and recent advances are mainly dominated by contrastive learning (**?**). Contrastive learning enables training by discriminating positive pairs of samples from negative pairs (Wu et al., 2018; Misra & van der Maaten, 2020; Ye et al., 2019; Zhuang et al., 2019; Tian et al., 2020; Chen et al., 2020; He et al., 2020; Li et al., 2020) or only narrowing the distance between positive pairs (Grill et al., 2020; Caron et al., 2020; 2021). The positive samples are usually from different augmentations of the same image while negative samples are sampled from all other images. These methods expect to obtain a feature space where different augmentations from the same image can be as close as possible, while just different in their approach to aligning positive pairs. For example, MoCo (He et al., 2020) uses a queue to store negative samples and discriminates positive pairs from negative samples with infoNCE (van den Oord et al., 2018) loss. SwAV (Caron et al., 2020) constructs clustering prototypes and narrows the cluster assignments between positive pairs. DINO (Caron et al., 2021) aligns positive pairs by minimizing the cross-entropy between the output of teacher and student network.

Existed methods expect to vary the redundancy among the positive samples with some semantic-irrelevant data augmentations so that narrowing positive samples can retain the common semantic information while eliminating noise and redundancy. However, such data augmentation is difficult to obtain. If we want to obtain a data augmentation with which only semantic information is preserved and all noise is removed, then this data augmentation will perfectly define semantic information and noise, which is unattainable. For example, if we want to learn a representation of a person, then our data augmentation should be able to generate data of this person wearing all clothes and in all scenes in order to separate semantic information and redundancy. Unfortunately, this is impossible. So the currently applied data augmentation implies the redundancy between the positive samples, which are ignored in previous methods.

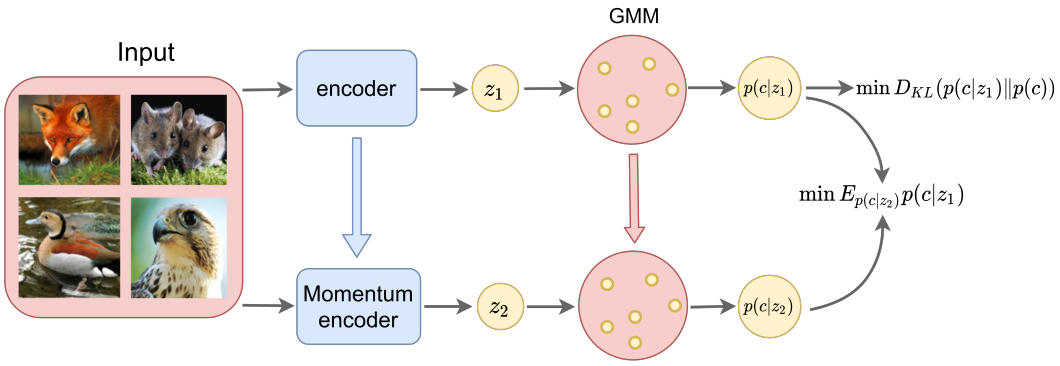

Figure 1: The framework of Self-supervised Variational Information Bottleneck.

Considering an episode, proposed by Redlich (1993) , where we are supplied with an article (in the case of English). This article lacks spaces between words, punctuation, and capitalization. We hope to recover the article without any prior information. These unseparated letters are regrouped into words that are similar to the human semantics by minimizing the entropy of each letter. It suggests that the learning process of humans is similar to entropy reduction, and the redundancy information will be eliminated before the semantic information during entropy reduction.

When coming back to self-supervised learning, we also need entropy reduction to reduce the redundancy between the positive samples while keeping the semantic information. In this paper, we make the first attempt to remove the redundancy between augmentations with the introduction of the information bottleneck (Tishby et al., 2000; Alemi et al., 2016) framework. Assume that the latent variable C is implicit and can generate features of two different augmentations Z_1, Z_2. Besides, we have the Markov chain as $Z_1 \leftrightarrow Z_2 \leftrightarrow C$. Information Bottleneck suggests a standard objective as $R_{IB} = I(C, Z_1) - \beta I(Z_2, C)$. To maximize this objective function, we need to maximize the first term, which means learn information to predict C from Z_1, while minimizing the last term, encouraging C to "forget" Z_2. This objective actually restricts the learning of shared information from two different augmentations and serves as a regularization term.

Moreover, directly calculate the information bottleneck with the latent variables will suffer intractable posterior distribution caused by high dimensional integration. Then variational distribution is introduced to approximate the intractable posterior distribution and results in a lower bound of IB, namely Variational Information Bottleneck (VIB).

Specifically, we apply Gaussian Mixture Model (GMM) here as the posterior parametric approximation of the feature distribution, from which we introduce the latent variable. Features from different augmentations are forced to infer the other one and the latent variable together. Features from different augmentations are forced to infer the other one and the latent variable together. Then, we propose variational information bottleneck as our objective, which is composed of two parts. The first is maximizing the mutual information between the inferred feature and the latent variable. The second is minimizing the mutual information between the other feature and the latent variable. In this way, we set an information bottleneck on the latent variable, with which we force model to learn a consistent representation for different augmentations with as little information as possible. In this process, redundancy information will be dropped first and induce to a better representation. The contributions of this paper can be summarized as follows:

- To our best knowledge, we are the first who attempts to constrain the mutual information between different views to facilitate representation learning via redundancy reduction. Previous works only focus on maximizing the mutual information between different views but neglect the entropy minimization principle and the redundancy shared by different augmentations. Our work provides a new perspective for the current self-supervised learning field.
- We introduce a theoretical framework with information bottleneck theory for self-supervised representation learning. Features from different augmentations are forced to infer the other one and the latent variable together. We set an information bottleneck on the latent variable

by maximizing mutual information between feature and latent variable while minimizing mutual information between the other one and latent variable. In this way, the model is forced to learn representations with minimal entropy.

- SVIB is evaluated on both convolutional network and vision transformer (Dosovitskiy et al., 2021) and shows consistent improvement compared to previous methods. Transfer tasks in semi-supervised, object detection, and semantic segmentation also demonstrate the generalize ability of SVIB.

2 Related work

Recent advances (Caron et al., 2021; 2020; Grill et al., 2020; He et al., 2020) in visual self-supervised learning show its potential in producing general representations for given images, and the key component of self-supervised learning is the design of a task to enable learning without any human annotations.

Doersch et al. (2015) address this issue by forcing the model to predict the positional relationship between two patches. Noroozi & Favaro (2016) treat this as a jigsaw puzzle further. Gidaris et al. (2018) use the rotation of images as the learning target. Pathak et al. (2016) and Zhang et al. (2016) consider this issue from reconstruction, and employee inpainting and colorization as their learning tasks respectively. These methods induce the model to concentrate on certain properties of the image but ignore others, which impair the representation ability. Besides that, they only focus on the intra-image information but neglect the inter-image relationship.

Contrastive learning enables self-supervised learning via discrimination tasks where different augmentations of the same image are regarded as positive pairs and others as negative pairs. Wu et al. (2018) build a memory bank for storing features in the previous step and regards features except itself in the memory bank except the current sample itself as negative samples. Misra & van der Maaten (2020); Ye et al. (2019); Zhuang et al. (2019); Tian et al. (2020); Chen et al. (2020) generates negative pairs using all samples within the current mini-batch. MoCo (He et al., 2020) apply a momentum encoder to generate negative pairs and a queue to store recent negative pairs cross mini-batch. PCL (Li et al., 2020) formulates the self-supervised learning with EM algorithm and generates negative pairs with k-means. These methods differ in their way to obtain negative pairs. From another point of view, BYOL (Grill et al., 2020) enables self-supervised learning by narrowing the distance between different views with an asymmetrical predictor and gets rid of the dependence on negative samples. SwAV (Caron et al., 2020) indirectly narrows two views through their cluster assignments. DINO (Caron et al., 2021) propose to apply trainable prototypes for end-to-end training and the centering operation to avoid trivial solutions. The difference between these approaches is how they construct the contrasting agents, such as predictor in BYOL, prototype in SwAV, and learnable parameters in DINO.

Mutual information maximization. There are also other works that consider the unsupervised representation learning tasks from an information perspective. Hjelm et al. (2018); Bachman et al. (2019) learns image representation via maximizing the mutual information between local and global features. Wu et al. (2020) derive a new lower bound on mutual information that supports sampling negative examples from a restricted distribution to facilitate representation learning. However, all of them propose to maximize the mutual information but neglect to reduce the entropy of representations.

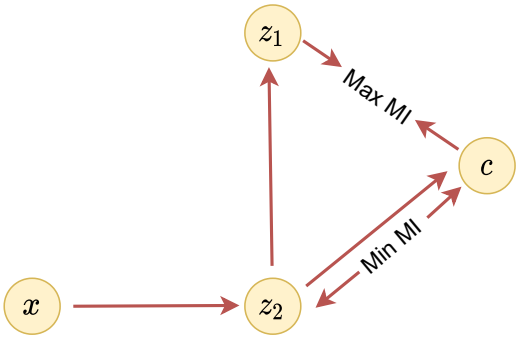

Figure 2: Illustration to the relationship between variables and our objective.

Semantic augmentation. Wang et al. (2021) attempts to generate semantic data augmentation, i.e., to generate different images with the same semantic information, such as cars from different viewpoints, animals in different scenes, etc. However, the method is only accessible by supervised learning or semi-supervised learning. In addition, Cai et al. (2020) proposes to estimate the distribution of the data augmentation in the feature space by sampling the current data augmentation multiple times, thus directly contrasting the distribution of positive samples instead of instances to improve the performance of self-supervised learning. However, the method is only applied to supervised learning or semi-supervised learning, and the method learns the distribution of images of the same category within the current dataset, which does not achieve the perfect data augmentation mentioned above. In addition, B also proposes to estimate the distribution of the data augmentation in the feature space by sampling the current data augmentation multiple times, thus directly comparing the distribution of positive samples to improve the performance of self-supervised learning. However, this approach is also limited to the currently applied data augmentation and does not solve the aforementioned problems.

3 METHODOLOGY

For clarity, we define our notions here. As illustrated in Fig 2, we regard output of the encoder as z_1 and output of the momentum encoder as z_2. The latent variable c is a categorical variable introduced by the GMM model.

3.1 VARIATIONAL CONTRASTIVE LEARNING

Following the standard in the Information Bottleneck literature, we assume the joint distribution can be factored as follows:

$$p(X, Z_1, Z_2, C) = p(C|Z_1, Z_2, X)p(Z_1|Z_2, X)P(Z_2|X)p(X) \tag{1}$$

With the probabilistic relationship between Z_1, Z_2, and C shown in Figure 2, we have $p(Z_1|Z_2, X) = p(Z_1|Z_2)$, $p(C|Z_1, Z_2, X) = p(C|Z_2)$. Then the joint distribution can be rewritten as:

$$p(X, Z_1, Z_2, C) = p(C|Z_2)p(Z_1|Z_2)p(Z_2|X)p(X) \tag{2}$$

Here, X is the observations, Z_1, Z_2 are features generated from different augmentation of X respectively. The latent variable C obeys a categorical distribution, and the number of categories keeps the same with prototypes. Then we have the conditional mutual information between C and Z_1:

$$I(C, Z_1|X) = \sum_x p(x)\mathbb{E}_{p(c,z_1|x)}[\log \frac{p(z_1|c, x)}{p(z_1|x)}] \tag{3}$$

The $p(z_1|c, x)$ is fully defined by Markov Chain as follows:

$$p(z_1|c, x) = \int p(z_1, z_2|x)dz_2 = \int p(z_1|z_2, x)p(z_2|c)dz_2 = \int \frac{p(z_1|z_2)p(c|z_2)p(z_2)}{p(c)}dz_2 \tag{4}$$

The calculation of the true posterior distribution is intractable, then we leverage a variational distribution $q(z_1|c)$ to approximate $p(z_1|c, x)$. Because that the Kullback Leibler divergence is always positive, we have:

$$\begin{aligned} I(C, Z_1|X) &\geq \sum_x p(x)\mathbb{E}_{p(c,z_1|x)}[\log \frac{q(z_1|c)}{p(z_1|x)}] \\ &= \sum_x p(x)\mathbb{E}_{p(c,z_1|x)}[\log q(z_1|c)] - \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log p(z_1|x)] \end{aligned} \tag{5}$$

Considering the first part of Eq. 5, we have:

$$\begin{aligned} \sum_x p(x)\mathbb{E}_{p(c,z_1|x)}[\log q(z_1|c)] &= \sum_x p(x)\mathbb{E}_{p(c|z_2)p(z_1|z_2)p(z_2|x)}[\log q(z_1|c)] \\ &= \sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}\mathbb{E}_{p(c|z_2)}[\log p(c|z_1, x) + \log \frac{q(z_1|c)}{p(c|z_1, x)}] \end{aligned} \tag{6}$$

With the Bayesian formula, we have:

$$p(c|z_1, x) = \frac{p(z_1|c, x)p(c)}{\sum_{i=1,\cdots,I} p(z_1|c_i, x)p(c = c_i)} \tag{7}$$

We take the uniform assumption for the prior distribution of $p(c = c_i) = 1/I$, where I is the number of categories for c. Then we have:

$$p(c|z_1) = \frac{p(z_1|c, x)\frac{1}{I}}{\sum_{i=1,\cdots,I} p(z_1|c_i, x)\frac{1}{I}} = \frac{p(z_1|c, x)}{\sum_i p(z_1|c_i, x)} \tag{8}$$

We further bring Eq. 8 into Eq. 6, then we can rewrite the second term in Eq. 6 as:

$$\sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}\mathbb{E}_{p(c|z_2)}[\log \frac{q(z_1|c)}{p(z_1|c, x)} \sum_i p(z_1|c_i, x)] \tag{9}$$

Because $q(z_1|c)$ is the approximation to $p(z_1|c, x)$, we can assume that $q(z_1|c) \approx p(z_1|c, x)$. Then the second term in Eq. 6 can be reduced to:

$$\sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}p(c|z_2)[\log \sum_i p(z_1|c_i, x)] \tag{10}$$

Then we bring Eq. 10 into Eq. 5 and Eq. 6, and get:

$$\begin{aligned} I(C, Z_1|X) &\geq \sum_x p(x)\mathbb{E}_{p(c,z_1|x)}[\log q(z_1|c)] - \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log p(z_1|x)] \\ &= \sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}\mathbb{E}_{p(c|z_2)}[\log p(c|z_1, x)] + \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log \frac{\sum_i p(z_1|c_i, x)}{p(z_1|x)}] \end{aligned} \tag{11}$$

For the second term in Eq. 11, we have:

$$\begin{aligned} \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log \frac{\sum_i p(z_1|c_i, x)}{p(z_1|x)}] &= \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log \frac{\sum_i p(z_1|c_i, x)p(c_i)}{p(z_1|x)p(c_i)}] \\ &= \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log \frac{p(z_1|x)}{p(z_1|x)\frac{1}{I}}] \\ &= \sum_x p(x)\mathbb{E}_{p(z_1|x)}[\log I] = \log I \end{aligned} \tag{12}$$

This term is a constant, then we can omit it in our optimization. Finally, we have:

$$I(C, Z_1|X) \geq \sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}\mathbb{E}_{p(c|z_2)}[\log p(c|z_1, x)] \tag{13}$$

We now consider the other term $I(C, Z_2|X)$ in our conditional information bottleneck.

$$I(C, Z_2|X) = \sum_x p(x)\mathbb{E}_{p(c,z_2|x)}[\log \frac{p(c|z_2, x)}{p(c|x)}] \tag{14}$$

The computation of $p(c|x) = \int p(c, z_2|x)dz_2$ is intractable. Thus, we let q(c) as a variational approximation. Then we have the upper bound:

$$\begin{aligned} I(C, Z_2|X) &\leq \sum_x p(x)\mathbb{E}_{p(c,z_2|x)}[\log \frac{p(c|z_2, x)}{q(c)}] \\ &= \sum_x p(x)\mathbb{E}_{p(z_2|x)}\mathbb{E}_{p(c|z_2,x)}[\log \frac{p(c|z_2, x)}{q(c)}] \\ &= \sum_x p(x)\mathbb{E}_{p(z_2|x)}[D_{\mathrm{KL}}(p(c|z_2, x)\|q(c)] \end{aligned} \tag{15}$$

Combining the proposed two bounds, we have the final information bottleneck:

$$VIB = I(C, Z_1|X) - \beta I(C, Z_2|X) \geq \sum_x p(x)\mathbb{E}_{p(z_1|z_2)p(z_2|x)}\mathbb{E}_{p(c|z_2)}[\log p(c|z_1, x)] \\ -\beta \sum_x p(x)\mathbb{E}_{p(z_2|x)}[D_{\mathrm{KL}}(p(c|z_2, x)\|q(c)] \tag{16}$$

In practice, we formulate the negative VIB as our loss function and enable end-to-end training with a mini-batch. The mathematical form of the loss function is given as:

$$Loss = p(c|z_2)\log p(c|z_1) + p(c|z_2)\log\frac{p(c|z_2)}{q(c)} \tag{17}$$

We assume each component of the GMM model follows an isotropy Gaussian distribution. Then we have:

$$p(z|c_i) = e^{-(z-c_i)^2/\sigma^2} \tag{18}$$

Combine Eq. 8 we have:

$$p(c_i|z) = \frac{e^{-(z-c_i)^2/\sigma^2}}{\sum_i e^{-(z-c_i)^2/\sigma^2}} \tag{19}$$

Since both z and the μ_{c_i} are normalized vector, we have $(z - c_i)^2 = 2 - 2z \cdot c_i$. Combining with Eq. 17, 8, we have:

$$Loss = \sum_i \frac{exp(c_i \cdot z_2/\sigma_i^2)}{\sum_j exp(c_i \cdot z_2/\sigma_j^2)} \log \frac{exp(c_i \cdot z_1/\sigma_i^2)}{\sum_j exp(c_i \cdot z_1/\sigma_j^2)} \\ + \frac{exp(c_i \cdot z_2/\sigma_i^2)}{\sum_j exp(c_i \cdot z_2/\sigma_j^2)} \log \frac{exp(c_i \cdot z_2/\sigma_i^2)}{\sum_j exp(c_i \cdot z_2/\sigma_j^2) \cdot q(c)} \tag{20}$$

4 EXPERIMENTS

We assess SVIB on ImageNet-1M (Deng et al., 2009) with the commonly used convolutional network ResNet50 (He et al., 2016) and the recently emerging advance VisionTransformer (ViT) (Dosovitskiy et al., 2021) to reveal its performance across architectures. We first train models with SVIB and evaluate pre-trained models in the linear classification task and the k-nearest neighbor (KNN) classification task on ImageNet. Then, we transfer the pre-trained ResNet50 to the semi-supervised scene to validate its potential in finetuning. Object detection and segmentation tasks are also applied to evaluate SVIB in dense pixels tasks.

4.1 IMPLEMENTATION DETAILS

We follow the standard implementation of ResNet50 (He et al., 2016) and ViT/S-16 (Dosovitskiy et al., 2021). The projection head is an MLP with 2048 hidden units and outputs a 256-D and L2-normalized feature. Data augmentation is composed of color jittering, Gaussian blur, and solarization which is the same as BYOL (Grill et al., 2020) and DINO (Caron et al., 2021). The multi-crop (Caron et al., 2020) operation is also applied by following the setting of DINO (10 local crops range from 0.05 to 0.25 for ViT and 6 local crops range from 0.05 to 0.14 for ResNet50 in default, unless otherwise stated). We use adamw optimizer (Loshchilov & Hutter, 2018) for ViT. The learning rate is linearly increasing in the first 10 epochs to 0.0005*batch size/256 (Goyal et al., 2017), and weight decay increases from 0.04 to 0.4 with a cosine scheduler. We use LARS optimizer (Ginsburg et al., 2018) for ResNet50. The base learning rate is 0.3, and weight decay is fixed on 1e-6. Others follow the same setting with ViT. We set $K = 65536$. The teacher network output more consistent representations for EMA updating, so we give a smaller variance to the teacher network. Specifically, we set $\sigma_t = 0.2$, $\sigma_s = 0.32$. We apply an exponentially decaying β which decays from 0.5 to 0.01 through the whole training process. We re-initialize the GMM model at the end of each epoch to avoid the parameters of the GMM model deviating too much from the true distribution of the current feature. Specifically, we randomly sample N instances from the dataset and re-initialize the mean

Method	architecture	epochs	Linear	KNN
Local aggregation (Zhuang et al., 2019)	ResNet50	200	60.2	49.4
PIRL (Misra & van der Maaten, 2020)	ResNet50	800	63.6	-
PCL (Li et al., 2020)	ResNet50	200	65.9	54.5
SimCLRv2 (Chen et al., 2020)	ResNet50	800	71.7	-
MoCo v2 (Chen et al., 2020)	ResNet50	300	71.1†	62.9†
BYOL (Grill et al., 2020)	ResNet50	300	72.7†	65.4†
SwAV (Caron et al., 2020)	ResNet50	300	74.1†	65.4†
DINO (Caron et al., 2021)	ResNet50	300	74.5	65.6
SVIB (Ours)	ResNet50	300	**74.8**	**69.4**
MoCo v2 (Chen et al., 2020)	ViT-S/16	300	71.6†	62.0†
BYOL (Grill et al., 2020)	ViT-S/16	300	71.4†	66.6†
SwAV (Caron et al., 2020)	ViT-S/16	300	71.8†	64.7†
DINO (Caron et al., 2021)	ViT-S/16	300	76.1	72.8
SVIB (Ours)	ViT-S/16	300	**76.4**	**73.5**

Table 1: Top-1 accuracy under linear classification and KNN 20 under K nearest neighbour classification on ImageNet with ResNet-50 as backbone. †: The result is reported by Caron et al. (2021).

of each component in GMM with their feature. The impact of this operation is discussed in the experiment section. We train SVIB for 300 epochs on ImageNet. In the ablation study, we adopt a more accessible configuration. We set $K = 8192$ and use 2 local crops. The training time is reduced to 100 epochs with a batch size of 512.

4.2 Image classification

Linear and KNN evaluation. Initially, we apply the linear classification and k-nearest neighbor (kNN) classification tasks on self-supervised learning models. We follow the standard evaluation protocol. We fix the encoder and attach a fully connected layer at the end of the encoder. The SGD optimizer and a batch size of 1024 are applied to train the FC layer on ImageNet for 100 epochs. We apply random size crops and horizontal flips augmentation during training and report the accuracy on a central crop. We use a 0.3 learning rate for ResNet50 and a 1e-3 learning rate for ViT-S/16. The kNN classifier matches the feature of an image to the k nearest stored features that vote for the label. Thus, kNN classification is less parametric compare to linear classification and is more stable (Caron et al., 2021). We apply the result of kNN 20 as our main result. Table 1 shows that the proposed method achieves a consistent improvement compared to previous methods in both linear evaluation and kNN classification. SVIB achieves a particularly significant improvement on the knn of resnet50 (3.8 improvement to DINO), which indicates that the semantically similar features will be closer in the feature space learned by SVIB.

Semi-supervised image classification. Then, We reveal the fine-tuning potential of our proposed methods with the semi-supervised setting on ImageNet. Follow the setting from Chen et al. (2020), we select the same subset (1% or %10) of ImageNet training data and fine-tune self-supervised models on these subsets. Table 2 report the top-1 and top-5 accuracy on ImageNet validation set. The results show that our method has good fine-tune potential with limited data (1%), even surpassing the model with 800 epochs of training.

4.3 Object detection and segmentation

We assess the performance of self-supervised models on the dense pixel tasks. We choose CoCo (Lin et al., 2014) as the experimental dataset and follow the setting from MoCov2 (Chen et al., 2020) to fine-tune the pre-trained model on COCO with the $1\times$ schedule. Table 3 report the result. The results show that SVIB pre-training generalizes well on the segmentation or detection task, and even better than methods trained for long epochs (SVIB 300 epochs, MoCo v2 800 epochs, BYOL 1000 epochs).

Method	architecture	#pretrain epochs	Top-1 Accuracy 1%	Top-1 Accuracy 10%	Top-5 Accuracy 1%	Top-5 Accuracy 10%
Random (Wu et al., 2018)	ResNet-50	-	-	-	22.0	59.0
Supervised baseline (Zhai et al., 2019)	ResNet-50	-	-	-	48.4	80.4
Semi-supervised learning methods:						
Pseudolabels (Zhai et al., 2019)	ResNet-50v2	-	-	-	51.6	82.4
S^4L Rotation (Zhai et al., 2019)	ResNet-50v2	-	-	-	53.4	83.8
Self-supervised learning methods:						
PIRL (Misra & van der Maaten, 2020)	ResNet-50	800	30.7	60.4	57.2	83.8
SimCLR Chen et al. (2020)	ResNet-50-MLP	1000	48.3	65.6	75.5	87.8
DINO (Caron et al., 2021)	ResNet-50-MLP	800	52.9†	66.6†	77.5†	87.6†
BYOL (Grill et al., 2020)	ResNet-50-MLP_{big}	1000	53.2	68.8	78.4	89.0
MoCov2 (Chen et al., 2020)	ResNet-50-MLP	800	52.4	65.3	78.4	86.6
SwAV (Caron et al., 2020)	ResNet-50-MLP	800	53.9	**70.2**	78.5	**89.9**
SVIB (Ours)	ResNet-50-MLP	300	**55.9**	68.6	**78.6**	88.6

Table 2: **Semi-supervised learning** on ImageNet. We report top-1 and top-5 accuracy on the ImageNet validation set of self-supervised models that are finetuned on 1% or 10% of labeled data. The accuracy is the mean of 5 independent runs. †: The result is produced by us with the released pre-trained model under the same setting.

Method	Mask R-CNN, R50-FPN, Detection						Mask R-CNN, R50-FPN, Segmentation					
	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP_S	AP_M	AP_L	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP_S	AP_M	AP_L
Supervised	38.9	59.6	42.0	23.0	42.9	49.9	35.4	56.5	38.1	17.5	38.2	51.3
MoCo v2 (Chen et al., 2020)	39.2	59.9	42.7	23.8	42.7	50.0	35.7	56.8	38.1	17.8	38.1	50.5
BYOL (Grill et al., 2020)	39.9	60.2	43.2	23.3	43.2	**52.8**	-	-	-	-	-	-
SVIB (Ours)	**40.0**	**61.7**	**43.4**	**24.8**	**43.9**	50.4	**36.7**	**58.6**	**39.1**	**18.4**	**39.8**	**51.8**

Table 3: Transfer learning performance (%) on COCO detection and COCO instance segmentation. MoCo v2 is trained for 800 epochs, and BYOL is trained for 1000 epochs. SVIB is trained for 300 epochs.

Method	ReIntial	IB	β	kNN-20
1	✓	×	×	66.3
2	×	✓	0.1	divergence
3	×	✓	0.5	61.3
4	✓	✓	0.5	64.2
5	✓	✓	0.1	66.8
6	✓	✓	$0.5 \rightarrow 0.01$	**67.3**

Table 4: Ablation study. We ablate the components in SVIB: the re-initialization operation, IB and the hyper-parameter β. Models are run for 100 epochs with ViT-S/16. We report the kNN 20 result.

4.4 Ablation study

In this section, we explore the role of each component in SVIB through ablation experiments. We discuss the impact of β and the re-initialization operation here and reveal their impact by removing them respectively or set various values.

We use ViT-S/16 as encoder and train the encoder on ImageNet for 100 epochs with a batch size of 512. As Table 4 shows, the re-initialization operation help model avoids trivial solution when β is small (Task 5 and Task 2) and brings performance improvement when using the same value of β (Task 3 and Task 4). We further illustrate the performance of models at 20, 40, 60, 80, and 100 epochs under different β in Fig 3, and the result shows that a large β is beneficial to model in the early phase, and a smaller β works better in the middle and later periods. An interpretation to it is that there is much redundancy in the representations in the early phase, thus a large β can help the model get rid of it. However, when the model is going to converge, a strong constraint on IB will force the model to discard part of semantic information and result in performance degradation. Therefore, we apply an exponentially decaying β in our experiments. Specifically, we initialize $\beta = 0.5$ at the initial stage and exponentially decrease it to 0.01 at the end. As presented in Fig 3, this strategy has

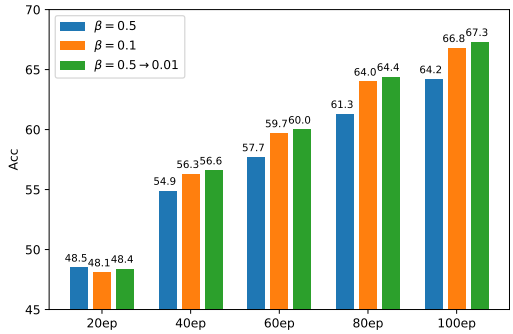

Figure 3: The performance in each epoch with different β

Figure 4: Visualization of attention maps from DINO and SVIB.

similar performance to large β in the early stage, and also has the growth rate of small β in the later training, ultimately achieving the best performance.

4.5 Visualization

We visualize the attention maps of DINO and SVIB in Fig 4. As shown in the figure, the attention map of SVIB is more compact and localized, and SVIB can better concentrate on independent semantic units than DINO.

5 Conclusion

Redundancy is naturally present between positive samples and is rarely mentioned in existing methods. The redundancy will impair the representation and we attempt to reduce it from the perspective of variational information bottleneck. We believe that a good representation should have minimal entropy, just as the process of human learning is forgetting. Thus, redundant information is discarded before semantic information in the entropy reduction process. So our information bottleneck can reduce the redundancy between positive samples and induce the model to learn a better representation. This is also demonstrated by our experimental results and visualization results. Methodologically, SVIB can be extended in many ways, such as different parameterization forms to the feature distribution and an adaptive β. More importantly, however, SVIB provides a question for self-supervised learning. whether the consistency we have relied on so far will be a bottleneck in performance at present? It is worthwhile to continue exploring.

REFERENCES

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *ICLR (Poster)*, 2016.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, volume 32, pp. 15509–15519, 2019.

Qi Cai, Yu Wang, Yingwei Pan, Ting Yao, and Tao Mei. Joint contrastive learning with infinite possibilities. *arXiv preprint arXiv:2009.14776*, 2020.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Thirty-fourth Conference on Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 9912–9924, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV 2021 - International Conference on Computer Vision*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML 2020: 37th International Conference on Machine Learning*, volume 1, pp. 1597–1607, 2020.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020.

Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1422–1430, 2015.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR 2021: The Ninth International Conference on Learning Representations*, 2021.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018.

Boris Ginsburg, Igor Gitman, and Yang You. Large batch training of convolutional networks with layer-wise adaptive rate scaling. 2018.

Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21271–21284, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9729–9738, 2020.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48.

Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. 2018.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6707–6717, 2020.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pp. 69–84, 2016.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2536–2544, 2016.

A. Norman Redlich. Redundancy reduction as a strategy for unsupervised learning. *Neural Computation*, 5(2):289–304, 1993.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *ECCV (11)*, volume 12356 of *Lecture Notes in Computer Science*, pp. 776–794. Springer, 2020. ISBN 978-3-030-58621-8.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv: Learning*, 2018.

Yulin Wang, Gao Huang, Shiji Song, Xuran Pan, Yitong Xia, and Cheng Wu. Regularizing deep networks with semantic data augmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Mike Wu, Chengxu Zhuang, Milan Mosse, Daniel Yamins, and Noah Goodman. On mutual information in contrastive learning for visual representations. *arXiv preprint arXiv:2005.13149*, 2020.

Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

Mang Ye, Xu Zhang, Pong C. Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, pp. 6210–6219. Computer Vision Foundation / IEEE, 2019.

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *ICCV*, pp. 1476–1485, 2019.

Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, pp. 649–666, 2016.

Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6002–6012, 2019.