Information-Theoretic Generative Model Compression with Variational Energy-based Model

Anonymous Author(s)
Affiliation
Address
email

Abstract

We propose an information-theoretic knowledge distillation approach for the compression of generative adversarial networks, which aims to maximize the mutual information between the teacher and student networks via a variational optimization based on an energy-based model. Because the direct computation of the mutual information in continuous domains is intractable, our approach alternatively optimizes the student network by maximizing the variational lower bound on the mutual information. To achieve a tight lower bound, we introduce an energy-based model relying on a deep neural network to represent a flexible variational distribution that deals with high-dimensional images and consider spatial dependencies between pixels, effectively. Since the proposed method is a generic optimization algorithm, it can be conveniently incorporated into arbitrary generative adversarial networks and even dense prediction networks, e.g., image enhancement models. We demonstrate that the proposed algorithm achieves outstanding performance in model compression of generative adversarial networks consistently when combined with several existing models.

1 Introduction

Generative adversarial networks (GANs) [1] have accomplished impressive achievements on various computer vision tasks including image synthesis [2, 3, 4, 5], image-to-image translation [6, 7, 8, 9], video prediction [10, 11, 12], and many others. However, despite the outstanding performance, their applicability to resource-hungry systems, *e.g.*, , edge or mobile devices, is limited due to their huge computational and energy costs. To deal with the issues, GAN compression, a task reducing the sizes of generator networks, has recently drawn attention to many machine learning researchers.

The straightforward way to compress generators is to directly employ traditional network compression techniques such as low-rank approximation [13, 14, 15], network quantization [16, 17, 18, 19], pruning [20, 21, 22, 23, 24], and knowledge distillation [25, 26, 27, 28, 29]. However, the compression of generative models typically deals with a higher-dimensional output space and requires to match the output distributions between the original and compressed models instead of the output instance pair given by the two models. Hence, as reported in [30, 31, 32], the synthesized images obtained from the compressed generators via the channel pruning [33, 22, 23] are often suboptimal—producing images with artifacts and suffering from image quality degradation, which implies that naïve application of the methods designed for discriminative models is not effective for GAN compressions.

To deal with the issue, existing approaches for compressing GANs [30, 34, 31, 35, 36, 37, 32, 38] utilize knowledge distillation to achieve competitive accuracy of a student model. Specifically, these methods transfer the knowledge of a teacher network to a student model via intermediate representations, final outputs, and embedded features [39, 40]. For the purpose, the existing algorithms

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

employ ℓ_1 or ℓ_2 norms to compute the distance between the representations of the two networks, but they are not necessarily effective for minimizing the distance between the two distributions.

We propose to compress GAN models by maximizing the mutual information for knowledge distillation between the outputs of a teacher and a students. For the compression of generative models, the use of the mutual information between two random variables makes more sense than the direct comparison between two outputs, which is the standard way to transfer knowledge from a teacher to a student in discriminative models. However, because the computation of the mutual information in continuous domain is not tractable in general, we employ a variational approach [41], where the lower bound of the mutual information is alternatively optimized using a variational distribution, an approximation for the conditional distribution of the teacher output given the student output. We further adopt an energy-based model for the flexible representation of the variational distribution and tighten the lower bound for the effective knowledge transfer. Note that the proposed energy-based model is convenient to deal with high-dimensional data (images) and effective to handle spatial dependency between pixels within an image. Combined with the existing approaches, our framework achieves substantial performance improvement on GAN compression. The main contributions of our work are summarized below:

- We propose a novel energy-based framework of GAN compression via knowledge distillation, which maximizes mutual information between a teacher and a student network.
- The proposed energy-based approach is effective to represent a flexible variational distribution and can be incorporated into many GAN compression methods due to its generality.
- Experimental results show that the proposed method can improve the performance when combined with the existing methods.

The rest of the paper is organized as follows. Section 2 discusses the related work to GAN compression. The details of our approach is described in Section 3 and 4, and the experimental results are presented in Section 5. Finally, we conclude this paper in Section 6.

2 Related Work

2.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) [1] consist of two components, a generator and a discriminator, and train the models using an adversarial loss; the generator aims to fool the discriminator by synthesizing realistic images from random noise samples while the discriminator wants to distinguish the real data from the fake ones. Unlike the vanilla GANs, conditional GANs [42, 6, 7] can control image generation conditioned on inputs such as a class label [42] or an image translated into another image [6, 7] (*e.g.*, Horse $\rightarrow$ Zebra). The conditional GANs are not confined to the conditional image generation tasks, but applied to various areas such as super-resolution [43], text-to-image [44], and image inpainting [45]. Since the conditional GAN models are typically large in terms of FLOPs, the proposed GAN compression algorithm is mainly tested on them but we also verify the effectiveness of our approach on the unconditional counterpart.

2.2 Knowledge Distillation

Knowledge distillation techniques aim to transfer knowledge from a powerful network (*i.e.*, teacher) to a compact network (*i.e.*, student) by matching logits [46], output distributions [25], intermediate activations [47], or attention maps [26]. Specifically, Hinton *et al*. [25] learn the representation of student by minimizing the relative entropy from its normalized softmax output to that of teacher while [46, 47] minimize the mean squared error between the representations of the teacher student networks. Both Variational Information Distillation (VID) [48] and Contrastive Representation Distillation (CRD) [49] employ the mutual information between teacher and student to estimate the similarity of the representations from the two networks. The former relies on a variational approach [41] for the tractable computation of mutual information between the two networks. On the other hand, the latter derives an InfoNCE-like loss function from the lower bound of the mutual information to learn the representation of a student. However, VID maximizes the variational lower bound using a strong assumption about the variational distribution, a fully factorized Gaussian distribution, and CRD is difficult to be employed to generative models due to its contrastive learning objective, which is prone

to be trivial in image generation tasks without semantic attribute annotations, *e.g.*, class labels. Our algorithm is related to VID and CRD since they all attempt to maximize the mutual information between teacher and student networks, but has clear advantage over them in terms of the flexibility of the variational distribution and the applicability to generative models.

2.3 GAN Compression

GAN compression methods typically aim to reduce the size of the generator without considering the size of the discriminator since it is not used for inference. Co-Evolution [30] utilizes an evolutionary algorithm [50] to prune the channel size of CycleGAN based on a fitness formulated based on FLOPs and the reconstruction error between the original and pruned generators. GAN-Compression [34] employs a once-for-all network training scheme [51] combined with the reconstruction loss as well as the intermediate feature distillation loss [47], and selects the best performing subnetwork given the target budget. Compression and Teaching (CAT) [37] designs an efficient building block in the generator, and prunes the network using the magnitude-based pruning method [23] while maximizing the similarity between the two networks based on centered kernel alignment [52]. Content-Aware GAN Compression (CAGC) [32] uses the feature distillation and reconstruction loss based only on the object part of a generated image while additionally considering LPIPS [40] as a perceptual distance for the synthesized ones. On the contrary, Online Multi-Granularity Distillation (OMGD) [38] iteratively trains a teacher network with original conditional GAN objectives and a student network using the total variation loss for spatial smoothness and a distillation loss that makes the student mimic the final outputs and channel-wise attention weights for the intermediate feature maps of the teacher networks. Generator-discriminator Cooperative Compression (GCC) [53] also adopts the online distillation scheme while selecting channels for the original discriminator to match the reduced capacity of the student generator at each iteration. To validate the effectiveness of our framework, we present how the proposed algorithm can improve performance via the combinations with existing methods such as OMGD, GCC, GAN-Compression, CAT, and CAGC.

3 Variational GAN Compression with Energy-Based Model

Our goal is to compress generators in GANs through knowledge distillation. Let S and T be random variables of the outputs from a student, $G^s(\cdot;\phi^s) : \mathcal{X} \rightarrow \mathcal{Y}$ and a teacher, $G^t(\cdot;\phi^t) : \mathcal{X} \rightarrow \mathcal{Y}$, respectively, where ϕ^s and ϕ^t are their model parameters. For knowledge transfer from the teacher to the student, we aim to maximize mutual information between T and S, $I(T;S)$, which is given by

$$I(T;S) = D_{\mathrm{KL}}\Big(p\big(\boldsymbol{t},\boldsymbol{s}\big)\,||\,p\big(\boldsymbol{t}\big)\cdot p\big(\boldsymbol{s}\big)\Big) = H(T) + \mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[\log p(\boldsymbol{t}|\boldsymbol{s})], \tag{1}$$

where $D_{\mathrm{KL}}(\cdot||\cdot)$ denotes the Kullback-Leibler (KL) divergence and $H(\cdot)$ means the entropy function. Note that all the expectations are taken over the joint data distribution, $p(\boldsymbol{t},\boldsymbol{s})$, where $\boldsymbol{t}$ and $\boldsymbol{s}$ indicate teacher and student output distributions, respectively.

The practical challenge in this objective is that the computation of the mutual information for multivariate random variables is typically intractable. To tackle the issue, a variational method [41] derives the lower bound on the mutual information, $\tilde{I}(T;S)$, using a variational distribution, $q(\boldsymbol{t}|\boldsymbol{s})$, which is given by

$$\begin{aligned} I(T;S) &= H(T) + \mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[\log q(\boldsymbol{t}|\boldsymbol{s})] + \mathbb{E}_{\boldsymbol{s}}[D_{\mathrm{KL}}(p(\boldsymbol{t}|\boldsymbol{s})||q(\boldsymbol{t}|\boldsymbol{s}))] \\ &\geq H(T) + \mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[\log q(\boldsymbol{t}|\boldsymbol{s})] \\ &:= \tilde{I}(T;S). \end{aligned} \tag{2}$$

To maximize the mutual information, we aim to increase the variational lower-bound by minimizing the KL-divergence between the true and variational distributions, which is defined as

$$\triangle I(T;S) := I(T;S) - \tilde{I}(T;S) = \mathbb{E}_{\boldsymbol{s}}[D_{\mathrm{KL}}(p(\boldsymbol{t}|\boldsymbol{s})||q(\boldsymbol{t}|\boldsymbol{s}))]. \tag{3}$$

Variational Information Distillation (VID) [48] optimizes the variational lower bound by employing the fully factorized Gaussian distribution as the variational distribution $q(\boldsymbol{t}|\boldsymbol{s})$. However, as noted in [54], the statistics of the natural images usually exhibit non-Gaussian property. Furthermore, optimizing the variational lower bound using the fully factorized distribution encourages the student

to generate the blurry outputs [55, 45, 56]. On the other hand, we employ an energy-based model as a flexible distribution for the variational distribution, which is given by

$$q_\theta(\boldsymbol{t}|\boldsymbol{s}) = \frac{1}{Z_\theta(\boldsymbol{s})} \cdot \exp\Big(- E_\theta\big(\boldsymbol{t}, \boldsymbol{s}\big)\Big), \tag{4}$$

where $E_\theta(\boldsymbol{t}, \boldsymbol{s}) : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is the energy function defined by a deep neural network with parameter θ and $Z_\theta(\boldsymbol{s})$ is the partition function. The variational distribution given by the energy-based model is effective for the student model to maximize the variational lower bound. In addition, contrary to VID [48], our approach employs the variational distribution that considers the spatial dependency among pixels in $\boldsymbol{t}$ and $\boldsymbol{s}$.

Overall, we optimize the energy-based model, the student network, and the teacher network, which are achieved by the following iterative procedure:

1. **Fix $\{\phi^s, \phi^t\}$ and optimize θ.**
   Under fixed parameters $\{\phi^s, \phi^t\}$, minimize the objective (3) with respect to θ.
2. **Fix $\{\theta, \phi^t\}$ and optimize ϕ^s.**
   Under fixed parameters $\{\theta, \phi^t\}$, maximize the objective (2) while minimizing standard knowledge distillation losses with respect to ϕ^s.
3. **Optimize ϕ^t.**
   Minimize the standard conditional GAN loss [6, 7] with respect to ϕ^t using a training dataset. This step is only required for online knowledge distillation. Otherwise, the teacher model should be learned offline in advance.

These steps are iterated until convergence.

4 Training

This section discusses our optimization procedure based on knowledge distillation, which iteratively maximizes the mutual information between a teacher and a student using the energy-based model.

4.1 Optimization for Energy-based Model

We now discuss how to optimize the energy-based model for minimizing $\mathbb{E}_{\boldsymbol{s}}[D_{\mathrm{KL}}(p(\boldsymbol{t}|\boldsymbol{s})||q_\theta(\boldsymbol{t}|\boldsymbol{s}))]$ with respect to θ given the fixed model parameters $\{\phi^s, \phi^t\}$.

4.1.1 Objective

Since the conditional data distribution $p(\boldsymbol{t}|\boldsymbol{s})$ is constant with respect to the model parameter θ, the energy-based model minimizes the following objective:

$$\min_\theta \triangle I(T; S) \iff \min_\theta \mathbb{E}_{\boldsymbol{s},\boldsymbol{t}}[-\log q_\theta(\boldsymbol{t}|\boldsymbol{s})]. \tag{5}$$

To optimize the energy-based model using the standard stochastic gradient descent method, we derive the gradient of the objective in (5), which is given by

$$\nabla_\theta \mathbb{E}_{\boldsymbol{s},\boldsymbol{t}}[-\log q_\theta(\boldsymbol{t}|\boldsymbol{s})] = \mathbb{E}_{\boldsymbol{s},\boldsymbol{t}}\Big[\nabla_\theta E_\theta(\boldsymbol{t}, \boldsymbol{s})\Big] - \mathbb{E}_{\boldsymbol{s}}\mathbb{E}_{\tilde{\boldsymbol{t}}\sim q_\theta(\boldsymbol{t}|\boldsymbol{s})}\Big[\nabla_\theta E_\theta(\tilde{\boldsymbol{t}}, \boldsymbol{s})\Big], \tag{6}$$

where the second expectation requires sampling from $q_\theta(\boldsymbol{t}|\boldsymbol{s})$ via a Markov Chain Monte Carlo (MCMC) method such as Gibbs sampling.

4.1.2 Improved Sampling for MCMC

The main drawback of the MCMC methods is high computational cost, so we rely on Langevin dynamics, which leads to the following recursive sample generation:

$$\tilde{\boldsymbol{t}}^k = \tilde{\boldsymbol{t}}^{k-1} - \frac{\lambda}{2}\nabla_{\boldsymbol{t}} E_\theta(\tilde{\boldsymbol{t}}^{k-1}, \boldsymbol{s}) + w^k, \tag{7}$$

where w^k is a sample drawn from $\mathcal{N}(0, \lambda I)$ and λ is a fixed step size. Theoretically, $\tilde{\boldsymbol{t}}^k$ becomes identical to a sample drawn from $q_\theta(\boldsymbol{t}|\boldsymbol{s})$ as $k \to \infty$ and $\lambda \to 0$ [57]. Following Short-Run MCMC [58], we also run K-step MCMC starting from an initial distribution, q_0, to approximate samples from $q_\theta(\boldsymbol{t}|\boldsymbol{s})$, where K is the number of MCMC steps.

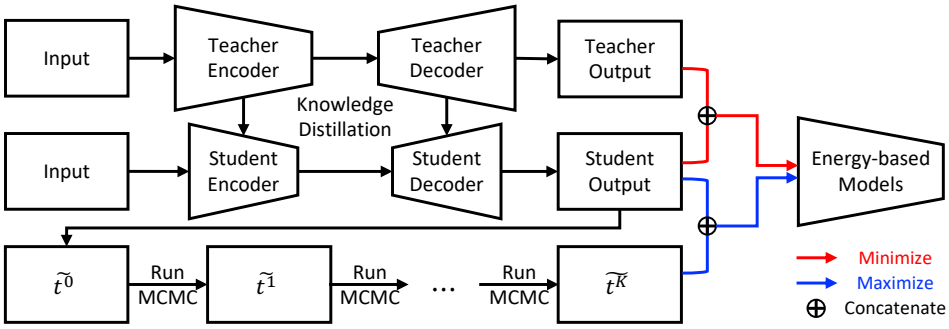

Figure 1: Overview of our framework for maximizing the variational lower bound with respect to the student model.

4.1.3 Initial Distribution

We initialize $\tilde{\boldsymbol{t}}^0$ as the output of the student network for the acceleration of the MCMC optimization via the Langevin sampling process. This initialization can be a good approximation to the model distribution. Intuitively, since the student model is supposed to mimic the samples drawn from the data distribution, the optimization of $q_\theta(\boldsymbol{t}|\boldsymbol{s})$ towards the data distribution $p(\boldsymbol{t}|\boldsymbol{s})$ from the current student output would be more efficient. As a result, the proposed initialization strategy often results in improved performance of student models compared to existing methods [58, 59, 60, 61, 62] even with a small number of MCMC steps. Note that Contrastive Divergence (CD) [59] uses the data distribution for the initialization of q_0 while Persistent Contrastive Divergence (PCD) [60, 61] utilizes the samples collected in the previous epoch and [58] initializes the uniform noise distribution. Since the data distribution is prone to induce bias in the CD estimate [62] and the random uniform distribution is far from the data distribution, the existing methods require many steps for MCMC and increase training time [58]. In addition, PCD updates all samples in the memory buffer, which leads to significantly increased memory overhead especially for generation tasks.

4.1.4 Inference

In contrast to the previous approaches for energy-based models [58, 62, 61], we do not require any MCMC sampling for inference but generate images via a single forward process of the student generator. Therefore, the proposed algorithm incur no extra computational cost at inference time.

4.2 Optimization for Student Generator

4.2.1 Objective for Mutual Information

The student generator learns to maximize the variational lower bound $\tilde{I}(T;S)$ in (2) with respect to ϕ_s, where the objective function is formally given by

$$\max_{\phi^s} \tilde{I}(T;S) \iff \max_{\phi^s} H(T) + \mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[\log q_\theta(\boldsymbol{t}|\boldsymbol{s})] \iff \max_{\phi^s} \mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[\log q_\theta(\boldsymbol{t}|\boldsymbol{s})]. \tag{8}$$

Note that the entropy, $H(T)$, is independent of the parameters in the student network and the objective is equivalent to maximize the expected log likelihood as shown in the above equation. Since the joint distribution $p(\boldsymbol{t},\boldsymbol{s})$ depends on ϕ_s, we employ a reparameterization trick using the function G^s. Then, the gradient of the objective function in (8) with respect to ϕ_s is derived as follows:

$$\begin{aligned}&\nabla_{\phi^s}\mathbb{E}_{\boldsymbol{t},\boldsymbol{s}}[-\log q_\theta(t|s)]\\ &\quad= \mathbb{E}_{\boldsymbol{x}}\Big[\nabla_{\phi^s}E_\theta\big(G^t(\boldsymbol{x};\phi^t),G^s(\boldsymbol{x};\phi^s)\big)\Big] - \mathbb{E}_{\boldsymbol{x}}\mathbb{E}_{\tilde{\boldsymbol{t}}\sim q_\theta\left(\boldsymbol{t}|G^s(x;\phi^s)\right)}\Big[\nabla_{\phi^s}E_\theta(\tilde{\boldsymbol{t}},G^s\big(\boldsymbol{x};\phi^s)\big)\Big].\end{aligned} \tag{9}$$

Note that we do not need to run MCMC additionally to obtain the samples from $q_\theta(\boldsymbol{t}|G^s_{\phi_s}(x))$ because they are already available from the training procedure for the energy-based model. Figure 1 illustrates the overview of our framework for maximizing the mutual information.

Table 1: Performance of VEM in comparison with the state-of-the-art methods for Pix2Pix model using the U-Net baseline on the Edges $\rightarrow$ Shoes and Cityscapes datasets. Methods with an asterisk (*) are our reproduction given by running the official codes.

Dataset	Method	MACs	# of Parameters	FID (↓)	mIoU
Edges $\rightarrow$ Shoes	Original [6]	18.60G (1.0×)	54.40M (1.0×)	34.31	-
	DMAD [36]	2.99G (6.2×)	2.13M (25.5×)	46.95	-
	OMGD* [38]	1.22G (15.3×)	3.40M (16.0×)	27.39	-
	OMGD* [38] + VID [48]	1.22G (15.3×)	3.40M (16.0×)	28.02	-
	OMGD* [38] + VEM (Ours)	1.22G (15.3×)	3.40M (16.0×)	**24.59**	-
Cityscapes	Original [6]	18.60G (1.0×)	54.40M (1.0×)	-	42.71
	DMAD [36]	3.96G (4.7×)	1.73M (31.4×)	-	40.53
	GCC [53]	3.09G (6.0×)	-	-	42.88
	OMGD* [38]	1.22G (15.3×)	3.40M (16.0×)	-	47.52
	OMGD* [38] + VID [48]	1.22G (15.3×)	3.40M (16.0×)	-	47.79
	OMGD* [38] + VEM (Ours)	1.22G (15.3×)	3.40M (16.0×)	-	**49.39**

4.2.2 Objectives Specialized for Compression Algorithms

The proposed variational lower bound maximization framework based on the energy-based model is incorporated into distillation methods such as OMGD [38], GCC [53], GAN-Compression [34], CAT [37], and CAGC [32]. The brief descriptions about these methods are presented in Section 2.3, and we will discuss the details of the algorithm-specific losses in the supplementary document.

4.2.3 Full Objective

The full objective function for the student network, which is given by

$$\mathcal{L}^s = \mathcal{L}_{\text{algo}} - \lambda_{\text{MI}} \cdot \tilde{I}(T; S), \tag{10}$$

where $\mathcal{L}_{\text{algo}}$ is an algorithm-specific loss, λ_{MI} is a hyperparameter, and $\tilde{I}(T; S)$ is the variational lower bound defined in (8).

4.3 Optimization for Teacher Generator

The teacher generator is trained using the standard methods of backbone algorithms. Optionally, the teacher model learns to minimize the ℓ_1 distance between the generated image and the ground-truth when the ground-truth is available. In the case of the unsupervised image-to-image translation tasks, the cycle consistency loss [7] based on the ℓ_1 distance enforces the instance mapping between two domains. To sum up, the teacher network is optimized by the adversarial loss, denoted by the GAN loss, $\mathcal{L}_{\text{GAN}}$, and the reconstruction loss, $\mathcal{L}_1$, as follows:

$$\min_{G^t} \max_{D^t} \mathcal{L}_{\text{GAN}} + \lambda_1 \mathcal{L}_1, \tag{11}$$

where λ_1 is a hyperparameter to balance the two loss terms. Note that, for unconditional GANs, there is nothing to reconstruct and we train the network based only on the GAN loss.

5 Experiments

To validate the effectiveness of the proposed compression algorithm based on variational energy-based model, denoted by VEM, we conduct extensive experiments on the standard datasets using various architectures.

5.1 Datasets and Models

Image-to-image translation We adopt Pix2Pix [6] on the Edge $\rightarrow$ Shoes [64] and Cityscapes [65] datasets while taking CycleGAN [7] on Horse $\rightarrow$ Zebra [7] and Summer $\rightarrow$ Winter [7]. Note that Pix2Pix is the image-to-image translation network for paired datasets while CycleGAN is designed for unpaired image-to-image translation. Following the original models [6, 7], we employ a U-Net-based generator [66] for Pix2Pix and a ResNet-based generator [67] for CycleGAN. For all the datasets, images are resized to 256×256 before feeding them into the models.

Table 2: Performance comparison between VEM and the state-of-the-art methods in CycleGAN model based on the ResNet backbones.

Dataset	Method	MACs	# of Parameters	FID (↓)
Horse → Zebra	Original [7]	56.80G (1.0×)	11.30M (1.0×)	61.53
	Co-Evolution [30]	13.40G (4.2×)	-	96.15
	GAN-Slimming [35]	11.3G (23.6×)	-	86.09
	Auto-GAN-Distiller [31]	6.39G (8.9×)	-	83.60
	GAN-Compression [34]	2.67G (21.3×)	0.34M (33.2×)	64.95
	DMAD [36]	2.41G (23.6×)	0.2M (40.0×)	62.96
	CAT [37]	2.55G (22.3×)	-	60.18
	GCC [53]	2.40G (23.7×)	-	59.31
	OMGD* [38]	1.41G (40.3×)	0.14M (82.5×)	57.14
	OMGD* [38] + VEM (Ours)	1.41G (40.3×)	0.14M (82.5×)	**50.83**
Summer → Winter	Original [7]	56.80G (1.0×)	11.30M (1.0×)	79.12
	Co-Evolution [30]	11.10G (5.1×)	-	78.58
	Auto-GAN-Distiller [31]	4.34G (13.1×)	-	78.33
	DMAD [36]	3.18G (17.9×)	0.30M (37.7×)	78.24
	OMGD* [38]	1.41G (40.3×)	0.14M (82.5×)	75.20
	OMGD* [38] + VEM (Ours)	1.41G (40.3×)	0.14M (82.5×)	**74.04**

Table 3: Performance comparison between VEM and the existing compression algorithms for unconditional GANs on the CelebA and FFHQ datasets using SAGAN and StyleGAN2, respectively.

Dataset	Model	Method	MACs	Compression Ratio	FID (↓)
CelebA	SAGAN [4]	Original	23.45M	-	24.87
		Slimming [23]	15.45M	34.12%	36.60
		GCC* [53]	15.45M	34.12%	27.91
		GCC* [53] + VID [48]	15.45M	34.12%	26.57
		GCC* [53] + VEM (Ours)	15.45M	34.12%	**25.27**
FFHQ	StyleGAN2 [63]	Original	45.1B	-	4.50
		GAN-Slimming [35]	5.0B	88.91%	12.40
		CAGC [32]	4.1B	90.91%	7.90
		CAGC [32] + VEM (Ours)	4.1B	90.91%	**7.48**

Image generation To evaluate the performance of our approach for unconditional GANs, we employ StyleGAN2 [63] on the Flickr-Faces-HQ (FFHQ) dataset [68] and Self-attention GAN (SAGAN) [4] on the CelebA dataset [69]. For image preprocessing, we resize images to 64×64 for CelebA and 256×256 for FFHQ datasets.

5.2 Evaluation Metrics

Following the protocol of the previous approaches, we report the FID scores on the Edge → Shoes, Horse → Zebra, Summer→Winter, CelebA, and FFHQ datasets while mIoU is selected for the Cityscapes dataset. The FID score is widely utilized for assessing the quality of the generated images by measuring the distance between the distributions of embedding features extracted from generated images and real images, which are given by the Inception V3 model [70] pretrained on ImageNet. In the case of the Cityscapes dataset, we adopt DRN-D-105 [71] as a segmentation model to derive the mIoU scores for the generated images.

5.3 Implementation Details

The proposed algorithm is implemented in PyTorch [72] based on the publicly available code. We incorporate the proposed mutual information maximization framework into existing approaches including GAN-Compression, CAT, CAGC, and GCC to validate its benefit and generality. For fair comparisons, we do not modify any experimental settings and network structures of the compared algorithms. Specifically, we set the batch size to 4 for Pix2Pix, 1 for CycleGAN, 64 for SAGAN, and 16 for StyleGAN2 while we adopt the Adam optimizer with an initial learning rate of 0.0002, which is linearly decayed to zero.

When we train our models integrated with the energy-based model, we employ the Adam optimizer with a learning rate of 0.0001. Note that we only run 10 steps of Langevin dynamics to draw samples

Table 4: Ablation study about the approaches maximizing mutual information. We compare the methods relying on the proposed energy-based model (VEM) with the approaches based on the fully-factorized Gaussian distribution (VID) and the InfoNCE-like contrastive-loss (CRD). This experiment is conducted on the Horse $\rightarrow$ Zebra dataset using CycleGAN.

Dataset	Method	MACs	#Parameters	FID (↓)
Horse $\rightarrow$ Zebra	OMGD* [38]	1.41G (40.3×)	0.14M (82.5×)	57.14
	OMGD* [38] + VID [48]	1.41G (40.3×)	0.14M (82.5×)	65.73
	OMGD* [38] + CRD [49]	1.41G (40.3×)	0.14M (82.5×)	70.59
	OMGD* [38] + VEM (Ours)	1.41G (40.3×)	0.14M (82.5×)	**50.83**
	CAT* [37]	2.56G (22.3×)	0.43M (26.3×)	64.79
	CAT* [37] + VID [48]	2.56G (22.3×)	0.43M (26.3×)	68.71
	CAT* [37] + CRD [49]	2.56G (22.3×)	0.43M (26.3×)	67.69
	CAT* [37] + VEM (Ours)	2.56G (22.3×)	0.43M (26.3×)	**52.44**
	GAN-Compression* [34]	2.55G (22.3×)	0.51M (22.2×)	59.13
	GAN-Compression* [34] + VID [48]	2.55G (22.3×)	0.51M (22.2×)	62.36
	GAN-Compression* [34] + CRD [49]	2.55G (22.3×)	0.51M (22.2×)	60.27
	GAN-Compression* [34] + VEM (Ours)	2.55G (22.3×)	0.51M (22.2×)	**50.01**

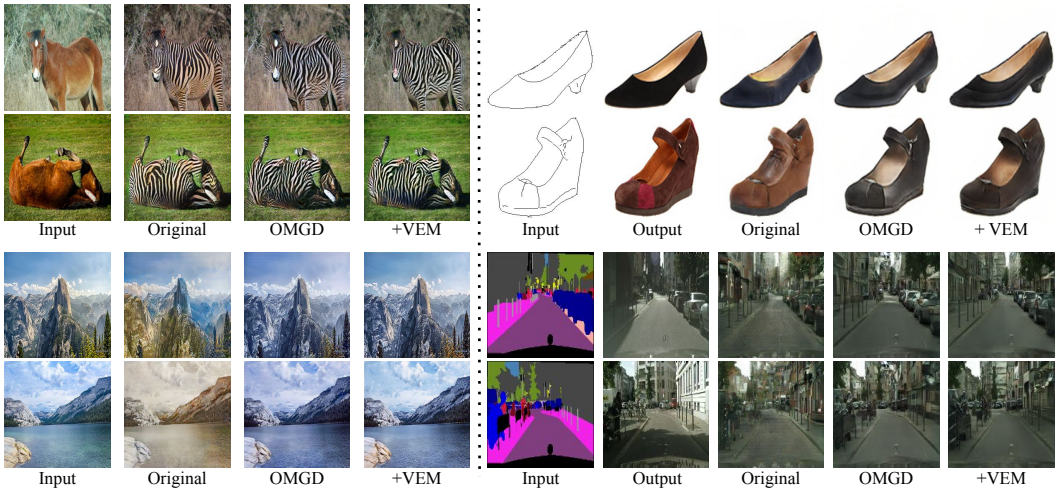

Figure 2: Qualitative results of CycleGAN and Pix2Pix on Horse $\rightarrow$ Zebra, Summer $\rightarrow$ Winter, Edges $\rightarrow$ Shoes, and Cityscapes datasets. Note that "output" denotes the ground-truth and "original" represents the generator without compression for Pix2Pix and CycleGAN.

from the energy-based model for saving training time. We set the standard deviation of random noise to 0.005 and use a step size of 100 for each gradient step of Langevin dynamics for all datasets except for the Cityscapes and FFHQ datasets. For the two datasets, we use a step size of 50 instead. Details about the hyperparameter setting will be provided in the supplementary document. In the case of the paired datasets such as Edges $\rightarrow$ Shoes and Cityscapes, we maximize the mutual information between the outputs given by the student model and the real outputs instead of the teacher outputs.

5.4 Results

Pix2Pix Table 1 presents the results from Pix2Pix on the Edges $\rightarrow$ Shoes and Cityscapes datasets by the proposed algorithm using the U-Net backbone in comparison to the original network and recent methods including OMGD and OMGD + VID. As shown in the table, OMGD + VEM achieves the best performance in terms of FID and MAC among all compared algorithms on Edges $\rightarrow$ Shoes. On Cityscapes, OMGD + VEM presents the highest mIoU by large margins even with the highest compression rate. Note that we report the results from OMGD based on our reproduction using the official code, which are denoted by asterisks (*) in all tables.

CycleGAN We test our compression algorithm using CycleGAN on Horse $\rightarrow$ Zebra and Summer $\rightarrow$ Winter. OMGD + VEM outperforms all the compared methods on the both datasets, especially by large margins in Horse $\rightarrow$ Zebra.

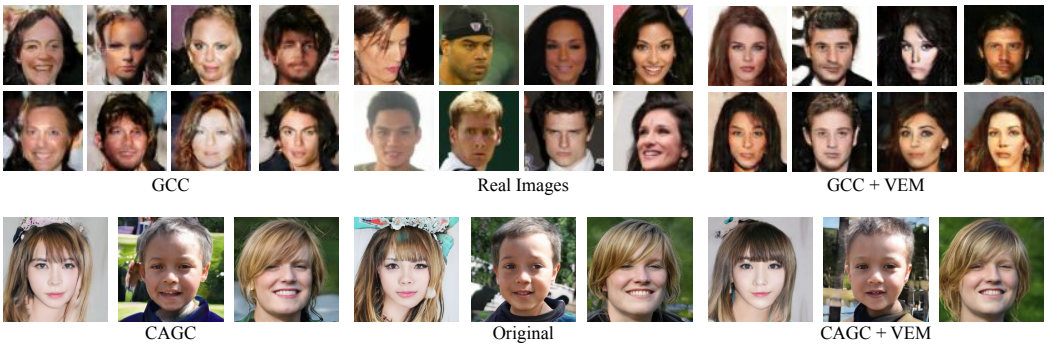

Figure 3: Qualitative results of SAGAN on the CelebA dataset (top) and StyleGAN2 on the FFHQ dataset (bottom). Note that “real images" denotes data samples in CelebA while “original" represents the images generated by uncompressed StyleGAN2.

Unconditional GANs We compress SAGAN and StyleGAN2 using GCC [53] and CAGC [32], respectively. Table 3 illustrates the proposed method is also effective to improve performance of the unconditional GAN compression techniques. In particular, our approach works well with a large-scale generator such as StyleGAN2.

5.5 Ablation Study about Mutual Information Maximization Techniques

For maximizing mutual information with the teacher and student networks, one can use a fully-factorized Gaussian distribution for the variational distribution followed by VID [48] or CRD [49] employing InfoNCE-like contrastive loss. To validate the effectiveness of our energy-based models compared with VID and CRD, we perform an ablation study to combine with the state-of-the-art GAN compression approaches such as GAN-Compression [34], CAT [37], and OMGD [38].

Table 4 demonstrates that VEM outperforms VID and CRD significantly in all the tested algorithms. This is because, compared to VID, we derive a tight lower bound of the mutual information theoretically and consider spatial dependencies among pixels thanks to the more flexible variational distribution based on the energy function. Also, note that the application of CRD to generative models is not effective since the instance-level contrastive loss is prone to be trivial due to large gaps between positive and negative pairs.

5.6 Qualitative Results

Figure 2 visualizes generated images given by the original uncompressed backbones and the compressed counterparts given by OMGD and OMGD + VEM (ours) with U-Net backbones for Pix2Pix and CycleGAN on the Horse $\rightarrow$ Zebra, Summer $\rightarrow$ Winter, Edges $\rightarrow$ Shoes, and Cityscapes datasets. According to the presented images, our approach is effective to enhance the generation quality. For example, OMGD fails to produce the fine details such as building in the background in the Cityscapes dataset while OMGD + VEM is successful. Figure 3 depicts the generation quality by unconditional GANs on the CelebA and FFHQ datasets. Compared with the uncompressed versions, the proposed method preserves or often improves the generation quality even with large compression ratios. Also, our approach generates visually improved images compared to the previous GAN compression methods such as GCC and CAGC when combined with them.

6 Conclusion

We presented a novel GAN compression approach via knowledge distillation based on the mutual information, which maximizes the variational lower bound by incorporating an energy-based variational distribution. Our energy-based model is straightforward to be incorporated into various existing model compression algorithms for generative models. It provides a more flexible variational distribution, which leads to theoretically tighter lower bounds and facilitates the consideration of the dependency between pixels. The proposed algorithm demonstrates outstanding performance in various GAN compression scenarios on multiple datasets compared to the state-of-the-art algorithms.

References

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In NIPS. (2014)

[2] Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein Generative Adversarial Networks. In ICML. (2017)

[3] Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral Normalization for Generative Adversarial Networks. In ICLR. (2018)

[4] Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-Attention Generative Adversarial Networks. In ICML. (2019)

[5] Brock, A., Donahue, J., Simonyan, K.: Large Scale GAN Training for High Fidelity Natural Image Synthesis. In ICLR. (2019)

[6] Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-Image Translation with Conditional Adversarial Networks. In CVPR. (2017)

[7] Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In ICCV. (2017)

[8] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.N.: Stackgan++: Realistic Image Synthesis with Stacked Generative Adversarial Networks. TPAMI (2018)

[9] Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic Image Synthesis with Spatially-Adaptive Normalization. In CVPR. (2019)

[10] Vondrick, C., Pirsiavash, H., Torralba, A.: Generating Videos with Scene Dynamics. In NIPS. (2016)

[11] Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: MoCoGAN: Decomposing Motion and Content for Video Generation. In CVPR. (2018)

[12] Xu, T., Wenliang, L.K., Munn, M., Acciaio, B.: COT-GAN: Generating Sequential Data via Causal Optimal Transport. In NeurIPS. (2020)

[13] Denton, E.L., Zaremba, W., Bruna, J., LeCun, Y., Fergus, R.: Exploiting Linear Structure within Convolutional Networks for Efficient Evaluation. In NIPS. (2014)

[14] Jaderberg, M., Vedaldi, A., Zisserman, A.: Speeding up Convolutional Neural Networks with Low Rank Expansions. In BMVC. (2014)

[15] Tai, C., Xiao, T., Zhang, Y., Wang, X., et al.: Convolutional Neural Networks with Low-Rank Regularization. In ICLR. (2016)

[16] Courbariaux, M., Bengio, Y., David, J.P.: Binaryconnect: Training Deep Neural Networks with Binary Weights during Propagations. In NIPS. (2015)

[17] Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A.: XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks. In ECCV. (2016)

[18] Zhu, C., Han, S., Mao, H., Dally, W.J.: Trained Ternary Quantization. In ICLR. (2017)

[19] Polino, A., Pascanu, R., Alistarh, D.: Model Compression via Distillation and Quantization. In ICLR. (2018)

[20] Han, S., Pool, J., Tran, J., Dally, W.: Learning both Weights and Connections for Efficient Neural Network. In NIPS. (2015)

[21] Wen, W., Wu, C., Wang, Y., Chen, Y., Li, H.: Learning Structured Sparsity in Deep Neural Networks. In NIPS. (2016)

[22] Li, Hao and Kadav, Asim and Durdanovic, Igor and Samet, Hanan and Graf, Hans Peter: Pruning Filters for Efficient ConvNets. In ICLR. (2017)

[23] Liu, Z., Li, J., Shen, Z., Huang, G., Yan, S., Zhang, C.: Learning Efficient Convolutional Networks through Network Slimming. In ICCV. (2017)

[24] He, Y., Liu, P., Wang, Z., Hu, Z., Yang, Y.: Filter Pruning via Geometric Median for Deep Convolutional Neural Networks Acceleration. In CVPR. (2019)

[25] Hinton, G., Vinyals, O., Dean, J.: Distilling the Knowledge in a Neural Network. arXiv preprint arXiv:1503.02531 (2015)

[26] Zagoruyko, S., Komodakis, N.: Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In ICLR. (2017)

[27] Passalis, N., Tefas, A.: Learning Deep Representations with Probabilistic Knowledge Transfer. In ECCV. (2018)

[28] Tian, Y., Krishnan, D., Isola, P.: Contrastive Representation Distillation. In ICLR. (2019)

[29] Chen, L., Wang, D., Gan, Z., Liu, J., Henao, R., Carin, L.: Wasserstein Contrastive Representation Distillation. In CVPR. (2021)

[30] Shu, H., Wang, Y., Jia, X., Han, K., Chen, H., Xu, C., Tian, Q., Xu, C.: Co-Evolutionary Compression for Unpaired Image Translation. In ICCV. (2019)

[31] Fu, Y., Chen, W., Wang, H., Li, H., Lin, Y., Wang, Z.: AutoGAN-Distiller: Searching to Compress Generative Adversarial Networks. In ICML. (2020)

[32] Liu, Y., Shu, Z., Li, Y., Lin, Z., Perazzi, F., Kung, S.Y.: Content-Aware GAN Compression. In CVPR. (2021)

[33] Hu, H., Peng, R., Tai, Y.W., Tang, C.K.: Network trimming: A Data-Driven Neuron Pruning Approach towards Efficient Deep Architectures. arXiv preprint arXiv:1607.03250 (2016)

[34] Li, M., Lin, J., Ding, Y., Liu, Z., Zhu, J.Y., Han, S.: Gan Compression: Efficient Architectures for Interactive Conditional GANs. In CVPR. (2020)

[35] Wang, H., Gui, S., Yang, H., Liu, J., Wang, Z.: GAN Slimming: All-in-One GAN Compression by A Unified Optimization Framework. In ECCV. (2020)

[36] Li, S., Lin, M., Wang, Y., Chao, F., Mao, X., Xu, M., Wu, Y., Huang, F., Shao, L., Ji, R.: Learning Efficient GANs for Image Translation via Differentiable Masks and co-Attention Distillation. arXiv preprint arXiv:2011.08382 (2020)

[37] Jin, Q., Ren, J., Woodford, O.J., Wang, J., Yuan, G., Wang, Y., Tulyakov, S.: Teachers Do More Than Teach: Compressing Image-to-Image Models. In CVPR. (2021)

[38] Ren, Y., Wu, J., Xiao, X., Yang, J.: Online Multi-Granularity Distillation for GAN Compression. In ICCV. (2021)

[39] Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In ECCV. (2016)

[40] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In CVPR. (2018)

[41] Barber, D., Agakov, F.: The im algorithm: A Variational Approach to Information Maximization. In NIPS. (2004)

[42] Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)

[43] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-Realistic Single Image Super-Resolution using a Generative Adversarial Network. In CVPR. (2017)

[44] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative Adversarial Text to Image Synthesis. In ICML. (2016)

[45] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context Encoders: Feature Learning by Inpainting. In CVPR. (2016)

[46] Ba, J., Caruana, R.: Do Deep Nets really Need to be Deep? In NIPS. (2014)

[47] Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for Thin Deep Nets. In ICLR. (2015)

[48] Ahn, S., Hu, S.X., Damianou, A., Lawrence, N.D., Dai, Z.: Variational Information Distillation for Knowledge Transfer. In CVPR. (2019)

[49] Tian, Y., Krishnan, D., Isola, P.: Contrasive representation distillation. In ICLR. (2020)

[50] Wang, Y., Xu, C., Qiu, J., Xu, C., Tao, D.: Towards evolutionary compression. In KDD. (2018)

[51] Cai, H., Gan, C., Wang, T., Zhang, Z., Han, S.: Once-for-All: Train One Network and Specialize it for Efficient Deployment. In ICLR. (2019)

[52] Cortes, C., Mohri, M., Rostamizadeh, A.: Algorithms for Learning Kernels based on Centered Alignment. JMLR (2012)

[53] Li, S., Wu, J., Xiao, X., Chao, F., Mao, X., Ji, R.: Revisiting Discriminator in GAN Compression: A Generator-discriminator Cooperative Compression Scheme. In NeurIPS. (2021)

[54] Wainwright, M.J., Simoncelli, E.P.: Scale Mixtures of Gaussians and the Statistics of Natural Images. In NIPS. (1999)

[55] Vemulapalli, R., Tuzel, O., Liu, M.Y.: Deep Gaussian Conditional Random Field Network: A Model-based Deep Network for Discriminative Denoising. In CVPR. (2016)

[56] Zhang, R., Isola, P., Efros, A.A.: Colorful Image Colorization. In ECCV. (2016)

[57] Welling, M., Teh, Y.W.: Bayesian Learning via Stochastic Gradient Langevin Dynamics. In ICML. (2011)

[58] Nijkamp, E., Hill, M., Zhu, S.C., Wu, Y.N.: Learning Non-Convergent Non-Persistent Short-Run MCMC Toward Energy-Based Model. In NeurIPS. (2019)

[59] Hinton, G.E.: Training Products of Experts by Minimizing Contrastive Divergence. Neural computation (2002)

[60] Tieleman, T.: Training Restricted Boltzmann Machines using Approximations to the Likelihood Gradient. In ICML. (2008)

[61] Du, Y., Mordatch, I.: Implicit Generation and Modeling with Energy Based Models. In NeurIPS. (2019)

[62] Gao, R., Lu, Y., Zhou, J., Zhu, S.C., Wu, Y.N.: Learning Generative Convnets via Multi-grid Modeling and Sampling. In CVPR. (2018)

[63] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and Improving the Image Quality of StyleGAN. In CVPR. (2020)

[64] Yu, A., Grauman, K.: Fine-Grained Visual Comparisons with Local Learning. In CVPR. (2014)

[65] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In CVPR. (2016)

[66] Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In MICCAI. (2015)

[67] He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In CVPR. (2016)

[68] Karras, T., Laine, S., Aila, T.: A Style-based Generator Architecture for Generative Adversarial Networks. In CVPR. (2019)

[69] Liu, Z., Luo, P., Wang, X., Tang, X.: Deep Learning Face Attributes in the Wild. In ICCV. (2015)

[70] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception Architecture for Computer Vision. In CVPR. (2016)

[71] Yu, F., Koltun, V., Funkhouser, T.: Dilated residual networks. In CVPR. (2017)

[72] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. In NeurIPS. (2019)

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See the abstract and Section 1.
   (b) Did you describe the limitations of your work? [Yes] See Section A.4 in the the supplemental material.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] we have read the guidelines and ensured that our paper conforms to them.

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We will attach the code and instructions in the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We ran one time.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We did not include them.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 5.
   (b) Did you mention the license of the assets? [No] We did not mention the license.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] No we did not introduce any new assets.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] See Section 5.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] No we did not.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]