LG-VQ: Language-Guided Codebook Learning

Guotao Liang[1,2], Baoquan Zhang[*1], Yaowei Wang[2], Xutao Li[1], Yunming Ye[1], Huaibin Wang[1]
Chuyao Luo[1], Kola Ye[3], Linfeng Luo[3]
[1]Harbin Institute of Technology, Shenzhen, [2]Peng Cheng Laboratory, [3]SiFar Company
{lianggt, wangyw}@pcl.ac.cn
{23B951062, 22S051022}@stu.hit.edu.cn
{baoquanzhang, lixutao, yeyunming}@hit.edu.cn
{luochuyao.dalian, kolaygm, llf10811020205}@gmail.com

Abstract

Vector quantization (VQ) is a key technique in high-resolution and high-fidelity image synthesis, which aims to learn a codebook to encode an image with a sequence of discrete codes and then generate an image in an auto-regression manner. Although existing methods have shown superior performance, most methods prefer to learn a single-modal codebook (*e.g.*, image), resulting in suboptimal performance when the codebook is applied to multi-modal downstream tasks (*e.g.*, text-to-image, image captioning) due to the existence of modal gaps. In this paper, we propose a novel language-guided codebook learning framework, called LG-VQ, which aims to learn a codebook that can be aligned with the text to improve the performance of multi-modal downstream tasks. Specifically, we first introduce pre-trained text semantics as prior knowledge, then design two novel alignment modules (*i.e.*, Semantic Alignment Module, and Relationship Alignment Module) to transfer such prior knowledge into codes for achieving codebook text alignment. In particular, our LG-VQ method is model-agnostic, which can be easily integrated into existing VQ models. Experimental results show that our method achieves superior performance on reconstruction and various multi-modal downstream tasks.

1 Introduction

In recent years, with the growing development of various multi-modal task scenarios [37, 36, 38], unified modeling of visuals and language has sparked considerable interest. Vector Quantization (VQ)-based image modeling technique, exemplified by VQ-VAE [43] and VQ-GAN [9], has emerged as a pivotal approach in the realm of unified modeling. The VQ methodology [43] typically follows a two-stage generation paradigm. In the initial stage, a trainable discrete codebook is employed to quantize continuous image features into a discrete token sequence to finish the reconstruction task. Subsequently, the codebook is utilized for various downstream tasks by generative models [42, 37].

Learning a robust codebook during the initial stage is crucial for optimizing performance in downstream tasks. At present, lots of VQ methods have been proposed to achieve robust code representation [15, 14, 7, 12]. For instance, VQ-GAN [9] introduces an adversarial training loss to learn a perceptually rich codebook. Some other works consider improving the codebook representation from the perspective of addressing the problem of codebook collapse [53, 52].

Although existing methods have shown superior performance, most methods only focus on learning a single-modal codebook contains more low-level information (*e.g.*, image's pixel, edge, and texture), resulting in suboptimal performance when the codebook is applied to multi-modal downstream tasks

[*]Corresponding Authors

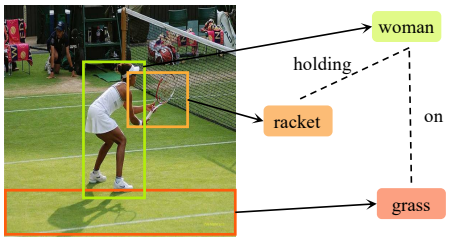
38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-4433

(*e.g.*, text-to-image [37], image captioning [36], VQA [24]). That is because the codebook lacks high-level semantics and the existence of modal gaps.

To address the above issue, we propose a novel codebook learning method (*i.e.*, multi-modal codebook learning), called ***L**anguage-**G**uided VQ* (LG-VQ). The novelty lies in utilizing pre-trained text semantics as supervised information to guide the codebook to learn abundant multi-modal knowledge.

Specifically, we first employ a cross-modal pre-trained model (*i.e.*, CLIP [32]) to encode text semantics. Then, we propose two novel semantic supervision modules to transfer the text semantics into codebook, *i.e.*, Semantic Alignment Module, and Relationship Alignment Module. Within the semantic alignment module, we enhance the consistency between the semantic representations of the codebook and text through global semantic alignment and masked text prediction. On the other hand, simply aligning the text and codebook in the holistic semantic space cannot satisfy more complex reasoning tasks like image captioning and VQA. Inspired by some VQA techniques [26, 40, 24], the semantic relationships between words play a very important role in various tasks of natural language processing (See Fig. 1). Based on this fact, we further propose to transfer the semantic relationships between words into codes to achieve better alignment between the codes and words. Such a text-aligned codebook helps alleviate modal gaps and improve codebook performance on cross-modal tasks.



Q: What is this woman holding?

Figure 1: To answer the question, one not only needs to identify "women" and "racket" but also understand the semantic relationship between them ("holding").

The contributions of this work are summarized as follows:

- We point out the limitations of existing methods in learning an expressive codebook since they learn a single-modal codebook. We propose a novel multi-modal codebook learning method, named LG-VQ, which can enable the codebook to effectively retain fine-grained reconstruction information while aligning with the text.
- Resorting to pre-trained text semantics, we propose two novel semantic supervision modules, *i.e.*, Semantic Alignment Module and Relationship Alignment Module, effectively learn text-aligned codebook. The advantage of such alignment modules is the abundant context and relationship semantics contained in pre-trained text can be sufficiently leveraged for enhancing multi-modal codebook learning.
- We conduct comprehensive experiments on four public datasets, which shows that our LG-VQ method outperforms various state-of-the-art models on reconstruction and various cross-modal tasks (*e.g.*, text-to-image, image captioning, VQA).

2 Related Works

2.1 Vector Quantization for Image Generation

Vector quantization (VQ) is designed to learn a codebook, which aims to encode continuous image features into a discrete sequence. Then, the learned codebook can be utilized for various downstream tasks. Oord et al. [43] first propose a novel VQ method called VQ-VAE. This method innovatively replaces the prior distribution of Variational Autoencoder (VAE) with a discrete deterministic distribution (*i.e.*, a codebook). To further improve the performance of VQ, various models are proposed to learn a more expressive codebook [9, 48, 2, 17, 7, 15, 14, 21]. For example, VQ-GAN [9] addresses the issue of image blur generated by VQ-VAE through the introduction of an adversarial training loss. However, the above methods do not tackle the codebook collapse issue. To address the issue, many novel methods are proposed from the perspective of regularization [33], codebook update [53], codebook transfer [51]. Recently, inspired by the large language models (LLMs), instead of mapping images to the visual code tokens, some works attempt to map the images to the word tokens of LLMs by viewing images as "foreign languages" [22, 50, 56]. However, because of the inherent differences between vision and language, these works have difficulty assigning correct semantic words to images.

https://doi.org/10.52202/079017-4433

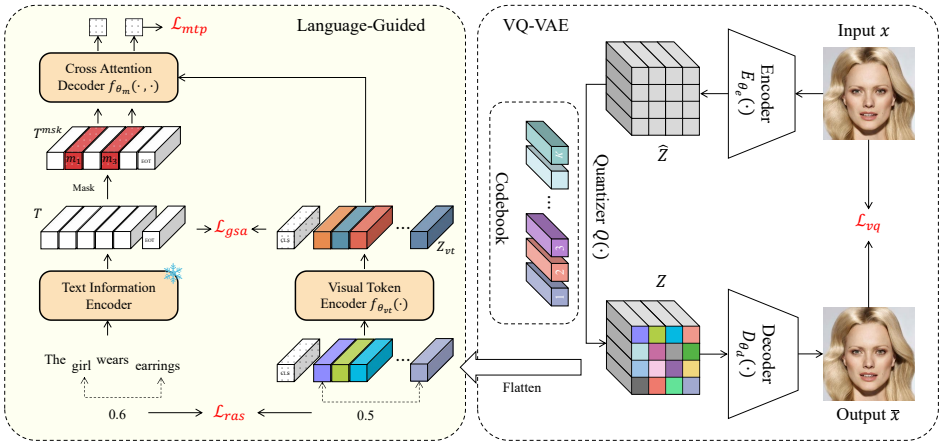


Figure 2: The overall architecture of the proposed LG-VQ method. The right part of the figure is the basic VQ-VAE module, left is our language-guided module, which consists of three losses: global semantic alignment ($\mathcal{L}_{gsa}$), masked text prediction ($\mathcal{L}_{mtp}$), and relationship alignment supervision ($\mathcal{L}_{ras}$). Here, pre-trained text information guides discrete code tokens of an image to learning rich semantic knowledge based on three losses.

Compared with the aforementioned methods, our approach focuses more on multi-modal alignment in feature space (*i.e.*, learning a text-aligned codebook). We use pre-trained text semantics to supervise the codebook learning. The advantage is that the rich semantic information from the text can be fully exploited for more robust codebook learning so that the codebook can not only retain more reconstruction information but also be able to understand and match text. More importantly, our method is model-agnostic, which can be easily integrated into existing VQ models.

2.2 Vision-Language Representation Learning

Vision-language Pre-training (VLP) aims to learn multi-modal representations from large-scale image-text pairs that can improve vision-language downstream tasks, for example, VQA[1]. Early methods such as LXMERT [41], UNITER [4] employ pre-trained object detectors to extract image region features, and fuse image features with text by a cross-modal encoder to achieve the vision-language representation learning. Although these methods achieve superior performance on downstream tasks, they require high-resolution input images and pre-trained object detectors. To remove the object detectors, a large number of researchers focus on learning two separate representations for image and text [32, 16, 18]. For instance, CLIP [32] learns a robust representation for each image and text using contrastive learning based on large-scale image-text pair data.

In this paper, we propose to employ pre-trained text semantics as supervised information to guide codebook learning. Its advantage is that abundant multi-modal knowledge contained in text can be fully leveraged for robust codebook learning. Additionally, we design a novel relationship alignment module to inject semantic relationships between words into codes.

3 Methodology

3.1 Preliminaries: VQ-VAE

VQ-VAE [43], as a pioneering work on the VQ research domain, aims to learn a discrete codebook to encode images into discrete token sequences through an Encoder-Decoder framework. As illustrated in Fig. 2 right, the VQ-VAE consists of a visual encoder $E_{\theta_e}(\cdot)$ with parameter θ_e, a token decoder $D_{\theta_d}(\cdot)$ with parameter θ_d, a quantizer $Q(\cdot)$, and a codebook is defined as $\mathcal{Z} = \{e_k\}_{k=1}^{K}$ that consists of learnable K entries $e_k \in \mathbb{R}^{d_z}$ with dimension d_z. Given an input image $x \in \mathbb{R}^{H\times W\times C}$, where H, W, and C represent the height, width, and channel of the image respectively. The visual encoder $E_{\theta_e}(\cdot)$ learns to convert the original image into grid features $\hat{Z} = E_{\theta_e}(x) \in \mathbb{R}^{\frac{H}{f}\times\frac{W}{f}\times d_z}$ and f is the down-sampling factor. The quantizer $Q(\cdot)$ looks up the nearest neighbor in the codebook for each

https://doi.org/10.52202/079017-4433

grid representation $\hat{z}_i \in \mathbb{R}^{d_z}$ in $\hat{Z}_i$ using the following equation:

$$z_i = Q(\hat{z}_i) = \underset{e_k \in \mathcal{Z}}{argmin} \|\hat{z}_i - e_k\|. \tag{1}$$

The token decoder $D_{\theta_d}(\cdot)$ is used to reconstruct the original image by $\widetilde{x} = D_{\theta_d}(Z)$, where Z is discrete code tokens of whole image obtained by Eq. 1. During training, the visual encoder $E_{\theta_e}(\cdot)$, codebook $\mathcal{Z}$, and token decoder $D_{\theta_d}(\cdot)$ are jointly optimized by minimizing the following objective:

$$\mathcal{L}_{vq} = \|x - \widetilde{x}\|_2^2 + \|sg[E_{\theta_e}(x)] - Z\|_2^2 + \omega\|E_{\theta_e}(x) - sg[Z]\|_2^2, \tag{2}$$

where, the first term is reconstruction loss, which measures the difference between the original image x and the reconstructed image $\widetilde{x}$. $sg[\cdot]$ represents the stop-gradient operator, and the second term is codebook loss, which encourages the codebook to be close grid features. The third term is the "commitment loss" [43], where ω serves as a hyper-parameter. However, existing VQ-based methods mainly focus on the learning of single-modal codebook, thereby limiting their applicability to multi-modal downstream tasks.

3.2 Proposed Method: LG-VQ

Existing works attempt to improve codebook reconstruction capabilities to obtain better performance on downstream tasks. However, ignoring modal differences results in suboptimal performance when the codebook is applied to cross-modal tasks. To address this issue, we propose to utilize the pre-trained text semantics as supervised information to learn a text-aligned codebook. Its advantage is abundant semantic information from text can be fully exploited for more robust codebook learning to improve the performance of reconstruction and cross-modal tasks. The comprehensive architecture of the proposed LG-VQ method is illustrated in Fig. 2 left. It consists of two supervision modules: Semantic Alignment Module (*i.e.*, $\mathcal{L}_{gsa}$ and $\mathcal{L}_{mtp}$), and Relationship Alignment Module (*i.e.*, $\mathcal{L}_{ras}$). The first module encourages global semantic consistency between the codebook and text. The second module aims to transfer the rich semantic relationship between words into codes. Next, we introduce these two modules in detail.

3.2.1 Semantic Alignment Module

Considering that paired image and text data have consistent semantic information and the missing information of masked data can be completed from the other modality, we propose global semantic alignment, which aims to enhance the consistency of global semantics between text and visual codes, and masked text prediction, which uses visual codes to restore the masked words. Next, we discuss how to align text and codebook in the semantic space.

Text Information Encoder: Instead of jointly training text and codebook from scratch, we employ a pre-trained cross-modal model CLIP [32] to encode text information. Its advantage is that such text information already has good cross-modal semantic knowledge and is beneficial for codebook learning. Specifically, for a given text description of an image $t = \{w_{SOT}, w_1, w_2, \cdots, w_{n-2}, w_{EOT}\}$, where w_i denotes the i-th word, w_{SOT} and w_{EOT} represent the $[start]$ token and $[end]$ token, respectively, and n is text sequence length. We use the text encoder of a pre-trained CLIP model to obtain whole sequence embedding $T \in \mathbb{R}^{n \times d_t}$:

$$T = \{e_{SOT}, e_{w_1}, e_{w_2}, \cdots, e_{w_n}, e_{EOT}\} = \text{CLIP}(t). \tag{3}$$

Similar to CLIP, we use the e_{EOT} to represent the global context feature of the sequence.

Global Semantic Alignment aims to align text and image visual codes in the global semantic space. For getting the global representation of visual codes, we employ a vision transformer (ViT) $f_{\theta_{vt}}$ [8] to encode the discrete codes of image. Specifically, given an image, we firstly obtain the discrete codes of image Z by Eq. 1. Then, we introduce a learnable global token $[CLS]$ at the beginning to form a token sequence Z_c, where global token $[CLS]$ is employed to capture the image's global context information. We feed the sequence into $f_{\theta_{vt}}$ to get a new visual code representation, that is:

$$Z_{vt} = \{e_{CLS}, e_1, e_2, \cdots, e_{\frac{H}{f} \times \frac{W}{f}}\} = f_{\theta_{vt}}(Z_c). \tag{4}$$

Finally, we employ InfoNCE [29], which maximizes the similarity between visual and text in the global representation, as our learning objective, where $\mathcal{B}$ is the batch size, $s(\cdot, \cdot)$ is cosine similarity:

$$\mathcal{L}_{gsa} = -\sum_{i \in \mathcal{B}} \log \frac{\exp(s(e^i_{CLS}, e^i_{EOT}))}{\sum_{j \in \mathcal{B}} \exp(s(e^i_{CLS}, e^j_{EOT}))}. \tag{5}$$

Masked Text Prediction: To further enhance the semantic alignment, we propose to use discrete visual codes to reconstruct the masked words from a more fine-grained perspective, refer to Fig. 2 left. Formally, for a given fixed-length text sequence of $n-2$, we first randomly sample the masking ratio r from a truncated Gaussian distribution [19]. Subsequently, we randomly mask out $r \cdot (n-2)$ words and replace them with learnable $[mask_i]$ tokens based on their positions i. Next, a self-attention module [44] is employed to learn adaptive masked word embeddings based on unmasked words. The resulting adaptive masked sequence is denoted as $T^{msk} = \{e_{SOT}, m_1, e_{w_2}, m_3, \cdots, e_{EOT}\}$, where m_i is the mask token embedding at the i-th position in the sequence. Following this, a cross attention decoder $f_{\theta_m}(\cdot,\cdot)$ is employed to predict the masked word tokens given the discrete visual codes Z_{vt} obtained by Eq. 4. Finally, we add a cross-entropy loss $H(\cdot,\cdot)$ between the ground-truth word tokens and the output of the decoder. Let y_{msk} denote a one-hot vocabulary distribution where the ground-truth word token has a probability of 1, $f_{\theta_m}(Z_{vt}, T^{msk})$ denote the predicted probability of model for masked word tokens. That is:

$$\mathcal{L}_{mtp} = -\mathbb{E}_{(Z_{vt},T^{msk})\sim\mathcal{B}} H(y_{msk}, f_{\theta_m}(Z_{vt}, T^{msk})). \tag{6}$$

3.2.2 Relationship Alignment Module

While the two aforementioned loss functions for achieving good alignment at holistic semantic space have demonstrated initial promise, they cannot satisfy more complex reasoning tasks like image captioning and VQA. Inspired by some VQA techniques [26, 40, 24, 1], the semantic relationships between pre-trained words play a very important role in complex text reasoning tasks. For instance, as shown in Fig 1, to answer question ("What is this woman holding?"), one needs to fully understand the visual objects "women", "racket", and semantic relationship between them ("holding"). Based on the above fact, we propose to transfer the semantic relationship between words into codes. Such semantic relationships enable the model to better understand the image for addressing complex reasoning tasks.

But unfortunately, there is an issue there is no alignment between words and codes. Thanks for the above two losses that have provided semantic alignment of text and visual codes. To achieve the above idea, as shown in Fig. 3, we first use Z_{vt} to align with words. Then, we inject semantic relationships between words into the initial codebook Z, instead of the Z_{vt}. Its advantage is it can prevent codes from collapsing into a single point for learning more diverse representations by relationship limiting. Then, Z_{vt} primarily serves the purpose of aligning words and codes, but it is a crucial step for subsequent processes. Specifically, given any two words of a sentence, we use pre-trained word embedding [32] to encode words, e_{w_i} and e_{w_j}. We employ cosine similarity to find the index of the code from Z_{vt} that is most similar to the word. Then, one can get code embedding from Z based on the index:

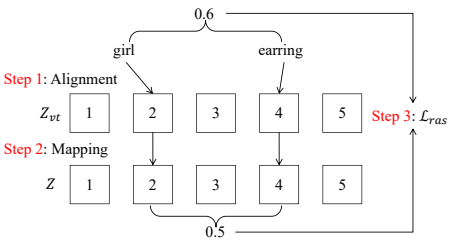


Figure 3: Illustration of relationship alignment module, we use Z_{vt} to align with two words, then inject the semantic relationship of two words into Z codes.

$$e_{z_i} = Z[\underset{e_z \in Z_{vt[1:]}}{argmax}\, s(e_{w_i}, e_z), :], \qquad e_{z_j} = Z[\underset{e_z \in Z_{vt[1:]}}{argmax}\, s(e_{w_j}, e_z), :]. \tag{7}$$

Next, we consider cosine similarity as a measure of semantic relationships between words and leverage it to establish corresponding relationships between codes achieving semantic relationship transfer. Finally, we utilize the following loss function as learning objective:

$$\mathcal{L}_{ras} = \sum_{(w_i,w_j)\in t} (s(e_{w_i}, e_{w_j}) - s(e_{z_i}, e_{z_j}))^2. \tag{8}$$

3.2.3 Training Objective

We use three hyperparameters (*i.e.*, α, β, and γ) to control three losses, respectively. Finally, the overall objective function is:

$$\mathcal{L} = \mathcal{L}_{vq} + \alpha\mathcal{L}_{gsa} + \beta\mathcal{L}_{mtp} + \gamma\mathcal{L}_{ras}. \tag{9}$$

https://doi.org/10.52202/079017-4433

Table 1: Results of image reconstruction on TextCaps, CelebA-HQ, CUB-200, and MS-COCO. "VQ-VAE+LG" denotes considering our method LG-VQ based on VQ-VAE.

Models	TextCaps		CelebA-HQ		CUB-200		MS-COCO	
	FID↓	PSNR↑	FID↓	PSNR↑	FID↓	PSNR↑	FID↓	PSNR↑
VQ-VAE	82.31	**21.96**	41.45	**25.57**	54.92	24.38	86.21	**23.55**
VQ-VAE+LG	**81.93**	21.95	**40.53**	25.04	**36.55**	**25.60**	**79.54**	23.40
VQ-GAN	24.08	19.64	5.66	**24.10**	3.63	22.19	14.45	20.21
VQ-GAN+LG	**20.35**	**19.92**	**5.34**	23.75	**3.08**	**22.47**	**10.72**	**20.50**
CVQ	16.35	**20.24**	5.19	23.15	3.61	22.29	9.94	20.48
CVQ+LG	**15.51**	20.21	**4.90**	**24.48**	**3.33**	**22.47**	**9.69**	**20.71**

4 Experiments

4.1 Experimental Settings

Evaluation Metrics. As our method is model-agnostic, we choose recent models, including VQ-VAE [43], VQ-GAN [9], and CVQ [53] as our backbone network. Following existing works [52, 53], we evaluate the reconstruction image quality on two evaluation metrics, *i.e.*, Fréchet Inception Distance (FID) [13] which evaluates the perceptual similarity of reconstructed images and original images, and Peak Signal-to-noise Ratio (PSNR) [10] is employed to measure the pixel-level similarity between the reconstructed and original images.

Dataset. We evaluate our method on four public datasets, including TextCaps [39], CelebA-HQ [23], CUB-200 [45], and MS-COCO [20]. For CelebA-HQ, CUB-200, and MS-COCO datasets, we use publicly available image captions, CelebA-HQ from [47], CUB-200 from [34], MS-COCO from [3]

Implementation Details. Following VQ-GAN [9], all images are reshaped 256×256 for reconstruction and generation. Down-sampling factor f is set to 16. The codebook size K is 1024. The batch size is 8. In our experiments, we maintain consistent parameter settings between our method LG-VQ and the chosen backbone networks (*i.e.*, VQ-VAE [43], VQ-GAN [9], and CVQ [53]) for a fair comparison. For each image, we randomly select a text from multi-text for training. Since our method introduces additional text and pre-trained CLIP model, for a fair comparison, we select VQCT [51] as the baseline for various downstream tasks. VQCT extracts many visual-related words from a large amount of text and designs a novel codebook transfer network based on the pre-trained CLIP model to learn the visual codebook.

Table 2: Ablation study of our three loss functions on TextCaps and CUB-200.

	Setting	TextCaps FID↓	CUB-200 FID↓
(i)	Baseline(VQ-GAN)	24.08	3.63
(ii)	+ $\mathcal{L}_{gsa}$	23.01	3.39
(iii)	+ $\mathcal{L}_{mtp}$	21.54	3.49
(iv)	+ $\mathcal{L}_{mtp}$ + $\mathcal{L}_{ras}$	20.77	3.32
(v)	+ $\mathcal{L}_{mtp}$ + $\mathcal{L}_{gsa}$	20.46	3.34
(vi)	+ $\mathcal{L}_{mtp}$ + $\mathcal{L}_{gsa}$ + $\mathcal{L}_{ras}$	20.35	3.08

Table 3: Results (Recall@1) of masked word prediction on CelebA-HQ and CUB-200. "Mask-1" denotes that text is randomly masked one word.

Dataset		Recall@1
CelebA-HQ	Mask-1	99.55
	Mask-3	99.24
CUB-200	Mask-1	83.65
	Mask-3	80.17

4.2 Discussion of Results

Table 1 illustrates the image reconstruction performance of our model compared to the backbone model on multiple datasets. It can be observed that our method LG-VQ outperforms all compared methods on most evaluations, which suggests that our method is extremely effective and has strong generality. Compared with FID, our PSNR improvement is marginal, this is reasonable in the VQ research domain, which widely exists in previous VQ methods [21, 52, 53]. The main reason is that PSNR only measures the pixel-level similarity of the images, while FID can effectively measure the diversity and semantic similarity of image generation. Compared with backbone models, the key difference lies in that our method introduces well pre-trained text semantics, which is beneficial to learning a more expressive codebook. This shows the effectiveness of our method. We also provide a qualitative comparison of the image reconstruction performance of different methods, please refer to

Appendix A.3 Fig. 10.

4.3 Ablation Study

Are our three loss functions both effective? In Table 2, we conduct an ablation study to show the effectiveness of the proposed three loss functions. Specifically, the VQ-GAN serves as the baseline model (*i.e.*, without introducing any loss). We do not conduct a separate experiment on $\mathcal{L}_{ras}$ because this module requires code and words to be well aligned. Based on the results from (i) $\sim$ (vi), we draw several key conclusions: Firstly, each loss function plays a crucial role in improving the performance of image reconstruction. Secondly, the performance of (iii) outperforms (ii) by a large margin on TextCaps. This is reasonable because TextCaps's texts are richer and more diverse than CUB-200, it can provide more knowledge for more fine-grained alignment between codes and text, which is useful for the learning of a more robust codebook. Thirdly, analyzing the results of (iii) and (iv), injecting word-level semantic relationships into codes is beneficial, which confirms our motivation. Furthermore, the performance of (v) outperforms (i), which is reasonable because the abundant semantic knowledge from pre-trained text can be fully exploited for learning more robust codebook representation. This supports the motivation of learning a multi-modal codebook (*i.e.*, aligned with text). Finally, comparing the results of (vi) with (i)$\sim$(v), fully considering all losses achieves the best performance, indicating the effectiveness of our method.

Can our global semantic supervision align vision and language? In Fig. 4, we provide several image-to-text retrieval cases on CelebA-HQ and CUB-200 datasets based on VQ-GAN+LG. From the figure, it can be observed that our method can accurately retrieve text very similar to the image content, achieving the alignment of vision and language. For example, row 2 examples show that our method can precisely understand some key attributes of images (*e.g.*, "gray hair", "necktie", "big nose" and "chubby") and retrieve similar text. This suggests that the codes learned through our method have obtained good alignment with the text, which verifies the effectiveness of our method. Moreover, such alignment is beneficial for learning robust code representations and enhancing performance in multi-modal downstream tasks.

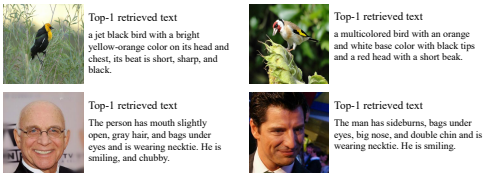


Figure 4: Examples of the top-1 most similar text selected on image-to-text retrieval task.

Figure 5: Examples of the top-1 word predicted on masked word prediction task.

Can our codebook accurately predict masked words? To answer this question, we conduct a word prediction task on test data based on VQ-GAN+LG by randomly masking one word or three words of text, as shown in Table 3. We use Recall@1 as the evaluation metric [18]. From the table, our method demonstrates accurate predictions of masked words, confirming the effectiveness of our approach. Fine-grained word prediction can help the codebook better understand the text semantics, which is crucial for improving the performance of the downstream task. Additionally, several examples in Fig. 5 demonstrate our method's ability that accurately predict subject words (*e.g.*, wings, eyes) and verbs (*e.g.*, has, is, and smiling), further affirming its strong multi-modal understanding capabilities.

Can our codebook learn the word semantic relationships? In Fig. 6, we visualize the cosine similarity between words and the cosine similarity between codes aligned with the words for a certain sample based on VQ-GAN+LG. From the figure, we can see our codes can learn consistent relationships with word semantics compared with VQ-GAN. For example, the similarity "code 33" vs "code 232" (0.46) resembles "wings" vs "chest" (0.49). In addition, we provide a quantitative similarity evaluation between codes and words in Table 4. From the results, we can find that our codes indeed achieve consistent semantic relationships with words.

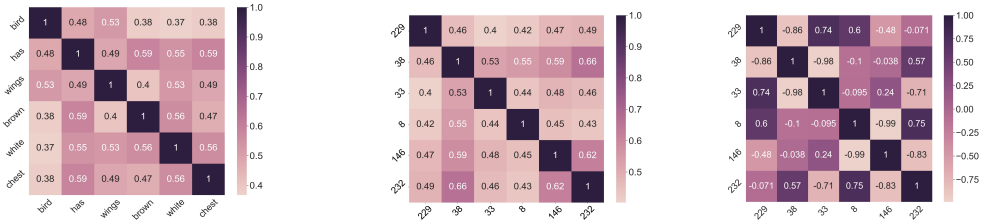
Table 4: Results of similarity evaluation between codes and words on CUB-200 all test data.

Method	VQ-GAN	VQ-GAN+LG
MSE↓	0.6374	0.0351

Is our method effectively learning more diverse code representation? Following [52], we directly feed each codebook embedding e_k (size: $1 \times 1 \times 256$) into the decoder $D_{\theta_d}(\cdot)$ to generate codebook

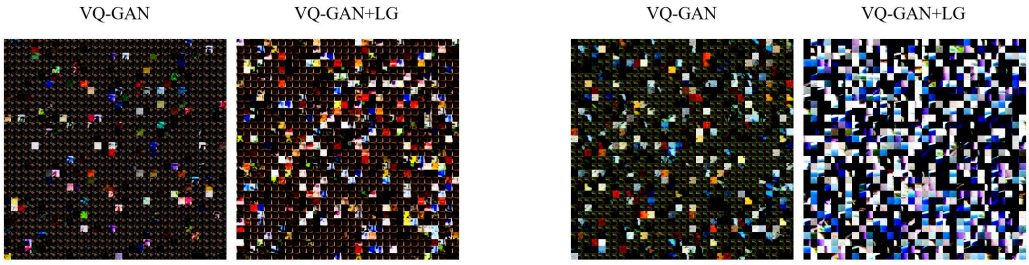
https://doi.org/10.52202/079017-4433

image (size: $16 \times 16 \times 3$). Then, we concatenate all codebook images to form a big image with 32×32 patches. Finally, we visualize the result of VQ-GAN and our LG-VQ on TextCaps and MS-COCO as shown in Fig. 7. This visualization suggests that our method enables the model to learn more diverse code representations and improve codebook usage.



(a) word similarity (b) LG-VQ code similarity (c) VQ-GAN code similarity

Figure 6: Visualization of words similarity and image codes similarity aligned with the word. We extract some representative words from the text as a demonstration.



(a) The usage of codebook on VQ-GAN is 18.62% and VQ-GAN+LG is 43.58% on TextCaps

(b) The usage of codebook on VQ-GAN is 40.09% and VQ-GAN+LG is 97.89% on MS-COCO

Figure 7: Visualization of the codebook of VQ-GAN and LG-VQ on TextCaps and MS-COCO.

4.4 Application

4.4.1 Image Generation

Following [9, 37, 11], we conduct image generation downstream tasks (*i.e.*, text-to-image, semantic synthesis, unconditional generation, and image completion) to fully validate the effectiveness of the learned codebook on CelebA-HQ.

Text-to-Image. In Table 5, we compare our LG-VQ with the state-of-the-art models on CelebA-HQ dataset for text-to-image. From the results, our LG-VQ method outperforms baseline methods by a large margin. This is reasonable due to the incorporation of pre-trained text knowledge enabling a comprehensive understanding of the text, which suggests our method's effectiveness. Moreover, we provide some synthesis examples comparing the results of our LG-VQ with baseline methods in Figure 8, showing the performance in the text-to-image task. From the figure, we can see our method not only comprehensively understands the given text conditions but also excels in generating realistic images compared with baseline methods. For instance, our method can capture the "glasses", "man", "long black hair", and "no beard" key attributions.

Semantic Synthesis. Following [9], we compare with existing semantic synthesis models in Table 6. Our method achieves the best performance, which suggests our method's effectiveness. We provide some examples in Appendix Figure 13.

Unconditional Generation and Image Completion. Following [9], we conduct unconditional image generation and Image Completion on CelebA-HQ dataset, as shown in Table 7 and Table 10. From the results, we can see that our method can significantly improve the performance of VQ-GAN, which is reasonable because pre-trained text can provide rich semantic knowledge for learning more robust codebook representation. This suggests the effectiveness of our method. We provide some examples in Appendix Figure 18 and Figure 17.

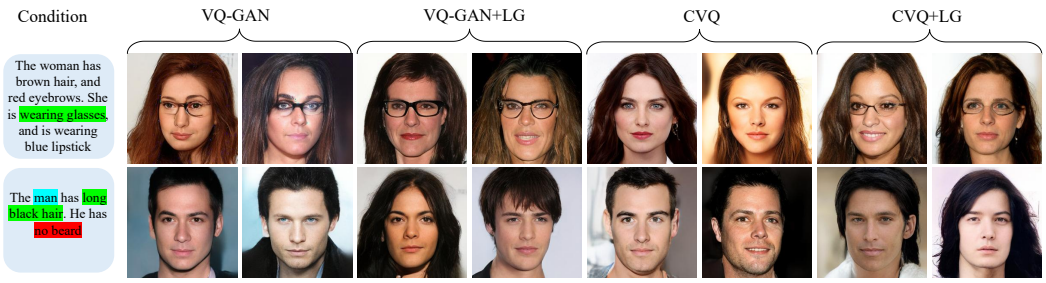


Figure 8: Text-to-image synthesis and semantic image synthesis on CelebA-HQ. Text with background color emphasizes generated details

4.4.2 Visual Text Reasoning

Follow [27, 6], we use the learned codebook to conduct two visual text reasoning tasks: 1) image captioning on CUB-200; and 2) visual question answering (VQA) on COCO-QA [35]. For the experimental setting, please refer to Appendix A.1.

Image captioning. Following [27], we conduct the image captioning task on the CUB-200 dataset. We compare two recent work V2L Tokenizer [56] and VQCT [51]. We select VQ-GAN as our backbone network. The results are shown in Table 11. For the results, we can see that our LG-VQ method outperforms the performance of VQ-GAN. This is reasonable because the pre-trained text provides rich context and relationship semantics for codebook learning, which verifies our motivation for learning a text-aligned codebook to improve the performance of the codebook on cross-modal tasks. On the other hand, the V2L Tokenizer and VQCT cannot achieve very good performance because it is difficult to assign correct semantic language tokens to images. Compared with the V2L Tokenizer, our method utilizes pre-trained text semantics as supervised information. Its advantage is can make the codebook learn semantic information consistent with the text (*i.e.*, learning a text-aligned codebook). And, our method is model-agnostic, which can be easily integrated into existing VQ models.

Visual Question Answering. We select VQ-GAN and VQCT [51] as the baseline. We conduct the VQA task on the COCO-QA [35] using the codebook trained on the MS-COCO dataset. The results are shown in Table 9. From the results, we can see that our LG-VQ method significantly improves the performance of VQ-GAN on VQA task (approximately 8.32%↑ on Accuracy). That is reasonable due to we introduce pre-trained text semantics to enable us to obtain a codebook aligned with the text, which is helpful for comprehensively understanding the given text question. This confirms our motivation and the effectiveness of our method.

Table 5: Results of text-to-image on CelebA-HQ.

Model	Text-to-Image FID↓
Unite and Conqu [4]	26.09
Corgi [54]	19.74
LAFITE [55]	12.54
VQ-GAN	15.29
CVQ	13.23
VQ-GAN+LG	12.61
CVQ+LG	12.33

Table 6: Result (FID↓) of semantic synthesis on CelebA-HQ.

Model	Semantic Synthesis FID↓
Reg-VQ [52]	15.34
VQCT [51]	14.47
VQ-GAN	11.53
CVQ	11.04
VQ-GAN+LG	11.46
CVQ+LG	11.03

4.4.3 Visual Grounding

We conduct a visual grounding task on refcoco dataset [49] to validate the effectiveness of the learned MS-COCO's codebook. Following the same metric used in [5], a prediction is right if the IoU between the grounding-truth box and the predicted bounding box is larger than 0.5. We select VQ-GAN and VQCT [51] as the baseline. The results are shown in Table 8. From the results, we can see that the performance of our method consistently outperforms VQ-GAN and VQCT, which suggests its effectiveness. We also provide a qualitative comparison in Appendix Figure 19. For the experimental setting, please refer to Appendix A.1.

https://doi.org/10.52202/079017-4433

Table 7: Result (FID↓) of unconditional image generation on CelebA-HQ.

Model	CelebA-HQ FID↓
Style ALAE [30]	19.2
DC-VAE [31]	15.8
VQ-GAN	10.2
LG-VQ	9.1

Table 8: Result (FID↓) of visual grounding on refcoco dataset using MS-COCO's codebook.

Model	Visual Grounding Accuracy(0.5)↑
VQ-GAN	9.14
VQCT [51]	9.46
LG-VQ	9.62

Table 9: Results of (Accuracy and WUPS [46]) VQA on COCO-QA [35] dataset using MS-COCO's codebook.

Setting	VQA	
	Accuracy↑	WUPS↑
VQCT [51]	40.42	82.06
VQ-GAN	37.82	83.22
LG-VQ	40.97	83.56

Table 10: Result (FID↓) of image completion on CelebA-HQ.

Model	CelebA-HQ FID↓
VQ-GAN	9.02
LG-VQ	8.14
Improve	**9.76%**

Table 11: Results of image captioning on CUB-200.

Model	Image Captioning			
	BLEU4↑	ROUGE-L↑	METEOR↑	CIDEr-D↑
VQ-GAN	1.29	33.40	24.47	93.62
V2L Tokenizer [56]	1.59	30.65	25.76	**104.14**
VQCT [51]	1.38	26.50	24.63	98.22
LG-VQ	**1.69**	**34.73**	**25.78**	102.77

5 Conclusions

In this paper, we propose a novel codebook learning method, named LG-VQ. LG-VQ is a model-agnostic method and can easily be integrated into existing VQ models. In particular, we propose to incorporate pre-trained text semantics into the codebook by two novel supervision modules, *i.e.*, semantic and relationship. Quantitative and qualitative experiments demonstrate the strong generality of our method, showing its ability to improve the performance of the codebook in cross-modal tasks.
Limitations. In our current paper, we suppose each word aligns with a code, but it fails to capture some more complex relationships between words and codes (*e.g.*, one code aligns with multiple words). In the future, we plan to investigate the relationships between codes and words. Moreover, although our results show that the performance of VQ in visual text reasoning tasks can be significantly improved, its results are still far lower than the performance of image captioning or VQA models.
Broader impact Our paper shows that learning a multi-modal codebook (*i.e.*, a text-aligned codebook) can not only significantly improve the performance of reconstruction but also the performance of the codebook on cross-modal tasks. The potential impact of our research lies in its influence on future studies, specifically in the area of unified modeling of multi-modal understanding and generation. For instance, our work can be extended to interact with LLMs to improve multi-modal understanding and generation capabilities. In particular, our model can be used to generate images or text. It may be exploited to produce some erroneous and unethical information, which needs to be handled carefully before employing our model in practical applications.

Acknowledgement

This work was supported by the Shenzhen Peacock Program under Grant No. ZX20230597, NSFC under Grant No. 62272130 and Grant No. 62376072, and the Shenzhen Science and Technology Program under Grant No. KCXFZ20211020163403005. It was also supported by the Major Key Project of PCL (PCL2023A08) and the National Science Foundation of China: Multi-source Cross-platform Video Analysis and Understanding for Intelligent Perception in Smart City: U20B2052.

References

[1] Stanislaw Antol and Aishwarya Agrawal. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

[2] Huiwen Chang and Han Zhang. Maskgit: Masked generative image transformer. In *CVPR*, pages 11315–11325, 2022.

[3] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

[4] Yen-Chun Chen and Linjie Li. Uniter: Universal image-text representation learning. In *ECCV*, pages 104–120. Springer, 2020.

[5] Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. Transvg: End-to-end visual grounding with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1769–1779, 2021.

[6] Ming Ding and Zhuoyi Yang. Cogview: Mastering text-to-image generation via transformers. *NeurIPS*, 34:19822–19835, 2021.

[7] Xiaoyi Dong and Jianmin Bao. Peco: Perceptual codebook for bert pre-training of vision transformers. In *AAAI*, volume 37, pages 552–560, 2023.

[8] Alexey Dosovitskiy and Lucas Beyer. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

[9] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, pages 12873–12883, 2021.

[10] Fernando A Fardo and Victor H Conforto. A formal evaluation of psnr as quality measurement parameter for image segmentation algorithms. *arXiv preprint arXiv:1605.07116*, 2016.

[11] Shuyang Gu and Dong Chen. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*, pages 10696–10706, 2022.

[12] Yuchao Gu and Xintao Wang. Rethinking the objectives of vector-quantized tokenizers for image synthesis. *arXiv preprint arXiv:2212.03185*, 2022.

[13] Martin Heusel and Hubert Ramsauer. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017.

[14] Mengqi Huang and Zhendong Mao. Not all image regions matter: Masked vector quantization for autoregressive image generation. In *CVPR*, pages 2002–2011, 2023.

[15] Mengqi Huang and Zhendong Mao. Towards accurate image coding: Improved autoregressive image generation with dynamic vector quantization. In *CVPR*, pages 22596–22605, 2023.

[16] Chao Jia and Yinfei Yang. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, pages 4904–4916. PMLR, 2021.

[17] Doyup Lee and Chiheon Kim. Autoregressive image generation using residual quantization. In *CVPR*, pages 11523–11532, 2022.

[18] Junnan Li and Ramprasaath Selvaraju. Align before fuse: Vision and language representation learning with momentum distillation. volume 34, pages 9694–9705, 2021.

[19] Tianhong Li and Huiwen Chang. Mage: Masked generative encoder to unify representation learning and image synthesis. In *CVPR*, pages 2142–2152, 2023.

[20] Tsung-Yi Lin and Michael Maire. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[21] Xinmiao Lin and Yikang Li. Catch missing details: Image reconstruction with frequency augmented variational autoencoder. In *CVPR*, pages 1736–1745, 2023.

[22] Hao Liu and Wilson and Yan. Language quantized autoencoders: Towards unsupervised text-image alignment. *arXiv preprint arXiv:2302.00902*, 2023.

[23] Ziwei Liu and Ping Luo. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015.

 https://doi.org/10.52202/079017-4433

[24] Jiasen Lu and Jianwei Yang. Hierarchical question-image co-attention for visual question answering. *NeurIPS*, 29, 2016.

[25] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.

[26] Tomas Mikolov and Kai Chen. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[27] Ron Mokady and Amir Hertz. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021.

[28] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 792–807. Springer, 2016.

[29] Aaron van den Oord and Yazhe Li. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[30] Gaurav Parmar, Dacheng Li, Kwonjoon Lee, and Zhuowen Tu. Dual contradistinctive generative autoencoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 823–832, 2021.

[31] Stanislav Pidhorskyi, Donald A Adjeroh, and Gianfranco Doretto. Adversarial latent autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14104–14113, 2020.

[32] Alec Radford and Jong Wook Kim. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021.

[33] Ali Razavi and Aaron Van den Oord. Generating diverse high-fidelity images with vq-vae-2. *NeurIPS*, 32, 2019.

[34] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 49–58, 2016.

[35] Mengye Ren and Ryan Kiros. Exploring models and data for image question answering. *NeurIPS*, 28, 2015.

[36] Yuchen Ren and Zhendong Mao. Crossing the gap: Domain generalization for image captioning. In *CVPR*, pages 2871–2880, 2023.

[37] Robin Rombach and Andreas Blattmann. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022.

[38] Robin Rombach and Andreas Blattmann. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022.

[39] Oleksii Sidorov and Ronghang Hu. Textcaps: a dataset for image captioningwith reading comprehension. 2020.

[40] Mohammed Suhail and Abhay Mittal. Energy-based learning for scene graph generation. In *CVPR*, pages 13936–13945, 2021.

[41] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[42] Aaron Van den Oord and Nal Kalchbrenner. Conditional image generation with pixelcnn decoders. *NeurIPS*, 29, 2016.

[43] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 30, 2017.

[44] Ashish Vaswani and Noam Shazeer. Attention is all you need. *NeurIPS*, 30, 2017.

[45] Catherine Wah and Steve Branson. The caltech-ucsd birds-200-2011 dataset. 2011.

[46] Zhibiao Wu and Martha Palmer. Verb semantics and lexical selection. *arXiv preprint cmp-lg/9406033*, 1994.

[47] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265, 2021.

[48] Jiahui Yu and Xin Li. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

[49] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016.

[50] Lijun Yu and Yong Cheng. Spae: Semantic pyramid autoencoder for multimodal generation with frozen llms. *arXiv preprint arXiv:2306.17842*, 2023.

[51] Baoquan Zhang, Huaibin Wang, Chuyao Luo, Xutao Li, Guotao Liang, Yunming Ye, Xiaochen Qi, and Yao He. Codebook transfer with part-of-speech for vector-quantized image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7757–7766, 2024.

[52] Jiahui Zhang and Fangneng Zhan. Regularized vector quantization for tokenized image synthesis. In *CVPR*, pages 18467–18476, 2023.

[53] Chuanxia Zheng and Andrea Vedaldi. Online clustered codebook. In *ICCV*, pages 22798–22807, 2023.

[54] Yufan Zhou, Bingchen Liu, Yizhe Zhu, Xiao Yang, Changyou Chen, and Jinhui Xu. Shifted diffusion for text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10157–10166, 2023.

[55] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Towards language-free training for text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17907–17917, 2022.

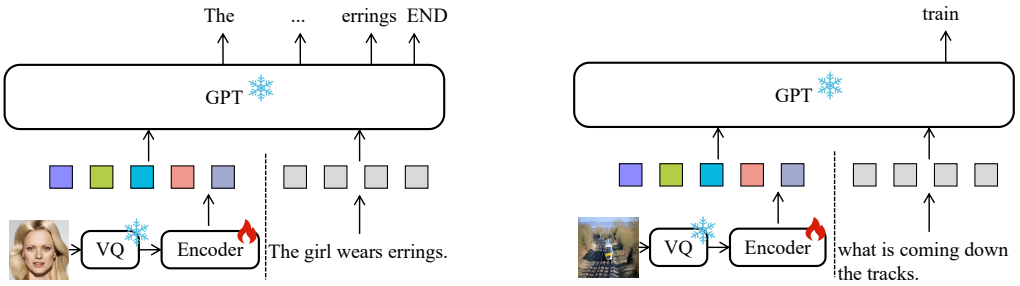
[56] Lei Zhu and Fangyun Wei. Beyond text: Frozen large language models in visual signal comprehension. *arXiv preprint arXiv:2403.07874*, 2024.

 https://doi.org/10.52202/079017-4433

A Appendix / supplemental material

A.1 Experiment Details

Semantic image synthesis, unconditional generation and image completion: All models follow the default setting of VQ-GAN-Transformer[2]. Specifically, the vocabulary size, embedding number, and input sequence length are 1024, 1024, and 512, respectively. The layers and heads of the transformer are both 16. The semantic image synthesis experiments are conducted on 1 4090 GPU with a batch size of 15 and one day of training time. The unconditional generation and image completion experiments are conducted on 2 4090 GPUs with a batch size of 36 and one day of training time.
Text-to-image generation: All models follow the default setting of VQ-Diffusion[3]. Specifically, the layers of the transformer are 19 with dimension of 1024. The diffusion step is 100. The training epoch is 90 for all models. The experiments are conducted on 1 4090 GPU with a batch size of 24 and two days of training time.
Image captioning: Inspired by ClipCap [27], we use the trained codes to replace the ClipCap's prefix embeddings. The model framework is shown in Fig. 9 (a). The training epoch is 100 for all models. The experiments are conducted on 2 4090 GPUs with a batch size of 60 and one day of training time.
Visual question answering: The COCO-QA [35] dataset is automatically generated from captions in the Microsoft COCO dataset [20]. There are 78,736 train questions and 38,948 test questions in the dataset, These questions are based on 8,000 and 4,000 images respectively. There are four types of questions including object, number, color, and location. Each type takes 70%, 7%, 17%, and 6% of the whole dataset, respectively. All answers in this data set are single word. Following the image captioning task, we use the last hidden embedding to do VQA, as shown in Fig. 9 (b). Following the [24], we report classification accuracy and Wu-Palmer similarity (WUPS). The training epoch is 50 for all models. The experiments are conducted on 2 4090 GPUs with a batch size of 60 and one day of training time.
Visual Grounding: The refcoco dataset [49] includes 19,994 images with 50,000 referred objects. Each object has more than one referring expression, and there are 142,210 referring expressions in this dataset. There are two commonly used split protocols for this dataset. One is RefCOCOg-google [25], and the other is RefCOCOgumd [28]. We follow RefCOCOgumd [28] to split the dataset. The train set has 42,404 expressions, the validation set has 3,811 expressions, and the test set has 3,785 expressions. Following [5], we concatenate the image codes and text tokens and feed them into a learnable transformer with coordinate regression layers (*i.e.*, FNN) to predict the object box. The training epoch is 100 for all models. The experiments are conducted on 2 4090 GPUs with a batch size of 30 and several hours of training time.



(a) Image captioning (b) VQA

Figure 9: The architecture of the visual text reasoning based on GPT.

[2]https://github.com/CompVis/taming-transformers
[3]https://github.com/microsoft/VQ-Diffusion

https://doi.org/10.52202/079017-4433

A.2 Experimental comparison with VQCT

We provide comparisons with VQCT [51] for image reconstruction in Table 12. Moreover, we also provide experimental results on further integrating our method into VQCT to verify its effectiveness and versatility. The results are shown in Table 13.

Table 12: Comparison of VQCT [51] and our method on reconstruction

Model	Codebook Size	#Tokens	CelebA-HQ	CUB-200	MS-COCO
VQCT	6207	512	5.02	**2.13**	9.82
VQ-GAN+LG	1024	256	5.34	3.08	10.72
CVQ+LG	1024	256	**4.90**	3.33	**9.69**

Table 13: Comparison of **reconstruction and VQA** on VQCT and VQCT+LG on the MS-COCO dataset.

	Image Reconstruction	VQA
Model	FID↓	Accuracy↑
VQCT	9.82	40.42
VQCT+LG	**9.57**	**40.64**

A.3 More Examples and Qualitative Results

We provide more examples of image reconstruction in Fig. 10, image-to-text retrieval in Fig. 11 and Fig. 12. We also provide more image synthesis results in Fig. 13 for semantic image synthesis, and text-to-image synthesis in Fig. 14. We provide some examples of image captioning in Fig. 15 and VQA in Fig. 16. We provide some examples of unconditional generation in Fig. 18, and image completion in Fig. 17. We also provide a qualitative comparison of visual grounding in Fig. 19.

Figure 10: Reconstruction from different models on four datasets. The red-color boxes highlight reconstruction details.

https://doi.org/10.52202/079017-4433

Top-5 retrieved text

a book with worn pages is open to page 160.
a book that is opened to page number 66
a book with pages that look aged, is open to a page that says the funeral at the top.
a book is opened up to a page with the heading first part.
a book is open to a page about book binding.

Ground Truth text

chapter of some book that is titled as the funeral.
open book on a page that says the young man dried up his tears.
the book is open to page 160 where the section header reads the funeral.
a book with pages that look aged, is open to a page that says the funeral at the top.
a book with worn pages is open to page 160.

Top-5 retrieved text

two samsung computer monitors with a blue screen showing "connecting".
a hyundai computer monitor with windows xp on the screen
a dell monitor is open to a blue sign in screen
white text appears on a blue background of a hyundai monitor.
a macbook air computer displaying a windows update screen.

Ground Truth text

a computer screen with the login page for windows 7
a dell monitor is open to a blue sign in screen
a dell computer is currently running windows 7 professional.
a dell monitor shows the log in screen for windows 7 professional.
dell monitor that shows the login and password to enter.

Top-5 retrieved text

a gold color coin depicting miguel del valle.
a closeup of the label on a bottle of louis roederer champagne.
a golden box of godiva chocolates wrapped with a gold ribbon.
an ornate bottle with yellow and gold has chambord across it.
bottle of veuve clicquot brut champagne from france.

Ground Truth text

a bottle of louis roederer champagne, brut premier.
a closeup of the label on a bottle of louis roederer champagne.
a label for louis roederer brut premier champagne.
the "louis roederer" champagne was created in reims, france.
the name louis is on the bottle of liquid

Top-5 retrieved text

closeup of the sign outside ben's chili bowl in bright yellow and red.
a yellow sign for beacon lighting hanging above the door.
yellow sign for station dunlop surrounding a clock with roman numerals
red and yellow sign for a restaurant which says ben's chili bowl.
a sign in yellow with the words beacon lighting in black.

Ground Truth text

a ben's chili bowl sign that is outside
red and yellow sign for a restaurant which says ben's chili bowl.
red and yellow sign that says ben's chilli bowl.
closeup of the sign outside ben's chili bowl in bright yellow and red.
a sign reading ben's chili bowl is lit up

Figure 11: Examples of the top-5 most similar text selected on Textcaps based on VQ-GAN+LG. The bold text means the same as the ground truth result.

Top-5 retrieved text

this is a black bird with a red crown on its head.
a large black bird with a long bill and bright red crown.
a black bird with a bright red crown and a pointed beak.
this is a black bird with a black and white head and a red nape.
this is a large black bird with a black cheek patch and a red crown.

Ground Truth text

the body of the bird is black while the crown of the bird is red.
this is a black bird with a red crown on its head.
this is a black bird with a black and white head and a red nape.
this bird has wings that are black and has a red crown
a small bird with a red crown and a pointed beak.

Top-5 retrieved text

a greenish yellow bird with a solid black crown and eyering.
a little yellow bird with green wings and tail and a black crown.
a yellow bird with green on the crown and black eyering
a yellow and green bird that has black throat and crown, yellow breast and belly, yellow cheek patch, and green wings and tail.
a bright yellow bird with green wings and nape, black crown, and black cheek patches.

Ground Truth text

a small bird with a black crown and a yellow breast and a small bill
this is a yellow bird with a blue crown and a black eyering
a greenish yellow bird with a solid black crown and eyering.
this bird has wings that are green and has a yellow belly
a small yellow bird, with a black crown, and short beak.

Top-5 retrieved text

a bright orange bird with black wings, black eyes and an orange beak.
a small bright orange bird with dark black wings and tail.
a colorful orange bird with black wings and tail, orange crown, black throat and a long black beak.
this is a beautiful orange bird with black wings and tail
this bright orange bird has black wings and a short pointed black bill.

Ground Truth text

a bright orange bird with black wings, black eyes and an orange beak.
this is an orange bird with a black wing and a long white beak.
this small bird has an orange belly and an orange back, with a yellow breast.
this bird is red with black and has a very short beak.
this bird has wings that are gray and has an orange belly

Top-5 retrieved text

a yellow and black bird with a black head, a yellow body, black wings with white wingbars, and a short, thin black bill.
a bright yellow and black bird perched on the end of a branch.
a bright yellow bird, with black primaries and a black crown.
a colorful bird with a yellow head, black neck and yellow body, with black wings.
long yellow and black downward curved beak on yellow and black feather covered bird.

Ground Truth text

a yellow and black bird with a black head, a yellow body, black wings with white wingbars, and a short, thin black bill.
this bird has a black crown, a yellow belly, and a short bill
the bird has a black crown and a yellow breast and belly.
this bird has wings that are black and grey and has a yellow belly
this bird has wings that are black and has a yellow belly

Figure 12: Examples of the top-5 most similar text selected on CUB-200 based on VQ-GAN+LG. The bold text means the same as the ground truth result.

https://doi.org/10.52202/079017-4433

Figure 13: Semantic image synthesis on CelebA-HQ.

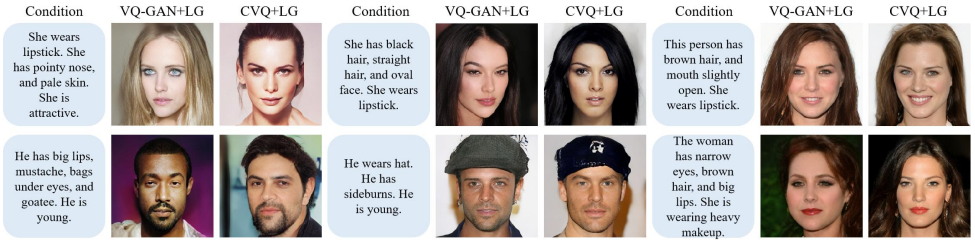


Figure 14: More text-to-image generation on CelebA-HQ.

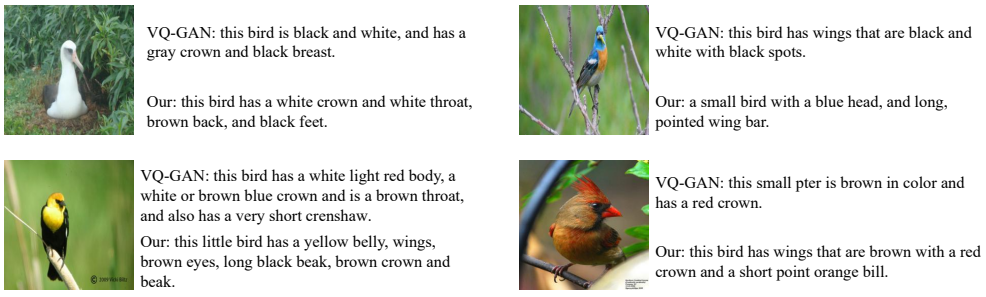


Figure 15: Image Captioning on CUB-200 based on VQ-GAN and VQ-GAN+LG.

https://doi.org/10.52202/079017-4433

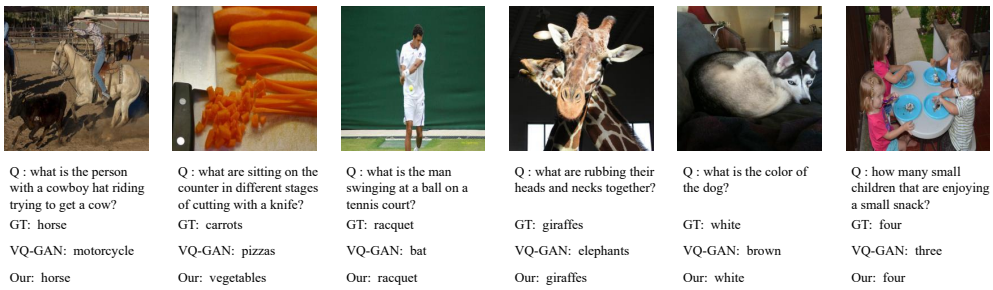


Figure 16: VQA on COCO-QA based on VQ-GAN and VQ-GAN+LG.

Figure 17: Image completion on CelebA-HQ.

Figure 18: Examples of **unconditional image generation** on CelebA-HQ based on VQ-GAN+LG.

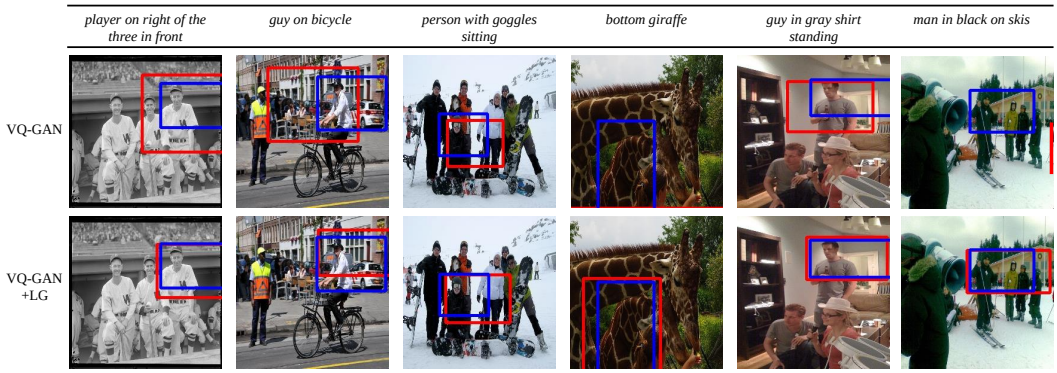


Figure 19: Examples of **visual grounding** on refcoco. Blue boxes are the ground-truth, red boxes are the model predictions.

https://doi.org/10.52202/079017-4433

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See the last paragraph in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See the second paragraph in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our work mainly involves empirical contributions.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide our code in the supplementary materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

https://doi.org/10.52202/079017-4433

Answer: [Yes]

Justification: We provide our code and dataset information in the supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide detailed experimental settings in Section 4.1 and Appendix A.1. In the supplementary materials, we provide our code, which contains more details of the model and parameters.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: See the third paragraph "Visual Question Answering" in Section 4.4.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We discuss our model's computational resources in Appendix A.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See the last paragraph in Section 5.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

https://doi.org/10.52202/079017-4433

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We are using publicly available datasets for all experiments.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all used public datasets in Section 4 and Appendix A.1. All datasets are publicly available. They are under a non-commercial license.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our code and pre-trained model will be released later for the assets. We are using publicly available datasets for all experiments. No personally identifiable information is involved.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No human subjects were involved in our experiments.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects were involved in our experiments.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-4433