GLIGEN: Open-set Grounded Text-to-Image Generation

Objective

Grounded Text-to-Image generation

Grounded Text-to-Image Generation

Condition the image generation with additional *grounding* condition, which specifies the spatial configuration of the object. The grounding condition might include

- Bounding box
- Keypoints
- Spatial-aligned condition: edge map, depth map, normal map, semantic map, etc.

Latent Diffusion Model (LDM)

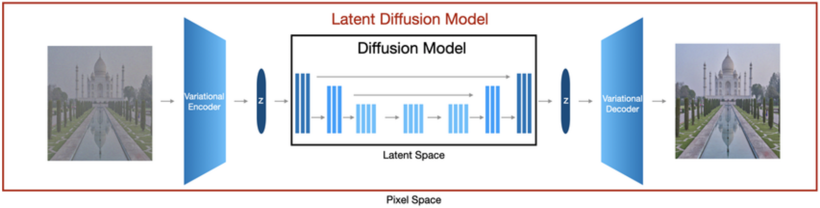


Figure 1: Latent Diffusion Model

Two stage image generation:

1. Latent representation $\mathbf{z}$ of an image $\mathbf{x}$
2. Diffusion model on the latent representation $\mathbf{z}$

Grounding Instruction Input

Instruction: $\mathbf{y} = (\mathbf{c}, \mathbf{e})$

- Caption: $\mathbf{c} = [c_1, \ldots, c_L]$
- Grounding: $\mathbf{e} = [(e_1, \mathbf{l}_1), \ldots, (e_N, \mathbf{l}_N)]$

where e is the semantic information of grounding entity, and $\mathbf{l}$ is the grounding spatial configuration.

Caption Token

The caption $\mathbf{c}$ is processed in the same way as in LDM:

$$\mathbf{h}^c = [h_1^c, \ldots, h_L^c] = f_{\textsf{text}}(\mathbf{c})$$

Grounding Token

Given an entity e and its grounding configuration $\mathbf{l}$, grounding information is processed by the same text encoder as with the caption token:

$$h^e = \mathsf{MLP}(f_{\mathsf{text}}(e), \mathsf{Fourier}(\mathbf{l})).$$

Then with N entities, the grounding token $\mathbf{h}^e$ is

$$\mathbf{h}^e = [h_1^e, \ldots, h_N^e].$$

Prior Works

Prior works only deals with a *closed-set* setting, where they have fixed number, say K, of concepts to consider. Typically, such concepts are encoded through a learned vector embeddings. In other words, $f_{\text{text}}(e)$ is replaced by a look-up table of K embeddings $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$. This approach has two major drawbacks:

1. Model can only ground the observed K entities in the generated image
2. No word or phrase is used in the model conditioning

From Closed-set to Open-set

GLIGEN uses a shared text encoder for both caption and grounding entity. Hence model can generate grounded entities that are not contained in the training dataset.

Architecture

To fully utilize the capability of large diffusion models, which is trained with web-scale large dataset, the authors propose to add an additional module to a frozen pretrained LDM.

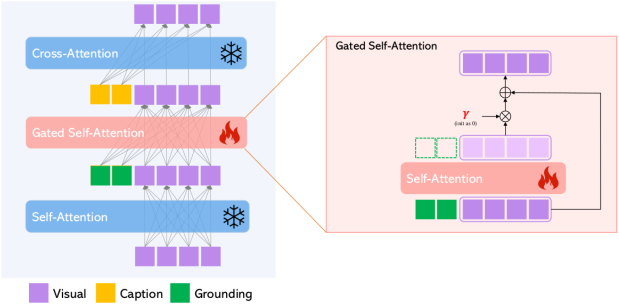


Figure 2: Gated self-attention

Attention Blocks

Let $\mathbf{v} = [v_1, \ldots, v_M]$ be the visual feature tokens of an image. The original attention blocks (or the transformer blocks) contains

- Self-attention layers for the visual tokens

$$\mathbf{v} = \mathbf{v} + \mathsf{SelfAttn}(\mathbf{v})$$

- Cross-attention layers for both visual and caption tokens

$$\mathbf{v} = \mathsf{CrossAttn}(\mathbf{v}, \mathbf{h}^c)$$

Gated Self-Attetion

Additional to two (frozen) attention layers, the authors add a new gated self-attention laeyrs inbetween to process the grounding condition:

$$\mathbf{v} = \mathbf{v} + \beta \cdot \mathsf{tanh}(\gamma) \cdot \mathsf{TS}(\mathsf{SelfAttn}([\mathbf{v}, \mathbf{h}^e])).$$

where

- β: magnitude of the grouding condition (default: $\beta = 1$ during training)
- γ: learnable scalar (default: $\gamma = 0$)
- $\mathsf{TS}(\cdot)$: token selection that selects only the visual tokens

Learning

$$\min_{\theta'} \mathcal{L}_{\mathsf{Grounding}} = \mathbb{E}_{\mathbf{z},\epsilon\sim\mathcal{N}(0,I),t}\left[\|\epsilon - f_{\theta,\theta'}(\mathbf{z}_t, t, \mathbf{y})\|_2^2\right]$$

Sampling

Setting $\beta = 1$ during inference yields suboptimal image generation quality. To cope with this the authors propose to use the scheduled sampling in inference:

$$\beta = \begin{cases} 1, & t \leq \tau * T \\ 0, & t > \tau * T \end{cases}$$

where

- T is the total number of timesteps
- $\tau \in [0, 1]$ is the hyperparameter for choosing the two-stage inference.

Experiments

Closed-set Grounded Text2Img Generation

Dataset

- COCO2014D: Detection Data
- COCO2014CD: Detection + Caption Data
- COCO2014G: Grounding Data

Evaluation metrics

- FID: measures image quality
- YOLO score: measures the grounding accuracy

Experiments

Model	Generation: FID (↓) Fine-tuned	Generation: FID (↓) Zero-shot	Grounding: YOLO (↑) $AP/AP_{50}/AP_{75}$
CogView [11]	-	27.10	-
KNN-Diffusion [2]	-	16.66	-
DALL-E 2 [51]	-	10.39	-
Imagen [56]	-	7.27	-
Re-Imagen [7]	5.25	6.88	
Parti [74]	3.20	7.23	-
LAFITE [82]	8.12	26.94	-
LAFITE2 [80]	4.28	8.42	-
Make-a-Scene [13]	7.55	11.84	-
NÜWA [69]	12.90	-	-
Frido [12]	11.24	-	-
XMC-GAN [77]	9.33	-	-
AttnGAN [70]	35.49	-	-
DF-GAN [65]	21.42	-	-
Obj-GAN [35]	20.75	-	-
LDM [53]	-	12.63	-
LDM*	5.91	11.73	0.6 / 2.0 / 0.3
GLIGEN (COCO2014CD)	5.82	-	21.7 / 39.0 / 21.7
GLIGEN (COCO2014D)	5.61	-	**24.0 / 42.2 / 24.1**
GLIGEN (COCO2014G)	6.38	-	11.2 / 21.2 / 10.7

- GLIGEN can successfully take the grounding conditions
- All grounding instruction types are useful

Experiments

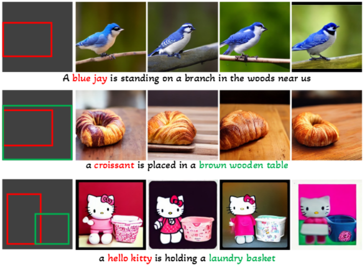
Model	FID (↓)	YOLO score (AP/AP$_{50}$/AP$_{75}$) (↑)
LostGAN-V2 [62]	42.55	9.1 / 15.3 / 9.8
OCGAN [64]	41.65	-
HCSS [25]	33.68	-
LAMA [40]	31.12	13.40 / 19.70 / 14.90
TwFA [71]	22.15	- / 28.20 / 20.12
GLIGEN-LDM	**21.04**	**22.4 / 36.5 / 24.1**

GLIGEN beats prior works on the layout2img.

Experiments

Open-set Grounded Text2Img Generation

Qualitative



Quantitative

Model	Training data	AP	AP_r	AP_c	AP_f
LAMA [40]	LVIS	2.0	0.9	1.3	3.2
GLIGEN-LDM	COCO2014CD	6.4	5.8	5.8	7.4
GLIGEN-LDM	COCO2014D	4.4	2.3	3.3	6.5
GLIGEN-LDM	COCO2014G	6.0	4.4	6.1	6.6
GLIGEN-LDM	GoldG,O365	10.6	5.8	9.6	13.8
GLIGEN-LDM	GoldG,O365,SBU,CC3M	11.1	9.0	9.8	13.4
GLIGEN-Stable	GoldG,O365,SBU,CC3M	10.8	8.8	9.9	12.6
Upper-bound	-	25.2	19.0	22.2	31.2

Thank You

Q & A