DEPENDENCY-AWARE DISCRETE DIFFUSION FOR SCENE GRAPH GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Scene graphs (SGs) represent objects and their relationships as structured graphs, enabling applications in image generation, robotics, and 3D understanding. Recent work suggests that conditioning image generation on scene graphs improves compositional fidelity compared to text-only prompting. However, since users typically provide text rather than structured graphs, a key challenge is to generate scene graphs from natural language. Prior work on discrete diffusion has demonstrated success in generating generic graphs such as molecules and encoding dependencies indirectly via asynchronous generation schedules, but does not account for the hierarchical structure and strong dependencies between objects, edges, and relations in scene graphs directly in the graph state. We address this limitation by introducing a dependency-aware, hierarchically constrained discrete diffusion model for scene graph generation. Our approach decouples structure and semantics across the forward and reverse processes, enabling the model to capture conditional dependencies. At inference time, we perform training-free conditioning to sample text-aligned scene graphs. We evaluate our method on standard SG benchmarks and demonstrate improvements over both continuous and discrete graph generation baselines across graph and layout metrics. When fed to downstream image generation, our approach yields improved compositional alignment compared to text-to-image models, particularly in multi-object scenarios.

1 INTRODUCTION

Text-to-image (T2I) models have made remarkable progress in generating images from user-specified prompts, but user expectations are rising just as quickly. As prompts become longer and more complex, multiple lines of attack are being explored to improve compositional fidelity and alignment. One promising direction is to generate images from scene graphs (SGs) (see Fig. 1) where visual objects are represented as nodes and their relationships as edges. Recent papers Xu et al. (2025); Farshad et al. (2023); Thanh-Nhan et al. (2025); Shen et al. (2025) suggest that SG-conditioned image generation $p(I|G)$ can be more effective, particularly in compositional settings. While these gains are largely empirical, they are intuitively justified: text-based conditioning lacks explicit relational constraints, often leading to compositional inconsistencies. In contrast, scene graphs provide an explicit factorization over objects and relations, offering a structured prior that is difficult to encode in text embeddings.

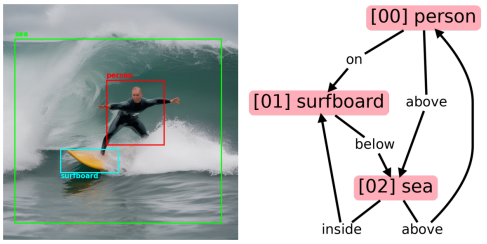

Figure 1: Image I and its scene graph G. Objects are pink nodes and relations are labels on edges.

Given this momentum, a natural question arises: how can scene graphs (SG) be generated from text prompts? While text-to-SG mapping may appear deterministic, it is inherently multi-modal due to the ambiguity and under-specification of natural language. For example, a prompt such as "a person and a surfboard" does not specify their relationship; the person could be on, beside, or holding the surfboard. Multiple valid SGs may correspond to the same prompt, necessitating modeling a conditional distribution $p(G|T)$ rather than predicting a single deterministic output.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Motivated by this, we propose to learn the manifold of scene graphs using discrete diffusion models, and then perform conditional generation of SGs from text prompts. However, this presents a fundamental challenge: scene graphs are significantly more complex than generic graphs. They exhibit long-tailed vocabularies, strong conditional dependencies between objects and relations, and global structural constraints. Injecting these properties into discrete diffusion remains underexplored.

A key limitation of existing discrete diffusion works like Vignac et al. (2023) is that they treat nodes and edges as independent categorical variables. With scene graphs, such independence assumptions break due to the inherent conditional dependencies, such as the hierarchical structure where objects determine edges and edges determine relations. Recent works GraphMaker Li et al. (2024a), MolDiff Chen et al. (2024), etc., encode dependencies indirectly via asynchronous corruption schedules without fundamental changes to the graph state space. Our key insight is that scene graphs require a *hierarchically constrained discrete diffusion process*, where structure and semantics are modeled with explicit conditional dependencies rather than independent categorical transitions.

We instantiate this insight by designing ❶ a hierarchical graph representation $G(V, E, R^+)$ in which relations are defined only on active edges (i.e., $e_{ij} = 0 \implies r_{ij} = 0$), ❷ an edge-gated forward noising process that respects this structure, and ❸ a factorized reverse sampler that generates objects, edges, and relations in a dependency-aware sequence:

$$p_\theta(\hat{V}_0, \hat{E}_0, \hat{R}_0 \mid x_t) = p_\theta(\hat{V}_0 \mid x_t) p_\theta(\hat{E}_0 \mid \hat{V}_0, x_t) p_\theta(\hat{R}_0 \mid \hat{E}_0, \hat{V}_0, x_t)$$

Finally, ❹ at inference time, we perform training-free conditioning via reward-tilted sampling, enabling the generation of SGs aligned with the input text, $p(G|T)$. We employ particle resampling (Sequential Monte Carlo): $p_\theta(G_{t-1}|G_t; T) \propto p_\theta(G_{t-1}|G_t) \exp(\beta R(\hat{G}_0, T))$, with CLIP's similarity metric Radford et al. (2021) to compute the reward $R(\hat{G}_0, T)$. In sum, our main contribution lies in replacing standard independent categorical diffusion with a dependency-aware hierarchical diffusion process tailored to the unique requirements of scene graphs. Fig. 2 illustrates our operation pipeline.

We implement our model **DiScGraph** using a Graph Transformer Dwivedi & Bresson (2021) as the denoising network and train on standard scene graph datasets including Visual Genome Krishna et al. (2016), COCO Lin et al. (2015), and LAION-SG Li et al. (2024b). We evaluate our approach on (i) graph metrics such as entity-MMD, Triplet-TV, and Rare-K TV, (ii) grounding metrics based on layout IoU, and (iii) image generation metrics including ImageReward, BLIP-VQA, and SG-IoU. Baselines include graph generation (DiGress Vignac et al. (2023), DiffuseSG Xu et al. (2025), etc.), layout models (LayoutDM Lian et al. (2024), LLMLayout-T2I Qu et al. (2023)), and text-based compositional image generators (CO3 Dutta et al. (2026) and ComposeDiffusion Liu et al. (2023)) on COCO-validation and CompSGBench Li et al. (2024b) benchmarks. Across these benchmarks, our method outperforms both continuous SG generation models and prior discrete graph baselines while achieving competitive performance on layout prediction. Finally, when the generated scene graphs are used to condition downstream image generation models, we observe improved alignment compared to text-only T2I systems, particularly in compositional and multi-object scenarios.

2 PRELIMINARIES

Discrete Diffusion. Continuous diffusion Song et al. (2021); Sohl-Dickstein et al. (2015), typically using a Gaussian noising process, is not well-suited for categorical or structured data such as graphs. This motivates diffusion directly in discrete state spaces. In discrete diffusion Austin et al. (2021), each entity takes values in a finite set of K categories. The forward process is defined via transition matrices $\{Q_t\}_{t=1}^T$, where, $q(x_t \mid x_{t-1}) = x_{t-1}Q_t$, and the marginal at time t is given by, $q(x_t \mid x_0) = x_0\bar{Q}_t, \quad \bar{Q}_t = \prod_{s=1}^t Q_s$. A common choice of Q_t yields a uniform stationary distribution. The posterior used for training admits a closed form, $q(x_{t-1} \mid x_t, x_0) \propto \left(x_t Q_t^\top\right) \odot \left(x_0 \bar{Q}_{t-1}\right)$.

Discrete Diffusion on Graphs. DiGress Vignac et al. (2023) extends discrete diffusion to graphs by modeling node and edge attributes as independent categorical variables. A graph is represented as $G = (V, E)$, and the forward process is defined by, $q(G_t \mid G_{t-1}) = \left(V_{t-1}Q_t^V,\ E_{t-1}Q_t^E\right)$, with closed-form marginals, $q(G_t \mid G_0) = \left(V_0\bar{Q}_t^V,\ E_0\bar{Q}_t^E\right)$. The reverse model predicts clean node and edge distributions, and sampling proceeds independently across nodes and edges,

$$p_\theta(G_{t-1} \mid G_t) = \prod_i p_\theta(v_{t-1}^i \mid G_t) \prod_{i,j} p_\theta(e_{t-1}^{ij} \mid G_t). \tag{1}$$

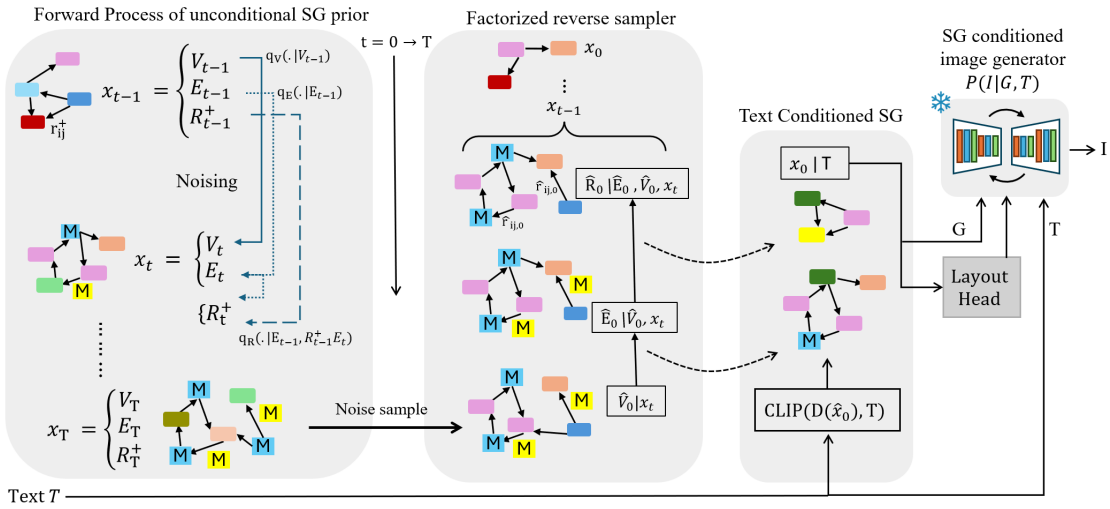

Figure 2: **DiScGraph** pipeline: (1) Forward noising process in which relations are edge-gated. (2) Factorized reverse sampling that generates dependency-aware objects, edges and relations. (3) Reward-tilting at inference using the CLIP similarity with the text prompt as reward; a layout head also generates bounding boxes for graph nodes. (4) SG and text conditioned off-the-shelf image generator.

Each term is computed by marginalizing over predicted clean states (nodes & edges),

$$p_\theta(v_{t-1}^i \mid G_t) = \sum_{v_0} q(v_{t-1}^i \mid v_0, x_t^i)\, \hat{p}_V^i(v_0), \qquad p_\theta(v_{t-1}^i \mid v_0, G_t) = \begin{cases} q(v_{t-1}^i \mid v_0, x_t^i), & \text{if } q(x_t^i \mid v_0) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

This formulation enforces permutation invariance, and treats nodes and edges as independent categorical variables. However, it assumes a categorical space where edge existence and semantics, edge identity are coupled, which is unsuitable for scene graphs' strong inter-entity dependence, large relation vocabularies, and long-tailed relation distribution.

3 Method

DiScGraph's redesign of discrete diffusion entails 4 parts, starting with a (1) hierarchical representation of SGs, (2) an edge-gated forward noising process (3) a factorized reverse sampler for unconditional generation, and (4) a text-conditioned SG generation at inference time (see Fig. 2).

3.1 Hierarchical Scene Graph Representation

In discrete diffusion, a graph is represented as a flat categorical state $G = (V, E)$, where node labels V and edge labels E are independently modeled Vignac et al. (2023). While this representation is effective for, say, molecules or transportation graphs (where edge relations are defined over a small, well-defined vocabulary with strict constraints), SGs involve large, long-tailed relational vocabularies and strong object–relation dependencies. Modeling the edges with a single categorical variable, say $e_{ij} \in \{noEdge, r_1, r_2, \ldots, r_K\}$, biases the learning towards edges rather than relations, especially for sparse scene graphs, large K, and the long-tailed distribution over relation classes r_k. Scene graphs are better modeled by separating edges (structure) from the relations (semantics), so that *relations can be conditioned only on active edges*, i.e., $e_{ij} = 0 \Rightarrow r_{ij} = 0$. This motivates our decoupled representation in which each edge is factorized into two: (i) edge existence, $e_{ij} \in \{0, 1\}$, and (ii) relation identity, $r_{ij}^+ \in \{1, \ldots, K_{rel}\}$. We define valid scene graph space as:

$$\mathcal{G}_{SG} = \{(V, E, R^+) : e_{ij} = 0 \Rightarrow r_{ij} = 0\}, \qquad \big(V, E, R^+\big) = \big((v_i), (e_{ij}), (r_{ij}^+)\big)_{i,j=1}^{N} \tag{3}$$

$$v_i \in \mathcal{V} = \{1, \ldots, K_{\text{obj}}\}, \quad e_{ij} \in \{0, 1\}, \quad r_{ij}^+ \in \{1, \ldots, K_{\text{rel}}\}. \tag{4}$$

Thus the abstract graph state space $\mathcal{G} := \mathcal{V}^N \times \{0, \ldots, K_{rel}\}^{N \times N}$ is modified to the constrained scene graph space, $\mathcal{SG} := \mathcal{V}^N \times \{0, 1\}^{N \times N} \times \{1, \ldots, K_{rel}\}^{N \times N}$. The factorized state explicitly encodes: (i) directedness, (ii) sparsity via E, (iii) semantic validity via R^+. This representation underlies all subsequent modeling components.

3.2 STRUCTURE-AWARE FORWARD PROCESS

We now define a discrete diffusion process over the factorized state as,

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}), \quad x_t = (V_t, E_t, R_t^+). \tag{5}$$

The forward corruption is redesigned for the factorized transition over the scene graph space $\mathcal{SG}$,

$$q(x_t \mid x_{t-1}) = q_V(V_t \mid V_{t-1})\, q_E(E_t \mid E_{t-1})\, q_R(R_t^+ \mid R_{t-1}^+, E_t, E_{t-1}). \tag{6}$$

where V and E corruptions follow standard discrete diffusion forward process Austin et al. (2021); Vignac et al. (2023). **Relation corruption (edge-aware).** Relations are only defined on active edges, so corruption must depend on edge state, as follows,

$$q_R(R_t \mid R_{t-1}, E_t) = \prod_{i \neq j} q_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t}). \tag{7}$$

$$q_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t}) = \begin{cases} \delta_0(r_{ij,t}), & e_{ij,t} = 0, \\ \bar{q}_R(r_{ij,t} \mid r_{ij,t-1}), & e_{ij,t} = 1, \end{cases} \tag{8}$$

For active edges ($e_{ij} = 1$) with relation prior π_R[1],

$$\tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) = (1 - \beta_t^R)\delta_{r_{ij,t-1}} + \beta_t^R \pi_R, \tag{9}$$

$$q_R(R_t \mid R_{t-1}, E_t) = \prod_{ij:e_{ij,t}=1} \tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) \prod_{ij:e_{ij,t}=0} \delta_0(r_{ij,t}). \tag{10}$$

Hence, the combined corruption process is:

$$q(x_t \mid x_{t-1}) = \prod_i q_V(v_{i,t} \mid v_{i,t-1}) \prod_{i \neq j} q_E(e_{ij,t} \mid e_{ij,t-1}) \prod_{ij:e_{ij,t}=1} \tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) \prod_{ij:e_{ij,t}=0} \delta_0(r_{ij,t}) \tag{11}$$

We derive the stationary distribution for the factorized discrete state in the Appendix Sec. A.1.1 and provide closed-form expressions for the marginals $q(x_t|x_0)$ in the Appendix Sec. A.1.2. In abstract graph works, the categorical edge prior is $\pi_E = \{\pi_E(0), \pi_E(1), \ldots, \pi_E(K_{rel})\}$, but due to sparsity, large K_{rel}, and long-tailed relation distribution in scene graphs, $\pi_E(0) \gg \pi_E(r), r \in \{1, \ldots, K_{rel}\}$. Thus, without factorization, under random corruption, many relations are pushed towards "no-edge". This implies that rare relations compete against both frequent relations and the overwhelmingly frequent no-edge class. In contrast, the factorized corruption happens over $\{1, \ldots, K_{rel}\}$ instead of $\{0, 1, \ldots, K_{rel}\}$. Thus, rare relations no longer compete with edge absence during relation denoising, resulting in better relation modeling.

In addition to random corruption, we also apply absorbing mask corruption on V and R^+, which preserves the SG structure and only hides the semantics information, allowing the model to learn SG semantics. For mask tokens $[\mathrm{MASK}]_V$ and $[\mathrm{MASK}]_R$, we employ a hybrid random + mask corruption strategy. The hybrid corruption kernel for objects is as follows (with more details in the Appendix Sec. A.1.3):

$$q_V(v_{i,t} \mid v_{i,t-1}) = (1 - \beta_t^V)\delta_{v_{i,t-1}} + \beta_t^V \left[\rho_V \delta_{[\mathrm{MASK}]_V} + (1 - \rho_V)\pi_V\right]. \tag{12}$$

3.3 FACTORIZED REVERSE SAMPLER

Given the constrained scene graph state at time t, $\{x_t = (V_t, E_t, R_t^+),\ e_{ij,t} = 0 \Rightarrow r_{ij,t} = 0\}$, we need to sample $x_{t-1} \sim p_\theta(x_{t-1} \mid x_t)$ while preserving this constraint. As in discrete diffusion, we parameterize the reverse process through a denoiser ϵ_θ prediction of the clean state x_0, and combine this prediction with the known forward kernels to obtain a posterior over x_{t-1}. For scene graphs, the clean state is $x_0 = (V_0, E_0, R_0)$. Following the structured state space, we factorize the clean-state prediction as

$$p_\theta(x_0 \mid x_t) = p_\theta(V_0, E_0, R_0 \mid x_t) = p_\theta(V_0 \mid x_t)\, p_\theta(E_0 \mid V_0, x_t)\, p_\theta(R_0 \mid V_0, E_0, x_t). \tag{13}$$

This factorization follows the semantic structure of scene graphs, where: (1) first, object identities define the semantic entities in the graph, then (2) directed edge existence determines which object pairs interact, and then (3) relation identities depend on objects and are meaningful only for active directed edges.
The combined object, edge, and relation posterior updates give the structured reverse transition is,

$$\begin{aligned} p_\theta(x_{t-1} \mid x_t) &= p_\theta(V_{t-1} \mid x_t)\, p_\theta(E_{t-1} \mid V_{t-1}, x_t)\, p_\theta(R_{t-1} \mid V_{t-1}, E_{t-1}, x_t). \\ &= \prod_i p_\theta(v_{i,t-1} \mid v_{i,t}, x_t) \prod_{i \neq j} p_\theta(e_{ij,t-1} \mid e_{ij,t}, V_{t-1}, x_t) \prod_{i \neq j} p_\theta(r_{ij,t-1} \mid r_{ij,t}, e_{ij,t-1}, V_{t-1}, E_{t-1}, x_t). \end{aligned} \tag{14}$$

[1]We slightly abuse notations and use R_t and R_t^+ interchangeably, but it implies relations on active edges.

DiScGraph redesigns the relation posterior to be compatible with the factorized state space $\mathcal{SG}$. This opens up three scenarios: (1) e_{t-1} does not exist, (2) both e_{t-1} and e_t exist, (3) e_{t-1} has come to exist,

$$p_\theta(r_{ij,t-1} \mid r_{ij,t}, e_{ij,t-1}, e_{ij,t}, x_t) = \begin{cases} \delta_0(r_{ij,t-1}), & e_{ij,t-1} = 0, \\ p_\theta^{\text{post}}(r_{ij,t-1} \mid r_{ij,t}, x_t), & e_{ij,t-1} = 1,\ e_{ij,t} = 1, \\ p_\theta^{\text{marg}}(r_{ij,t-1} \mid x_t), & e_{ij,t-1} = 1,\ e_{ij,t} = 0. \end{cases} \tag{15}$$

where $p_\theta^{\text{post}}(r_{ij,t-1} \mid r_{ij,t}, x_t)$ denotes the standard relation posterior distribution and $p_\theta^{\text{marg}}(r_{ij,t-1} \mid x_t)$ denotes the marginal distribution obtained by moving the denoiser's clean prediction to time $t-1$. Please refer to Appendix Sec. A.2.1 for the detailed derivation of the posterior. Thus, the factorized reverse sampler is the reverse-time counterpart of the structure-aware forward process. It uses the known discrete posterior for categorical diffusion, but modifies appropriately with *edge-gating*, needed for our constrained scene graph state.

In addition to the base sampler, we also explore three types of sampler refinements for robust sampling, such as (1) Split Gibbs refinement Vono et al. (2019); Chu et al. (2025), (2) Soft masked refinement, and (3) Rare relation refinement for improving semantic consistency, correcting uncertain predictions, and improving rare relation generation, respectively. Appendix Sec. A.2.2 provides details on each strategy and their schedules. Following the observations in Table 13, the factorized reverse sampler utilizes Split Gibbs ($T = 10 - 20$ early timesteps) and Soft-mask ($T = 90 - 100$, late timesteps) refinements.

3.4 Text-Conditioned Scene Graph Generation via Reward-Tilted Inference

The structured diffusion model described above defines an unconditional generative prior over scene graphs, $p_\theta(G_0)$, G_0 is the clean sampler output. We aim to generate scene graphs conditioned on a text prompt T, i.e., $P(G_0 \mid T)$. Instead of training a conditional model $p_\theta(G_0 \mid T)$, we perform *training-free alignment* by sampling from a reward-tilted distribution, following works like Kim et al. (2025); Yeh et al. (2024),

$$\tilde{p}(G_0 \mid T) \propto p_\theta(G_0)\, \exp\big(\beta R(G_0, T)\big), \tag{16}$$

where $R(G_0, T)$ is a CLIP Radford et al. (2021) similarity score between the scene graph and the text prompt, and $\beta > 0$ controls the strength of conditioning (Appendix Sec. A.3 gives more information on the reward). The reward depends only on the final state G_0, while sampling proceeds sequentially from $t = T$ to $t = 0$. So we can define intermediate timestep reward approximations $R_t := R(\hat{G}_0^{(t)}, T)$ using the denoiser's ϵ_θ predictions of the clean graph at t, $\hat{G}_0^{(t)} = \epsilon_\theta(G_t, t)$. **Tilted Sampling:** Ideally, we would sample from the tilted distribution,

$$\tilde{p}_\theta(G_{t-1} \mid G_t, T) \propto p_\theta(G_{t-1} \mid G_t)\, \exp\big(\beta R(\hat{G}_0(G_{t-1}), T)\big), \tag{17}$$

but this distribution is intractable, so we use Sequential Monte Carlo (See Kim et al. (2025) and Appendix Sec. A.3). Thus, by leveraging the intermediate clean-state predictions $\hat{G}_0 \mid G_t$, we can guide the reverse diffusion trajectory toward samples that are aligned with the input text, while generating valid scene graphs.

4 Experiments

■ **Datasets**: We train the scene graph generator on popular scene graph datasets: Visual Genome (VG) Krishna et al. (2016), COCO-Stuff Lin et al. (2015), and a large-scale dataset LAION-SG Li et al. (2024b). Both quantitative and qualitative results reported are from evaluations on VG/COCO-validation and CompSGBench Li et al. (2024b) benchmark, consisting of SGs describing complex multi-object-relation visual scenes.

■ **Image Generators**: For T2I generation task, we use SG-to-image generator SDXL-SG Li et al. (2024b), and Layout-to-image generator, GLIGEN Li et al. (2023).

■ **Baselines**: (1) *Graph Generators*: We compare generated unconditional SGs against generic diffusion graph generators trained on SG datasets: DiGress Vignac et al. (2023), GraphGDP Huang et al. (2022), D3PM Austin et al. (2021), autoregressive SG generators: SceneGraphGen Garg et al. (2021) and VarScene Verma et al. (2022), and diffusion SG work: DiffuseSG Xu et al. (2025). (2) *Layout Generators*: We compare layouts extracted from **DiScGraph**'s layout head against Transformer model: BLT Kong et al. (2022), discrete diffusion works: LayoutDiffusion Zhang et al. (2023a), LayoutDM Inoue et al. (2023) that generate discretized layouts trained on layouts from SG datasets, and DiffuseSG Xu et al. (2025) that generates layouts along with scene graphs. (3) *T2I Image Generation*: We compare text-conditioned SG to image generation performance against text-only compositional works: ComposedDiff Liu et al. (2023), CO3 Dutta et al. (2026) on SDXL, and base SDXL Podell et al. (2023). We also compare layout-to-image generation with LLM augmented works: LayoutLLM-T2I Qu et al. (2023), and LDM Lian et al. (2024).

■ **Metrics**: We use (1) *Unconditional Graph Metrics*: N-MMD, R-MMD, ID-MMD, OD-MMD, Attach-TV (AT-TV) Xu et al. (2025) (for baselines, relations and edges are synonymous). A triplet

is defined as <object, relation, subject>, and Triplet-TV measures if semantic consistency is preserved. RARE-K-TV (R-K-TV) measures the model's capability to generate long-tailed relations. (2) *Layout Metrics*: F1 scores between generated and ground truth layouts. (3) *Image Quality Metrics*: FID , CLIP-I2I , CLIP-I2T Radford et al. (2021), BLIP-VQA Huang et al. (2023), and ImageReward (IR) Xu et al. (2023). (4) Scene-graph-to-image metrics: SG-IoU , Entity-IoU , Relation-IoU Shen et al. (2024), that measure structural overlap between generated and real images; quantified via IoU of SGs, objects, and relations extracted by an LLM from the image pairs.

■ **Training Details**: We use the graph transformer Yun et al. (2020) backbone for the denoiser ϵ_θ, following DiGress Vignac et al. (2023). However, the embeddings and denoiser prediction heads are modified for the SG factorized formulation. Additionally, we also use a layout head that extracts layouts from sampled scene graphs. GIoU layout loss ensures a geometrically consistent scene graph generator. Please refer to Appendix Sec. A.4 for more details on metrics and implementation details.

4.1 RESULTS

DiScGraph's two main performance goals are: (1) scene graph generation from a learnt structural prior $P(G)$, and (2) improving text-to-image (T2I) generation with text-conditioned scene graphs.

❶ Scene Graph Generation

Unconditional scene graphs: Table 1 reports results on $N = 1000$ generated SGs. The reported distributional metrics, MMD and TV, analyze their statistics against ground truth. The corresponding qualitative scene graphs are visualized in Figures 5, 6, 7, 8. Results show that **DiScGraph** achieves superior performance across all graph metrics. Particularly, the factorized formulation boosts relation-learning significantly, as evidenced by the large gap between **DiScGraph** and the second-best baselines (DiffuseSG and DiGress) on Relation-MMD (R-MMD) and Attach-TV (AT-TV). R-MMD quantifies **DiScGraph**'s ability at learning true relation distribution, while AT-TV specifically measures semantic consistency via $p(R|V)$ i.e., given V, can a model predict right R. Please note that the ID/OD-MMD metrics are 0 for D3PM in Xu et al. (2025) as they are unavailable in the paper.

Table 1: Comparing graph methods on sampled graph metrics on val/test datasets (Best , 2nd best).

Method	N-MMD (↓)	R-MMD (↓)	ID-MMD (↓)	OD-MMD (↓)	TRIP-TV (↓)	R-K-TV (↓)	AT-TV (↓)
Visual Genome							
GraphGDP	$1.3e^{-2}$	$3.40e^{-2}$	$8.67e^{-2}$	$9.35e^{-2}$	0.8980	0.999	0.5904
D3PM[2]	$7.69e^{-3}$	$2.00e^{-2}$	$3.07e^{-2}$	$3.07e^{-2}$	0.8160	-	-
DiGress	$7.63e^{-3}$	$7.26e^{-3}$	$9.27e^{-3}$	$9.66e^{-3}$	0.7730	0.8622	0.4561
SceneGraphGen	$1.28e^{-2}$	$9.66e^{-3}$	$1.54e^{-2}$	$1.29e^{-2}$	0.9540	0.984	0.3264
VarScene	$3.68e^{-2}$	$2.55e^{-2}$	$2.45e^{-2}$	$2.51e^{-2}$	0.9880	0.997	0.2348
$\text{DiffuseSG}_{\text{sg}}$	$8.21e^{-3}$	$1.11e^{-2}$	$7.05e^{-3}$	$6.01e^{-3}$	0.6390	0.826	0.2916
$\text{DiScGraph}_{\text{sg}}$	$4.59e^{-3}$	$5.78e^{-3}$	$4.32e^{-3}$	$4.46e^{-3}$	0.4920	0.5194	0.1766
COCO							
GraphGDP	$1.55e^{-2}$	$7.13e^{-3}$	$5.00e^{-4}$	$7.00e^{-4}$	0.7855	0.982	0.5388
D3PM[2]	$4.92e^{-4}$	$1.29e^{-4}$	0	0	0.305	-	-
DiGress	$7.81e^{-4}$	$8.86e^{-4}$	$9.01e^{-6}$	$1.95e^{-5}$	0.4147	0.855	0.3775
SceneGraphGen	$6.06e^{-4}$	$9.92e^{-5}$	$9.00e^{-4}$	$9.00e^{-4}$	0.9486	0.977	0.2822
VarScene	$6.23e^{-2}$	0.104	0.1040	0.0994	0.8684	0.961	0.2902
$\text{DiffuseSG}_{\text{sg}}$	$3.93e^{-4}$	$6.45e^{-5}$	$8.47e^{-6}$	$9.91e^{-6}$	0.3106	0.814	0.2916
$\text{DiScGraph}_{\text{sg}}$	$3.97e^{-4}$	$5.39e^{-5}$	$6.37e^{-6}$	$5.76e^{-6}$	0.2100	0.6661	0.1204
LAION-SG							
DiGress	$3.42e^{-3}$	$9.78e^{-3}$	$7.35e^{-3}$	$8.00e^{-3}$	0.8188	0.9164	0.5526
$\text{DiffuseSG}_{\text{sg}}$	$1.36e^{-3}$	$9.66e^{-3}$	$6.50e^{-3}$	$6.90e^{-3}$	0.7025	0.8973	0.5440
$\text{DiScGraph}_{\text{sg}}$	$9.81e^{-4}$	$6.35e^{-3}$	$5.90e^{-3}$	$5.80e^{-3}$	0.5693	0.7746	0.4238

*Does **DiScGraph** learn rare relations when plausible object pairs are presented to it?* Rare-K-TV evaluates this behavior and results show that by decoupling structure E and semantics R, **DiScGraph** is uniquely suited for scene graphs' long-tailed relation distributions. Moreover, the In/Out Degree MMD metrics show that learning edge-existence structure is not compromised in favor of relations. The factorized sampler that predicts $V \rightarrow E \rightarrow R$ results in **DiScGraph**'s sampled graphs to adhere strongly to semantic consistency, thereby improving Triplet-TV (<object, relation, subject>). Thus, the factorized constrained discrete space and reverse sampler together ensure **DiScGraph** learns the language semantics (strong conditional dependencies) and the long-tailed vocabularies. This makes **DiScGraph** a more robust SG generator over a continuous diffusion baseline (DiffuseSG), and over a generic graph discrete framework (DiGress).

[2]We are unable to replicate the results of D3PM reported in Xu et al. (2025) so we report them as provided.

Table 2: Comparing layout methods w/ layout head output (Best , 2nd best).

Visual Genome				
Method	F1-std (↑)	F1-Area (↑)	F1-Freq (↑)	F1-Box (↑)
BLT	0.1673	0.3294	0.3716	0.7110
LayoutDiff.	0.1459	0.3095	0.3744	0.7716
LayoutDM	0.1496	0.2521	0.3951	0.7843
$\text{DiffuseSG}_{\text{lay}}$	0.1753	0.3296	0.3939	0.7210
DiScGraph$_{\text{lay}}$	0.1790	0.3302	0.3950	0.7540
COCO				
BLT	0.4166	0.5056	0.6280	0.8000
LayoutDiff.	0.3586	0.5099	0.6290	0.8040
LayoutDM	0.3719	0.5095	0.6330	0.8110
$\text{DiffuseSG}_{\text{lay}}$	0.4644	0.5153	0.6960	0.8023
DiScGraph$_{\text{lay}}$	0.4724	0.5371	0.7130	0.8010

Table 3: Comparing results of SG completion task (Best).

	Single Object (↑)				Single Relation (↑)			
Baselines	w_1	w_{10}	w_{50}	w_{100}	w_1	w_{10}	w_{50}	w_{100}
Visual Genome								
DiGress	8.5	14.9	62.7	86.6	9	27.8	64.1	88.8
$\text{DiffuseSG}_{\text{sg}}$	10.2	20.4	73.6	90	11.3	36.1	74.4	93.5
DiScGraph$_{\text{sg}}$	16.7	28.1	82.9	95.3	19.4	43.6	87	98.7
CompSGBench								
DiGress	5.9	10.9	55.5	82.5	7.7	19.8	50.1	70.5
$\text{DiffuseSG}_{\text{sg}}$	8.6	19.3	66.9	85	8.8	27.9	70.3	83.5
DiScGraph$_{\text{sg}}$	14.4	25.4	72.1	92.6	14	34.4	85.2	94.6

Unconditional SG to Layout: Recall from Fig. 2 that a layout head extracts layouts from the graph generator. Similar to graph evaluation, we sample $N = 1000$ graphs and obtain their corresponding layouts. We restrict evaluation to scene graph datasets that have ground truth layout records. Table 2 illustrates that **DiScGraph** achieves comparable performance to dedicated layout-baselines, namely LayoutDM and LayoutDiffusion. Of particular note is that **DiScGraph** beats DiffuseSG, despite the latter modeling layout as a diffusion state. The authors use the mismatched continuous diffusion framework to model scene graphs (and later encode them back to categorical results) for explicitly modeling layout as a continuous variable. However, these results indicate that by pursuing gains from layouts, DiffuseSG does not fully extract the gains from scene graphs.

Scene graph completion: We report results on auxiliary scene graph tasks: single node and single relation completion. Following Xu et al. (2025), Table 3 reports win rates over $N = \{1, 10, 50, 100\}$ graph samples, i.e., we mask one object or relation at random, and for such a masked sample, the model predicts N completions. The win rate is computed as the number of graph samples out of N that match the ground truth object and relation label. **DiScGraph** outperforms both DiffuseSG and DiGress. This is not surprising because **DiScGraph** is designed to explicitly learn object-relation semantics during the diffusion training. Figures 9, 10, 11, 12 plot the corresponding qualitative completion results.

❷ Text-to-Image Generation

Text-conditioned scene graphs, $G|T$, are sampled from the CLIP-reward tilted distribution (Eqn. 16). The sampled graphs are passed as conditioning to an SG-Image generator, SDXL-SG. The additional gain from SGs, over text-prompts alone, is evaluated by comparing **DiScGraph** against text-only image generation models: (1) SDXL, (2) ComposedDiff, an inference-time approach that breaks text prompts into concepts and guides image diffusion sampling with an algebraic score function composition of individual concepts, and (3) CO3, also an inference-time intervention, that hypothesizes that image alignment collapses due to concept-mode collisions, and proposes a re-sampler/corrector algorithm to mitigate the issue.

Table 4 demonstrates **DiScGraph**'s performance gains over text-only solutions, on popular composition metrics, ImageReward (IR) and BLIP-VQA. Figure 3 shows some qualitative results, while Figure 4 visualizes corresponding text-conditioned generated SGs. Table 5 extends Table 4's analysis to metrics measuring more granular structural attributes of the generated images for challenging scenarios (like long-prompts, multi-object multi-relational spatial and semantic scenarios) in the CompSGBench benchmark. SG/E/R-IoU compare the structural components (scene graphs, objects, entities), detected in each method's final image, using an LLM. Gains are substantive across IoUs, ImageReward, and BLIP-VQA, highlighting in essence that specialized text-based interventions are limited, and SGs, by constructing a more visual-scene compatible, structural representation of text, offer robust conditioning signals for image synthesis. In particular, **DiScGraph** exhibits significant relation learning performance (Rel-IoU) without compromising object learning or structure, reinforcing the two key innovations: decoupling structure and semantics $E \to (E, R^+)$, and factorized sampler $V \to E \to R^+$. Appendix A.6 reports additional results, comparisons with ground-truth SGs, LLM SG generation, and extensions to FLUX models. A.6.1 explains the FID regression. Additional qualitative results, including SGs for Fig. 3, are in A.9.1.

Finally, we evaluate the grounding signals extracted by **DiScGraph**'s layout-head by using the GLIGEN backbone for layout-to-image generation. Baselines models, LayoutLLM-T2I and LDM, first use an LLM to translate text to layouts and then employ GLIGEN to perform layout-to-image generation. Table 6 and Figure 18 (qualitative) reveal that **DiScGraph** can keep up with more powerful LLMs in extracting structural cues from input text. Thus, **DiScGraph** captures the inherent visual structure, generating visually consistent SGs.

❸ Ablation Studies

(1) *Factorized Formulation*: We ablate on our main contributions: (1) the factorized state, and (2) the dependency-aware reverse sampler, $V \to E \to R$. As expected, a joint E, R model for abstract graphs (using an independent

Table 4: Comparing against text-only methods (Best , 2nd best).

Method	FID (↓)	CLIP-I2T (↑)	CLIP-I2I (↑)	BLIP-VQA (↑)	ImageReward (↑)
CompSGBench Li et al. (2023)					
SDXL	31.34	0.7415	0.3361	0.6425	1.1286
ComposeDiff	49.72	0.6515	0.2844	0.3860	-0.9213
CO3	36.06	0.7348	0.3347	0.6724	1.2070
DiScGraph$_{sg}$	33.85	0.7399	0.3350	0.7854	1.4512
COCO Lin et al. (2015)					
SDXL	43.02	0.7190	0.3177	0.6001	0.6409
ComposeDiff	41.94	0.6518	0.2702	0.2981	-1.2087
CO3	40.26	0.7229	0.3218	0.6282	0.8857
DiScGraph$_{sg}$	38.68	0.8228	0.3386	0.6734	1.1028

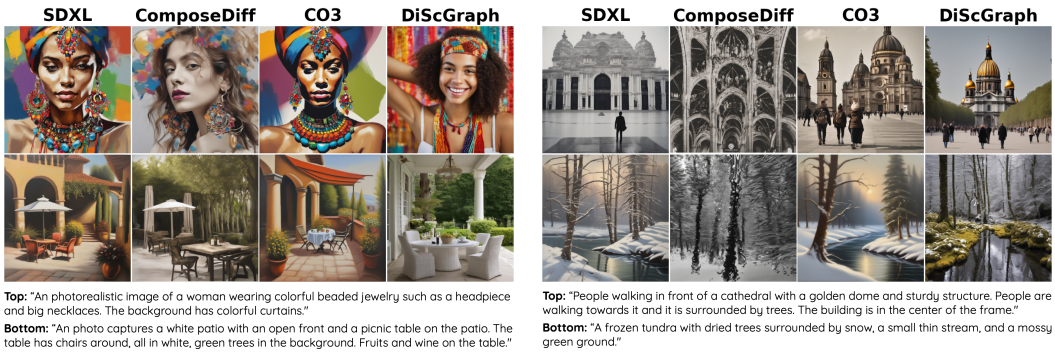

Figure 3: Qualitative results: Comparing T2I generation between SDXL, ComposeDiff, CO3, and **DiScGraph**.

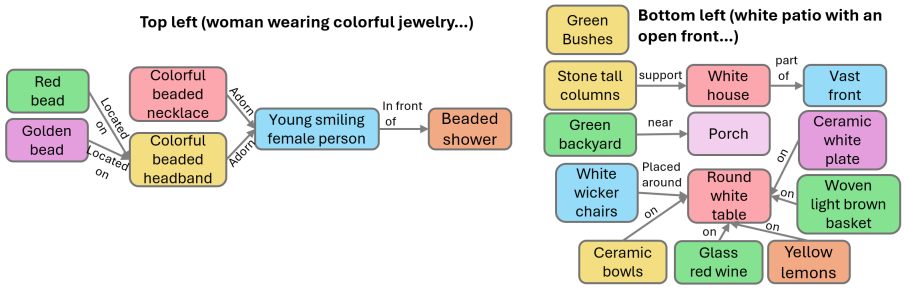

Figure 4: Qualitative Results: Text-conditioned SGs used in SG-to-image results in Figure 3.

V, E sampler) achieves comparable performance to DiGress (Table 1). This joint model is then replaced by the factorized state $\mathcal{SG} := \{(V, E, R^+), e_{ij} = 0 \implies r_{ij} = 0\}$ and the corresponding hybrid corruption strategy is applied; however, the sampler is not dependency-aware and only respects the factorization i.e., $P(V_{t-1} \mid x_t)P(E_{t-1} \mid x_t)P(R_{t-1} \mid x_t, E_{t-1})$. The metrics, particularly over structure (ID/OD-MMD) and relation (R-MMD, RARE-K-TV), improve substantially, validating the importance of E, R^+ factorization. Next, when the reverse sampler includes object-relation dependencies (Eqn. 13), Triplet-TV improves, implying that the model captures semantic consistency between V, E, R without compromising structure or relations. Finally, grounding cues can be extracted by adding the layout head, without compromising SG quality.

(2) We ablate over random and mask corruption ratios (Table 8) and observe best performance when R/M=0.8. With pure random corruption (R/M=1), we see a drop in relation metrics (R-MMD, RARE-K-TV) and Triplet-TV, supporting our claims that absorbing-mask corruption boosts semantic learning by preserving structure and only deleting semantic information. On the other hand, with only mask corruption (R/M=0), the graph metrics are poor, indicating that the corruption does not meaningfully destroy the graph structure, impeding graph learning. When each corruption is applied equally (R/M=0.5), performance improves over R/M=1, but graph metrics are still poor. Thus, partial masking improves semantic learning, but it must only supplement random corruption, not replace it. Appendix Sec. A.7 presents additional ablations.

5 Related Work

■ **Deep Graph Generative Models:** Pre-diffusion autoregressive, VAE, and GAN graph generators You et al. (2018); Liao et al. (2019); Samanta et al. (2019); Li et al. (2018b); Ingraham et al. (2019),De Cao & Kipf

Table 5: Comparing text-only methods over structural image metrics (Best , 2nd best).

CompSGBench			
Method	SG-IoU (↑)	E-IoU (↑)	R-IoU (↑)
SDXL	0.3260	0.7620	0.7188
ComposeDiff	0.3250	0.7801	0.7169
CO3	0.4271	0.7945	0.7203
DiScGraph$_{sg}$	0.5629	0.8357	0.8299

Table 6: **Comparing layout head's results with layout-to-image methods (Best , 2nd best).**

COCO					
Method	FID (↓)	CLIP-I2T (↑)	CLIP-I2I (↑)	BLIP-VQA (↑)	IR (↑)
LayoutDM	72.32	0.5204	0.2955	0.6519	0.9206
LayoutLLM-T2I	69.01	0.6789	0.5373	0.8714	1.2966
LDM	68.84	0.6828	0.5386	0.8286	1.3552
DiScGraph$_{lay}$	71.59	0.6802	0.5400	0.8599	1.3548

Table 7: Ablation on various components of DiScGraph: **1**=Joint, **2**=Factorized state, **3**=2+Dependency-aware sampler, **4**=3+Layout Head on VG (Best).

ID	N-MMD	E-MMD	ID-MMD	OD-MMD	TRIP-TV	R-K-TV
1	0.0077	0.0078	0.0068	0.0062	0.6910	0.8118
2	0.0066	0.0066	0.0056	0.0056	0.5822	0.6099
3	0.0046	0.0058	0.0043	0.0046	0.4878	0.5199
4	0.0046	0.0058	0.0043	0.0046	0.4920	0.5194

Table 8: **Ablation on the different ratios of random and absorbing mask corruptions on Visual Genome (Best).**

R/M	N-MMD	R-MMD	ID-MMD	OD-MMD	TRIP-TV	R-K-TV
1	0.0051	0.0118	0.0045	0.0049	0.6369	0.8326
0.8	0.0046	0.0058	0.0043	0.0046	0.4920	0.5194
0.5	0.0058	0.0091	0.0063	0.0060	0.5735	0.7006
0	0.4533	0.6099	0.9751	1.0000	0.9213	0.9992

(2018),Simonovsky & Komodakis (2018) impose special constraints like node ordering, unlabelled, fixed-size graphs, poor scaling, and training instabilities. Concurrently, many works explored scene graph generation (VarScene, SceneGraphGen) Verma et al. (2022); Garg et al. (2021) and prediction Newell et al. (2017); Li et al. (2018a); Xu et al. (2017); Yang et al. (2018); Zellers et al. (2018) tasks.

■ **Diffusion Graph Generation.** ***(1) Continuous:*** Diffusion models have replaced autoregressive approaches as the SOTA graph modeling paradigm. Early graph diffusion models Niu et al. (2020),Jo et al. (2022), Huang et al. (2022) applied thresholding to snap continuous generations to categorical. DiffuseSG Xu et al. (2025) adapts this idea for grounded scene graphs. Grounding augments standard scene graphs' node-object representation to include layouts. The rationale is layout is continuous while graph components are discrete, so grounded SGs are treated as continuous. ***(2) Discrete diffusion models:*** Following the success of discrete diffusion for text, audio, point clouds etc. Trippe et al. (2023); Hoogeboom et al. (2022); Wu & Li (2023), DiGress Vignac et al. (2023) designs a discrete diffusion model for abstract graphs (molecules, traffic, etc.). The forward process corrupts nodes and edges independently. The training enforces permutation invariance, and DiGress produces exchangeable graph distributions. Some works, GraphMaker Li et al. (2024a), MolDiff Peng et al. (2023), EDGE Chen et al. (2023) have explored encoding entity dependencies in generation through asynchronous generation schedules, but diffusion is still defined over the original graph state. The underlying diffusion support and state space remain unchanged. While this works well for abstract graphs, scene graphs are inherently directed, have large vocabularies, long-tailed relation distributions, and expect semantic consistency between objects and relations. **DiScGraph** addresses these scene graph-specific requirements with a hierarchical constrained factorized state space and corresponding factorized reverse sampler. Appendix A.5 offers a detailed comparison.

6 Limitations & Future Scope

We designed a scene graph generator as a factorized discrete diffusion model, **DiScGraph**, that has a factorized state space decoupling structure from semantics for better relation learning and a dependency-aware reverse sampler to sample SGs with strong entity dependence. Furthermore, we facilitated text-conditioned scene graph generation by sampling from a text-reward tilted distribution. The text-conditioned model was then used downstream in image generation. We provided empirical evidence on T2I generation improvements, unconditional scene graph generation metrics, and layout metrics.

While **DiScGraph** achieves good scene graph generation and T2I generation performance, there is room for improvement, (1) Our text conditioning framework operates at inference time through reward-tilted sampling, avoiding conditional model retraining. While this works well for textual abstractions (SGs), the quality of conditioning depends heavily on the reward function (CLIP). Future work could explore stronger learned reward models. (2) Graph transformers scale quadratically with the number of scene entities and relations. While practical for current scene graph datasets, scaling to very large structured scenes may require techniques like sparse attention. (3) Discrete frameworks are restricted by sampler quality, requiring sampling refinements (Sec. A.2.2, A.4.2). For future work, we plan to address it in a more principled way by designing a robust sampling algorithm. (4) Scene graph quality is dependent on the text input and is encumbered by unexpressive text prompts, limiting improvements to image generation. Limited availability of large-scale scene graph data and scene graph conditioned image generators also limits development; however, recent efforts like LAION-SG Li et al. (2024b) are a step in the right direction.

AI USE STATEMENT

In this work, we used generative AI tools for LLM-based scene graph generation baseline results outlined in Appendix A.6 and SG/R/E-IoU metrics outlined in Li et al. (2024b). We have not used generative AI tools for any other required tasks such as help with developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refine hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, interpreting results, and generating synthetic data, assisting with translation, clean and reformat dataset, supporting qualitative and thematic data analysis do not apply to this work. Additionally, we used generative AI tools for formatting tables. We have reviewed all AI-assisted work. We have reviewed all AI results through manual spot checks and manually verified the content of the tables. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

DiScGraph improves text-to-image generation by learning a structured scene graph prior using a factorized discrete diffusion model. At the same time, it introduces certain risks. It could be misused to create deceptive or misleading visuals, contributing to the spread of misinformation. When applied to depictions of public figures, it may compromise personal privacy. Moreover, the generated content can raise concerns around copyright and intellectual property.

REPRODUCIBILITY STATEMENT

The full code, training, and inference pipelines and scripts are included in the supplementary material. This includes instructions for environment setup, dependencies, and reproducible random seeds. The datasets used in our experiments are publicly available. We provide detailed descriptions of the datasets in Sec. 4. Hyperparameters, metrics, model architectures, and inference samplers are described in the Appendix A.4.

REFERENCES

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=h7-XixPCAL.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 843–852. IEEE, 2023. doi: 10.1109/CVPRW59228.2023.00091.

Siyi Chen, Huijie Zhang, Minzhe Guo, Yifu Lu, Peng Wang, and Qing Qu. Exploring low-dimensional subspaces in diffusion models for controllable image editing, 2024. URL https://arxiv.org/abs/2409.02374.

Xiaohui Chen, Jiaxing He, Xu Han, and Li-Ping Liu. Efficient and degree-guided graph generation via discrete diffusion modeling, 2023. URL https://arxiv.org/abs/2305.04111.

Wenda Chu, Zihui Wu, Yifan Chen, Yang Song, and Yisong Yue. Split gibbs discrete diffusion posterior sampling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=U7ab14f4YQ.

Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation, 2023. URL https://arxiv.org/abs/2201.11460.

Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.

Debottam Dutta, Jianchong Chen, Rajalaxmi Rajagopalan, Yu-Lin Wei, and Romit Roy Choudhury. Steer away from mode collisions: Improving composition in diffusion models, 2026. URL https://arxiv.org/abs/2509.25940.

Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs, 2021. URL https://arxiv.org/abs/2012.09699.

Azade Farshad, Yousef Yeganeh, Yu Chi, Chengzhi Shen, Björn Ommer, and Nassir Navab. Scenegenie: Scene graph guided diffusion models for image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, 2023.

Sarthak Garg, Helisa Dhamo, Azade Farshad, Sabrina Musatian, Nassir Navab, and Federico Tombari. Unconditional scene graph generation. In *ICCV*, 2021.

Emiel Hoogeboom, Victor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *ICML*, 2022.

Han Huang, Leilei Sun, Bowen Du, Yanjie Fu, and Weifeng Lv. Graphgdp: Generative diffusion processes for permutation invariant graph generation. In *ICDM*, 2022.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747, 2023.

John Ingraham, Vikas K. Garg, Regina Barzilay, and Tommi Jaakkola. Generative models for graph-based protein design. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Naoto Inoue, Kotaro Kikuchi, Edgar Simo-Serra, Mayu Otani, and Kota Yamaguchi. Layoutdm: Discrete diffusion model for controllable layout generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10167–10176, June 2023.

Jaehyeong Jo, Seul Lee, and Sung Ju Hwang. Score-based generative modeling of graphs via stochastic differential equations. In *ICML*, 2022.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. *arXiv preprint arXiv:2501.05803*, 2025.

Xiang Kong, Lu Jiang, Huiwen Chang, Han Zhang, Yuan Hao, Haifeng Gong, and Irfan Essa. Blt: Bidirectional layout transformer for controllable layout generation. In *ECCV*, 2022.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations, 2016. URL https://arxiv.org/abs/1602.07332.

Mufei Li, Eleonora Kreacic, Vamsi K. Potluru, and Pan Li. Graphmaker: Can diffusion models generate large attributed graphs?, 2024a. URL https://openreview.net/forum?id=1edQ1BCrwc.

Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. Factorizable net: An efficient subgraph-based framework for scene graph generation. In *European Conference on Computer Vision (ECCV)*, 2018a.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation, 2023. URL https://arxiv.org/abs/2301.07093.

Yujia Li, Oriol Vinyals, Chris Dyer, Razvan Pascanu, and Peter Battaglia. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018b.

Zejian Li, Chenye Meng, Yize Li, Ling Yang, Shengyuan Zhang, Jiarui Ma, Jiayi Li, Guang Yang, Changyuan Yang, Zhiyuan Yang, Jinxiong Chang, and Lingyun Sun. Laion-sg: An enhanced large-scale dataset for training complex image-text models with structural annotations, 2024b. URL https://arxiv.org/abs/2412.08580.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. Llm-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models, 2024. URL https://arxiv.org/abs/2305.13655.

Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, Charlie Nash, William L. Hamilton, David Duvenaud, Raquel Urtasun, and Richard Zemel. Efficient graph generation with graph recurrent attention networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. URL https://arxiv.org/abs/1405.0312.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models, 2023. URL https://arxiv.org/abs/2206.01714.

Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8edd72158ccd2a879f79cb2538568fdc-Paper.pdf.

Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling, 2020. URL https://arxiv.org/abs/2003.00638.

Xingang Peng, Jiaqi Guan, Qiang Liu, and Jianzhu Ma. MolDiff: Addressing the atom-bond inconsistency problem in 3D molecule diffusion generation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 27611–27629. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/peng23b.html.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. *arXiv preprint arXiv:2308.05095*, 2023. URL https://arxiv.org/abs/2308.05095.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763. PMLR, 2021.

Bidisha Samanta, Abir De, Gourhari Jana, Pratim Kumar Chattaraj, Niloy Ganguly, and Manuel Gomez Rodriguez. Nevae: A deep generative model for molecular graphs. In *AAAI Conference on Artificial Intelligence*, 2019.

Guibao Shen, Luozhou Wang, Jiantao Lin, Wenhang Ge, Chaozhe Zhang, Xin Tao, Yuan Zhang, Pengfei Wan, Zhongyuan Wang, Guangyong Chen, Yijun Li, and Ying-Cong Chen. Sg-adapter: Enhancing text-to-image generation with scene graph guidance, 2024. URL https://arxiv.org/abs/2405.15321.

Guibao Shen, Luozhou Wang, Jiantao Lin, Wenhang Ge, Chaozhe Zhang, Xin Tao, Di Zhang, Pengfei Wan, Guangyong Chen, Yijun Li, and Ying-cong Chen. Scene graph guided generation: Enable accurate relations generation in text-to-image models via textural rectification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15437–15446, October 2025.

Martin Simonovsky and Nikos Komodakis. Graphvae: Towards generation of small graphs using variational autoencoders. *arXiv preprint arXiv:1802.03480*, 2018.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Vo Thanh-Nhan, Nguyen Trong-Thuan, Nguyen Tam V., and Tran Minh-Triet. Autoregressive image generation guided by scene graphs. *arXiv preprint arXiv:2508.14502*, 2025.

Brian L. Trippe, Jason Yim, Doug Tischer, Tamara Broderick, David Baker, Regina Barzilay, and Tommi Jaakkola. Diffusion probabilistic modeling of protein backbones. In *ICLR*, 2023.

Tathagat Verma, Abir De, Yateesh Agrawal, Vishwa Vinay, and Soumen Chakrabarti. Varscene: A deep generative model for realistic scene graph synthesis. In *ICML*, 2022.

Clement Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=UaAD-Nu86WX.

Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. Split-and-augmented gibbs sampler—application to large-scale inference problems. *IEEE Transactions on Signal Processing*, 67(6):1648–1661, March 2019. ISSN 1941-0476. doi: 10.1109/tsp.2019.2894825. URL http://dx.doi.org/10.1109/TSP.2019.2894825.

Stephen J. Wright. Coordinate descent algorithms, 2015. URL https://arxiv.org/abs/1502.04759.

Fang Wu and Stan Z. Li. Diffmd: A geometric diffusion model for molecular dynamics simulations, 2023. URL https://arxiv.org/abs/2204.08672.

Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. URL https://arxiv.org/abs/2307.10816.

Bicheng Xu, Qi Yan, Renjie Liao, Lele Wang, and Leonid Sigal. Joint generative modeling of grounded scene graphs and images via diffusion models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=2cxxZI2LOL.

Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pp. 15903–15935, 2023.

Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018.

Ling Yang, Zhilin Huang, Yang Song, Shenda Hong, Guohao Li, Wentao Zhang, Bin Cui, Bernard Ghanem, and Ming-Hsuan Yang. Diffusion-based scene graph to image generation with masked contrastive pre-training, 2022. URL https://arxiv.org/abs/2211.11138.

Po-Hung Yeh, Kuang-Huei Lee, and Jun-Cheng Chen. Training-free diffusion model alignment with sampling demons. *arXiv preprint arXiv:2410.05760*, 2024.

Jiaxuan You, Rex Ying, Xiang Ren, William L. Hamilton, and Jure Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 5708–5717, 2018.

Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J. Kim. Graph transformer networks, 2020. URL https://arxiv.org/abs/1911.06455.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018.

Junyi Zhang, Jiaqi Guo, Shizhao Sun, Jian-Guang Lou, and Dongmei Zhang. Layoutdiffusion: Improving graphic layout generation by discrete diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7226–7236, October 2023a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023b. URL https://arxiv.org/abs/2302.05543.

Xu Zhang, Jin Yuan, Hanwang Zhang, Guojin Zhong, Yongsheng Zang, Jiacheng Lin, and Zhiyong Li. Sgdiff: Scene graph guided diffusion model for image collaborative segcaptioning, 2025. URL https://arxiv.org/abs/2512.01975.

Zhiyuan Zhang, Dongdong Chen, and Jing Liao. Sgedit: Bridging llm with text-to-image models for scene graph-based image editing. *ACM Transactions on Graphics*, 2024.

A Appendix

Appendix Table of Contents

A.1 Structure-aware Forward Process – Additional Details 15
A.2 Factorized Reverse Sampler – Additional Details 18
A.3 Text-Conditioned Scene Graph Generation via Reward Tilting – Additional Details 23
A.4 Experimental Details 23
A.5 Related Works 26
A.6 Additional Results 26
A.7 Ablation Studies 28
A.8 Scene Graph Generation – Qualitative results 31
A.9 Text-Image Generation – Qualitative Results 31

A.1 Structure-aware Forward Process – Additional Details

In this section, we provide a detailed analysis of the forward corruption steady state convergence, marginals over object, edge, and relation entities, and different corruption strategies.

A.1.1 Structure-Aware Forward Diffusion Convergence

We derive the convergence of the factorized forward process to a stationary distribution and show that it matches the scene graph data distribution formulation.

We define a forward diffusion process over the structured scene graph state

$$x_t = (V_t, E_t, R_t^+), \tag{18}$$

where object labels V_t, edge existence E_t, and relation labels R_t^+ satisfy the constraint

$$e_{ij,t} = 0 \;\Rightarrow\; r_{ij,t} = 0. \tag{19}$$

The forward process is defined as a Markov chain:

$$q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1}), \tag{20}$$

with factorized transition kernel:

$$q(x_t \mid x_{t-1}) = q_V(V_t \mid V_{t-1}) q_E(E_t \mid E_{t-1}) q_R(R_t \mid R_{t-1}, E_t). \tag{21}$$

Categorical Corruption Kernel. For a categorical variable $z_t \in \{1, \dots, K\}$, we define

$$Q_t = (1 - \beta_t) I + \beta_t \mathbf{1}\pi^\top, \tag{22}$$

where $\pi \in \Delta^{K-1}$ is a fixed prior distribution. The transition is

$$q(z_t = b \mid z_{t-1} = a) = [Q_t]_{ab}. \tag{23}$$

In the time-homogeneous case, $Q = (1 - \beta) I + \beta \mathbf{1}\pi^\top$, we have $\pi^\top Q = \pi^\top$, so π is a stationary distribution.

Moreover, $p_t^\top - \pi^\top = (1 - \beta)^t (p_0^\top - \pi^\top)$, so $p_t \to \pi \quad$ as $t \to \infty$.

In the time non-homogeneous case,

$$Q_t = (1 - \beta_t) I + \beta_t \mathbf{1}\pi^\top, \tag{24}$$

we obtain

$$p_t^\top - \pi^\top = \left(\prod_{s=1}^{t} (1 - \beta_s) \right) (p_0^\top - \pi^\top), \tag{25}$$

and convergence holds if

$$\prod_{s=1}^{t} (1 - \beta_s) \to 0. \tag{26}$$

Object and Edge Corruption. Object labels and edge existence follow standard discrete diffusion corruption,

$$q_V(v_{i,t} \mid v_{i,t-1}) = (1 - \beta_t^V) \delta_{v_{i,t-1}} + \beta_t^V \pi_V, \tag{27}$$

$$q_V(V_t \mid V_{t-1}) = \prod_i q_V(v_{i,t} \mid v_{i,t-1}), \tag{28}$$

$$Q_t^V = (1 - \beta_t^V) I + \beta_t^V \mathbf{1}\pi_V^\top. \tag{29}$$

where π_V is a prior distribution over object labels, can be chosen as uniform or the empirical training data distribution, and δ is a point-mass distribution.

Each directed edge is, $e_{ij,t} \in \{0, 1\}$.

$$q_E(e_{ij,t} \mid e_{ij,t-1}) = (1 - \beta_t^E) \delta_{e_{ij,t-1}} + \beta_t^E \pi_E, \tag{30}$$

with prior from training data statistics on edge existence, $\pi_E(1) = \rho_E, \quad \pi_E(0) = 1 - \rho_E$.

$$q_E(E_t \mid E_{t-1}) = \prod_{i \neq j} q_E(e_{ij,t} \mid e_{ij,t-1}). \tag{31}$$

$$Q_t^E = (1 - \beta_t^E) I + \beta_t^E \mathbf{1}\pi_E^\top. \tag{32}$$

Under the above conditions on β_t^V and β_t^E, we obtain

$$q(V_t) \to \pi_V(V) = \prod_i \pi_V(v_i), \tag{33}$$

$$q(E_t) \to \pi_E(E) = \prod_{i \neq j} \pi_E(e_{ij}). \tag{34}$$

Structure-Aware Relation Corruption. Relation corruption is conditioned on edge existence,

$$q_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t}) = \begin{cases} \tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}), & e_{ij,t} = 1, \\ \delta_0(r_{ij,t}), & e_{ij,t} = 0, \end{cases} \tag{35}$$

where δ_0 is the point mass at 0, and

$$\tilde{Q}_t^R = (1 - \beta_t^R)I + \beta_t^R \mathbf{1}\pi_R^\top. \tag{36}$$

Thus, for active edges,

$$\tilde{q}_R(r_{ij,t} \mid e_{ij,t} = 1) \to \pi_R(r_{ij}), \tag{37}$$

while for inactive edges,

$$q(r_{ij,t} = 0 \mid e_{ij,t} = 0) = 1. \tag{38}$$

Structured Stationary Distribution. The limiting distribution over scene graphs is therefore

$$\pi_{\mathrm{SG}}(V, E, R) = \prod_i \pi_V(v_i) \prod_{i \neq j} \pi_E(e_{ij}) \prod_{i \neq j : e_{ij}=1} \pi_R(r_{ij}) \prod_{i \neq j : e_{ij}=0} \delta_0(r_{ij}). \tag{39}$$

Equivalently,

$$\pi_{\mathrm{SG}}(V, E, R) = \pi_V(V)\, \pi_E(E)\, \pi_R(R \mid E), \tag{40}$$

where

$$\pi_R(R \mid E) = \prod_{e_{ij}=1} \pi_R(r_{ij}) \prod_{e_{ij}=0} \delta_0(r_{ij}). \tag{41}$$

This distribution is supported only on valid scene graphs.

Lemma A.1. *Factorized diffusion state convergence: Assume that*

$$\prod_{s=1}^{t}(1 - \beta_s^V) \to 0, \quad \prod_{s=1}^{t}(1 - \beta_s^E) \to 0, \quad \prod_{s=1}^{t}(1 - \beta_s^R) \to 0. \tag{42}$$

Then the forward process satisfies

$$q(x_t \mid x_0) \to \pi_{\mathrm{SG}}(V, E, R). \tag{43}$$

Proof. From categorical diffusion convergence,

$$q(V_t \mid V_0) \to \pi_V(V), \quad q(E_t \mid E_0) \to \pi_E(E). \tag{44}$$

For each pair (i, j), conditional on $e_{ij,t}$,

$$q(r_{ij,t} \mid r_{ij,0}, e_{ij,t}) \to \begin{cases} \pi_R(r_{ij}), & e_{ij,t} = 1, \\ \delta_0(r_{ij}), & e_{ij,t} = 0. \end{cases} \tag{45}$$

Thus,

$$q(R_t \mid R_0, E_t) \to \prod_{i \neq j : e_{ij,t}=1} \pi_R(r_{ij,t}) \prod_{i \neq j : e_{ij,t}=0} \delta_0(r_{ij,t}). \tag{46}$$

Combining all components,

$$q(x_t \mid x_0) = q(V_t \mid V_0) q(E_t \mid E_0) q(R_t \mid R_0, E_t) \to \pi_{\mathrm{SG}}(V, E, R). \tag{47}$$

The data distribution can be factorized as

$$p_{\text{data}}(G) = p_{\text{data}}(V)\, p_{\text{data}}(E \mid V)\, p_{\text{data}}(R^+ \mid E, V), \tag{48}$$

which exposes the hierarchical visual scene structure: objects $\to$ edges $\to$ relations.

Additionally,

Proposition. If $x_{t-1} \in \mathcal{G}_{SG}$, then $x_t \sim q(x_t \mid x_{t-1})$ also lies in $\mathcal{G}_{SG}$.

From the forward corruption formulation,

$$\tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t} = 0) = \delta_0(r_i j, t) \tag{49}$$

Then the only non-zero probability relation is $r_{ij,t} = 0$. Thus, If $x_{t-1} \in \mathcal{G}_{SG}$, then $x_t \sim q(x_t \mid x_{t-1})$ also lies in $\mathcal{G}_{SG}$, thus the valid scene graph factorized space is preserved at all timesteps in the forward process.

A.1.2 Closed-Form Marginals and Corruption Variants

A key property of discrete diffusion is the ability to directly sample from the marginal distribution $q(x_t \mid x_0)$ via closed-form transition matrices. We now state these marginals for the factorized scene-graph state.

Closed-form marginal for objects. Following discrete diffusion formulation Austin et al. (2021), let $\{Q_s^V\}_{s=1}^t$ denote the object transition matrices and define the cumulative transition

$$\bar{Q}_t^V = Q_1^V Q_2^V \cdots Q_t^V. \tag{50}$$

Then the marginal distribution at time t is

$$q(v_{i,t} \mid v_{i,0}) = \mathrm{Cat}\Big(v_{i,0}\, \bar{Q}_t^V\Big). \tag{51}$$

If $v_{i,0}$ is represented as a one-hot vector, then

$$p(v_{i,t} = c \mid v_{i,0}) = \Big[v_{i,0}\, \bar{Q}_t^V\Big]_c. \tag{52}$$

Closed-form marginal for edges. Similarly, let $\bar{Q}_t^E = Q_1^E Q_2^E \cdots Q_t^E$. Then

$$q(e_{ij,t} \mid e_{ij,0}) = \mathrm{Cat}\Big(e_{ij,0}\, \bar{Q}_t^E\Big). \tag{53}$$

For binary edges,

$$p(e_{ij,t} = b \mid e_{ij,0}) = \Big[e_{ij,0}\, \bar{Q}_t^E\Big]_b. \tag{54}$$

Closed-form marginal for relations. The relation marginals need to be reworked for the factorized scene graph space $\mathcal{SG}$. Relation marginals depend on the edge state at time t. Conditioned on an active edge,

$$q(r_{ij,t} \mid r_{ij,0}, e_{ij,t} = 1) = \mathrm{Cat}\Big(r_{ij,0}\, \bar{Q}_t^R\Big), \tag{55}$$

where $\bar{Q}_t^R = Q_1^R Q_2^R \cdots Q_t^R$.

If the edge is inactive,

$$q(r_{ij,t} = 0 \mid r_{ij,0}, e_{ij,t} = 0) = 1. \tag{56}$$

Thus, the structural components, edges, are decoupled from the semantics, relations, but the dependency structure ensures we are operating in valid scene graph space.

$$q(r_{ij,t} \mid r_{ij,0}, e_{ij,t}) = \begin{cases} \mathrm{Cat}\big(r_{ij,0}\, \bar{Q}_t^R\big), & e_{ij,t} = 1, \\ \delta_0(r_{ij,t}), & e_{ij,t} = 0. \end{cases} \tag{57}$$

Full closed-form marginal. Combining all components,

$$q(x_t \mid x_0) = q_V(V_t \mid V_0)\, q_E(E_t \mid E_0)\, q_R(R_t \mid R_0, E_t). \tag{58}$$

$$\begin{aligned} q(x_t \mid x_0) = &\prod_i q_V(v_{i,t} \mid v_{i,0}) \prod_{i \neq j} q_E(e_{ij,t} \mid e_{ij,0}) \\ &\cdot \prod_{ij: e_{ij,t}=1} q_R(r_{ij,t} \mid r_{ij,0}, e_{ij,t} = 1) \prod_{ij: e_{ij,t}=0} \delta_0(r_{ij,t}). \end{aligned} \tag{59}$$

This enables direct sampling of noisy states of the factorized space during training.

A.1.3 Forward Corruption Choices

Absorbing-mask corruption. Given the success of absorbing masks for diffusion language tasks, apart from random corruption, we apply absorbing mask corruption, which preserves the structure and only deletes information (object and relation), allowing the model to specifically learn semantics, which is beneficial for scene graphs. We define mask tokens $[\mathrm{MASK}]_V$ and $[\mathrm{MASK}]_R$.

For objects:

$$q_V(v_{i,t} \mid v_{i,t-1}) = (1 - \beta_t^V)\delta_{v_{i,t-1}} + \beta_t^V \delta_{[\mathrm{MASK}]_V}. \tag{60}$$

For relations on active edges:

$$\tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) = (1 - \beta_t^R)\delta_{r_{ij,t-1}} + \beta_t^R \delta_{[\mathrm{MASK}]_R}. \tag{61}$$

The edge-aware kernel becomes

$$q_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t}) = \begin{cases} (1 - \beta_t^R)\delta_{r_{ij,t-1}} + \beta_t^R \delta_{[\mathrm{MASK}]_R}, & e_{ij,t} = 1, \\ \delta_0(r_{ij,t}), & e_{ij,t} = 0. \end{cases} \tag{62}$$

This restricts masking to valid relations.

Hybrid random + mask corruption. Performing masking in isolation would not allow the model to learn structure; we need a hybrid kernel combining random replacement and mask absorption, thereby learning both structure and semantics.

For objects:

$$q_V(v_{i,t} \mid v_{i,t-1}) = (1-\beta_t^V)\delta_{v_{i,t-1}} + \beta_t^V\big[\rho_V \delta_{[\text{MASK}]_V} + (1-\rho_V)\pi_V\big]. \tag{63}$$

For active relations:

$$\tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) = (1-\beta_t^R)\delta_{r_{ij,t-1}} + \beta_t^R\big[\rho_R \delta_{[\text{MASK}]_R} + (1-\rho_R)\pi_R\big]. \tag{64}$$

The edge-aware relation kernel remains

$$q_R(r_{ij,t} \mid r_{ij,t-1}, e_{ij,t}) = \begin{cases} \tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}), & e_{ij,t}=1, \\ \delta_0(r_{ij,t}), & e_{ij,t}=0. \end{cases} \tag{65}$$

Thus, we defined a structure-aware forward process over $x_t = (V_t, E_t, R_t^+)$ such that $q(x_t \mid x_{t-1}) = q_V q_E q_R$, with,

$$q_R(R_t \mid R_{t-1}, E_t) = \prod_{ij:e_{ij,t}=1} \tilde{q}_R(r_{ij,t} \mid r_{ij,t-1}) \prod_{ij:e_{ij,t}=0} \delta_0(r_{ij,t}). \tag{66}$$

This construction: (i) preserves validity of the scene-graph state space, (ii) decouples structural sparsity from relation semantics, and (iii) prevents rare relations from competing with the no-edge class.

A.2 Factorized Reverse Sampler – Additional Details

In this section, we discuss posterior derivation for the factorized scene graph model and refinements to the sampler at inference.

A.2.1 Factorized Reverse Model and Structured Posterior Sampler

Discrete posterior for a categorical variable. Austin et al. (2021); Vignac et al. (2023) Consider a categorical variable $z_t \in \{1, \dots, K\}$ with forward transition matrices

$$Q_t, \qquad \bar{Q}_t = Q_1 Q_2 \cdots Q_t. \tag{67}$$

The one-step posterior conditioned on the clean variable z_0 is

$$q(z_{t-1}=a \mid z_t=b, z_0=c) = \frac{q(z_t=b \mid z_{t-1}=a)\, q(z_{t-1}=a \mid z_0=c)}{q(z_t=b \mid z_0=c)}. \tag{68}$$

Using the transition matrices, this becomes

$$q(z_{t-1}=a \mid z_t=b, z_0=c) = \frac{[Q_t]_{ab}\,[\bar{Q}_{t-1}]_{ca}}{[\bar{Q}_t]_{cb}}. \tag{69}$$

Since z_0 is unknown at sampling time, the model predicts a distribution

$$p_\theta(z_0=c \mid x_t). \tag{70}$$

The learned reverse posterior marginalizes over z_0:

$$p_\theta(z_{t-1}=a \mid z_t=b) = \sum_c q(z_{t-1}=a \mid z_t=b, z_0=c)\, p_\theta(z_0=c \mid x_t). \tag{71}$$

Equivalently,

$$p_\theta(z_{t-1}=a \mid z_t=b) \propto q(z_t=b \mid z_{t-1}=a) \sum_c p_\theta(z_0=c \mid x_t) q(z_{t-1}=a \mid z_0=c). \tag{72}$$

Reverse factorization for scene graphs. For scene graphs, the clean state is

$$x_0 = (V_0, E_0, R_0). \tag{73}$$

Following the structured state space, we factorize the clean-state prediction as

$$p_\theta(x_0 \mid x_t) = p_\theta(V_0, E_0, R_0 \mid x_t). \tag{74}$$

A scene-graph-aware factorization is

$$p_\theta(V_0, E_0, R_0 \mid x_t) = p_\theta(V_0 \mid x_t)\, p_\theta(E_0 \mid V_0, x_t)\, p_\theta(R_0 \mid V_0, E_0, x_t). \tag{75}$$

This factorization follows the semantic structure of scene graphs:

1. object identities define the semantic entities in the graph,
2. directed edge existence determines which object pairs interact,
3. relation identities are meaningful only for active directed edges.

Thus, relation prediction is conditioned on both object identity and edge existence.

Object reverse posterior. For each object variable v_i, the model predicts, $p_\theta(v_{i,0} = c \mid x_t)$. The reverse posterior is

$$p_\theta(v_{i,t-1} = a \mid v_{i,t} = b, x_t) = \sum_c q_V(v_{i,t-1} = a \mid v_{i,t} = b, v_{i,0} = c)\, p_\theta(v_{i,0} = c \mid x_t). \tag{76}$$

Using the forward kernels,

$$q_V(v_{i,t-1} = a \mid v_{i,t} = b, v_{i,0} = c) = \frac{[Q_t^V]_{ab}\,[\bar{Q}_{t-1}^V]_{ca}}{[\bar{Q}_t^V]_{cb}}. \tag{77}$$

Therefore,

$$p_\theta(v_{i,t-1} = a \mid v_{i,t} = b, x_t) = \sum_c \frac{[Q_t^V]_{ab}\,[\bar{Q}_{t-1}^V]_{ca}}{[\bar{Q}_t^V]_{cb}} p_\theta(v_{i,0} = c \mid x_t). \tag{78}$$

Edge reverse posterior. For each directed edge variable $e_{ij} \in \{0, 1\}$, the model predicts, $p_\theta(e_{ij,0} = c \mid V_0, x_t)$. The reverse posterior is

$$p_\theta(e_{ij,t-1} = a \mid e_{ij,t} = b, V_0, x_t) = \sum_c q_E(e_{ij,t-1} = a \mid e_{ij,t} = b, e_{ij,0} = c) p_\theta(e_{ij,0} = c \mid V_0, x_t), \tag{79}$$

where

$$q_E(e_{ij,t-1} = a \mid e_{ij,t} = b, e_{ij,0} = c) = \frac{[Q_t^E]_{ab}\,[\bar{Q}_{t-1}^E]_{ca}}{[\bar{Q}_t^E]_{cb}}. \tag{80}$$

Thus,

$$p_\theta(e_{ij,t-1} = a \mid e_{ij,t} = b, V_0, x_t) = \sum_c \frac{[Q_t^E]_{ab}\,[\bar{Q}_{t-1}^E]_{ca}}{[\bar{Q}_t^E]_{cb}} p_\theta(e_{ij,0} = c \mid V_0, x_t). \tag{81}$$

Edge-gated relation reverse posterior. Relations are only meaningful on active directed edges. Since the forward relation corruption is conditioned on the current edge state $e_{ij,t}$, the reverse relation posterior must distinguish three cases depending on $(e_{ij,t-1}, e_{ij,t})$.

Let the denoiser output clean relation logits after edge and object predictions,

$$\epsilon_\theta^R(x_t, t, V_{t-1}, E_{t-1})_{ij} \in \mathbb{R}^{K_{\text{rel}}}, \tag{82}$$

and the predicted clean relation distribution on active edges is,

$$p_\theta(r_{ij,0} = c \mid x_t, V_{t-1}, E_{t-1}) = \text{softmax}\left(\epsilon_\theta^R(x_t, t, V_{t-1}, E_{t-1})_{ij}\right)_c. \tag{83}$$

Case 1: Edge absent at $t-1$. If $e_{ij,t-1} = 0$, then the relation is deterministically null:

$$p_\theta(r_{ij,t-1} = 0 \mid e_{ij,t-1} = 0, x_t) = 1. \tag{84}$$

Case 2: Edge active at both $t-1$ and t. If $e_{ij,t-1} = 1, \quad e_{ij,t} = 1$, then relation corruption was active in the forward process. Therefore we can use the standard discrete posterior over positive relation labels:

$$p_\theta(r_{ij,t-1} = a \mid r_{ij,t} = b, e_{ij,t-1} = 1, e_{ij,t} = 1, x_t) = \sum_c q_R(r_{ij,t-1} = a \mid r_{ij,t} = b, r_{ij,0} = c)\, p_\theta(r_{ij,0} = c \mid x_t, V_{t-1}, E_{t-1}), \tag{85}$$

where

$$q_R(r_{ij,t-1} = a \mid r_{ij,t} = b, r_{ij,0} = c) = \frac{[Q_t^R]_{ab}\,[\bar{Q}_{t-1}^R]_{ca}}{[\bar{Q}_t^R]_{cb}}. \tag{86}$$

Equivalently,

$$p_\theta(r_{ij,t-1} = a \mid r_{ij,t} = b, e_{ij,t-1} = 1, e_{ij,t} = 1, x_t) \propto q_R(r_{ij,t} = b \mid r_{ij,t-1} = a) \sum_c p_\theta(r_{ij,0} = c \mid x_t, V_{t-1}, E_{t-1}) q_R(r_{ij,t-1} = a \mid r_{ij,0} = c). \tag{87}$$

Case 3: Edge reactivates in the reverse step. The key sampler modification for the factorized setting is in addressing this case. If $e_{ij,t-1} = 1, \quad e_{ij,t} = 0$, then the current relation $r_{ij,t} = 0$ is caused deterministically by the absence of the current edge. Hence $r_{ij,t}$ carries no semantic information about the positive relation at $t-1$. We therefore use the predictive marginal obtained by propagating the denoiser's clean relation prediction to timestep $t-1$:

$$p_\theta(r_{ij,t-1} = a \mid e_{ij,t-1} = 1, e_{ij,t} = 0, x_t) = \sum_c q_R(r_{ij,t-1} = a \mid r_{ij,0} = c) p_\theta(r_{ij,0} = c \mid x_t, V_{t-1}, E_{t-1}). \tag{88}$$

Using the relation marginal transition matrix,

$$q_R(r_{ij,t-1} = a \mid r_{ij,0} = c) = [\bar{Q}^R_{t-1}]_{ca}, \tag{89}$$

so

$$p_\theta(r_{ij,t-1} = a \mid e_{ij,t-1} = 1, e_{ij,t} = 0, x_t) = \sum_c [\bar{Q}^R_{t-1}]_{ca} p_\theta(r_{ij,0} = c \mid x_t, V_{t-1}, E_{t-1}). \tag{90}$$

Thus the full edge-gated relation sampler is

$$p_\theta(r_{ij,t-1} \mid r_{ij,t}, e_{ij,t-1}, e_{ij,t}, x_t) = \begin{cases} \delta_0(r_{ij,t-1}), & e_{ij,t-1} = 0, \\ p_\theta^{\text{post}}(r_{ij,t-1} \mid r_{ij,t}, x_t), & e_{ij,t-1} = 1,\ e_{ij,t} = 1, \\ p_\theta^{\text{marg}}(r_{ij,t-1} \mid x_t), & e_{ij,t-1} = 1,\ e_{ij,t} = 0. \end{cases} \tag{91}$$

Structured reverse transition. Combining the object, edge, and relation posterior updates gives the structured reverse transition,

$$p_\theta(x_{t-1} \mid x_t) = p_\theta(V_{t-1} \mid x_t)\, p_\theta(E_{t-1} \mid V_{t-1}, x_t)\, p_\theta(R_{t-1} \mid V_{t-1}, E_{t-1}, x_t). \tag{92}$$

$$\begin{aligned} p_\theta(x_{t-1} \mid x_t) = \prod_i p_\theta(v_{i,t-1} \mid v_{i,t}, x_t) \prod_{i \neq j} p_\theta(e_{ij,t-1} \mid e_{ij,t}, V_{t-1}, x_t) \\ \prod_{i \neq j} p_\theta(r_{ij,t-1} \mid r_{ij,t}, e_{ij,t-1}, V_{t-1}, E_{t-1}, x_t). \end{aligned} \tag{93}$$

The final relation factor in Eqn. 93 is edge-gated (Eqn. 91).

Proposition. Validity preservation of the reverse sampler:

If the reverse sampler uses the edge-gated relation posterior above, then every sampled state x_{t-1} satisfies

$$e_{ij,t-1} = 0 \Rightarrow r_{ij,t-1} = 0. \tag{94}$$

For any directed pair (i, j), suppose $e_{ij,t-1} = 0$. By definition of the edge-gated relation posterior,

$$p_\theta(r_{ij,t-1} = 0 \mid e_{ij,t-1} = 0, x_t) = 1. \tag{95}$$

Therefore the only relation value with nonzero probability is $r_{ij,t-1} = 0$. Hence

$$e_{ij,t-1} = 0 \Rightarrow r_{ij,t-1} = 0. \tag{96}$$

Thus the sampled state remains in the constrained scene graph space.

Sampling Process. At each reverse timestep t, we perform:

1. Predict object clean-state probabilities:
$$p_\theta(V_0 \mid x_t). \tag{97}$$
Sample or decode
$$V_{t-1} \sim p_\theta(V_{t-1} \mid x_t) \tag{98}$$
using the object posterior.
2. Predict edge clean-state probabilities conditioned on recovered objects:
$$p_\theta(E_0 \mid V_{t-1}, x_t). \tag{99}$$
Sample
$$E_{t-1} \sim p_\theta(E_{t-1} \mid V_{t-1}, x_t) \tag{100}$$
using the edge posterior.

3. Predict relation clean-state probabilities conditioned on recovered objects and edges:

$$p_\theta(R_0 \mid V_{t-1}, E_{t-1}, x_t). \tag{101}$$

For each pair (i, j):

$$r_{ij,t-1} = \begin{cases} 0, & e_{ij,t-1} = 0, \\ \text{sample from active relation posterior}, & e_{ij,t-1} = 1, e_{ij,t} = 1. \\ \text{sample from relation marginal} & e_{ij,t-1} = 1, e_{ij,t} = 0 \end{cases} \tag{102}$$

Thus, the reverse sampler follows the semantic order

$$V_{t-1} \to E_{t-1} \to R_{t-1}, \tag{103}$$

and relation labels are sampled only on active directed edges.

A.2.2 Structure-Aware Sampler Refinements

The factorized reverse sampler defined in Sections A.2.1, 3.3 produces samples by approximating the reverse diffusion distribution,

$$p_\theta(x_{t-1} \mid x_t), \quad x_t = (V_t, E_t, R_t^+). \tag{104}$$

While this sampler respects the structural constraints of scene graphs, it may still produce locally inconsistent or semantically implausible configurations due to approximation errors in the learned model. To address this, we introduce a family of *structure-aware refinement kernels* that can be applied at inference time. These refinements operate as additional transition kernels that modify intermediate samples while preserving the constrained scene-graph state space.

Let the original, unrefined reverse sampling be, $y_{t-1} \sim p_\theta(x_{t-1} \mid x_t)$. A refinement step applies a transition kernel

$$x_{t-1} \sim K_t(\cdot \mid y_{t-1}, x_t), \tag{105}$$

The resulting modified sampler is,

$$\tilde{p}_\theta(x_{t-1} \mid x_t) = \sum_y K_t(x_{t-1} \mid y, x_t)\, p_\theta(y \mid x_t). \tag{106}$$

We design refinement kernels that exploit the structured factorization of scene graphs to improve semantic consistency and long-tail relation modeling. All refinement kernels are constructed to preserve the constraint, $e_{ij} = 0 \Rightarrow r_{ij} = 0$.

■ Split Gibbs Refinement

Following the success of split Gibbs samplers Vono et al. (2019); Chu et al. (2025), we consider a entity-wise Gibbs-style refinement over the factorized state, $x = (V, E, R^+)$. Let $S \subseteq \{V, E, R^+\}$ be a subset of entities, and denote its complement by $x_{\neg S}$. A Gibbs update replaces the selected subset by sampling from the conditional distribution over k steps,

$$x_S^{(k+1)} \sim p_\theta(x_S \mid x_{\neg S}^{(k)}, x_t). \tag{107}$$

This defines the corresponding refinement kernel,

$$K_{\text{Gibbs}}(x' \mid x) = p_\theta(x'_S \mid x_{\neg S}, x_t)\, \mathbf{1}[x'_{\neg S} = x_{\neg S}]. \tag{108}$$

For scene graphs, the choices are, $S \in \{V, E, R^+\}$, yielding updates such as

$$R^+ \sim p_\theta(R^+ \mid V, E, x_t), \quad E \sim p_\theta(E \mid V, R^+, x_t), \quad V \sim p_\theta(V \mid E, R^+, x_t). \tag{109}$$

In particular, relation resampling

$$r_{ij} \sim p_\theta(r_{ij} \mid v_i, v_j, e_{ij} = 1, x_t) \tag{110}$$

directly enforces object–relation compatibility. Gibbs refinement corresponds to coordinate ascent algorithm Wright (2015), $x_S^{(k+1)} = \arg\max_{x_S} p_\theta(x_S \mid x_{\neg S}^{(k)}, x_t)$. Thus, Gibbs iteratively moves the sample toward a local fixed point of the model's conditional distributions:

$$x_S \approx \arg\max p_\theta(x_S \mid x_{\neg S}, x_t). \tag{111}$$

■ Soft Mask Refinement

While Gibbs refinement updates entire entities, it may unnecessarily modify already correct entities and, therefore, might be too aggressive. To address this, we introduce a selective resampling mechanism based on model uncertainty.

Let M denote a subset of low-confidence entities. We temporarily mask them, $x_M \leftarrow \texttt{MASK}$ while keeping the remaining entities fixed. The masked entities are then resampled:

$$x'_M \sim p_\theta(x_M \mid x_{\neg M}, x_M = \texttt{MASK}, x_t). \tag{112}$$

This defines the mask refinement kernel,

$$K_{\text{mask}}(x' \mid x) = p_\theta(x'_M \mid x_{\neg M}, \text{mask}(x_M), x_t)\, \mathbf{1}[x'_{\neg M} = x_{\neg M}]. \tag{113}$$

To determine the entities to mask, confidence is derived from the model's predictions. For a entitiy z, we can define confidence based on entropy,

$$\lambda(z) = -\sum_c p_\theta(z = c \mid x_t) \log p_\theta(z = c \mid x_t). \tag{114}$$

We select, $M = \{z : \lambda(z) < \tau\}$ or the top-k most uncertain entities.

For relations, masking is applied only on active edges:

$$M_R = \{(i, j) : e_{ij} = 1,\ \lambda(r_{ij}) < \tau_R\}. \tag{115}$$

Soft mask refinement, therefore, focuses on resampling uncertain entities, improving local consistency while preserving reliable structure.

■ **Rare-Relation Refinement**

Finally, we introduce a refinement mechanism targeting long-tailed relation distributions. Standard relation sampling uses, $p_\theta(r_{ij} \mid V, E, x_t)$, which tends to favor frequent relations. To counteract this, we define a tilted distribution,

$$\tilde{p}_\theta(r_{ij} \mid V, E, x_t) \propto p_\theta(r_{ij} \mid V, E, x_t) \exp(\beta_{\text{rare}}\, s(r_{ij})), \tag{116}$$

where $s(r)$ is a rarity score, $s(r) = -\log p_{\text{data}}(r)$.

Unlike Gibbs refinement, which follows the scene graph factorization $V \to E \to R$, rare-relation refinement inverts the dependency structure locally, $R \to E \to V.$, i.e., we sample rare relations from the rarety tilted distribution in Eqn. 116, force edges, and then revise objects based on whether they match the rare relations sampled. For a subset S_R of selected edges, we perform:

$$\begin{aligned} R'_{S_R} &\sim \tilde{p}_\theta(R_{S_R} \mid V, E, x_t), \\ E' &\sim p_\theta(E \mid V, R', x_t), \\ V' &\sim p_\theta(V \mid E', R', x_t). \end{aligned}$$

This defines the kernel

$$K_{\text{rare}}(x' \mid x) = \tilde{p}_\theta(R'_{S_R} \mid V, E, x_t)\, p_\theta(E' \mid V, R', x_t)\, p_\theta(V' \mid E', R', x_t). \tag{117}$$

This reversed conditioning allows rare relations to induce compatible objects and edges.

■ **Combined Refinement Strategy**

At each reverse timestep t, we apply a composition of refinement kernels (or a subset of them):

$$x_{t-1} \sim K_{\text{mask},t} \circ K_{\text{rare},t} \circ K_{\text{Gibbs},t}\, (p_\theta(x_{t-1} \mid x_t))\,. \tag{118}$$

Refinements are applied selectively over different phases of the reverse process:

- Gibbs refinement in early-to-mid timesteps to improve structural coherence,
- rare-relation tilting in mid timesteps to enhance long-tail relation generation (when needed).
- soft mask refinement in late timesteps to correct low-confidence entities.

They provide a flexible and structure-aware mechanism for improving sample quality without modifying the underlying diffusion model.

A.3 Text-Conditioned Scene Graph Generation via Reward Tilting – Additional Details

Sequential Monte Carlo (SMC) approximation. We approximate sampling from text-conditioned SG distribution, $\tilde{p}(G_{0:T} \mid T)$ using Sequential Monte Carlo with D particles. At timestep t, we maintain a set of D particles, $\{G_t^{(d)}\}_{k=1}^{D}$.

1. Particle Propagation step:

$$G_{t-1}^{(d)} \sim p_\theta(G_{t-1} \mid G_t^{(d)}). \tag{119}$$

2. Weight computation step: For each particle, we compute the predicted clean graph and to distribute the reward across timesteps, incremental weights are used, $w_t = \exp\big(\beta(R_{t-1} - R_t)\big)$,

$$\hat{G}_0^{(d)} = \epsilon_\theta(G_{t-1}^{(d)}, t-1), \tag{120}$$

$$w_{t-1}^{(d)} = \exp\big(\beta(R(\hat{G}_0^{(d)}, T) - R_t^{(d)})\big), \tag{121}$$

where $R_t^{(d)} = R(\hat{G}_0^{(d)}(t), T)$ is the previous reward and T is the text prompt (not to be confused with time T). Followed by normalization,

$$\tilde{w}_{t-1}^{(d)} = \frac{w_{t-1}^{(d)}}{\sum_{j=1}^{D} w_{t-1}^{(j)}}. \tag{122}$$

3. Resampling Step: $\{G_{t-1}^{(d)}\}_{d=1}^{D}$ are the propagated particles with normalized weights $\{\tilde{w}_{t-1}^{(d)}\}_{d=1}^{D}$, where $\sum_{d=1}^{D} \tilde{w}_{t-1}^{(d)} = 1$. We perform resampling by drawing particle indices from a categorical distribution based on reward alignment,

$$I^{(d)} \sim \text{Categorical}\big(\tilde{w}_{t-1}^{(1)}, \ldots, \tilde{w}_{t-1}^{(D)}\big), \tag{123}$$

and setting

$$G_{t-1}^{(d)} \leftarrow G_{t-1}^{(I^{(d)})}. \tag{124}$$

Equivalently, this step samples particles from the empirical distribution, thereby replicating high-weight particles (high text similarity) and discarding low-weight ones.

$$\hat{p}(G_{t-1}) = \sum_{d=1}^{D} \tilde{w}_{t-1}^{(d)}\, \delta_{G_{t-1}^{(d)}}, \tag{125}$$

Reward function. The reward $R(G, T)$ measures compatibility between a scene graph G and a text prompt T. defined as,

$$R(G, T) = \cos\big(f_{\text{CLIP}}(T),\ f_{\text{CLIP}}(\text{serialize}(G))\big), \tag{126}$$

where f_{CLIP} is a pretrained CLIP encoder Radford et al. (2021) and $\text{serialize}(G)$ converts the scene graph into a textual description. Under the SMC procedure, the resulting samples approximate:

$$\tilde{p}(G_0 \mid T) \propto p_\theta(G_0)\, \exp\big(\beta R(G_0, T)\big), \tag{127}$$

thus enabling text-conditioned scene graph generation without retraining the diffusion model.

A.4 Experimental Details

A.4.1 Metrics

■ **Graph Metrics.**

A scene graph is defined as $G = (V, E, R)$, where $V = \{v_i\}_{i=1}^{N_V}$ are node labels with $v_i \in \mathcal{V}$, $E = \{e_{ij}\}_{i,j=1}^{N_E}$ is binary edge existence with $e_{ij} \in \{0, 1\}$, and $R = \{r_{ij}\}_{i,j=1}^{N_R}$ are relation labels with $r_{ij} \in \mathcal{R}$ and $r_{ij} = 0$ indicating no relation. Let p_{data} denote the empirical distribution of real graphs and p_θ denote the distribution of generated graphs.

Maximum Mean Discrepancy (MMD). We evaluate distributional similarity using Maximum Mean Discrepancy (MMD) over graph statistics You et al. (2018); Liao et al. (2019). Given a feature map $\phi(G)$ and a positive-definite kernel (RBF) $k(\cdot,\cdot)$, MMD is defined as,

$$\mathrm{MMD}^2(\phi) = \mathbb{E}_{G,G'\sim p_{\text{data}}}[k(\phi(G),\phi(G'))] + \mathbb{E}_{\hat{G},\hat{G}'\sim p_\theta}[k(\phi(\hat{G}),\phi(\hat{G}'))] - 2\mathbb{E}_{G\sim p_{\text{data}},\hat{G}\sim p_\theta}[k(\phi(G),\phi(\hat{G}))]. \tag{128}$$

We instantiate $\phi(G)$ using different graph statistics:

- **Node MMD (N-MMD):** $\phi_V(G)$ is the histogram of node labels v_i.
- **Relation MMD (R-MMD):** $\phi_R(G)$ is the histogram of relation labels over active edges ($e_{ij}=1$).
- **In-degree MMD (ID-MMD):** $\phi_{\text{in}}(G)$ is the histogram of node in-degrees $d_i^{\text{in}} = \sum_j e_{ji}$.
- **Out-degree MMD (OD-MMD):** $\phi_{\text{out}}(G)$ is the histogram of node out-degrees $d_i^{\text{out}} = \sum_j e_{ij}$.

Triplet-TV. We measure the discrepancy between distributions over <object, relation, subject> triplets. Considering only active edges ($e_{ij}=1$),

$$p(v_i, r, v_j) = \mathbb{P}(v_i, r_{ij} = r, v_j \mid e_{ij} = 1). \tag{129}$$

$$\mathrm{TV}_{\text{triplet}} = \frac{1}{2}\sum_{v_i\in\mathcal{V}}\sum_{r\in\mathcal{R}}\sum_{v_j\in\mathcal{V}} \left|p_\theta(v_i,r,v_j) - p_{\text{data}}(v_i,r,v_j)\right|. \tag{130}$$

Attach-TV. We introduce a new metric, Attach-TV, which evaluates semantic attachment patterns given objects, by comparing relation distributions conditioned on object pairs. For active edges ($e_{ij}=1$),

$$p(r \mid v_i, v_j) = \mathbb{P}(r_{ij} = r \mid v_i, v_j, e_{ij} = 1). \tag{131}$$

$$\mathrm{TV}_{\text{attach}} = \frac{1}{2}\sum_{v_i\in\mathcal{V}}\sum_{v_j\in\mathcal{V}} p_{\text{data}}(v_i,v_j)\sum_{r\in\mathcal{R}} \left|p_\theta(r \mid v_i,v_j) - p_{\text{data}}(r \mid v_i,v_j)\right|. \tag{132}$$

Tail-weighted Conditional TV (RARE-K-TV). To evaluate long-tailed relation modeling, we consider the distribution of relation labels conditioned on active edges, $p(r \mid e=1)$. We define a tail-weighted total variation distance,

$$\mathrm{TV}_{\text{tail}} = \frac{1}{2}\sum_{r\in\mathcal{R}} w_r \left|p_\theta(r \mid e=1) - p_{\text{data}}(r \mid e=1)\right|. \tag{133}$$

The weights emphasize rare predicates:

$$w_r \propto \frac{1}{p_{\text{data}}(r \mid e=1)^\alpha}, \quad \alpha \in [0.5, 1], \tag{134}$$

with normalization $\sum_r w_r = 1$.

■ **Layout Metrics.**

From Xu et al. (2025), for a generated layout, we calculate the F1 score between this generated layout and every ground-truth layout. Assume there are K object categories. Given a pair of generated and ground-truth layouts, the F1 score is calculated as $\text{F1} = \sum_{k\in K} w_k \text{F1}_k$, where F1_k is the F1 score for that class k and weight w_k. Calculating F1_k is based on IoU metric. We use 10 different bounding box IoU thresholds ranging from 0.05 to 0.5 with a step size of 0.05, $\text{F1}_k = \frac{1}{10}\sum_{iou\in[0.05:0.05:0.5]} \text{F1}(iou \mid k)$, where $\text{F1}(iou \mid k)$ means F1 score between two layouts given a specific object category k and a IoU threshold iou. We calculate 4 different types of F1 scores: (1) F1-Vanilla (F1-V), where $w_k = \frac{1}{|K|}$ for every object category; (2) F1-Area (F1-A), where $w_k = \frac{\text{Area}(k)}{\sum_{k\in K}\text{Area}(k)}$ and $\text{Area}(k)$ is the average bounding box area in the validation set for the object category k; (3) F1-Frequency (F1-F), where$w_k = \frac{\text{Freq}(k)}{\sum_{k\in K}\text{Freq}(k)}$ and $\text{Freq}(k)$ is the frequency of the object category k in the validation set; (4) F1-Box, where the F1 calculation is purely based on the bounding box locations, that is, we treat all bounding boxes as having a single object category ($|K|=1$ and $w_k=1$).

We compute results on 300 data points from the CompSGBench dataset for Table 5 as these metrics require expensive LLM inference to analyze the image and extract SG, objects, and relations.

A.4.2 Implementation Details

Denoising network. We use DiGress Vignac et al. (2023) denoiser built on Graph transformer as the backbone but unlike DiGress we do not inject any global features. At timestep t, the input graph state is

$$x_t = (V_t, E_t, R_t),$$

where V_t denotes object labels, E_t, directed edge existence, and R_t, relation labels. Given our factorized setup, we embed each entity separately,

$$h_i^{V,0} = \mathrm{Emb}_V(v_{i,t}) + \mathrm{Emb}_t(t),$$
$$h_{ij}^{E,0} = \mathrm{Emb}_E(e_{ij,t}) + \mathrm{Emb}_t(t),$$
$$h_{ij}^{R,0} = \begin{cases} \mathrm{Emb}_R(r_{ij,t}) + \mathrm{Emb}_t(t), & e_{ij,t} = 1, \\ \mathbf{0}, & e_{ij,t} = 0. \end{cases}$$

The edge-relation pair representation is initialized as,

$$h_{ij}^0 = \phi_E(h_{ij}^{E,0}, h_{ij}^{R,0}),$$

and the graph transformer updates node and pair edge-relation features jointly,

$$(H^{V,L}, H^{P,L}) = \mathrm{GraphTransformer}_\theta(H^{V,0}, H^{P,0}, t).$$

where $L = 1, \ldots, \mathcal{L}$ is the transformer layers.

Structured prediction heads. The factorized discrete space also requires modifying the prediction heads. The denoiser predicts clean graph entities with separate heads for objects and edges,

$$p_\theta(v_{i,0} \mid x_t) = \mathrm{softmax}(W_V h_i^{V,L}),$$
$$p_\theta(e_{ij,0} \mid x_t) = \mathrm{softmax}(W_E h_{ij}^{P,L}),$$

For predicting relations, we condition the relation head on the edge logits. Let $\ell_{ij}^E = W_E h_{ij}^{P,L}$ denote the edge logits, and let $\phi(\cdot)$ be a learned projection of these logits. The relation distribution is defined as

$$p_\theta(r_{ij,0} \mid x_t) = \mathrm{softmax}\Big(W_R\big[\, h_{ij}^{P,L},\ \phi(\ell_{ij}^E)\,\big]\Big).$$

To enforce structural validity, we apply edge-gating at the output,

$$p_\theta(r_{ij,0} \mid x_t) = \begin{cases} \delta_0(r_{ij,0}), & e_{ij} = 0, \\ \mathrm{softmax}\Big(W_R\big[\, h_{ij}^{P,L},\ \phi(\ell_{ij}^E)\,\big]\Big), & e_{ij} = 1. \end{cases}$$

Thus, edge logits provide a soft conditioning signal for relation prediction during training, while hard gating ensures that no semantic relations are assigned to inactive edges.

Layout head. For generating grounded scene graphs and for downstream image generation, we optionally predict continuous layout bounding boxes from the final object-node representations using a layout head.

$$\hat{b}_i = \sigma(W_B h_i^{V,L}) \in [0,1]^4.$$

Training. For each training scene graph $x_0 = (V_0, E_0, R_0)$, we sample, $t \sim \mathrm{Uniform}\{1, \ldots, T\}$ and construct, $x_t \sim q(x_t \mid x_0)$ using the structure-aware forward corruption process. The model is trained to reconstruct x_0. The base diffusion loss is cross-entropy loss between predictions $\{v, e, r\}_{ij,0}$ and clean entities $\{v, e, r\}_{ij,0}^*$,

$$\mathcal{L}_{\mathrm{diff}} = \mathcal{L}_V + \lambda_E \mathcal{L}_E + \lambda_R \mathcal{L}_R.$$

The object loss and directed edge loss are,

$$\mathcal{L}_V = -\frac{1}{\sum_i m_i} \sum_i m_i \log p_\theta(v_{i,0} = v_{i,0}^* \mid x_t) \quad \mathcal{L}_E = -\frac{1}{\sum_{i \neq j} M_{ij}} \sum_{i \neq j} M_{ij} \log p_\theta(e_{ij,0} = e_{ij,0}^* \mid x_t)$$

The relation loss is computed only on active clean edges:

$$\mathcal{L}_R = -\frac{1}{\sum_{i \neq j} M_{ij} e_{ij,0}} \sum_{i \neq j} M_{ij} e_{ij,0} w_{r_{ij,0}} \log p_\theta(r_{ij,0} = r_{ij,0}^* \mid x_t).$$

To handle the long-tailed relation distribution, we use relation class weights:

$$w_r = \frac{1}{(p_{\text{data}}(r) + \epsilon)^{\alpha}}, \qquad \alpha \in [0, 1].$$

For layout head supervision and performing grounding-aware training,

$$\mathcal{L}_{\text{box}} = \frac{1}{\sum_i m_i} \sum_i m_i \left[\|\hat{b}_i - b_i\|_1 + \lambda_{\text{giou}} \big(1 - \text{GIoU}(\hat{b}_i, b_i)\big) \right].$$

The total objective is $\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda_{\text{box}} \mathcal{L}_{\text{box}}$.

Optionally, we also include intermediate reverse-step supervision:

$$\mathcal{L}_{\text{rev}} = \lambda_{\text{rev},V} \mathcal{L}_{\text{rev},V} + \lambda_{\text{rev},E} \mathcal{L}_{\text{rev},E} + \lambda_{\text{rev},R} \mathcal{L}_{\text{rev},R}, \tag{135}$$

where the target is x_{t-1} sampled from the known forward process. Finally,

$$\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda_{\text{box}} \mathcal{L}_{\text{box}} + \lambda_{\text{rev}} \mathcal{L}_{\text{rev}}.$$

Optimization. We optimize the model with AdamW, LR 10^{-4}, weight decay 10^{-2}, gradient clipping, and an exponential moving average of model parameters for sampling. We use the training data distribution as the prior for the forward noising process instead of a uniform distribution. The training is performed on a single NVIDIA A6000 GPU (48GB). The number of diffusion steps for sampling is $T = 100$. The training and sampling scripts are available here[3].

A.5 Related Works

■ **Entity-dependent discrete diffusion graph works:** Some works have explored encoding entity dependencies in generation. GraphMaker Li et al. (2024a) studies large attributed undirected graphs by asynchronously generating node attributes and binary graph topology. However, the underlying diffusion support and state space remain the same as generic graphs. MolDiff Peng et al. (2023) similarly introduces atom–bond consistency through the denoising architecture and asynchronous corruption schedules, but diffusion is still defined over the original molecular state (A, R, B). The underlying diffusion support and state space remain unchanged; there is no constrained state or state-dependent posterior. EDGE Chen et al. (2023) also retains the standard graph diffusion state space and studies diffusion over binary adjacency matrices through absorbing edge corruption and degree-guided generation. It neither augments nor restructures the graph state space, and therefore does not require modifications to the underlying forward or reverse diffusion process. In contrast, **DiScGraph** redefines the diffusion state itself (a foundational upgrade) from a flat graph state to a constrained hierarchical state $\mathcal{SG} = \{(V, E, R), e_{ij} = 0 \to R_{ij} = 0\}$. This changes the support of the Markov chain and consequently requires new forward transition kernels, stationary distribution, marginal transitions, and dependency-aware reverse posterior. The dependency between graph topology and relation semantics is encoded directly in the diffusion process itself, rather than being imposed only through the generation schedule or collapsing into a single categorical variable. This also allows SG structure, edge existence and the relation semantics to be handled coherently without compromising each's generation quality. Thus, our contribution is not simply a scheduling or architectural factorization, but a new discrete diffusion formulation over scene graphs.

■ **Controllable Image Generation:** ***(1) Layout & LLM-augmented methods:*** Layout generation works (Xie et al., 2023; Kong et al., 2022; Inoue et al., 2023; Zhang et al., 2023a) use autoregressive and discrete diffusion paradigms to generate layouts or leverage LLM reasoning to infer layouts from text (Qu et al., 2023; Lian et al., 2024), With backbones like GLIGEN Li et al. (2023), ControlNet Zhang et al. (2023b), they can generate layout-to-image. ***(2) T2I Composition works:*** such as Composed-Diffusion (Liu et al., 2023) and CO3 Dutta et al. (2026) enable algebraic composition of text concepts to rectify misaligned attributes. ***(3) SG-to-Image:*** Several works Johnson et al. (2018); Yang et al. (2022); Farshad et al. (2023) train specific SG-to-image generators. SG-Adapter Shen et al. (2024) introduces SG IP-adapter that activates SG conditioning in existing image models. SDXL-SG Li et al. (2024b) trains a large-scale SOTA image generator SDXL conditioned on scene graphs. SGEdit Zhang et al. (2024) performs SG-to-image editing, and SGDiff Zhang et al. (2025) generates SG to image captions.

A.6 Additional Results

A.6.1 Explanations addressing Table 4

We make two notes here regarding Table 4:

[3] https://annonymous4565.github.io/annon/

(1) The observed regression in FID, CLIP metrics from base SDXL to SDXL-SG is consistent with prior work on fine-tuning pretrained T2I models for structured generation Shen et al. (2024); Li et al. (2024b); Shen et al. (2025). It is noted that SG-conditioning improves compositional generation and spatial understanding but regresses modestly on FID and CLIP. This happens because: (a) Reference-based metrics like FID evaluate similarity to the true image distribution (ground truth) and cannot perceive the abstract Relation Correspondence concept Shen et al. (2024) in complex compositional scenes (CompSGBench). (b) CLIP trained on image-caption pairs, does not transfer its semantic alignment sensitivity to fine-grained spatial relations (the gap to SDXL is very marginal). In contrast, BLIP-VQA and ImageReward evaluate prompt faithfulness and compositional correctness. In SDXL-SG, SG conditioning explicitly biases generation toward satisfying object-relation compositions. Consequently, the generated distribution deviates slightly from the pretrained SDXL distribution, leading to a modest degradation in FID, CLIP while substantially improving compositional correctness.

(2) **DiScGraph**'s differing performance between COCO and CompSGBench for T2I is because CompSGBench, comprising of SGs with over four objects and multi relations, is a benchmark designed specifically to evaluate complex scene composition performance. In contrast, the COCO dataset is not specifically biased for composition, and contains simpler scenes of common objects and relations.

A.6.2 Comparison with Ground Truth Scene Graphs

To isolate the quality of the SG condition under a fixed SDXL-SG backbone, we report SG-IoU, E-IoU, and R-IoU in Table 9 for ground truth SGs from CompSGBench Li et al. (2024b), SGs from **DiScGraph**, DiffuseSG Xu et al. (2025), and DiGress Vignac et al. (2023) evaluated on CompSGBench. Since DiffuseSG and DiGress are unconditional generators, we apply the same training-free reward-tilted text conditioning (Sec. 3.4) to each. Firstly, we observe that **DiScGraph** learns an SG prior that mostly matches the ground truth performance and complements the distributional matching highlighted in Table 1. Secondly, **DiScGraph** outperforms DiffuseSG and DiGress, supporting our insight that scene graphs require a hierarchically constrained discrete diffusion process. Lastly, comparisons to T2I SDXL show the usefulness of structured priors (SG) in compositional image generation.

Table 9: Scene graph conditioned image generation performance comparing different scene graph priors on IoU metrics.

SG prior	Base $P(I/G)$	SG-IoU (↑)	E-IoU (↑)	R-IoU (↑)
GT SG	SDXL-SG	0.5690	0.8528	0.8536
DiScGraph	SDXL-SG	0.5629	0.8357	0.8299
DiffuseSG w/ text-cond	SDXL-SG	0.2733	0.7281	0.6197
DiGress w/ text-cond	SDXL-SG	0.1460	0.6605	0.6088
–	SDXL	0.3260	0.7620	0.7188

Table 10 reports the corresponding performance in metrics: FID, CLIP-I2T, CLIP-I2I, BLIP-VQA, and ImageReward. We observe that, (1) Even Ground truth SGs with SDXL-SG backbone do not beat SDXL in FID and CLIP, supporting the observation (Shen et al. (2024); Li et al. (2024b); Shen et al. (2025)) that the modest degradation in these metrics arises from the SDXL-SG pipeline itself, while the SG-conditioned backbone improves spatial understanding and compositional correctness (BLIP-VQA and ImageReward). (2) Similar to the trend of IoU metrics, **DiScGraph**'s SGs achieve similar downstream performance to GT SGs and outperform DiffuseSG and DiGress.

Table 10: Comparison of image generation performance using different scene graph priors on image quality metrics.

SG prior	Base $P(I/G)$	FID (↓)	CLIP-I2T (↑)	CLIP-I2I (↑)	BLIP-VQA (↑)	ImageReward (↑)
GT SG	SDXL-SG	33.13	0.7404	0.3354	**0.7991**	**1.4653**
DiScGraph	SDXL-SG	33.85	0.7399	0.3350	0.7854	1.4512
DiffuseSG w/ text-cond	SDXL-SG	38.67	0.6235	0.2599	0.6608	1.0848
DiGress w/ text-cond	SDXL-SG	45.72	0.5980	0.2413	0.6075	0.9535
–	SDXL	**31.34**	**0.7415**	**0.3361**	0.6425	1.1286

A.6.3 LLM-based Scene Graph Generation

To ensure **DiScGraph**'s contributions are comparable to current advances, we compare **DiScGraph** against Anthropic Claude Sonnet 4.5, serving as an LLM-based SG generator that converts text prompts to scene graphs,

and the SDXL-SG image generator is used for downstream image generation. Table 11 reports results on 100 random samples from CompSGBench (for feasible LLM inference). We observe that **DiScGraph**'s hierarchical factorized diffusion state and reverse sampler offer a much more reliable text to SG generator $P(G/T)$ than LLM parsing. LLM parsing does not explicitly adhere to SG constraints on edge-relations and object-relation dependence and is also prone to hallucinations.

Table 11: Comparison of image generation performance using **DiScGraph** and LLM-generated SGs.

Method	SG-IoU (↑)	E-IoU (↑)	R-IoU (↑)
SDXL-SG w/ **DiScGraph**	0.6131	0.8622	0.8544
SDXL-SG w/ Sonnet 4.5	0.5570	0.8083	0.8111

A.6.4 Zero-Shot SG-conditioned generation with FLUX.1

In this work, SDXL-SG only serves as the generator $P(I/G)$ that produces images conditioned on the scene graphs for the factorization $P(I/T) = P(G/T)P(I/G)$. **DiScGraph** is agnostic to the modelling of $P(I/G)$. Our main contribution is to design a hierarchically constrained discrete diffusion-based scene graph generator $P(G/T)$. We believe **DiScGraph** will improve any modern T2I generator if SG conditioning support is available, and we chose SDXL-SG for this reason. We also acknowledge this limitation in Sec. 6. However, to prove our understanding that **DiScGraph**'s SGs can improve T2I generation in recent models, We enable zero-shot SG-conditioning in T2I FLUX.1 dev model by using universal guidance Bansal et al. (2023) and evaluate on CompSGBench. The process is as follows:

(1) For a given text prompt, T, we sample $SG \leftarrow P(G/T)$ from **DiScGraph**.

(2) We finetune an off-the-shelf Img→SG converter ReITR Cong et al. (2023) on a subset of LAION-SG ($100K$ samples from the training set 40% randomly sampled + 60% complex scenes with > 4 relations). This will serve as the **<Img,SG>** compatibility loss (offers classifier guidance to latents).

(3) We use the same T in FLUX to get a clean image, score it against ReITR with the sampled **DiScGraph** SG, this loss gradient guides the FLUX latents, resulting in **FLUX-SG-test**.

Table 12 reports the inference-only compositional gains in FLUX from enabling **DiScGraph**'s SG conditioning through steering latents, demonstrating that the structural priors generated by **DiScGraph** are model-agnostic and transfer to other SOTA T2I models.

Table 12: Comparison of FLUX.1-Dev and zero-shot SG conditioned FLUX-SG-test.

Method	FID (↓)	SG-IoU (↑)	E-IoU (↑)	R-IoU (↑)
FLUX.1-Dev	38.27	0.4816	0.7703	0.7922
FLUX-SG-test	37.22	0.5266	0.7998	0.8375

A.7 Ablation Studies

In this section, we provide additional ablation studies.

A.7.1 Sampler Refinements

We ablate over various sampler refinements discussed in Secs. 3.3 & A.2.2 and their schedules. Table 13 (row 1) (on Visual Genome) demonstrates that the factorized reverse sampler, even without the sampler refinements, enforces sampling that adheres to the semantic structure of SGs; objects inform edges, which in turn influence relations. Comparing this to the ablation Table 7 rows 1,2, we can observe: (1) the gain from replacing the joint graph model with the factorized forward state and (2) the additional improvements from the reverse sampler design that is aware of the SG structure.

Table 13 also confirms our understanding that Gibbs refinement is beneficial at early timesteps but not too early as it may be too noisy to effect any improvements. Soft mask refinement is best as a last-mile refinement strategy to quickly resample uncertain predictions. Rare refinement, on the other hand, degrades graph performance at the expense of rare relation sampling. It samples from the distribution defined in Eqn. 116. It explicitly upweights relations in inverse proportion to their empirical frequency and skews the sampling to very rarely occurring relations. RARE-K-TV measures how closely the generated distribution matches the corresponding

data distribution (Eqn 133) over mainly rare relations. The refinement assigns more probability to rare relations that would typically be under-generated by the base sampler. Hence RARE-K-TV falls.

Rare-relation refinement inverts the dependency structure locally, $R \to E \to V$., i.e., we sample rare relations first (Eqn. 116), force edges, and then revise objects based on whether they match the rare relations sampled. Rare relations such as "*dipped in*", "*surrounded by*", or "*bathed in*" occur with a narrow set of compatible object pairs. Once a rare relation is chosen, the sampler must find objects satisfying p_θ. However, because these relations have few training examples, the estimated conditional is imprecise. The relation-first update (Eqn 117) can therefore produce one of two failures: (1) retain the original objects even though the newly sampled relation is not compatible with them (Triplet-TV is worse), or (2) alter the objects toward a rare-relation-compatible configuration, shifting the generated object distribution away from the actual dataset marginal ($\tilde{p}_\theta \neq p_{data}$), increasing N-MMD. Thus, rare-relation refinement intentionally trades fidelity to the full relation (common and rare) distribution of all SGs (p_{data}) for better coverage of its tail (very rare relations).

Thus, we use early Gibbs ($T = 10 - 20$ early timesteps), late soft mask ($T = 90 - 100$, late timesteps) refinements, and no rare refinement as the final setting.

Table 13: Ablation study over different sampling refinement schedules (start - $T_{\text{Gibbs}}, T_{\text{mask}}, T_{\text{rare}}$). Each refinement is applied for $\Delta = 10$ timesteps.

T_{Gibbs}	T_{mask}	T_{rare}	N-MMD (↓)	R-MMD (↓)	ID-MMD (↓)	OD-MMD (↓)	TRIP-TV (↓)	RARE-K-TV (↓)
-	-	-	0.0053	0.0061	0.005	0.0051	0.5513	0.5466
10	-	-	0.0048	0.0058	0.0043	0.0047	0.5367	0.5310
25	-	-	0.0046	0.0058	0.0043	0.0047	0.5288	0.5218
50	-	-	0.0055	0.0063	0.0054	0.0053	0.6182	0.5536
-	50	-	0.0053	0.0058	0.0044	0.0048	0.5560	0.5514
-	75	-	0.0051	0.0060	0.0045	0.0048	0.5548	0.5466
-	90	-	0.0051	0.0060	0.0044	0.0047	0.5479	0.5347
25	90	50	0.0175	0.0424	0.0053	0.0044	0.7235	0.2627
25	**90**	**-**	**0.0046**	**0.0058**	**0.0043**	**0.0046**	**0.4920**	**0.5194**

A.7.2 Entity training loss scales

We perform ablations on different per-entity loss scales i.e., $\lambda_{O,E,R}$ in the training loss (Eqn. 135) on the VG dataset in Table 14. Best performance is observed for $\lambda_O, \lambda_E, \lambda_R = (1, 0.5, 0.5)$.

Table 14: Ablation study on different per-entity (V, R, E) loss scales.

Node scale	Relation scale	Edge scale	N-MMD (↓)	R-MMD (↓)	ID-MMD (↓)	OD-MMD (↓)	TRIP-TV (↓)
1	1	1	0.0012	0.0091	0.0064	0.0060	0.5882
0.5	1	1	0.0045	0.0090	0.0065	0.0060	0.5950
1	0.5	1	0.0011	0.0095	0.0068	0.0062	0.5789
1	1	0.5	0.0011	0.0096	0.0068	0.0062	0.5802
0.5	0.5	1	0.0012	0.0095	0.0068	0.0062	0.5940
1	**0.5**	**0.5**	**0.0046**	**0.0058**	**0.0043**	**0.0046**	**0.4920**
0.5	1	0.5	0.0012	0.0095	0.0065	0.0061	0.5906

A.7.3 Rare-relation Class Weighting

Table 15 ablates on the VG dataset over different class weighting strategies such as, (1) simple $w_k = 1$ (2) Effective num, $w_k = \frac{1-\beta}{1-\beta^{f_k}}$, $\beta \in (0, 1)$ and (3) inverse frequency, $w_k = \frac{1}{f_k}$, where k is a relation class and f_k is class frequency. The effective num approach performs the best, as it is based on diminishing emphasis as class count increases, while inverse frequency is too aggressive.

Table 15: Ablation study on different relation weighting strategy for long-tail relations.

Rel-K class weight	N-MMD (↓)	R-MMD (↓)	ID-MMD (↓)	OD-MMD (↓)	TRIP-TV (↓)	RARE-K-TV (↓)
Simple	0.0024	0.0828	0.0155	0.0155	0.7422	0.5802
Inverse freq	0.0088	0.0105	0.0091	0.0093	0.5257	0.5437
Effective_num	**0.0046**	**0.0058**	**0.0043**	**0.0046**	**0.4920**	**0.5194**

A.7.4 CLIP CONDITIONING STRENGTH

Table 16 reports the effect of the reward guidance scale, β, on image generation performance. The SG-Text similarity CLIP metric, data distributional graph metric Triplet-TV, and downstream image quality metric ImageReward are shown for different values of β . The results demonstrate that, increasing β initially improves SG-text alignment and downstream image quality while preserving the SG data distribution. Beyond a moderate guidance strength, graph-text similarity saturates whereas Triplet-TV deteriorates rapidly, indicating over-guidance that shifts samples away from the learned graph prior without further downstream gains. $\beta = 15$ was used for the paper results.

Table 16: Effect of reward guidance strength β.

β	Graph-Text CLIP (↑)	Triplet-TV (↓)	ImageReward (↑)
0	0.514	0.568	-1.32
5	0.683	0.569	-0.99
10	0.809	0.571	0.71
15	**0.955**	**0.575**	**1.45**
20	0.984	0.587	1.56
50	0.992	0.851	1.54

A.7.5 SAMPLING COST ANALYSIS

Table 17 reports the total sampling cost for generation, inclusive of the reward tilting cost.

Table 17: Sampling cost and memory usage as a function of the number of particles.

Particles D	Reverse Steps T	Refinement Steps (Split-Gibbs, soft-mask), R	Particle Evaluations	Mean Resamples	Time / 100 SG (s)	Peak GPU (GB)
1	100	20	120	0	4	2
16	100	20	1920	7	72	5
24	100	20	2880	7	102	6

Particles are processed jointly as a single batch. Therefore, although the total particle evaluations scale as $D(T+R)$, the sampler performs only $T+R = 120$ batched denoiser forward passes. Runtime increases roughly proportionally with the number of particles (all results reported in the paper use $D = 16$) and quadratically with graph size. Since our method has no architectural limit on the number of nodes, the practical graph size is determined by available GPU memory rather than the algorithm itself and scene graphs typically have a max size around 30, substantiated by CompSGBench, whose largest SG contains 38 objects.

A.7.6 DEGENERATE GRAPH STATISTICS

DiScGraph does not use any acceptance/rejection or post-hoc quality filtering if a sampled scene graph is degenerate/low-quality. Each sampled scene graph is obtained from one complete reverse trajectory. The refinements are in-trajectory transition kernels; they do not reject or replace the final graph based on a separate quality score.

Unlike proteins and molecules, scene graphs do not have a canonical degeneracy definition. We therefore define degeneracy as,

1. SGs with no objects but > 0 relations
2. SGs with entities not within valid class labels ($O, E, R \notin N_o, N_E, N_R$).
3. SGs with active edges but no relations $e_{ij} = 1, r_{ij} = 0$
4. SGs with in-active edges but active relations $e_{ij} = 0, r_{ij} = 1$
5. SGs with the same object class when there are > 10 objects
6. SGs with the same relation class when there are > 5 relations.

We perform ablation experiments on degeneracy, and Table 18 lists the degeneracy counts in the Visual Genome dataset and **DiScGraph** generations per 1000 graphs. We observe that the **DiScGraph** generations do not

produce degenerate graphs disproportionately and perform similarly, with slight differences to the dataset metrics.

Table 18: Frequency of graph degeneracy criteria in the dataset and generated samples.

Degeneracy criteria	Dataset (per 1000)	Generated (per 1000)
1	10.12	0
2	5.68	6.44
3	12.50	11.98
4	0	0
5	8.07	9.21
6	6.66	7.82

A.8 Scene Graph Generation – Qualitative results

A.8.1 Sampled Scene Graphs

Figures 5, 6 are graphs sampled from **DiScGraph** trained on Visual Genome data. Figure 7 is correspondingly for COCO, and Figure 8 shows graphs sampled from **DiScGraph** trained on LAION-SG data.

A.8.2 Graph Completion Tasks

Figures 9, 10 illustrate qualitative results for the single object and single relation completion task. Likewise, Figures 11 and 12 illustrate completion examples on CompSGBench benchmark data.

A.9 Text-Image Generation – Qualitative Results

A.9.1 Text-to-SG-to-Image Generation

Figure 13 shows the generated text-conditioned scene graphs, which were then used for graph-conditioned image generation in Figure 3 in the main text.

Figures 14, 16 illustrate additional qualitative examples of text-to-image generation, comparing text-based composition methods: SDXL, ComposedDiffusion, CO3, and ours, **DiScGraph**. Figures 15, 17 show the corresponding text-conditioned scene graphs sampled from **DiScGraph**, which were in turn used to generate the graph-conditioned images.

A.9.2 Text-to-SG-to-Layout to Image Generation

We highlight **DiScGraph**'s capability to learn grounding signals from scene graphs in Figure 18. The text-conditioned scene graphs are converted to layouts by the layout head, and these layouts are used to generated images using GLIGEN Li et al. (2023) model. **DiScGraph**'s derived layouts thus convert text into a plausible explanation of the visual scene and achieve performance comparable to layout generation methods such as LDM Lian et al. (2024). LDM uses an LLM to infer layouts from text and uses the same downstream model GLIGEN to generate images. Thus, this result highlights that **DiScGraph** learns comparable grounding layouts derived from its good understanding of scene graphs. We specifically use prompts with multiple objects and attributes to highlight the performance.

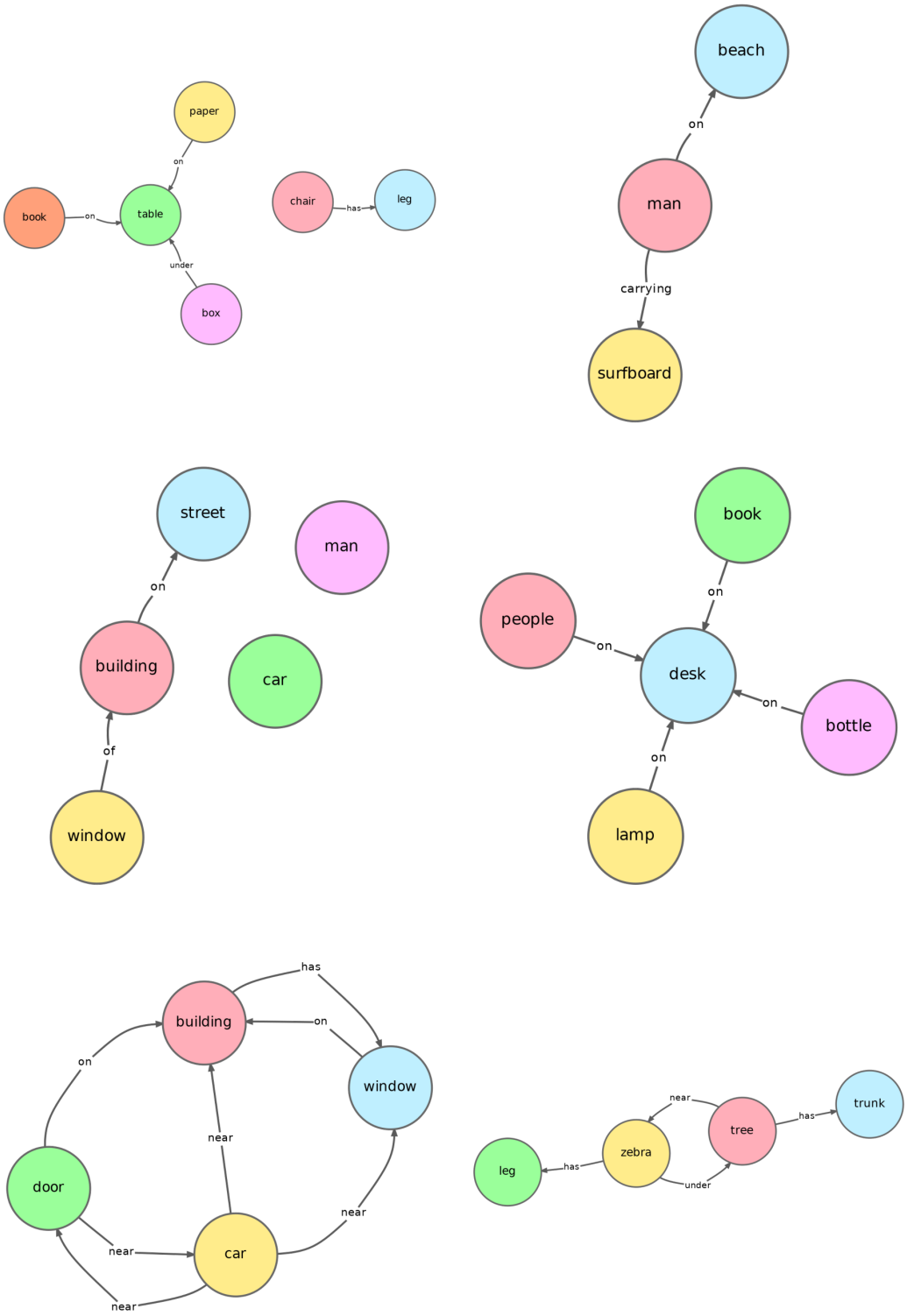

Figure 5: Qualitative Results: Scene graphs sampled from **DiScGraph** trained on Visual Genome Krishna et al. (2016).

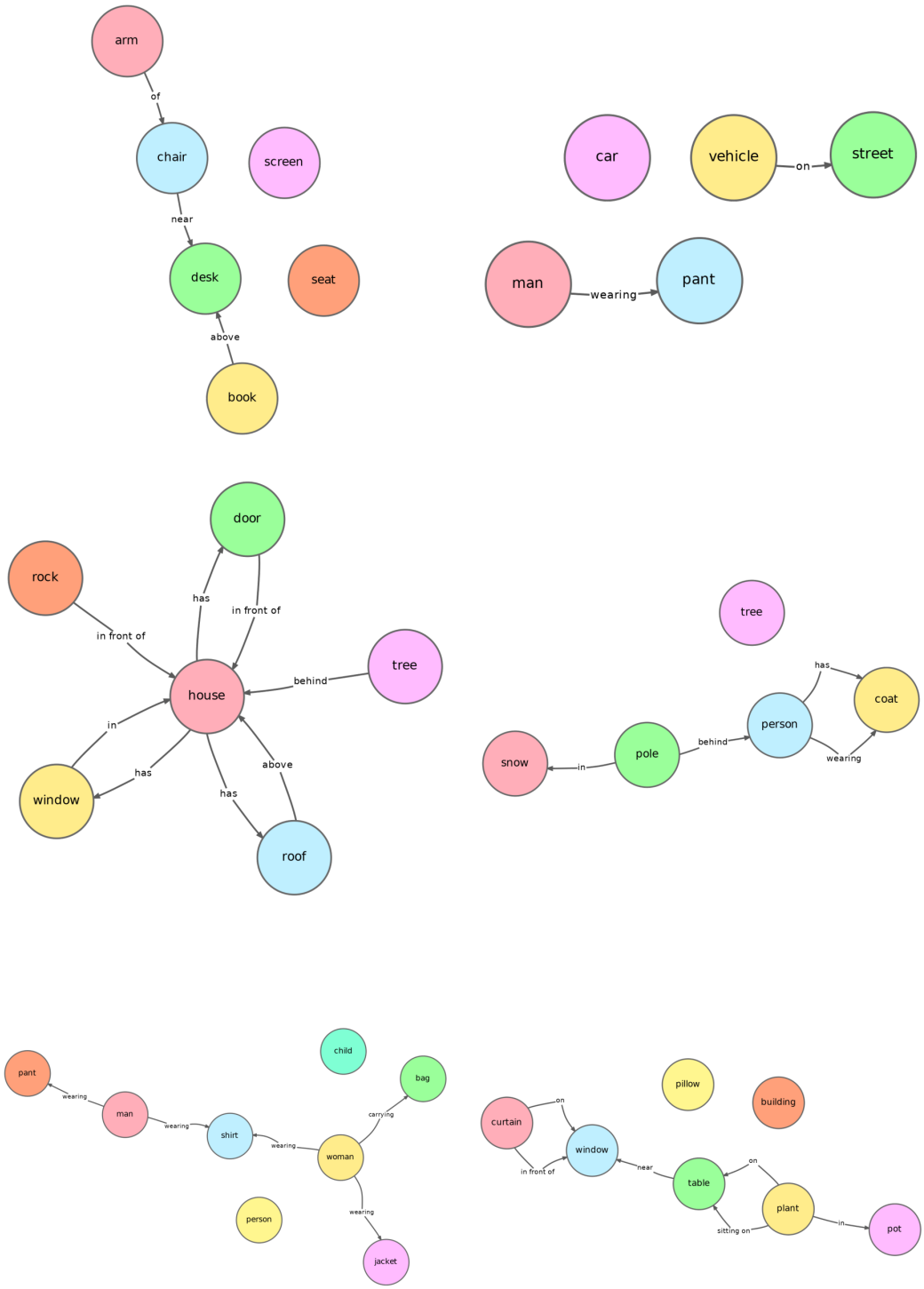

Figure 6: Qualitative Results: Scene graphs sampled from **DiScGraph** trained on Visual Genome.

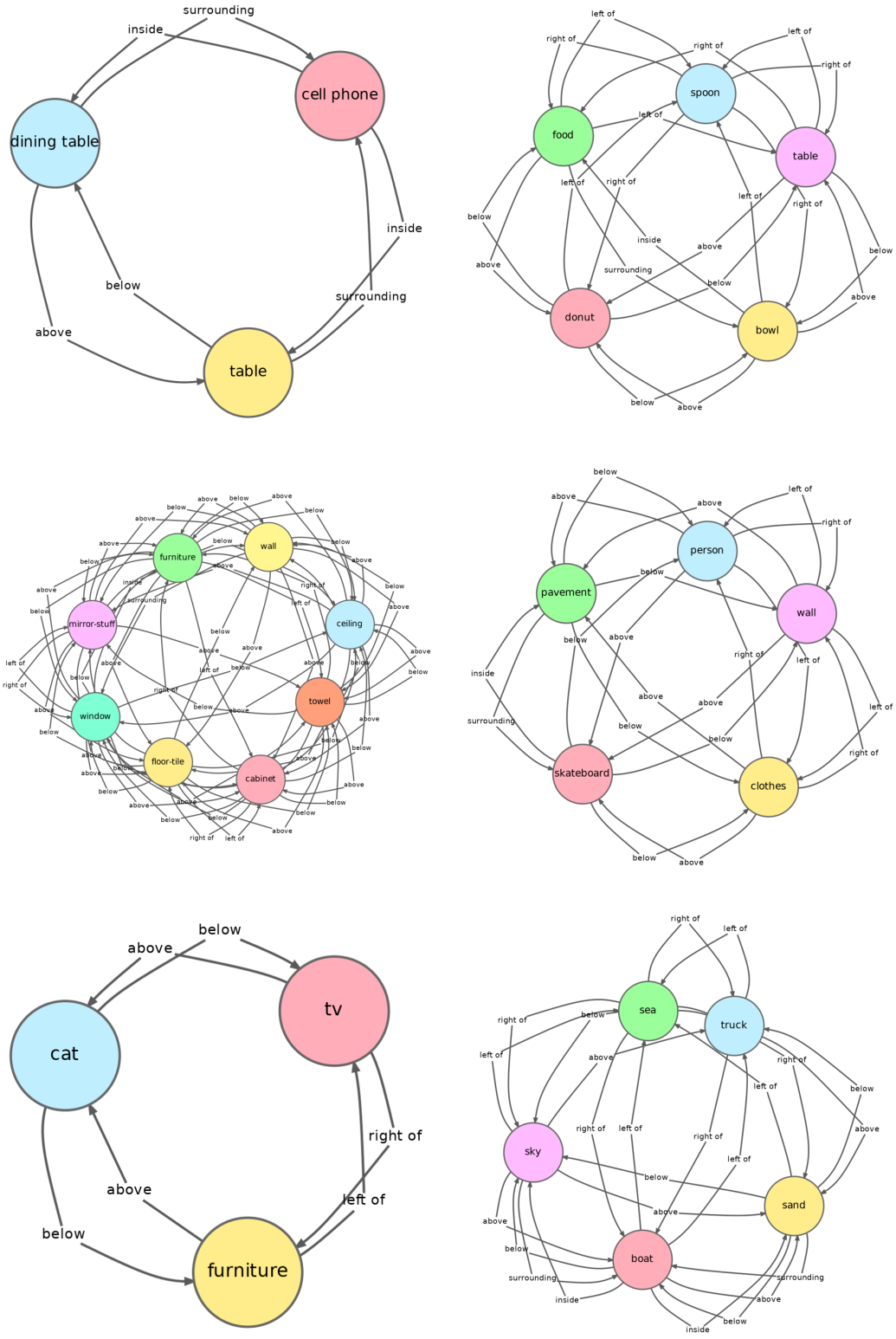

Figure 7: Qualitative Results: Scene graphs sampled from **DiScGraph** trained on COCO dataset Lin et al. (2015). *Note: The COCO dataset represents scenes with symmetric edges.*

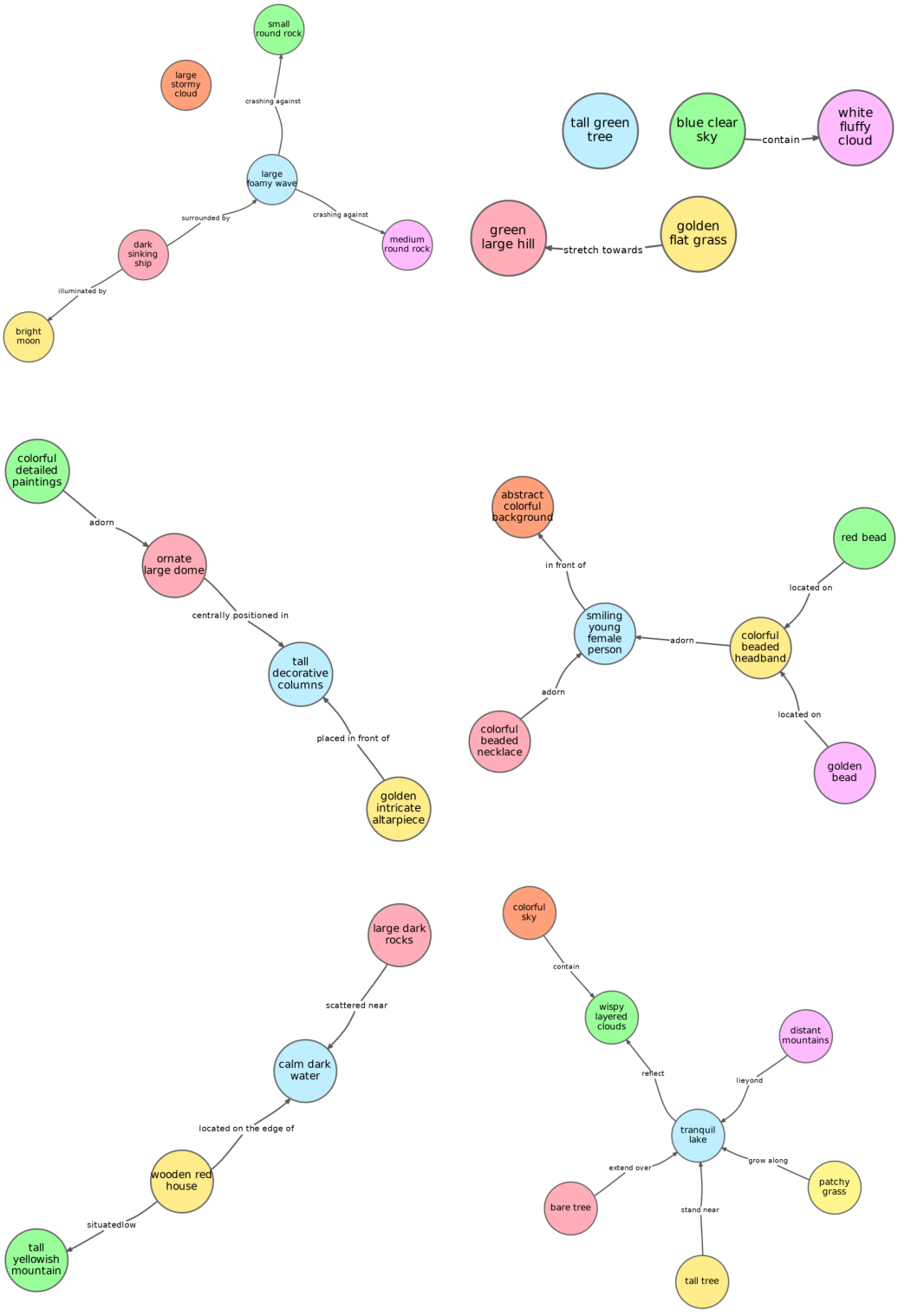

Figure 8: Qualitative Results: Scene graphs sampled from **DiScGraph** trained on LAION-SG Li et al. (2024b).

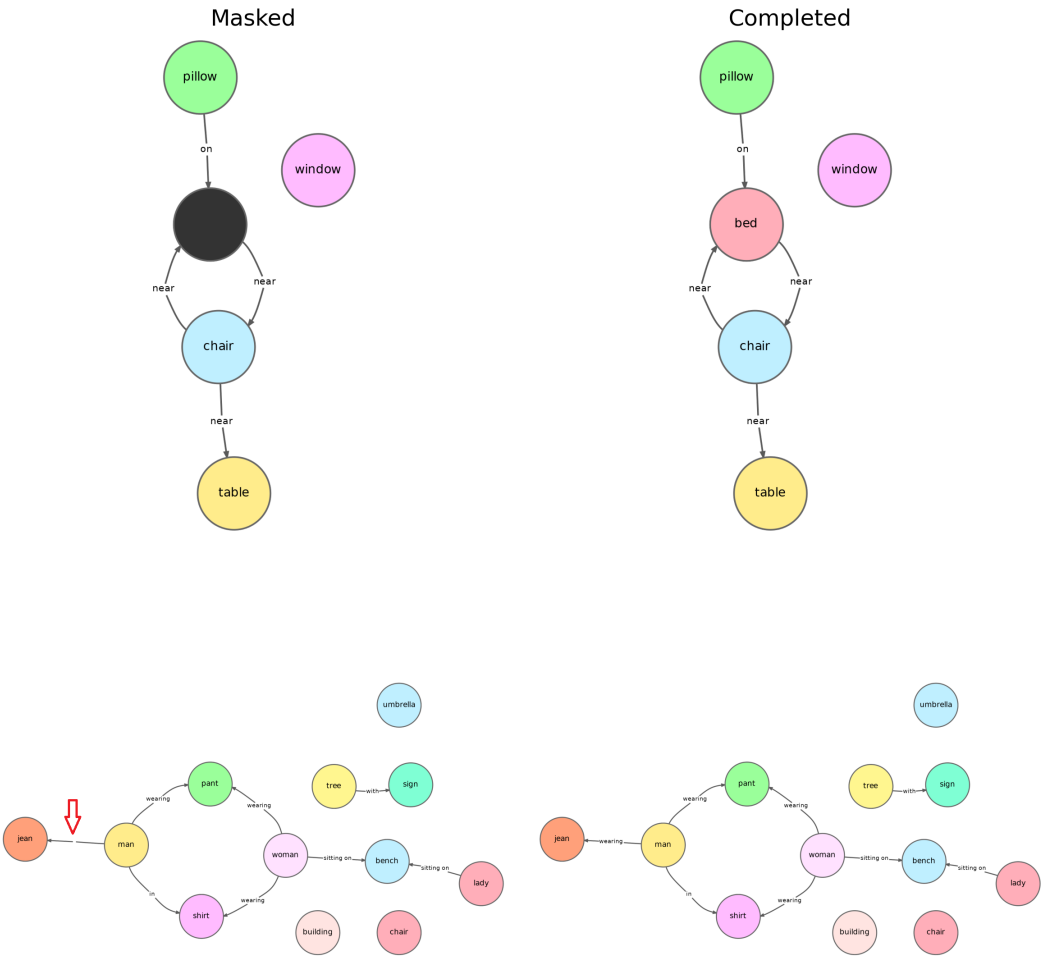

Figure 9: Qualitative Results: **DiScGraph** scene graph single object/relation completion task on Visual Genome. Masked objects are in black, and masked relations are indicated by red arrows.

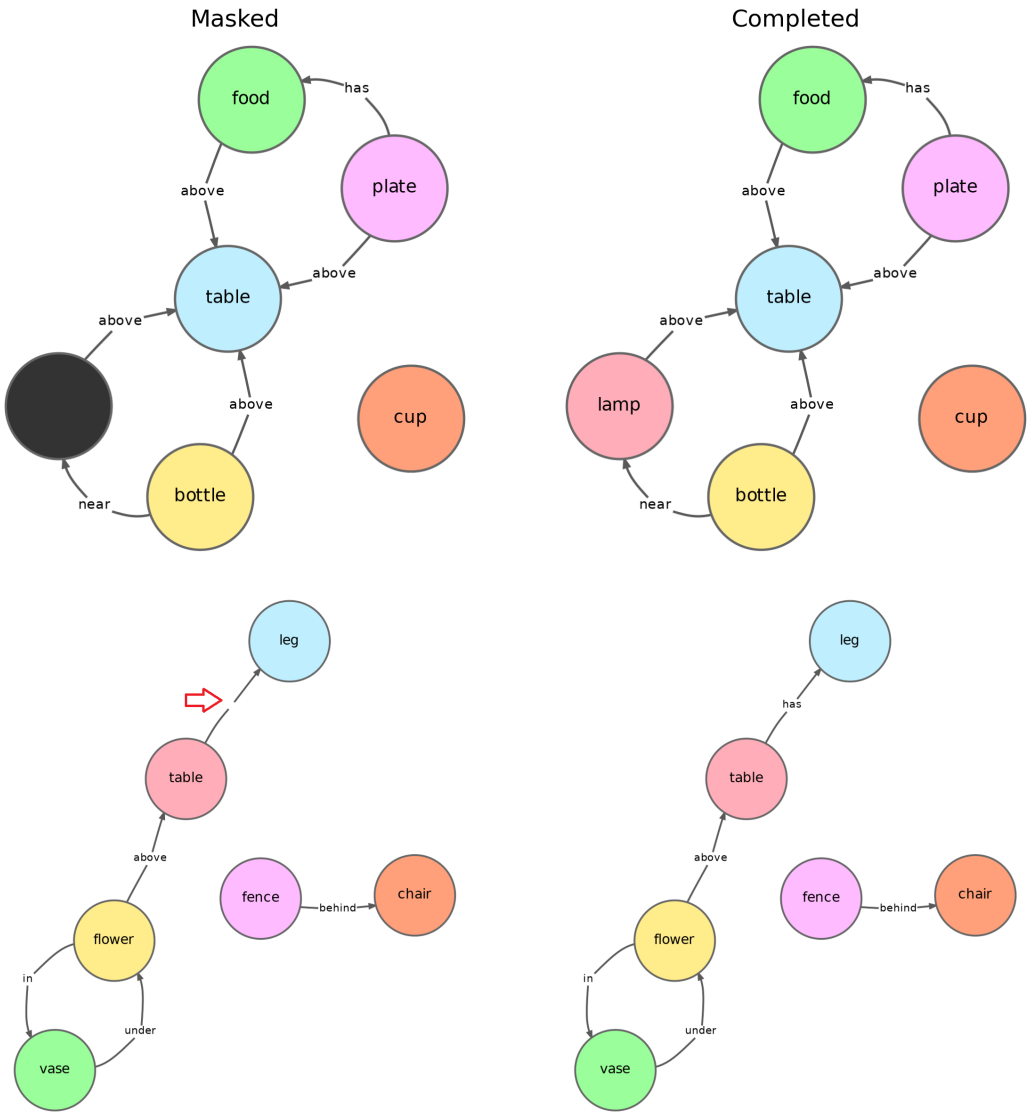

Figure 10: Qualitative Results: **DiScGraph** scene graph single object/relation completion task on Visual Genome dataset. Masked objects are in black, and masked relations are indicated by red arrows.

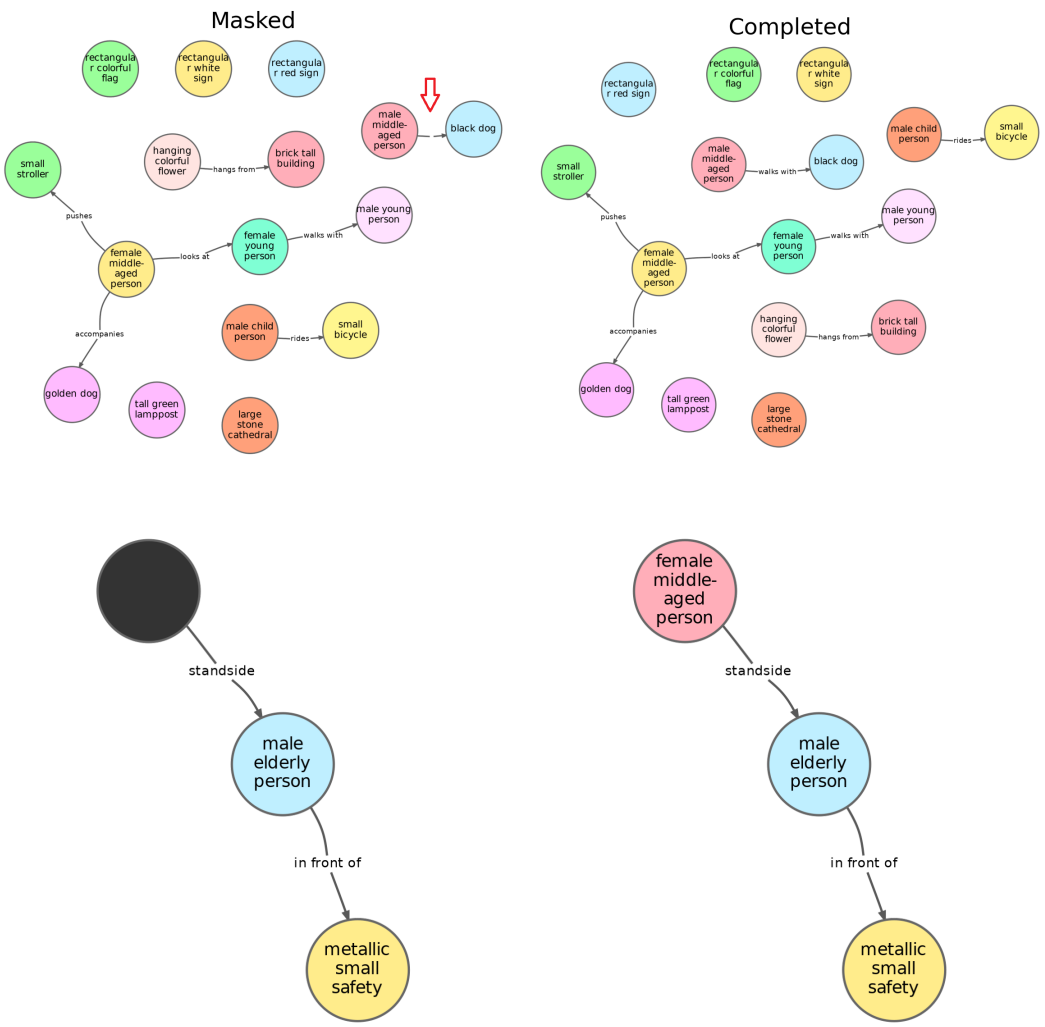

Figure 11: Qualitative Results: **DiScGraph** scene graph single object/relation completion task on CompSGBench Li et al. (2024b). Masked objects are in black, and masked relations are indicated by red arrows.

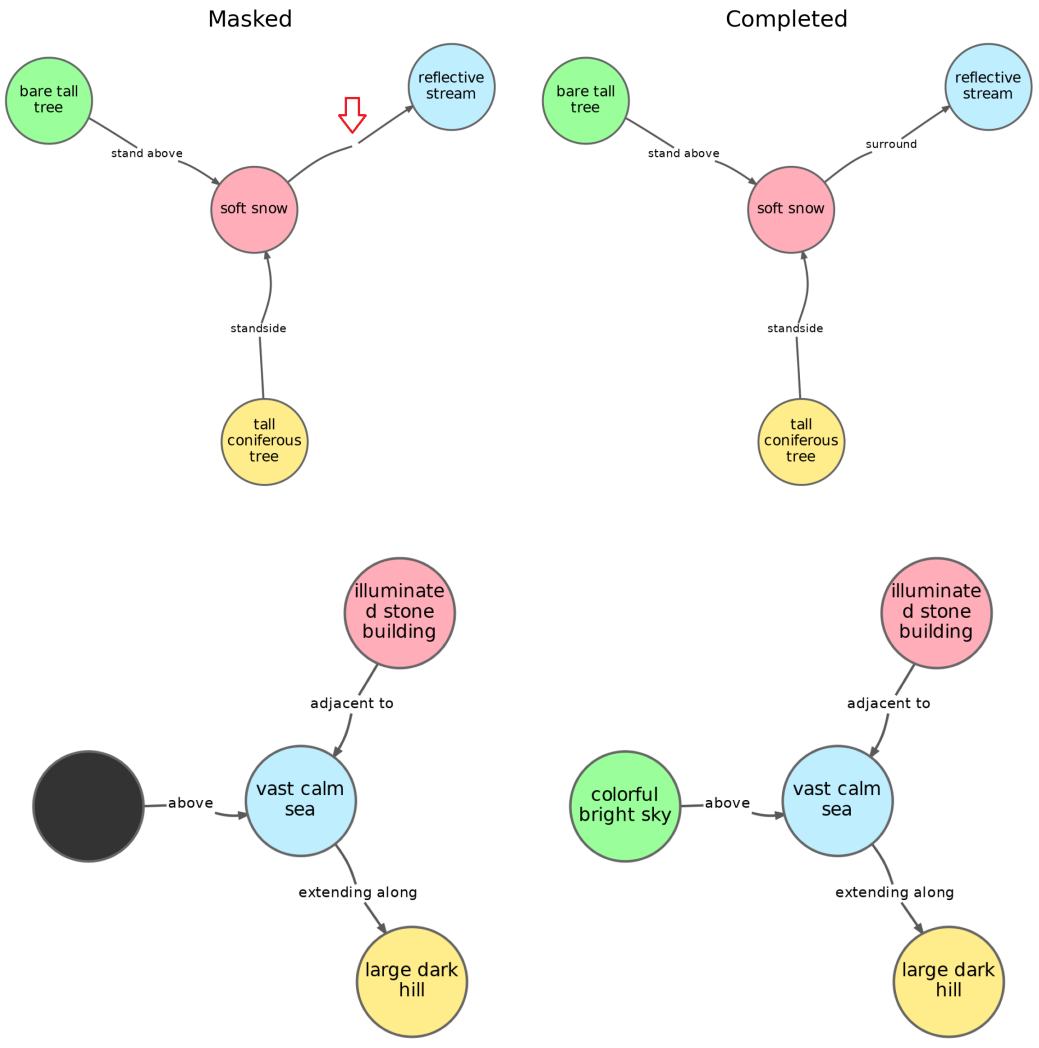

Figure 12: Qualitative Results: **DiScGraph** scene graph single object/relation completion task on CompSGBench. Masked objects are in dark gray, and masked relations are indicated with red arrows.

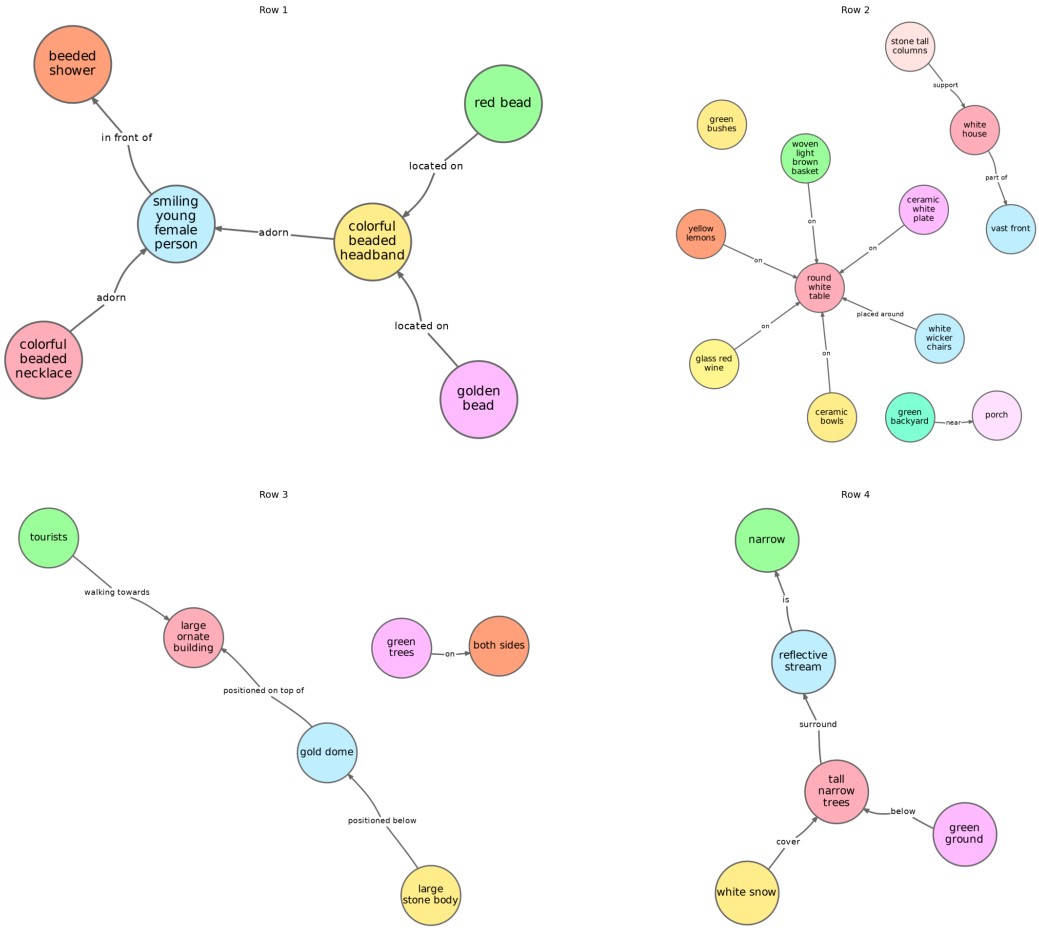

Figure 13: Qualitative Results: Sampled text-conditioned scene graphs used for generating image results of **DiScGraph** in Figure 3 (Main text).

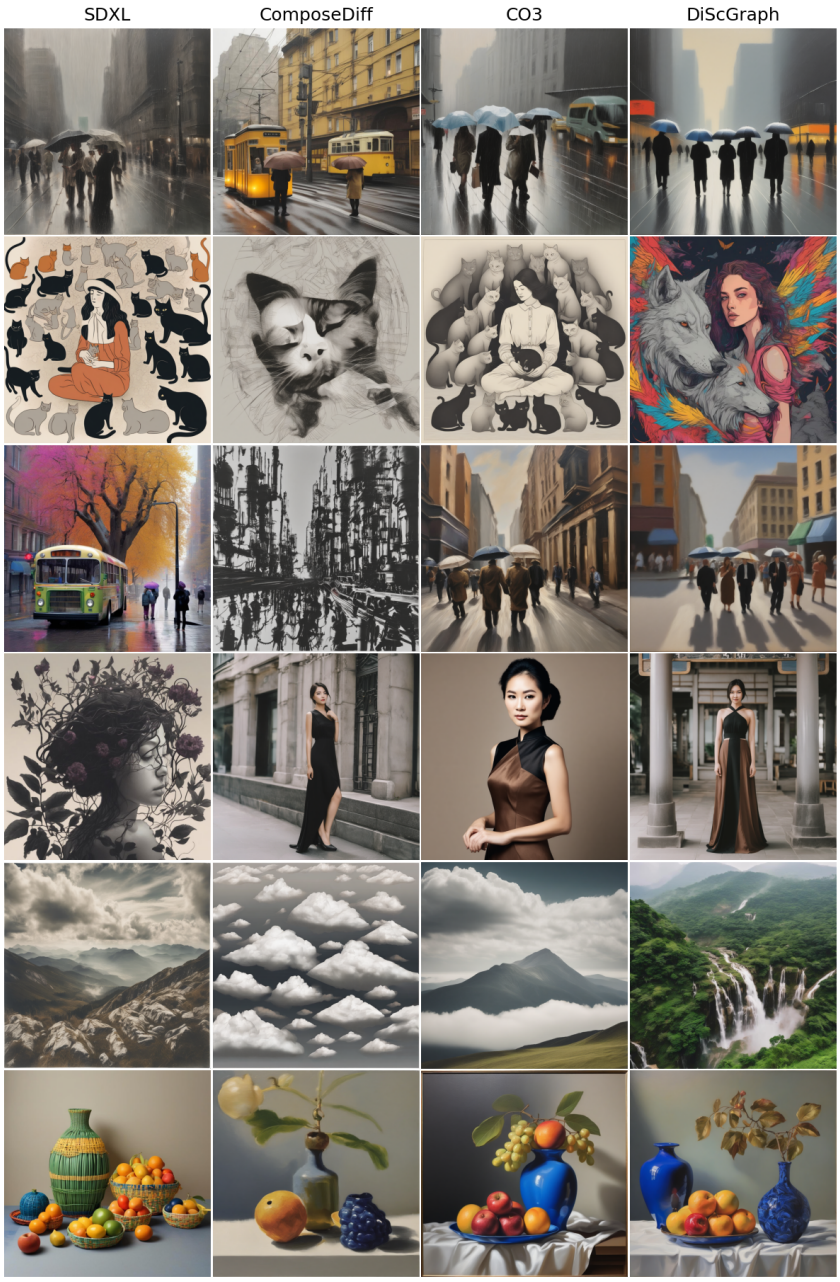

Figure 14: Qualitative Results: T2I generation comparison of methods, SDXL, ComposeDiff, CO3, and **DiScGraph**. For prompts: *"Five people are walking on a wet street with buildings. Two people are wearing skirts and carrying blue umbrellas while the others are wearing pants and long coats while carrying umbrellas. It is gloomy with some bright illumination from lights", "a girl has wolves near her and she has colorful background around her with feathers of different colors.", "Four people consisting of two men and women are walking down a street which is surrounded by buildings on sides. One man is old and another man is young. The women are mother and daughter. They are carrying blue umbrellas.", "An Asian woman in a black and gold dress is standing in front of a large stone archway and is posing for a camera.","a far away perspective shot of a mountainous area with lush green trees and a dramatic waterfall in the middle.", "An oil painting of various yellow and orange fruits on a blue plate that is on a table with table cloth. Two blue vases are on the table.",*

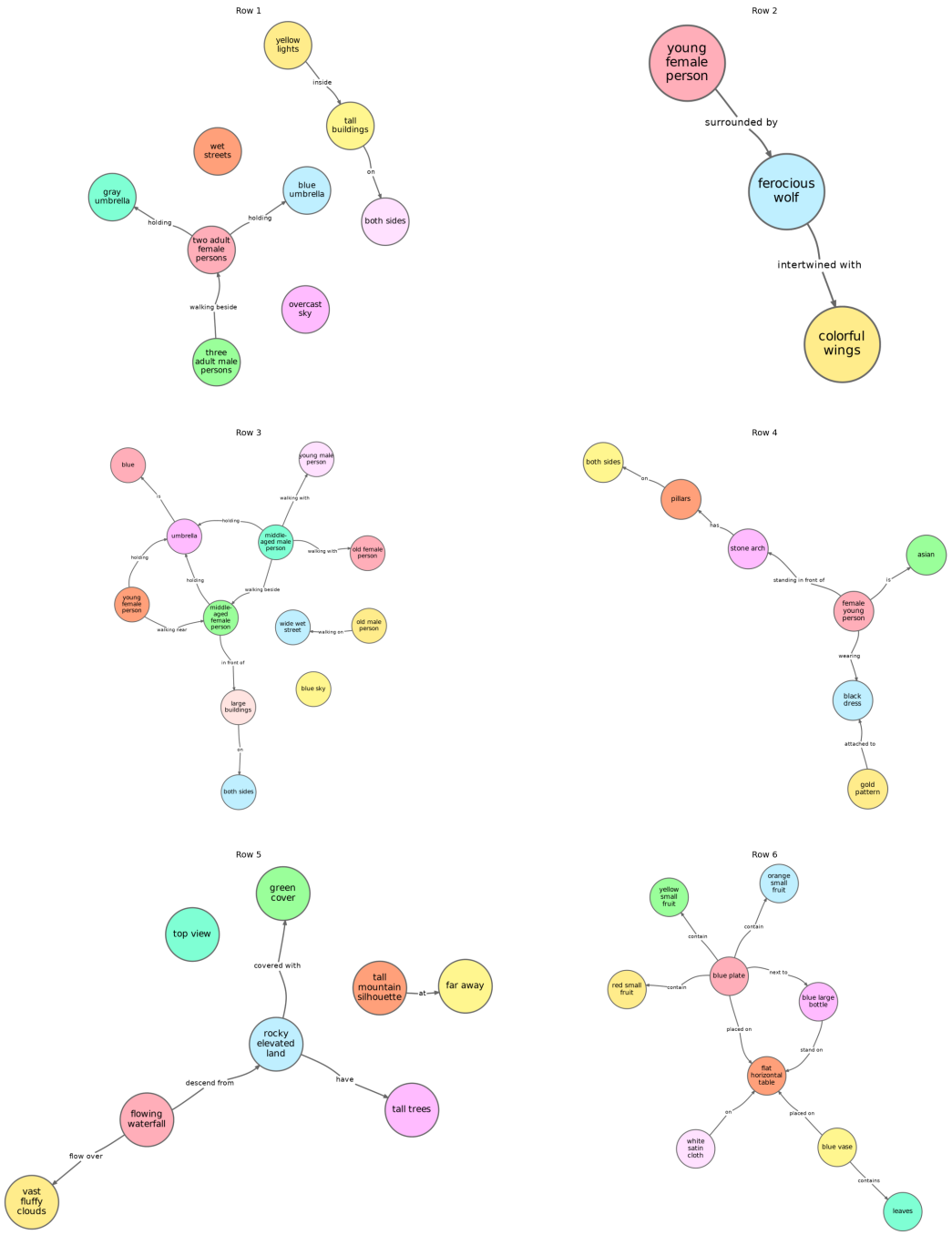

Figure 15: Qualitative Results: Sampled text-conditioned scene graphs used for generating image results of **DiScGraph** in Figure 14



Figure 16: Qualitative Results: T2I generation comparison of methods, SDXL, ComposeDiff, CO3, and **DiScGraph**. For prompts:*"An old woman in a white hat holding an old camera with a blurred car and buildings in the background.", "A woman in traditional Japanese dress walking down a narrow path with trees. The woman's back is turned away.", "A starry twilight sky and a rocky area with a boulder", "An old kitchen that is worn out has large windows with frayed curtains on the right. The kitchen counter has many pots and pans, and cabinets above the stove. The kitchen floor has checkered pattern. A dog sits in front of a dining table and an old woman is sitting at the table cooking.", "The interior of a cavernous building with many arches forming a pattern, golden and made of light blue ,marble."*

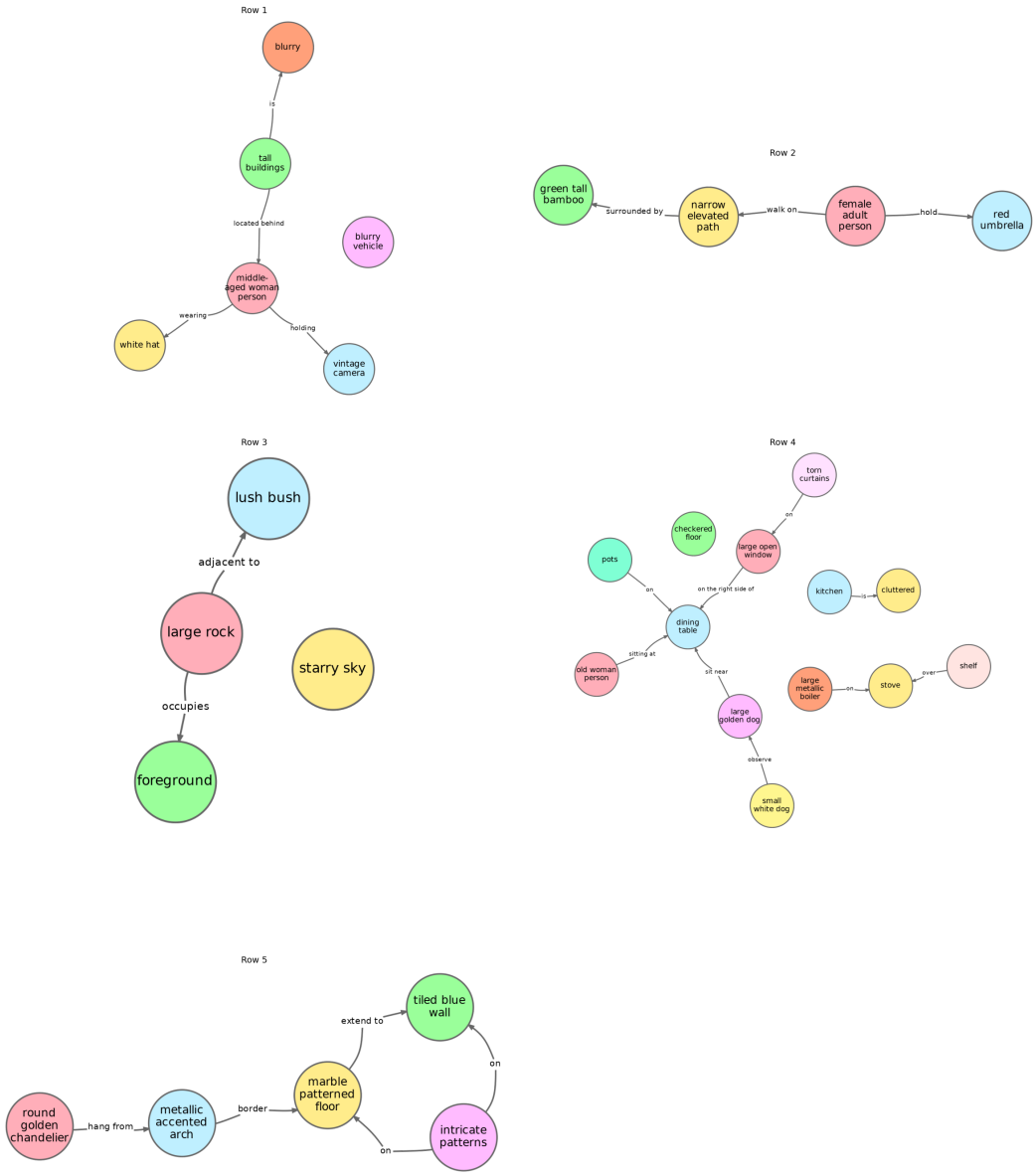

Figure 17: Qualitative Results: Sampled text-conditioned scene graphs used for generating image results of **DiScGraph** in Figure 16



Figure 18: Qualitative Results: Text -> SG -> Layout -> image results of LDM Lian et al. (2024) and **DiScGraph**. For prompts:*:"a bathtub with four plastic bear toys in bright colors", "a man and a woman each throwing a frisbee while jumping in the air, "three egg sandwiches on a white plate","a baby surrounded by many bears", "a white horse and a brown horse near a stable,"a church with a big turret and a clock tower surrounded by trees", "three giraffes standing in front of a barn","three zebras in grasslands","a tractor and two boats on a white sand beach", "a man and a dog on a surfboard riding a wave", "a boy on a skateboard on a road", "a bathroom with a big bath tub, a large window on the right, a big mirror above a sink."*