Semantic Routing: Exploring Multi-Layer LLM Feature Weighting for Diffusion Transformers

Bozhou Li [1 2 *] **Yushuo Guan** [2] **Haolin Li** [3] **Bohan Zeng** [1] **Yiyan Ji** [4] **Yue Ding** [5] **Pengfei Wan** [2] **Kun Gai** [2]
Yuanxing Zhang [2] **Wentao Zhang** [1]

Code: https://github.com/zooblastlbz/SemanticRouting

Abstract

Recent DiT-based text-to-image models increasingly adopt LLMs as text encoders, yet text conditioning remains largely static and often utilizes only a single LLM layer, despite pronounced semantic hierarchy across LLM layers and non-stationary denoising dynamics over both diffusion time and network depth. To better match the dynamic process of DiT generation and thereby enhance the diffusion model's generative capability, we introduce a unified normalized convex fusion framework equipped with lightweight gates to systematically organize multi-layer LLM hidden states via time-wise, depth-wise, and joint fusion. Experiments establish Depth-wise Semantic Routing as the superior conditioning strategy, consistently improving text–image alignment and compositional generation (e.g., +9.97 on the GenAI-Bench Counting task). Conversely, we find that purely time-wise fusion can paradoxically degrade visual generation fidelity. We attribute this to a train–inference trajectory mismatch: under classifier-free guidance, nominal timesteps fail to track the effective SNR, causing semantically mistimed feature injection during inference. Overall, our results position depth-wise routing as a strong and effective baseline and highlight the critical need for trajectory-aware signals to enable robust time-dependent conditioning.

1. Introduction

Diffusion models have established dominance in image and video generation (Esser et al., 2024; Gao et al., 2025; Wu et al., 2025; Kong et al., 2024; Ma et al., 2025; Cai et al., 2025; Wan et al., 2025), particularly via Diffusion Transformers (DiTs; Peebles & Xie, 2023). As the pivotal component guiding visual synthesis, the text encoder has recently witnessed a paradigm shift: moving from encoder-only architectures like T5 (Raffel et al., 2020) and CLIP (Radford et al., 2021) to decoder-only LLMs to leverage their superior semantic expressiveness (BehnamGhader et al., 2024; Ma et al., 2024; Wang et al., 2025a; Li et al., 2025b).

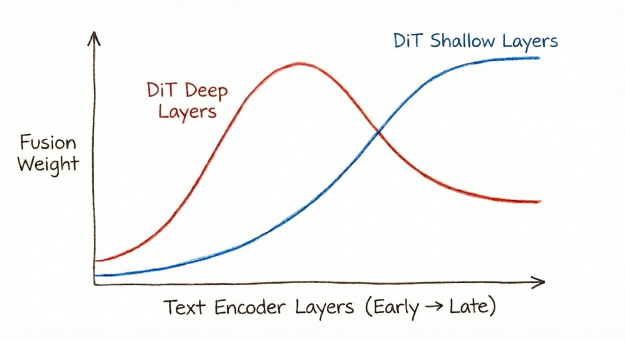


Figure 1. **Learned Fusion Weights.** We observe distinct weight distributions for shallow versus deep DiT blocks, indicating that different generative stages utilize distinct semantic levels from the text encoder.

However, while text encoders have become increasingly potent, the underlying conditioning mechanisms remain static: most prevailing methods inject a fixed text representation, failing to account for the **evolving conditioning needs across both diffusion time and network depth**.

This discrepancy becomes particularly salient when considering the multi-faceted heterogeneity in both LLMs and DiTs. Figure 1 illustrates this functional stratification, depicting distinct semantic preferences between shallow and deep DiT blocks. From the LLM perspective, representations are inherently hierarchical: layers at varying depths capture distinct semantic granularities and levels of conceptual abstraction (Liu et al., 2024b; Jin et al., 2025; Fan et al., 2024; Barbero et al., 2025; Skean et al., 2025). Concurrently, from the DiT perspective, the diffusion trajectory is intrinsi-

* Work done during an internship at Kling Team, Kuaishou Technology. [1]Peking University [2]Kling Team, Kuaishou Technology [3]Fudan University [4]Nanjing University [5]School of Artificial Intelligence, University of Chinese Academy of Sciences. Correspondence to: Wentao Zhang <wentao.zhang@pku.edu.cn>.

Preprint. February 3, 2026.

cally non-uniform across dual axes: (i) temporally, denoising progresses through a coarse-to-fine evolution, where earlier timesteps prioritize low-frequency global structures while later stages focus on high-frequency textural refinement; and (ii) structurally, DiT blocks exhibit functional stratification, contributing unevenly to structural formation versus detail synthesis (Chen et al., 2024).

These observations raise a fundamental research question: To fully unleash the **immense capacity of billion-parameter LLMs**, how can we **adaptively route and aggregate** hierarchical signals through a mechanism **conditioned on diffusion timesteps and distinct DiT network depths**, thereby extracting superior text representations to enhance the overall generative pipeline?

In this work, we take a systematic empirical perspective on text conditioning with multi-layer LLM features. We investigate feature fusion along two orthogonal axes: (i) time-wise adaptivity, where fusion weights vary with the diffusion timestep, and (ii) depth-wise adaptivity, where weights vary with the DiT block index. We further study their combination to understand whether these axes are complementary or introduce undesirable interactions. To enable rigorous, controlled comparisons under matched settings, we develop a unified and lightweight framework that instantiates these alternatives under a single interface with minimal architectural changes, thereby isolating the effects of different fusion strategies. Our contributions are as follows:

- **Unified Framework for Semantic Routing.** We propose a unified formulation that generalizes text conditioning by dynamically weighting multi-layer LLM features. This lightweight framework instantiates time-wise, depth-wise, and joint gating mechanisms, enabling rigorous, controlled comparisons of adaptive strategies.

- **Superiority of Depth-wise Fusion. We establish Depth-wise Semantic Routing as the optimal strategy.** By aligning LLM hierarchy with DiT functional depth, it significantly enhances compositional generation, yielding a substantial **+9.97** improvement on GenAI-Bench Counting over the penultimate-layer baseline and **+5.45** over uniform averaging.

- **Diagnosis of Trajectory Mismatch.** We identify a critical failure mode in time-wise fusion: a **train–inference trajectory mismatch**. We demonstrate that nominal timesteps fail to track the effective SNR during iterative sampling, causing semantic misalignment. This mechanistic insight motivates future designs for trajectory-aware conditioning to enable robust time-dependent fusion.

2. Related Work

2.1. Text Conditioning and Text Encoders

Text conditioning has evolved from U-Net cross-attention (Ronneberger et al., 2015) to DiT's adaLN-Zero (Peebles & Xie, 2023). To enable fine-grained control beyond global adaLN, PixArt-α reintroduced cross-attention (Chen et al., 2023), while MMDiT further unified modalities via joint self-attention (Esser et al., 2024).

Regarding text encoders, early diffusion models predominantly employed encoder-only architectures such as CLIP (Radford et al., 2021) or T5 (Raffel et al., 2020). With the rapid evolution of LLMs, recent research has pivoted toward leveraging decoder-only LLMs to obtain stronger semantics for diffusion conditioning (BehnamGhader et al., 2024; Ma et al., 2024; Wang et al., 2025a; Li et al., 2025b). Due to the suboptimal performance of early iterations like LLaMA-2 (Touvron et al., 2023) in this context, LiDiT introduced a refiner module to enhance extracted text features (Ma et al., 2024). However, with the advent of more capable foundation models, the reliance on such auxiliary components has diminished (Seedream et al., 2025; Wu et al., 2025; Cai et al., 2025; Gao et al., 2025; Kong et al., 2024; Ma et al., 2025). Consequently, the field has gravitated toward direct utilization of LLM representations, where standard practice typically adopts single-layer features, predominantly from the penultimate layer.

2.2. Semantic Heterogeneity and Multi-Layer Fusion

LLMs exhibit distinct representational characteristics across their hierarchy. Probing techniques (Liu et al., 2024b) have elucidated that shallow layers primarily capture lexical semantics, whereas deeper layers are increasingly shaped by next-token prediction objectives. Complementarily, Jin et al. (2025) observed that complex conceptual abstractions are typically acquired in deeper layers. Regarding task specificity, the utility of different layers varies across downstream applications (Fan et al., 2024), while Gurnee & Tegmark (2023) revealed that spatial and temporal information can be encoded in distinct layers. Notably, intermediate layers offer unique advantages, such as attenuated attention sinks (Barbero et al., 2025) and high semantic compression (Skean et al., 2025), highlighting the potential of leveraging multi-layer semantics to enhance text conditioning.

Such heterogeneity highlights the potential of multi-layer conditioning, while recent evidence confirms its superiority over single-layer baselines (Wang et al., 2025a). For instance, Playground v3 (Liu et al., 2024a) and Tang et al. (2025) adopt a deep fusion strategy where internal attention K/V states from the LLM are directly reused in DiT's cross-attention, in pursuit of a deeper fusion between the text encoder and the DiT backbone. However, the former

lacks rigorous controlled experiments to isolate the source of its improvements, while the latter primarily evaluates its approach against the final LLM layer rather than the typically used penultimate layer. While normalizing multi-layer features (Wang et al., 2025a) or introducing learnable weights for adaptive fusion (Li et al., 2025b) has shown promise, these conditioning mechanisms typically remain static, applying a uniform fusion strategy regardless of the diffusion timestep or the DiT block index.

2.3. Temporal and Depth-wise Dynamics in DiT

The diffusion generative process is characterized by distinct, non-uniform dynamics along both temporal and structural dimensions. From a temporal perspective, the denoising trajectory follows a coarse-to-fine paradigm: earlier timesteps prioritize the recovery of low-frequency global structures, whereas later stages transition toward the refinement of high-frequency textures and details (Hertz et al., 2022; Liu et al., 2023; Wang & Vastola, 2023). This temporal non-stationarity suggests that the demand for textual guidance may evolve across different denoising stages. From a depth-wise perspective, DiTs exhibit functional stratification along the network hierarchy: shallower blocks are primarily responsible for structural formation, while deeper blocks contribute more to detail synthesis (Chen et al., 2024). These variations across time and depth prompt us to consider whether multi-layer LLM semantics can be effectively fused via time-wise adaptivity, depth-wise adaptivity, or their combination, aimed at enhancing generative models from the perspective of text conditioning.

3. Method

3.1. Problem Setup

This work investigates text conditioning mechanisms within DiTs under the flow matching formulation (Lipman et al., 2022). Specifically, the conditioning signal is synthesized by aggregating hidden-state sequences across multiple layers of a pretrained LLM. Building upon the observations in Section 2, we identify two primary sources of variation that govern the utility of different semantic abstractions: the flow time $t \in [0, 1]$ and the DiT block index $d \in \{1, \dots, D\}$ (representing network depth). Consequently, we systematically evaluate fusion mechanisms where weights are conditioned on t (time-wise), d (depth-wise), or both (jointly). To ensure a controlled comparison, all variants employ an identical DiT backbone and training protocol, differing exclusively in the design of the fusion module.

3.2. Preliminaries and Notation

Flow matching and timestep. Let $x(t)$ denote the sample state (or latent) at continuous time $t \in [0, 1]$ along the flow, with marginal distribution $x(t) \sim p_t$, where p_0 is a simple base distribution (e.g., Gaussian noise) and p_1 is the target data distribution. A DiT-based backbone parameterizes a text-conditioned vector field $v_\theta\big(x(t), t, c\big)$, where c denotes the text condition. Given c and an initial condition $x(0) \sim p_0$, sampling is performed by integrating the ODE

$$\frac{dx(t)}{dt} = v_\theta\big(x(t), t, c\big), \tag{1}$$

which transports the sample from p_0 toward p_1. For consistency, we refer to t as the timestep throughout this paper.

DiT depth and conditioning site. The DiT backbone consists of D Transformer blocks. We use $d \in \{1, \dots, D\}$ to index the specific block where the conditioning signal is applied. In our experimental setup, the fused text representation $H_{\text{cond}}(t, d)$ provides the conditioning sequence of text hidden states that is fed to the cross-attention module in block d.

Multi-layer LLM features. Let the pretrained LLM provide hidden-state sequences from its entire hierarchy. Let $\mathcal{L}$ denote this set of layers, with $|\mathcal{L}| = L$. We represent the sequence output from layer $l \in \mathcal{L}$ as $H^{(l)} \in \mathbb{R}^{N \times C}$, where N is the text sequence length and C is the LLM hidden dimension.

3.3. A Unified Formulation for Multi-layer Fusion

The hierarchical nature of LLM representations facilitates the capture of complementary linguistic information across a continuum of abstraction levels. Concurrently, the conditioning demands of a generative model are inherently non-stationary: they evolve both across the flow time t and across DiT depth d due to the functional stratification of Transformer blocks. To investigate and mitigate this potential misalignment between semantic supply and conditioning demand, we propose a unified formulation that parameterizes the interaction between the LLM hierarchy and DiT dynamics. This framework enables flexible semantic routing across both temporal and structural axes, subsuming the diverse set of fusion strategies investigated in this study as specific instances.

We instantiate the text condition using a normalized convex fusion of multi-layer features. Specifically, we apply LayerNorm (Ba et al., 2016) to each layer-wise feature to mitigate scale discrepancies across the hierarchy (Kim et al., 2025; Li et al., 2025a). The final fused representation is formed via a softmax-normalized convex combination, which ensures the resulting feature remains within the convex hull of the normalized layer representations, rendering the learned weights directly interpretable:

$$H_{\text{cond}}(t, d) = \sum_{l \in \mathcal{L}} \alpha_{t,d}^{(l)} \cdot \text{LN}\Big(H^{(l)}\Big), \tag{2}$$

where the weights $\alpha_{t,d} = \{\alpha_{t,d}^{(l)}\}_{l\in\mathcal{L}} \in \mathbb{R}^L$ satisfy $\sum_l \alpha_{t,d}^{(l)} = 1$ and $\alpha_{t,d}^{(l)} \geq 0$. These weights are derived by applying a softmax function to the logits $z_{t,d} \in \mathbb{R}^L$:

$$\alpha_{t,d} = \mathrm{Softmax}(z_{t,d}). \tag{3}$$

Different fusion strategies correspond to distinct parameterizations of the logits $z_{t,d}$.

3.4. Fusion Weight Parameterizations

We evaluate the following parameterization schemes under the framework of Eq. (2).

(B1) Penultimate-layer baseline. We utilize only the penultimate LLM layer for conditioning:

$$H_{\text{cond}}(t,d) = \mathrm{LN}\Big(H^{(l^\star)}\Big), \quad l^\star = \text{penultimate}. \tag{4}$$

(B2) Uniform normalized averaging. We aggregate all layers via a uniform average after normalization, without introducing learnable fusion weights:

$$H_{\text{cond}}(t,d) = \frac{1}{L}\sum_{l\in\mathcal{L}} \mathrm{LN}\Big(H^{(l)}\Big). \tag{5}$$

(B3) Static learnable fusion. We learn a single global logit vector shared across all pairs of (t,d):

$$z_{t,d} = \beta, \quad \beta \in \mathbb{R}^L \text{ (learnable)}. \tag{6}$$

Time-conditioned fusion gate (TCFG). To facilitate time-dependent adaptivity, we introduce a lightweight gating module that maps the flow time t to fusion logits. We first embed the continuous time using a sinusoidal encoding $\phi(t)$ and subsequently compute the logits via a small MLP:

$$z_t = g_\psi\big(\phi(t)\big), \quad g_\psi = \mathrm{MLP}(\cdot), \tag{7}$$

where $z_t \in \mathbb{R}^L$ yields fusion weights via Eq. (3). The TCFG serves as the fundamental building block for both time-wise and joint fusion strategies. More details can be seen in Appendix A.

(S1) Time-wise fusion. We apply a shared TCFG across all DiT blocks, making the fusion dependent on t but invariant to d:

$$z_{t,d} = z_t = g_\psi\big(\phi(t)\big). \tag{8}$$

(S2) Depth-wise fusion. We learn block-specific logits that depend on the depth index d but remain static over time t:

$$z_{t,d} = z_d = \beta_d, \quad \beta_d \in \mathbb{R}^L \text{ (learnable for each } d). \tag{9}$$

(S3) Joint time-and-depth fusion. We model the dependency on both t and d by employing a depth-specific TCFG for each DiT block. Concretely, each block d has its own gating function g_{ψ_d}:

$$z_{t,d} = g_{\psi_d}\big(\phi(t)\big). \tag{10}$$

The weights $\alpha_{t,d}$ are then obtained via Eq. (3) to compute $H_{\text{cond}}(t,d)$ via Eq. (2).

4. Experiments

4.1. Experimental Setup

Models. We employ Qwen3-VL-4B-Instruct (Bai et al., 2025) as the text encoder and the pretrained VAE from Stable Diffusion 3 (SD3) (Esser et al., 2024). The diffusion backbone is a cross-attention-based DiT comprising $D = 24$ Transformer blocks and approximately 2.24B parameters. The architectural design of the backbone follows the implementation of FuseDiT (Tang et al., 2025): we apply 1D RoPE (Su et al., 2024) to text prompts, 2D RoPE (Heo et al., 2024) to image latents, and use QK-Norm (Henry et al., 2020) in attention. The only architectural difference from FuseDiT is that we do not use Sandwich Norm (Gong et al., 2022). Unless otherwise specified, all conditioning variants in Section 3 share this identical backbone and differ only in how multi-layer text features are fused. For the TCFG module, we utilize a 128-dimensional sinusoidal encoding for the timestep input.

Dataset. We train all models on a high-quality subset of LAION-400M (Schuhmann et al., 2021), comprising approximately 30 million image-text pairs. We replace the original texts with dense synthetic captions generated by Qwen3-VL-32B-Instruct (Bai et al., 2025). Images are resized to 256×256, and text prompts are tokenized with a maximum length of 512 tokens.

Training. We train all models using AdamW (Loshchilov & Hutter, 2017) with $(\beta_1, \beta_2) = (0.9, 0.999)$, a learning rate of 1×10^{-4}, weight decay of 1×10^{-4}, and a constant learning rate scheduler. The batch size is 512, and all models are trained for 500k steps. The prompt drop ratio is set to 0.1 to enable unconditional generation. For timestep sampling, we follow the logit-normal distribution used in SD3.

Baselines. We evaluate our proposed strategies against two categories of baselines: (i) the **Standard Baselines** (B1–B3) introduced in Section 3.4; and (ii) **FuseDiT** (Tang et al., 2025), a representative deep-fusion approach that reuses internal LLM attention K/V states directly within the DiT attention layers to facilitate a more intrinsic integration between the text encoder and the visual backbone.

Evaluation. Unless otherwise specified, images are generated using 50 sampling steps with a CFG (Ho & Salimans, 2022) scale of 6.0 and the FlowMatch Euler scheduler from diffusers (von Platen et al., 2022). We use GenAI-Bench (Li et al., 2024) and GenEval (Ghosh et al., 2023) to assess text–image alignment in generated samples. For GenAI-Bench, evaluation is conducted using Qwen3-VL-235B-A22B-Instruct as the judge model. To quantify aesthetic appeal, we report the style dimension scores from UnifiedReward-2.0 (Wang et al., 2025b) on samples generated using the DrawBench prompt set (Saharia et al., 2022).

4.2. Main Results

Table 1. Performance of different fusion strategies on three benchmarks. Best in each column is in **bold**, and second best is underlined.

Method	GenEval ↑	GenAI ↑	UnifiedReward ↑
Baselines			
B1: Penult.	64.54	74.96	3.02
B2: Uniform	<u>66.51</u>	76.82	**3.06**
B3: Static	64.77	76.31	<u>3.05</u>
Deep-fusion baseline			
FuseDiT	60.95	75.02	<u>3.05</u>
Our fusion strategies			
S1: Time	63.41	76.20	2.97
S2: Depth	**67.07**	**79.07**	**3.06**
S3: Joint	66.05	<u>77.44</u>	**3.06**

We report the overall performance in Table 1 and the granular capability breakdown on GenAI-Bench in Table 2. The empirical results summarized in these tables reveal several key insights:

Limits of Static Aggregation. First, aggregating multi-layer features (B2–S3) consistently outperforms the penultimate-layer baseline (B1). This trend suggests that LLM hierarchies contain complementary semantic signals that are largely underutilized by conventional single-layer conditioning. Furthermore, the learnable static fusion (B3) fails to surpass uniform normalized averaging (B2), indicating that without explicit adaptivity, a fixed set of learned weights is not robust enough to outperform a strong uniform prior.

Interplay in Deep Fusion Architectures. Regarding the deep-fusion baseline, FuseDiT, Table 1 reveals that its architecture struggles to effectively extract essential textual information for high-quality synthesis. We hypothesize that this performance gap stems from an inherent architectural constraint: by directly reusing internal LLM key/value states within cross-attention, FuseDiT imposes a restrictive coupling that deprives the DiT backbone of the flexibility to dynamically re-contextualize text features. These observations offer a useful perspective for future unified architectures, highlighting the importance of carefully considering the interplay between internal state-sharing mechanisms and the task-specific feature extraction capabilities required for high-fidelity generative modeling.

Superiority of Depth-wise Semantic Routing. Among our proposed strategies, **depth-wise fusion (S2)** delivers the most robust and significant overall gains. Notably, introducing learnable weights in S2 yields clear improvements over B2 (which can be viewed as a fixed-weight depth-wise scheme). This contrast with the static setting (where learnable B3 fails to surpass B2) highlights a critical divergence: purely global parameterization is ineffective; rather, the value of learnability is effectively unlocked only when aligned with the depth-wise structural hierarchy.

This advantage implies that hierarchical LLM semantics are indispensable for navigating intricate prompts. As shown in Table 2, performance gains are disproportionately pronounced within "Advanced" categories. Taking the "Counting" task as a representative case, S2 achieves a substantial improvement of **+9.97** over B1 and **+5.45** over B2. This disparity underscores a pivotal insight: naive aggregation of multiple layers is insufficient to unlock the full potential of text conditioning. Rather, the synergy created by allowing functional blocks at different DiT depths to selectively route and aggregate task-relevant LLM-layer semantics is the key to mastering compositional reasoning and fine-grained constraint following.

Instability of Time-Awareness and Joint Mitigation. In contrast, purely time-wise fusion (S1) does not provide consistent benefits and often leads to degraded generation quality, manifesting as noticeable blurriness and loss of fine details (see Appendix B). We provide a detailed mechanistic diagnosis of this phenomenon in Section 5.2, attributing it to optimization conflicts arising from fundamental train–inference inconsistencies along the temporal axis. Joint fusion (S3) remains competitive but is slightly less effective than S2. Notably, S3 avoids the blurriness characteristic of S1 by incorporating depth-specificity, a phenomenon further analyzed in Section 5.2.

5. Analysis

We analyze the proposed fusion strategies from three aspects: (i)the distribution patterns of fusion weights across flow time t and DiT depth d, validating the interpretability of the learned semantic routing; (ii) a mechanistic diagnosis of the degradation observed in purely time-wise fusion, examining how the mismatch between training-time temporal representations and the inference denoising trajectory

Table 2. Fine-grained GenAI-Bench performance. **Basic skills** include Attribute, Scene, Spatial relations, Action relations, and Part relations. **Advanced skills** include Counting, Comparison, Differentiation, Negation, and Universal. We report average scores for each group; signed changes relative to B1 are colored (green/red). **Bold** and underlined denote best and second-best results, respectively.

	Summary			Basic					Advanced				
Method	Avg	Basic	Adv.	Attr.	Scene	Spat.	Action	Part	Count.	Comp.	Differ.	Neg.	Uni.
Baselines													
B1: Penult.	74.96	80.05	70.55	79.04	85.38	81.27	83.28	73.03	64.60	66.70	65.39	71.21	72.76
	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00	+0.00
B2: Uniform	76.82	81.28	72.95	81.10	86.07	81.98	85.01	73.13	69.12	68.19	70.49	71.51	71.10
	+1.86	+1.23	+2.40	+2.06	+0.69	+0.71	+1.73	+0.10	+4.52	+1.49	+5.10	+0.30	-1.66
B3: Static	76.31	80.36	72.88	79.41	84.93	80.59	84.13	75.78	68.32	70.52	70.57	74.73	68.21
	+1.35	+0.31	+2.33	+0.37	-0.45	-0.68	+0.85	+2.75	+3.72	+3.82	+5.18	+3.52	-4.55
Deep-fusion baseline													
FuseDiT	75.02	77.65	72.65	77.45	83.49	78.69	81.39	70.52	67.23	66.15	68.48	74.16	73.31
	+0.06	-2.40	+2.10	-1.59	-1.89	-2.58	-1.89	-2.51	+2.63	-0.55	+3.09	+2.95	+0.55
Our fusion strategies													
S1: Time	76.20	79.69	73.16	79.17	83.50	81.03	83.18	72.38	66.37	71.74	71.49	**75.79**	70.98
	+1.24	-0.36	+2.61	+0.13	-1.88	-0.24	-0.10	-0.65	+1.77	+5.04	+6.10	**+4.58**	-1.78
S2: Depth	**79.07**	82.68	**76.03**	81.67	**88.33**	83.08	86.10	**77.89**	**74.57**	**72.29**	**74.31**	74.05	**76.09**
	+4.11	+2.63	**+5.48**	+2.63	**+2.95**	+1.81	+2.82	**+4.86**	**+9.97**	**+5.59**	**+8.92**	+2.84	**+3.33**
S3: Joint	77.44	**82.92**	72.71	**82.16**	87.90	**85.13**	**87.08**	74.16	67.55	69.44	70.79	72.54	72.20
	+2.48	**+2.87**	+2.16	**+3.12**	+2.52	**+3.86**	**+3.80**	+1.13	+2.95	+2.74	+5.40	+1.33	-0.56

leads to semantic misalignment; and (iii) the computational overhead introduced by the additional fusion modules.

5.1. Weight Dynamics over Time and Depth

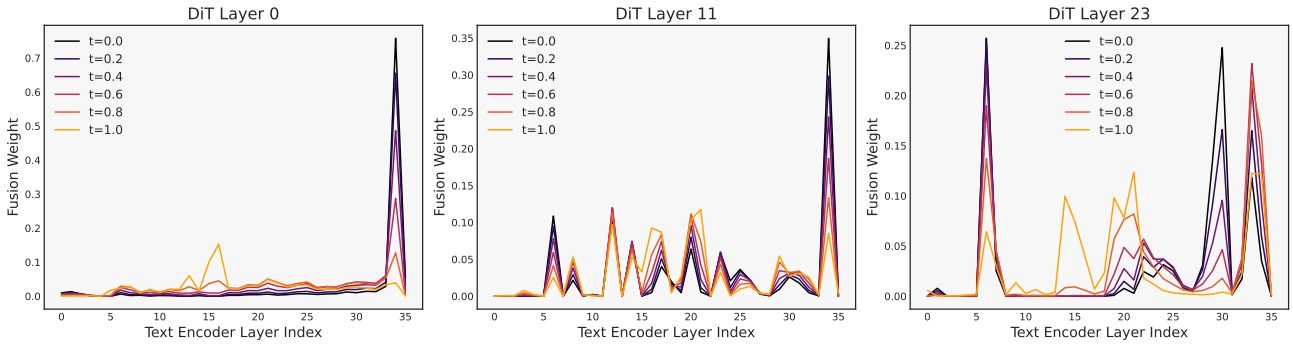
To validate that the learned fusion weights capture meaningful semantic preferences rather than arbitrary noise, we analyze their evolution across timestep t and DiT depth d in Figure 2. More details are provided in Figure C.2.

Text Encoder Layer Specificity. According to Figure 2, the learned weights exhibit clear selectivity: the initial and final text encoder layers consistently receive negligible attention, confirming that effective semantics reside within the model's internal depth. Notably, the penultimate text encoder layer dominates primarily during early timesteps but fades as generation progresses, as shown in Figure 2 (a-b). This suggests it serves as a high-level semantic anchor for initial structural layout, while lacking the fine-grained features required for later-stage texture refinement.

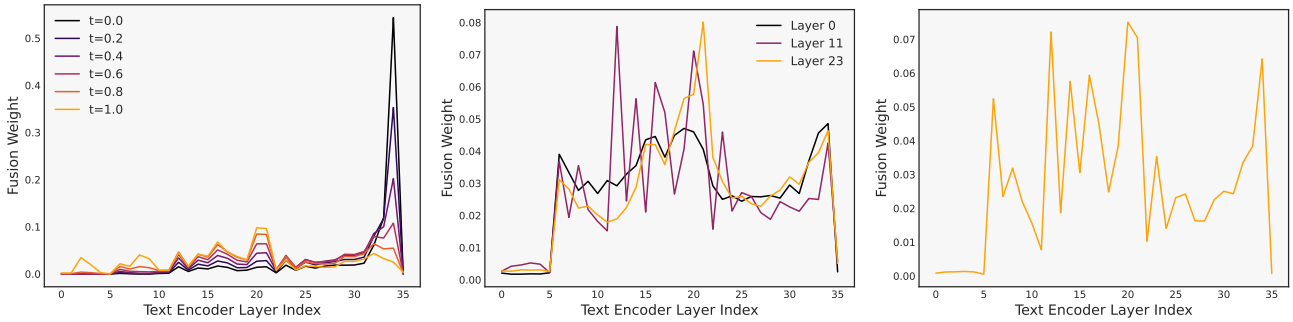
Neighbor Inhibition and Information Selection. For the intermediate layers of text encoder, as visualized in Figure 2, we observe an interesting pattern of weight fluctuation across the layer index. This is characterized by local peaks where specific layers receive high weights while their immediate neighbors are noticeably suppressed. We attribute this to an implicit redundancy reduction strategy learned by the model. Since LLM hidden states typically possess high inter-layer similarity due to residual connections, the gating mechanism tends to select the most representative layer within a local neighborhood to avoid redundant information infusion. This selective inhibition is particularly pronounced in the Joint strategy (Figure 2a), which exhibits much sharper peaks and higher local contrast compared to the smoother distributions observed in the decoupled Time-wise and Depth-wise settings.

Spatiotemporal Dynamism. As shown in Figure 2b, the learned weights shift significantly across timesteps, reflecting the evolving semantic demands of the denoising process. Notably, a comparison between the depth-only setting (Figure 2a) and the joint setting (Figure 2c) reveals that S3 exhibits a stronger depth-dependent reallocation. This suggests that coupling time and depth enables a more nuanced orchestration of feature selection beyond what is possible with independent dimensions.

Emergent Local Smoothness. Similarity visualizations (Figure 3) confirm that these variations are highly structured. We observe clear smoothness across neighboring timesteps and adjacent DiT blocks. In the depth-only setting, where no explicit cross-block constraints are enforced, this emergent locality provides robust evidence that the learned routing is driven by coherent semantic signals rather than stochastic optimization noise.



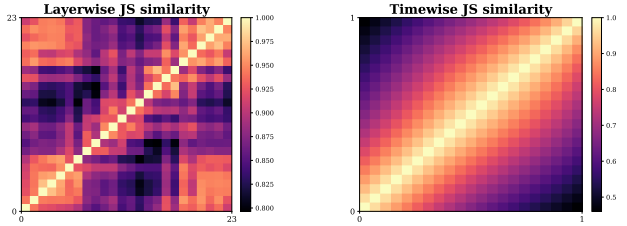
(a) Joint Fusion Weights



(b) Time-wise Fusion Weights *(c)* Depth-wise Fusion Weights *(d)* Static Fusion Weights

Figure 2. Weight distributions under different fusion-weight parameterizations. The x-axis denotes the text encoder layer index l, and the y-axis denotes the normalized fusion weight $\alpha_{t,d}$. For time-wise fusion, we sample $t \in [0, 1]$ with a step size of 0.2. For joint fusion and depth-wise fusion, we report representative DiT block indices $d \in \{0, 11, 23\}$. Additional visualizations are provided in the appendix C.1

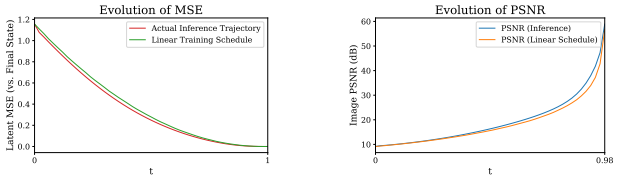
.



(a) JS similarity matrix of fusion-weight distributions across DiT blocks.

(b) JS similarity matrix of fusion-weight distributions across diffusion timesteps.

Figure 3. Visualizing the local smoothness of learned fusion weights. We compute pairwise JS similarity between normalized fusion-weight distributions along two axes: (a) across DiT blocks (depth) and (b) across diffusion timesteps (time).



(a) Latent MSE vs. timestep. *(b)* PSNR vs. timestep.

Figure 4. Trajectory analysis on GenAI-Bench: Latent MSE (left) and PSNR (right) evolution.

5.2. Trajectory Misalignment in Time-wise Fusion

Our experiments reveal that time-wise fusion leads to degraded generation quality, manifesting as blurriness and detail loss. We attribute this failure to a fundamental train–inference instability regarding the evolution of the SNR.

Empirical Evidence on GenAI-Bench. To quantify diffusion dynamics, we analyze the divergence between the actual inference trajectory and the theoretical forward process on GenAI-Bench. We generate images using a CFG scale of 6.0 and 50 steps. By constructing a reference trajectory from the final generated latent $\hat{x}_1$ via the training forward process, we evaluate the deviation of the intermediate latent x_t. As shown in Figure 4, the actual inference trajectory consistently outpaces the training schedule, exhibiting lower MSE and higher PSNR at identical nominal

timesteps. This effectively decouples the real SNR from the nominal timestep t.

Mechanism: Iterative vs. Static Sampling. The root cause of this drift lies in the distinct data-construction mechanisms. During training, x_t is sampled from a pre-defined static interpolation, ensuring a rigid bijection between t and SNR. Conversely, inference is an iterative process where x_t is the recursive output of preceding steps. Under CFG, which sharpens the vector field, the model restores structural information faster than the linear training assumption predicts, causing the cumulative denoising progress to run ahead of the fixed schedule.

Consequence: Semantic Lag and Joint Stability. This misalignment renders the time-wise gating network $g_\psi(t)$ ineffective. Conditioned solely on the nominal t, the gate remains oblivious to the fact that the latent x_t has already reached a "cleaner" state (semantic lag), thus rigidly injecting coarse-grained training priors that hamper high-frequency detail formation.

In contrast, the joint strategy (S3) mitigates this by exhibiting higher temporal stability. As visualized in Figure 9, S3 weights undergo much smaller variations across timesteps than the S1 weights. This inherent stability, achieved by coupling time with depth, effectively dampens the synchronization errors caused by trajectory misalignment, explaining why S3 remains robust while S1 falters.

Counterfactual Validation: Shifted Timestep. To rigorously verify this hypothesis, we conduct a counterfactual experiment: we artificially advance the timestep input to the TCFG to "catch up" with the accelerated inference trajectory. We introduce a heuristic shift function modulated by a cosine window active in $t \in (0.2, 1]$:

$$t' = t + \delta(t), \quad \delta(t) = 0.01 \cdot \left(1 - \cos\left(\pi \cdot \frac{t - 0.2}{0.8}\right)\right). \tag{11}$$

As shown in Table 3, this simple calibration yields consistent performance recovery across all metrics (e.g., +0.24 on GenEval). Crucially, this positive signal validates the mechanism of our mismatch hypothesis. However, such a rigid manual shift is evidently insufficient to perfectly rectify the complex nonlinear deviations of the inference trajectory. This limitation suggests that while the diagnosis is correct, developing a truly robust time-aware fusion strategy remains an open challenge for future exploration.

5.3. Compute Overhead

We summarize model complexity and inference overhead in Table 4. Relative to standard baselines (B1–B3), our adaptive variants (S1–S3) add negligible cost. In particular, the best depth-wise strategy (S2) introduces essentially no extra parameters, increases end-to-end latency by only $\sim$8%, indicating that gating is not a computational bottleneck.

Table 3. Effect of manual timestep recalibration on S1. The slight recovery validates the mismatch mechanism.

Method	GenEval ↑	GenAI ↑	UnifiedReward ↑
S1	63.41	76.20	2.97
S1 + Shift	**63.65**	**76.46**	**2.98**

Table 4. Overhead of fusion strategies (relative to the DiT backbone). Lower is better. Values are rounded to the nearest integer.

Method	Params (M)↓	FLOPs (T)↓	Latency (ms)↓
Baselines			
B1: Penult.	2247	454	**2339**
B2: Uniform	2247	454	2370
B3: Static	2247	454	2373
Deep-fusion baseline			
FuseDiT	**1712**	**357**	2575
Our fusion strategies			
S1: Time	2247	454	2391
S2: Depth	2247	470	2515
S3: Joint	2249	470	2523

FuseDiT attains lower FLOPs primarily by reusing LLM hidden states and adopting a lighter self-attention design. As shown in Table 1, this reduction in FLOPs is accompanied by a clear degradation in generative quality. We hypothesize that reusing LLM K/V states, may restrict conditioning expressiveness and limit the model's ability to adapt semantic cues. In contrast, our modular routing preserves semantic expressiveness and yields a more favorable quality efficiency trade-off under comparable latency.

6. Conclusion

This paper investigates how to leverage multi-layer LLM representations for text conditioning in DiT-based generative models. We propose a unified fusion formulation that supports controlled implementation and fair comparison of time-wise adaptive fusion, depth-wise adaptive fusion, and their combinations within a single framework. Experiments show that multi-layer fusion consistently outperforms single-layer conditioning, and that depth-wise adaptive fusion delivers the most robust and substantial improvements among the studied strategies. By contrast, purely time-wise fusion can hurt performance, which we attribute to a train–inference mismatch between nominal timesteps and inference-time denoising dynamics. Interestingly, the learned time-wise weights remain structured across timesteps, suggesting that effective time-adaptive conditioning may be possible when driven by trajectory-aligned signals.

7. Impact Statements

This paper aims to advance research in machine learning and generative modeling. We introduce a semantic routing mechanism for diffusion Transformers that performs lightweight and interpretable fusion of multi layer large language model hidden states, better matching the generation process across network depth and optional temporal dynamics, thereby improving text image alignment and compositional instruction following.

Potential positive impacts include improved semantic alignment between generated content and textual inputs, enabled by stronger textual conditioning. This can enhance controllability and instruction consistency in text to image generation, benefiting applications such as controllable synthesis, content editing, and human AI co creation. The proposed lightweight gated fusion also provides a more interpretable interface for analysis, which can support scientific understanding of how textual conditioning operates during generation and motivate more robust conditioning designs.

As with many advances in high fidelity text to image generation, our approach could be misused to produce misleading or deceptive imagery, potentially amplifying misinformation. Moreover, if training data or the underlying text encoder contains societal biases, stronger alignment may reproduce such biases more consistently and could exacerbate stereotyping or unfair representations. Enhanced compliance with fine grained prompts may also lower the barrier to generating harmful or infringing content.

We do not expect this work to introduce fundamentally new categories of safety risks. In deployment or release settings, existing safety and compliance practices commonly used for generative models can still be applied to mitigate potential misuse, bias, and infringement risks, such as content safety filtering, sensitive concept blocking, bias and robustness evaluation, and, when appropriate, provenance mechanisms including watermarking.

References

Ba, J. L., Kiros, J. R., and Hinton, G. E. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.

Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., and Zhu, K. Qwen3-vl technical report. arXiv preprint arXiv:2511.21631, 2025.

Barbero, F., Arroyo, A., Gu, X., Perivolaropoulos, C., Bronstein, M., Veličković, P., and Pascanu, R. Why do llms attend to the first token? arXiv preprint arXiv:2504.02732, 2025.

BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N., and Reddy, S. Llm2vec: Large language models are secretly powerful text encoders. arXiv preprint arXiv:2404.05961, 2024.

Cai, H., Cao, S., Du, R., Gao, P., Hoi, S., Huang, S., Hou, Z., Jiang, D., Jin, X., Li, L., et al. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. arXiv preprint arXiv:2511.22699, 2025.

Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., and Li, Z. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. URL https://arxiv.org/abs/2310.00426.

Chen, P., Shen, M., Ye, P., Cao, J., Tu, C., Bouganis, C.-S., Zhao, Y., and Chen, T. δ-dit: A training-free acceleration method tailored for diffusion transformers. ArXiv, abs/2406.01125, 2024. URL https://api.semanticscholar.org/CorpusID:270215326.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. Scaling rectified flow transformers for high-resolution image synthesis. In Forty-first international conference on machine learning, 2024.

Fan, S., Jiang, X., Li, X., Meng, X., Han, P., Shang, S., Sun, A., Wang, Y., and Wang, Z. Not all layers of llms are necessary during inference. arXiv preprint arXiv:2403.02181, 2024.

Gao, Y., Guo, H., Hoang, T., Huang, W., Jiang, L., Kong, F., Li, H., Li, J., Li, L., Li, X., et al. Seedance 1.0: Exploring the boundaries of video generation models. arXiv preprint arXiv:2506.09113, 2025.

Ghosh, D., Hajishirzi, H., and Schmidt, L. Geneval: An object-focused framework for evaluating text-to-image alignment. Advances in Neural Information Processing Systems, 36:52132–52152, 2023.

Gong, X., Chen, W., Chen, T., and Wang, Z. Sandwich batch normalization: A drop-in replacement for feature distribution heterogeneity. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2494–2504, 2022.

Gurnee, W. and Tegmark, M. Language models represent space and time. arXiv preprint arXiv:2310.02207, 2023.

Henry, A., Dachapally, P. R., Pawar, S. S., and Chen, Y. Query-key normalization for transformers. In Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 4246–4253, 2020.

Heo, B., Park, S., Han, D., and Yun, S. Rotary position embedding for vision transformer. In European Conference on Computer Vision, pp. 289–305. Springer, 2024.

Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., and Cohen-Or, D. Prompt-to-prompt image editing with cross attention control. arXiv preprint arXiv:2208.01626, 2022.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598, 2022.

Jin, M., Yu, Q., Huang, J., Zeng, Q., Wang, Z., Hua, W., Zhao, H., Mei, K., Meng, Y., Ding, K., et al. Exploring concept depth: How large language models acquire knowledge and concept at different layers? In Proceedings of the 31st International Conference on Computational Linguistics, pp. 558–573, 2025.

Kim, J., Lee, B., Park, C., Oh, Y., Kim, B., Yoo, T., Shin, S., Han, D., Shin, J., and Yoo, K. M. Peri-ln: Revisiting normalization layer in the transformer architecture. arXiv preprint arXiv:2502.02732, 2025.

Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al. Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603, 2024.

Li, B., Lin, Z., Pathak, D., Li, J., Fei, Y., Wu, K., Ling, T., Xia, X., Zhang, P., Neubig, G., et al. Genai-bench: Evaluating and improving compositional text-to-visual generation. arXiv preprint arXiv:2406.13743, 2024.

Li, B., Xue, X., Yang, S., Shi, Y., Chen, X., Guan, Y., Zhang, Y., and Zhang, W. The unseen bias: How norm discrepancy in pre-norm mllms leads to visual information loss. arXiv preprint arXiv:2512.08374, 2025a.

Li, B., Yang, S., Guan, Y., An, R., Chen, X., Shi, Y., Wan, P., Zhang, W., et al. Gran-ted: Generating robust, aligned, and nuanced text embedding for diffusion models. arXiv preprint arXiv:2512.15560, 2025b.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. arXiv preprint arXiv:2210.02747, 2022.

Liu, B., Akhgari, E., Visheratin, A., Kamko, A., Xu, L., Shrirao, S., Lambert, C., Souza, J., Doshi, S., and Li, D. Playground v3: Improving text-to-image alignment with deep-fusion large language models. arXiv preprint arXiv:2409.10695, 2024a.

Liu, E., Ning, X., Lin, Z., Yang, H., and Wang, Y. Omsdpm: Optimizing the model schedule for diffusion probabilistic models. In International Conference on Machine Learning, pp. 21915–21936. PMLR, 2023.

Liu, Z., Kong, C., Liu, Y., and Sun, M. Fantastic semantics and where to find them: Investigating which layers of generative llms reflect lexical semantics. arXiv preprint arXiv:2403.01509, 2024b.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

Ma, B., Zong, Z., Song, G., Li, H., and Liu, Y. Exploring the role of large language models in prompt encoding for diffusion models. arXiv preprint arXiv:2406.11831, 2024.

Ma, G., Huang, H., Yan, K., Chen, L., Duan, N., Yin, S., Wan, C., Ming, R., Song, X., Chen, X., et al. Step-video-t2v technical report: The practice, challenges, and future of video foundation model. arXiv preprint arXiv:2502.10248, 2025.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 4195–4205, 2023.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pp. 8748–8763. PmLR, 2021.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research, 21 (140):1–67, 2020.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pp. 234–241. Springer, 2015.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems, 35:36479–36494, 2022.

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., and Komatsuzaki, A. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv preprint arXiv:2111.02114, 2021.

Seedream, T., Chen, Y., Gao, Y., Gong, L., Guo, M., Guo, Q., Guo, Z., Hou, X., Huang, W., Huang, Y., et al. Seedream 4.0: Toward next-generation multimodal image generation. arXiv preprint arXiv:2509.20427, 2025.

Skean, O., Arefin, M. R., Zhao, D., Patel, N., Naghiyev, J., LeCun, Y., and Shwartz-Ziv, R. Layer by layer: Uncovering hidden representations in language models. arXiv preprint arXiv:2502.02013, 2025.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. Neurocomputing, 568:127063, 2024.

Tang, B., Zheng, B., Paul, S., and Xie, S. Exploring the deep fusion of large language models and diffusion transformers for text-to-image synthesis. In Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 28586–28595, 2025.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023.

von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Nair, D., Paul, S., Berman, W., Xu, Y., Liu, S., and Wolf, T. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022.

Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.-W., Chen, D., Yu, F., Zhao, H., Yang, J., et al. Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314, 2025.

Wang, A. Z., Ge, S., Karras, T., Liu, M.-Y., and Balaji, Y. A comprehensive study of decoder-only llms for text-to-image generation. In Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 28575–28585, 2025a.

Wang, B. and Vastola, J. J. Diffusion models generate images like painters: an analytical theory of outline first, details later. arXiv preprint arXiv:2303.02490, 2023.

Wang, Y., Zang, Y., Li, H., Jin, C., and Wang, J. Unified reward model for multimodal understanding and generation. arXiv preprint arXiv:2503.05236, 2025b.

Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.-m., Bai, S., Xu, X., Chen, Y., et al. Qwen-image technical report. arXiv preprint arXiv:2508.02324, 2025.

A. TCFG Implementation Details

The Time-Conditioned Fusion Gate (TCFG) serves as the core module for adaptively aggregating multi-layer LLM features based on the diffusion timestep. Our implementation follows a lightweight Multi-Layer Perceptron (MLP) design with specific initialization strategies to ensure training stability.

Sinusoidal Timestep Embedding. The continuous timestep t is first mapped to a high-dimensional feature vector using a sinusoidal embedding, similar to the positional encoding in Transformers. For a time embedding dimension D_t, the embedding $\phi(t) \in \mathbb{R}^{D_t}$ is computed as:

$$\phi(t) = [\dots, \cos(t \cdot \omega_i), \sin(t \cdot \omega_i), \dots], \quad \text{where } \omega_i = \frac{1}{10000^{2i/D_t}}. \tag{12}$$

In our experiments, we set $D_t = 128$.

Gating Network Architecture. The embedding $\phi(t)$ is processed by a two-layer MLP to generate the fusion logits $z_t \in \mathbb{R}^L$, where L is the number of LLM layers. The network structure consists of:

1. **Input Projection:** A linear layer mapping from D_t to a hidden dimension of $4 \times D_t$.
2. **Activation:** A Sigmoid Linear Unit (SiLU) activation function.
3. **Output Projection:** A linear layer mapping from $4 \times D_t$ to L.

Zero-Initialization Strategy. To facilitate a smooth starting point for optimization, we employ a zero-initialization strategy for the final output projection layer. Specifically, both the weight matrix and the bias vector of the second linear layer are initialized to zero. Consequently, at the beginning of training, the output logits z_t are zero vectors, which results in a uniform probability distribution after the Softmax operation (i.e., $\alpha_t^{(l)} = 1/L$ for all l). This ensures that the model initially utilizes an average of all layers before learning specific routing preferences.

Feature Normalization and Aggregation. Given a set of hidden states $\{H^{(l)}\}_{l=1}^{L}$ from the text encoder, we strictly apply Layer Normalization (LayerNorm) *before* fusion to handle scale discrepancies across layers. The final aggregated feature H_{cond} is computed as:

$$H_{\text{cond}} = \sum_{l=1}^{L} \text{Softmax}(z_t)^{(l)} \cdot \text{LayerNorm}(H^{(l)}). \tag{13}$$

This formulation ensures that the aggregated feature remains within the convex hull of the normalized representations, maintaining numerical stability throughout the training process.

B. Image Examples

Figure 5. Qualitative comparisons across strategies under multiple prompts. Columns: B1/B2/B3, FuseDiT baseline, and three fusion strategies (S1/S2/S3). All images use identical sampling settings for fair comparison.

Prompt: “Among a group of pastel-colored balloons, one stands out in vibrant red.”

Penult Uniform Static FuseDiT Time Depth Joint

Prompt: “A vase with five purple roses on a kitchen table.”

Penult Uniform Static FuseDiT Time Depth Joint

Prompt: “A pilot with aviator sunglasses.”

Penult Uniform Static FuseDiT Time Depth Joint

Prompt: “Five colorful balloons floating against a clear blue sky.”

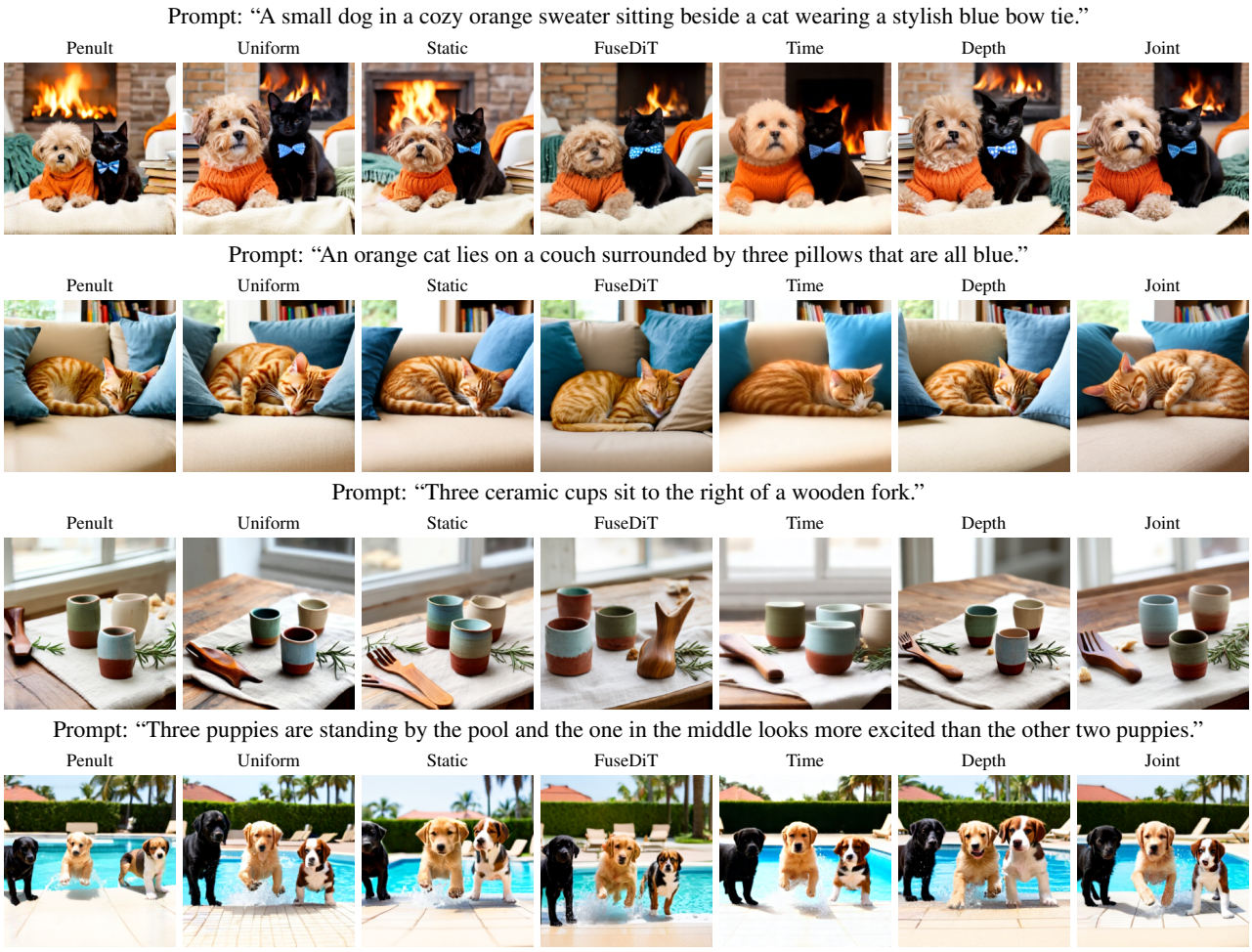
Penult Uniform Static FuseDiT Time Depth Joint

Prompt: “A large pizza with pepperoni on the left half and mushrooms on the right half.”

Penult Uniform Static FuseDiT Time Depth Joint

Prompt: “A large teddy bear wearing a bow tie next to a small teddy bear wearing a party hat.”

Penult Uniform Static FuseDiT Time Depth Joint



C. Additional Visualizations of Fusion Weight

In this section, we provide more detailed visualizations of the fusion weights (App. C.1) and an analysis of their variation trends (App. C.2).

C.1. Detailed Visualization of Fusion Weights

Figures 6 and 7 visualize the fusion-weight distributions of the text encoder across different layers and diffusion timesteps under the depth-wise and joint strategies, respectively.

C.2. Weight Evolution and Trend Analysis

To quantify how the fusion weights over text-encoder layers evolve under different diffusion timesteps and depth settings, we treat each fusion-weight vector as a discrete probability distribution supported on uniformly spaced points over $[0, 1]$. We then compute its mean and variance as summary statistics, and plot their trends in Figure 8.

Given a fusion-weight vector $\mathbf{w} = \{w_i\}_{i=0}^{L-1}$ of length L, we first normalize it as

$$p_i = \frac{w_i}{\sum_{j=0}^{L-1} w_j + \epsilon}, \tag{14}$$

where ϵ is a small positive constant for numerical stability.

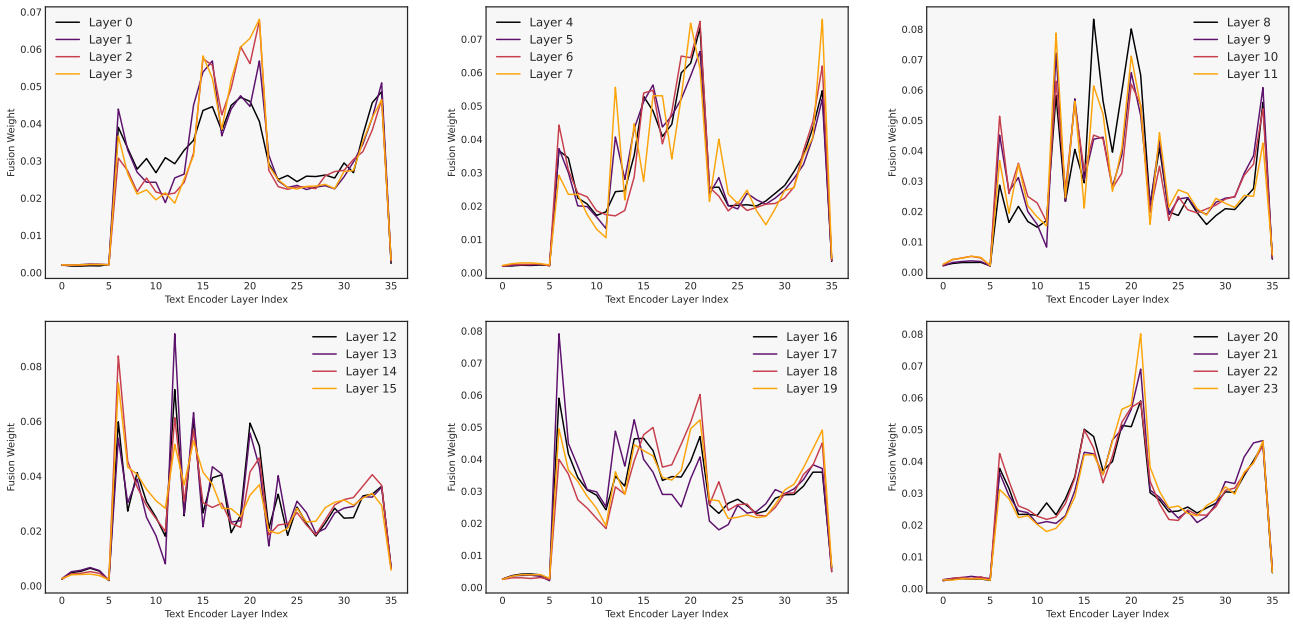


Figure 6. Depth-wise fusion weights across layers.

We define the uniformly spaced support locations by

$$l_i = \begin{cases} \frac{i}{L-1}, & L > 1, \\ 0, & L = 1, \end{cases} \qquad i \in \{0, 1, \ldots, L-1\}. \tag{15}$$

The mean (semantic center) is defined as

$$\hat{\mu} = \sum_{i=0}^{L-1} p_i \, l_i, \tag{16}$$

and the variance (semantic dispersion) is defined as

$$\hat{\sigma}^2 = \sum_{i=0}^{L-1} p_i \, (l_i - \hat{\mu})^2. \tag{17}$$

Intuitively, this interpretation views fusion weights as a distribution along a "semantic hierarchy" axis: $\hat{\mu}$ indicates where the mass concentrates on the axis (favoring earlier vs. later layers), while $\hat{\sigma}^2$ measures how dispersed the weights are (larger values imply a less concentrated, more spread-out allocation).

To better understand how fusion weights are allocated across diffusion timesteps and encoder depth, we visualize the statistics in Figure 8. Under the joint strategy, each sample yields a 2D weight map $w(t, l)$ over timestep t and text-encoder layer l. To obtain interpretable 1D trends, we form marginal distributions by normalizing along the complementary axis and compute the semantic center $\hat{\mu}$ and dispersion $\hat{\sigma}^2$ on each marginal.

The results suggest that joint fusion maintains a largely stable semantic center over time: $\hat{\mu}$ stays at a moderately deep-layer regime and only shifts slightly toward shallower layers at later timesteps, while the consistently wide band indicates that the weights do not collapse onto a few layers but preserve broad multi-layer coverage throughout sampling. In contrast, time-wise fusion exhibits pronounced monotonic drift: $\hat{\mu}$ decreases substantially as timesteps progress, indicating a systematic shift from deeper-layer emphasis toward shallower-layer emphasis, which reflects a more time-dependent and less stable allocation of semantic levels. Depth-wise fusion yields smoother curves with smaller fluctuations, suggesting a more consistent and controllable allocation pattern across layers. Overall, joint fusion preserves broad layer coverage while substantially mitigating the deep-to-shallow drift observed in purely time-wise fusion, providing a statistical explanation for its more stable generation behavior.

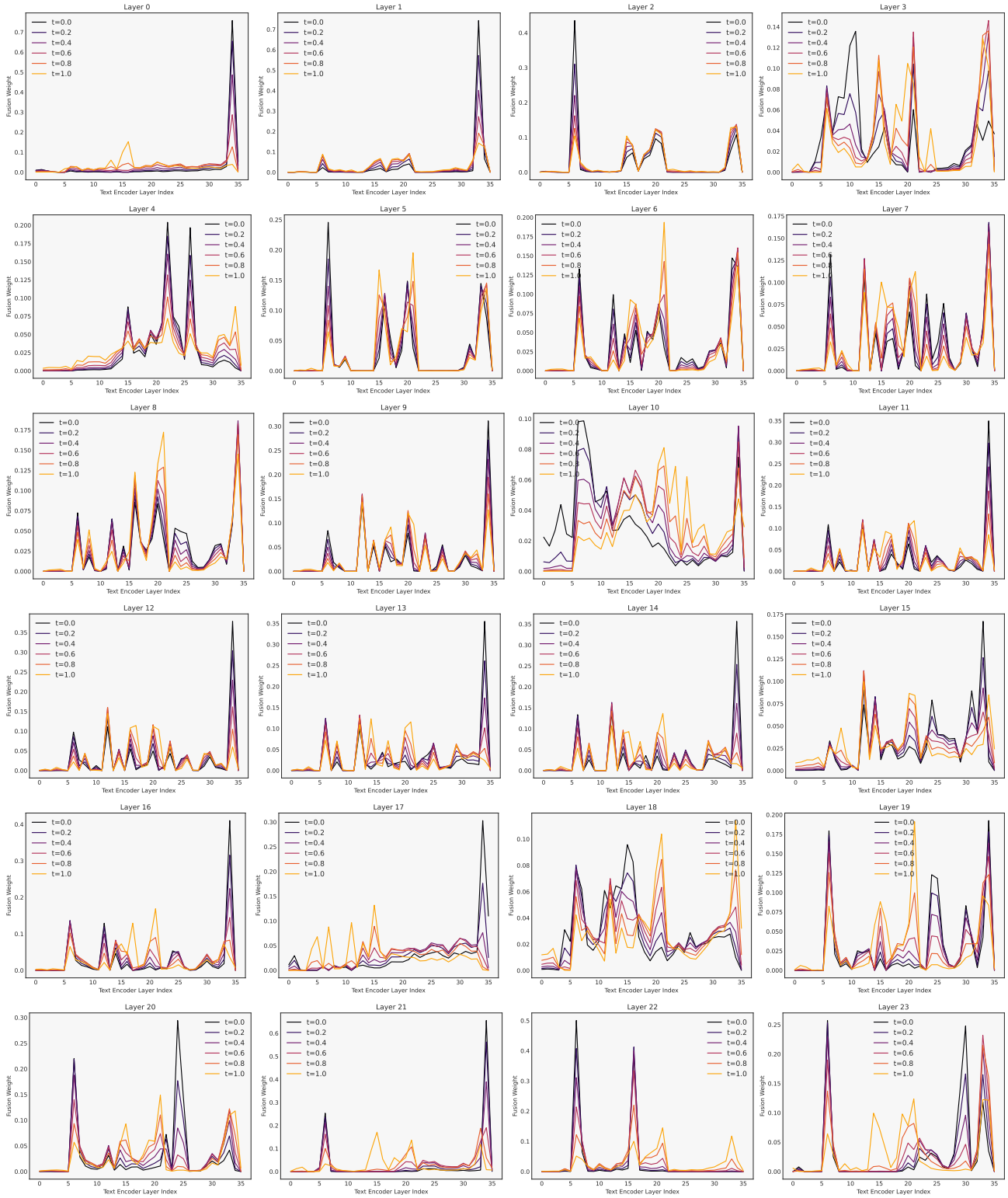
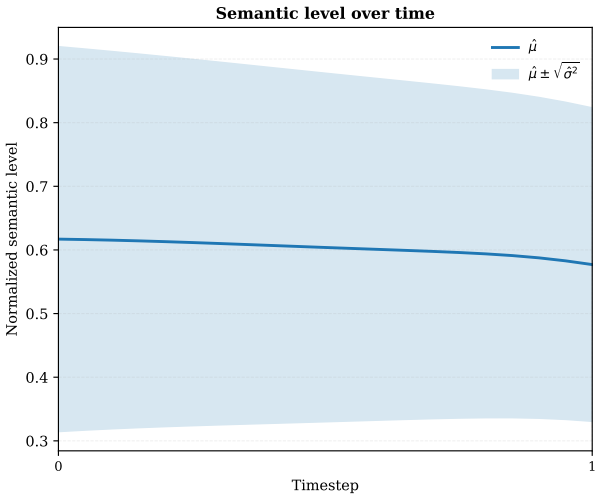
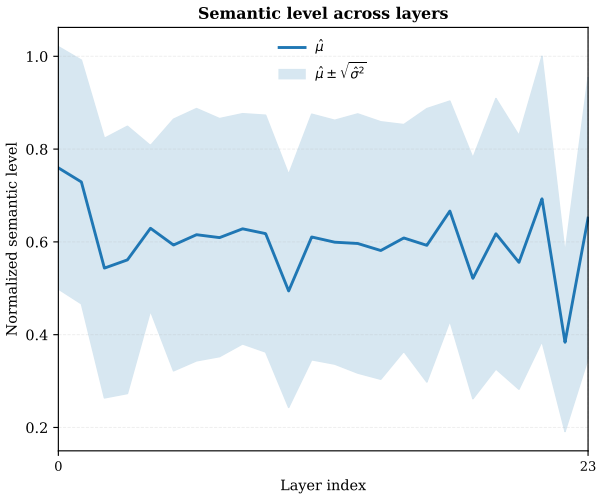


Figure 7. Joint fusion weights (depth × time) across layers.

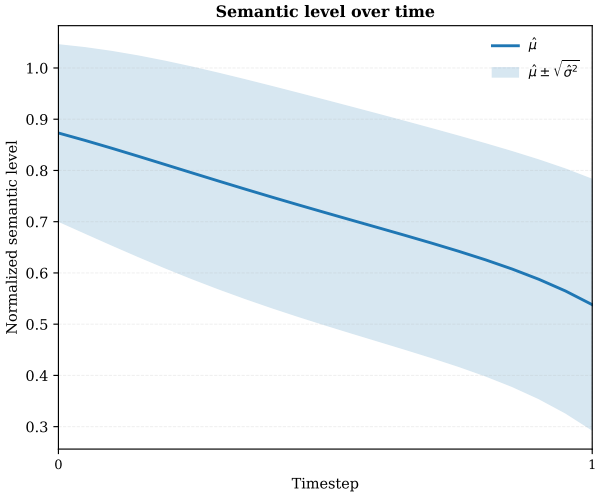
To compare how fast the fusion weights evolve over time under the time-wise and joint strategies, and to further complement the analysis in Section 5.2, we quantify the discrepancy between weight distributions at consecutive timesteps. Concretely,



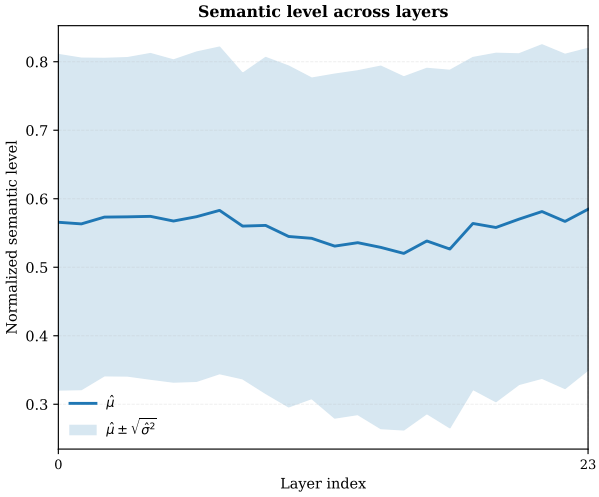
(a) Joint fusion: mean and variance computed on the time-marginal weight distribution.



(b) Joint fusion: mean and variance computed on the depth-marginal weight distribution.



(c) Time-wise fusion: mean and variance trends of the fusion weights across diffusion timesteps.

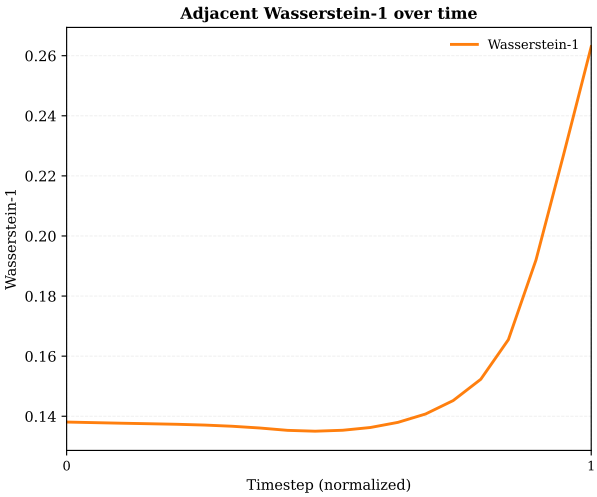


(d) Depth-wise fusion: mean and variance trends of the fusion weights across text-encoder layers.

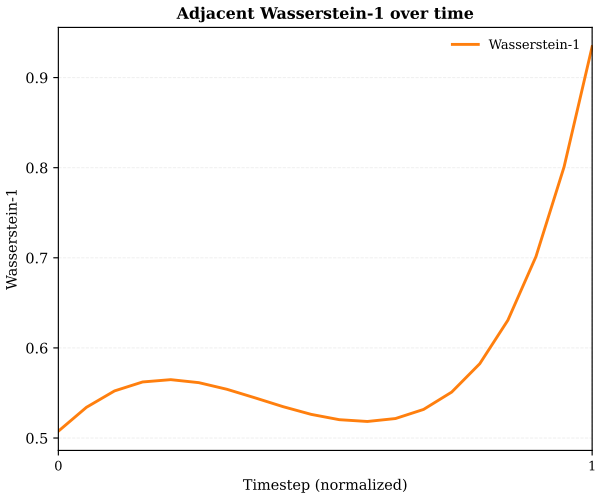
Figure 8. Weight-trend statistics under different fusion strategies. Top row: joint fusion with marginalization over timesteps (left) and depths (right). Bottom row: the corresponding statistics under time-wise (left) and depth-wise (right) fusion.

for each sampled timestep t, we treat its corresponding weight vector as a distribution and compute the 1-Wasserstein distance to the distribution at the previous timestep t-1. For the joint strategy, since the weights are defined over both timesteps and layers, we first marginalize over the layer dimension to obtain a 1D marginal distribution for each timestep, and then compute the 1-Wasserstein distance between consecutive timesteps. We sample 21 timesteps in total, and report the results in Figure 9.

As shown in Figure 9, the joint strategy yields consistently smaller 1-Wasserstein distances between consecutive timesteps than the time-wise strategy, indicating a substantially smoother temporal evolution of the weight allocation. In light of the train–inference timestep-trajectory mismatch discussed in Section 5.2, a simple interpretation is as follows: under a time-wise-only scheme, rapid weight changes across adjacent timesteps can amplify the mismatch-induced deviation, making the semantic conditioning less stable during inference and thus more prone to blur; joint fusion markedly slows down such temporal drift, leading to more coherent conditioning along the inference trajectory and mitigating the resulting



(a) Time-wise fusion: 1-Wasserstein distance between consecutive timesteps.



(b) Joint fusion (time-marginal): 1-Wasserstein distance between consecutive timesteps.

Figure 9. Temporal variation speed of fusion weights, measured by the 1-Wasserstein distance between weights at timesteps spaced by 0.05. For joint fusion, we compute the distance on the time-marginal distributions.

degradation.