# DiTCarbon: Predictive Carbon Footprint Estimation for Diffusion Transformer Inference

**Anonymous Authors**[1]

## Abstract

Diffusion Transformers (DiTs) are moving into consumer image and video products, where high per-request inference cost makes carbon emissions a major deployment concern. Existing works either measure carbon at runtime or predict lifecycle carbon for autoregressive transformers. However, none estimates full-lifecycle carbon directly from a DiT architecture and generation configuration. We propose DiTCarbon, the first framework that combines parameter, FLOP, hardware efficiency, and lifecycle accounting models to predict carbon for class-conditional, text-conditional, dual-stream, and video DiT variants. Across 83 V100 configurations, hardware efficiency varies by 6.4×. A pooled saturation fit predicts hardware efficiency within 7.8% MAPE in the production regime (resolution $\geq 512^2$). Predicted operational carbon agrees with measured ground truth (from wall-clock runtime and NVML GPU power) within 10.7% MAPE in the production regime. For a representative deployment, regional carbon varies $45\times$, making deployment region the largest lever for reducing per-output emissions as DiT services move into consumer products.

## 1. Introduction

Diffusion Transformers (DiTs) are moving into consumer image and video products, and their per-request inference energy is already far above text generation. WAN2.1 14B (Wan-AI, 2025) consumes over 415 Wh per short video on H100 hardware (Delavande et al., 2025), thousands of times the energy of a text generation query (Luccioni et al., 2024). At scale, carbon accounting becomes part of deployment planning. A useful estimate must combine workload energy with datacenter power usage effectiveness (PUE), regional grid carbon intensity, and embodied hardware emissions.

No existing framework predicts lifecycle carbon for DiT inference across the broader DiT family from architecture and generation specifications. Tracking libraries (Schmidt et al., 2021) estimate emissions only after the workload runs, missing pre-deployment planning and embodied emissions. LLMCarbon (Faiz et al., 2024) predicts lifecycle carbon before execution but targets autoregressive transformers rather than DiT inference. Recent video diffusion work (Li et al., 2024; Delavande et al., 2025) measures specific models on specific hardware and does not generalize across unmeasured DiT variants. DiTCarbon addresses all three gaps: pre-run estimation, lifecycle accounting, and broader DiT coverage.

We propose DiTCarbon, the first framework that takes a DiT architecture specification and a generation configuration as input and returns lifecycle carbon estimates without running the model. The pipeline consists of four sequential stages (Figure 1): parameter estimation produces $P$, FLOP computation combines $P$ with the generation settings to produce $F_{\text{total}}$, hardware efficiency maps $F_{\text{total}}$ to runtime and energy through $\mu$, and lifecycle accounting adds datacenter and hardware factors to produce operational and embodied carbon.

Unlike prior work, DiTCarbon combines predictive estimation, lifecycle accounting, and broad architectural coverage in one framework. Pre-run estimates let operators compare model, resolution, batching, and region choices before spending GPU time. Lifecycle accounting captures both operational emissions and amortized hardware manufacturing emissions. Unified parameter and FLOP models cover class-conditional, text-conditional, dual-stream, and video variants without a separate estimator for each backbone. After hardware efficiency calibration, CPU prediction supports inexpensive configuration sweeps.

Evaluation on a Tesla V100 gives three headline results. End-to-end carbon predictions show a $45\times$ regional swing for a representative PixArt-$\alpha$ $1024^2$ deployment, reaching 559 kg $CO_2$eq per day at one million requests under the US-average grid. On the V100 corpus, predicted opera-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

tional carbon agrees with carbon computed from measured wall-clock runtime and NVML GPU power within 10.7% MAPE in the production regime (15.4% across the full 83-configuration sweep). Hardware efficiency $\mu$ varies $6.4\times$ across 83 configurations, ruling out a single transferred efficiency constant. In the production regime (resolution $\geq 512^2$), a pooled saturation fit predicts $\mu$ within 7.8% MAPE and bounds leave-one-architecture-out error at 12%. The analytical FLOP model agrees with manual instrumentation within 0.5% on the denoising path, and the parameter model matches published counts within 3% across five models.

We make four contributions: (1) an architecture-aware parameter model whose three parameter shape flags ($s$, $c$, $a$) cover four DiT variant families; (2) an operator-level FLOP model extending Delavande et al. (2025) with the same parameter shape flags plus a joint attention flag ($j$) specific to FLOP counting; (3) a V100 hardware-efficiency characterization that separates production from sub-production configurations, enabling a single per-hardware calibration; and (4) a lifecycle carbon pipeline producing operational and embodied estimates before execution.

## 2. Related Work

The environmental cost of large machine learning has been studied through training measurement, lifecycle accounting, and deployment profiling. Strubell et al. (2019) measured neural architecture search for a transformer and reported emissions equivalent to five passenger cars over their lifetimes. Patterson et al. (2021) decomposed the problem into datacenter efficiency, accelerator selection, and grid carbon intensity. Location alone shifts emissions by 5–10×. Specialized accelerators are 2–5× more efficient than commodity hardware. Luccioni et al. (2023) measured the BLOOM lifecycle at 50.5 tonnes $CO_2$eq including embodied hardware, and characterized inference through an API endpoint. Luccioni et al. (2024) followed with *Power Hungry Processing* and showed that generative tasks consume orders of magnitude more energy than discriminative ones.

Existing ML carbon tools split into post-hoc characterization and predictive estimation. CodeCarbon (Schmidt et al., 2021) instruments power draw at runtime. ML CO2 Impact (Lacoste et al., 2019) estimates from user-supplied hardware and runtime details. LLMCarbon (Faiz et al., 2024) predicts emissions from architecture specifications. Faiz et al. decompose the estimate into parameter counting, FLOP estimation, hardware efficiency modeling, and lifecycle carbon accounting with operational and embodied components. They calibrate LLMCarbon for autoregressive transformer inference. Diffusion Transformers do not match this profile. Generation repeats a full denoising network for 20–50 steps, and attention scales quadratically with spatial resolution rather than linearly with sequence length. Architectural variants compound the mismatch: class-conditional DiT (Peebles & Xie, 2023), text-conditional PixArt-$\alpha$ (Chen et al., 2024b), and dual-stream MMDiT (Esser et al., 2024) yield distinct compute profiles. Across 83 DiT configurations on V100 we measure $\mu$ from 0.070 to 0.448, a $6.4\times$ spread. DiT inference on tensor core hardware cannot be summarized by a single efficiency constant transferred across architectures.

Two recent papers focus on DiT energy. Carbon in Motion (Li et al., 2024) characterizes Open-Sora and isolates denoising steps, resolution, and video duration as the dominant drivers. Delavande et al. (2025) built an operator-level FLOP decomposition for WAN2.1, validated it on H100, and benchmarked six text-to-video models. We build on this decomposition with ($s$, $c$, $a$) parameter shape flags that extend coverage to the broader DiT family. Both studies measure rather than predict. Neither models embodied carbon. DiTCarbon integrates these approaches. It adapts LLMCarbon's lifecycle accounting, extends Delavande et al.'s operator-level FLOP decomposition using unified architectural flags spanning the broader DiT family (class-conditional, text-conditional, dual-stream, and video), and predicts carbon before execution.

## 3. Methodology

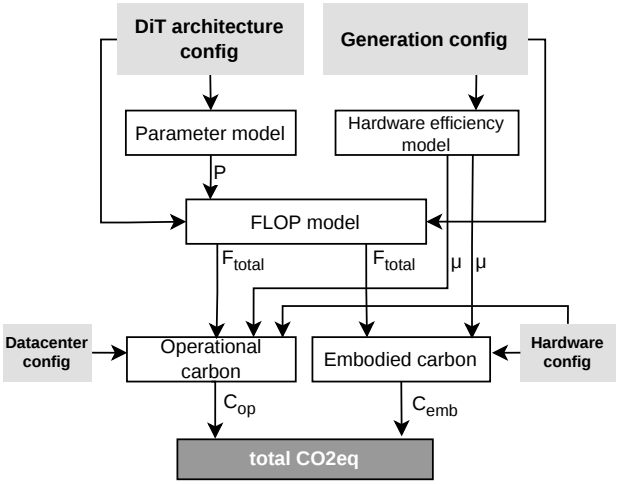


*Figure 1.* DiTCarbon overview. The framework predicts lifecycle carbon for diffusion transformer inference from architecture, generation, hardware, and datacenter configurations using four sequential model stages, producing total $CO_2$eq as the final output.

DiTCarbon predicts lifecycle carbon from four components computed in sequence (Figure 1). The framework takes four input configurations: the model architecture, generation request, hardware platform, and datacenter environment. These inputs drive four sequential stages: architecture parameters (Section 3.1), inference FLOPs (Section 3.2),

hardware efficiency and runtime (Section 3.3), and operational plus embodied carbon (Section 3.4). Stages pass $P$, $F_{\text{total}}$, and $\mu$ to carbon accounting, which outputs $C_{\text{op}}$ and $C_{\text{emb}}$. Once hardware efficiency is calibrated for a device, the pipeline runs on CPU and outputs total $CO_2$eq without executing the target DiT workload.

### 3.1. Architecture-Aware Parameter Model

DiT variants share a common transformer block: self-attention, a feed-forward MLP, and adaptive layer normalization for conditioning. Three architectural choices drive the parameter count differences across the family. We capture this parameter shape variation with three binary architectural switches ($s$, $c$, $a$) and two continuous auxiliary parameters ($n_{\text{ada}}$, $t$) that generalize the adaLN term, yielding the leading-order approximation:

$$P \approx \Big( \big(2f + 4 + 6a \cdot n_{\text{ada}} \cdot \tfrac{t}{d}\big)(1+s) + 4c \Big) \cdot N \cdot d^2 \quad (1)$$

where $N$ is the number of transformer blocks; $d$ is the hidden dimension, equal to the product of attention head count and per-head dimension; and $f$ is the MLP expansion ratio, typically 4. The block contributions are $4d^2$ for the self-attention $Q$, $K$, $V$, and output projections and $2fd^2$ for the MLP up and down projections.

$s \in \{0, 1\}$ marks dual-stream processing. MMDiT (Esser et al., 2024) runs image and text tokens through duplicated attention and MLP weights, doubling the per-block cost via the $(1 + s)$ factor.

$c \in \{0, 1\}$ indicates an explicit cross-attention sublayer with its own $Q$, $K$, $V$, $O$ projections ($4d^2$). PixArt-$\alpha$ (Chen et al., 2024b) uses this design. MMDiT does not. Text conditioning in MMDiT enters through joint attention, so $c = 0$ for SD3 even though it is text-conditioned.

$a \in \{0, 1\}$ encodes per-block adaLN-Zero. When $a = 1$, $n_{\text{ada}}$ modulation modules per block project from a conditioning width $t$ to $6t$, producing shift, scale, and gate vectors for attention and MLP. The default ($n_{\text{ada}} = 1$, $t = d$) contributes $6d^2$ and applies to DiT (Peebles & Xie, 2023) and SD3. CogVideoX-2B (Yang et al., 2025) inherits the DiT backbone but uses expert adaptive LayerNorm with separate modulation projections for text and visual tokens ($n_{\text{ada}} = 2$) over a smaller conditioning width ($t = 512$ vs. $d = 1920$, per the released model configuration), giving an effective adaLN cost of $6 \cdot 2 \cdot (512/1920) \cdot d^2 \approx 3.2d^2$ per block. PixArt-$\alpha$ uses adaLN-Single, sharing one global projection across blocks. Its parameters scale as $\mathcal{O}(d^2)$ globally rather than $\mathcal{O}(N \cdot d^2)$, so $a = 0$.

The $4c$ term assumes text embeddings are projected to dimension $d$. The conditioning width $t$ in the adaLN term is the time embedding dimension, not the text encoder embedding width. The formula omits the patchify convolution, timestep embedding MLP, caption embeddings, final output projection, biases, and layer norms. Each is subleading in $N$ and contributes $\mathcal{O}(d^2)$ or less globally. We keep the repeated transformer block terms and omit non-repeated lower-order terms.

Table 1 lists the flag assignments and leading coefficients for the five reference DiTs used to validate the parameter model. Section 4.1 verifies the approximation against measured parameter counts.

*Table 1.* Architecture configuration flags ($s$, $c$, $a$) and coefficient values for the five parameter reference DiT models. CogVideoX-2B is a video generation model, and the remaining four are image generation models. CogVideoX-2B uses $n_{\text{ada}} = 2$, $t = 512$, and the other rows use the defaults $n_{\text{ada}} = 1$, $t = d$.

| Model | $s$ | $c$ | $a$ | Coeff. |
|---|---|---|---|---|
| DiT-XL/2 | 0 | 0 | 1 | 18 |
| PixArt-$\alpha$ | 0 | 1 | 0 | 16 |
| SD3 Medium | 1 | 0 | 1 | 36 |
| SD3.5 Large | 1 | 0 | 1 | 36 |
| CogVideoX-2B | 0 | 0 | 1 | 15.2 |

### 3.2. Operator-Level FLOP Model

Inference FLOPs decompose into a denoising loop and two one-shot passes for the text encoder and VAE decoder:

$$F_{\text{total}} = B\big(F_{\text{text}} + F_{\text{VAE}}\big) + B \cdot g \cdot S \cdot N \cdot F_{\text{block}} \quad (2)$$

where $B$ is the batch size, $g$ is the number of classifier-free guidance passes (typically 2), $S$ is the number of denoising steps, and $N$ is the number of transformer blocks. $F_{\text{text}}$ and $F_{\text{VAE}}$ are per-generation costs. They scale linearly with $B$ and do not depend on $g$, $S$, or $N$.

$F_{\text{block}}$ breaks down by operator (Table 2). We build on the operator-level derivations of Delavande et al. (2025) for self-attention, cross-attention, and MLP, and extend them via the ($s$, $c$, $a$) parameter shape flags from Section 3.1. Because those flags describe weight structure, CogVideoX requires an additional flag $j$, specific to FLOP counting, for joint attention with single projections.

For images, $\ell = \frac{H}{v_s \cdot p} \cdot \frac{W}{v_s \cdot p}$, where $v_s$ is the VAE spatial downsample factor and $p$ is the patch size. For video, $\ell$ includes an additional temporal factor from the frame count.

For dual-stream architectures ($s = 1$), self-attention operates on the concatenated sequence of length $(\ell + m)$, and the MLP weights are duplicated across streams. MMDiT does not have a separate cross-attention sublayer, so the cross-attention row is absent when $s = 1$.

CogVideoX-2B is a hybrid case. Text and video tokens are concatenated into a sequence of length $(\ell + m)$, but self-attention and MLP use a single set of projections. The flag

*Table 2.* Per-block FLOP counts under the convention that one fused multiply-add equals two FLOPs. $\ell$ is the latent token count, $m$ is the text token count, $d$ is the hidden dimension, and $f$ is the MLP expansion ratio. Bias additions, layer normalization, activation functions, and softmax are subleading and omitted.

| Operator | Single-stream ($s = 0$) | Dual-stream ($s = 1$) |
|---|---|---|
| Self-attention | $8\ell d^2 + 4\ell^2 d$ | $8(\ell+m)d^2 + 4(\ell+m)^2 d$ |
| Cross-attention ($c = 1$) | $4\ell d^2 + 4md^2 + 4\ell md$ | — |
| MLP | $4f\ell d^2$ | $4f(\ell+m)d^2$ |
| adaLN-Zero ($a = 1$) | $12d^2$ | $24d^2$ |

$j$ routes both operators through $(\ell + m)$ without duplicating weights, differing from $s = 1$ (duplicated projections) and $s = 0$ (attention on $\ell$ alone). Because $j$ changes token flow rather than weight shape, it is omitted from Equation (1). CogVideoX's parameter count follows the $s = 0$ pattern, with adaLN cost handled by $n_{\text{ada}}$ and $t$, while its FLOP count uses the joint $(\ell + m)$ sequence.

The one-shot terms use the standard $2 \times$ params $\times$ tokens convention from scaling-law work (Kaplan et al., 2020; Hoffmann et al., 2022). $F_{\text{text}} = 2 \cdot P_{\text{enc}} \cdot m$ where $P_{\text{enc}}$ is the text encoder parameter count (4.7 B for T5-XXL, 0 for class-conditional DiT). $F_{\text{VAE}} = 2 \cdot P_{\text{VAE}} \cdot \ell_{\text{latent}}$ where $P_{\text{VAE}} \approx 50\,\text{M}$ and $\ell_{\text{latent}}$ is the latent pixel count. Validation of these analytical counts against manual operator instrumentation is reported in Section 4.2.

### 3.3. Hardware Efficiency

We define hardware efficiency $\mu$ as the fraction of peak FP16 throughput that the workload actually delivers. LLMCarbon (Faiz et al., 2024) provides lifecycle accounting but not hardware efficiency calibrations for single-GPU DiT inference. To estimate $\mu$ without direct measurement at every batch size $B$ and latent token count $\ell$, we fit a saturation model with three parameters $(\mu_{\text{sat}}, \ell^*, B_0)$:

$$\mu(B, \ell) = \mu_{\text{sat}} \cdot \frac{B}{B + B_0} \cdot \frac{\ell}{\ell + \ell^*/B}. \tag{3}$$

The form encodes two saturation effects. The factor $B/(B + B_0)$ captures batch-wise saturation as launch overhead amortizes, while $\ell/(\ell + \ell^*/B)$ captures sequence-length saturation of the tensor cores. The effective threshold $\ell^*/B$ decreases as batch grows because larger batches produce larger matrix multiplications. As $B$ and $\ell$ grow, $\mu \to \mu_{\text{sat}}$, the maximum fraction of peak achievable in the production regime. A batch-discrete alternative has one additional parameter, but we use this batch–generation–resolution (BGR) form for simplicity. Fitted parameters, residuals, and leave-one-architecture-out cross-validation are reported in Section 4.3.

### 3.4. Carbon Accounting

We adapt the lifecycle carbon accounting framework of Faiz et al. (2024) to single-GPU DiT inference. Given the FLOP estimate $F_{\text{total}}$ from Section 3.2 and the hardware efficiency $\mu$ from Section 3.3, runtime, energy, and carbon follow as:

$$\begin{aligned} \tau &= F_{\text{total}} \,/\, (\mu \cdot \Theta_{\text{peak}}) \\ E &= \tau \cdot \text{TDP} \,/\, (3.6 \times 10^6) \\ C_{\text{op}} &= E \cdot \text{PUE} \cdot I_{\text{grid}} \\ C_{\text{emb}} &= (\tau / L) \cdot C_{\text{chip}} \\ C_{\text{total}} &= C_{\text{op}} + C_{\text{emb}} \end{aligned} \tag{8}$$

where $\Theta_{\text{peak}}$ is the device peak throughput, TDP is rated thermal design power in watts, $E$ is energy in kWh after conversion from joules, PUE is the datacenter power usage effectiveness, $I_{\text{grid}}$ is grid carbon intensity in $gCO_2eq/kWh$, $L$ is GPU operational lifetime, and $C_{\text{chip}}$ is embodied manufacturing carbon. All carbon terms are expressed in $gCO_2eq$. $\tau$ and $L$ are expressed in the same time units.

The accounting equations follow LLMCarbon. DiTCarbon contributes the DiT-specific $\mu$ estimate that the accounting equations use. The pipeline accepts the production pooled BGR fit from Equation (3) for production inference, a $\mu$ value calibrated from a short measurement sweep when running below the production regime, or a directly measured $\mu$ when calibration data is available.

We make three simplifying assumptions. First, rated TDP replaces measured power draw. On our V100 corpus, mean inference power was 206 to 243 W compared to a 250 W TDP, overestimating operational carbon by roughly 5 to 15%. Second, embodied carbon is amortized linearly over GPU lifetime. We do not model duty cycle or end-of-life recycling. Third, the hardware efficiency $\mu$ fitted on the denoiser is applied to $F_{\text{total}}$. This can add at most 4% runtime error at typical configurations, because the one-shot terms account for less than 4% of total FLOPs. The current implementation targets single-GPU inference.

For current grid mixes, operational carbon dominates total emissions. As grids decarbonize, embodied carbon becomes the larger fraction, and accurate FLOP modeling (Sections 3.1 and 3.2) enters both terms directly.

# 4. Evaluation

**Hardware and baselines.** All evaluations use Tesla V100-PCIE-16GB inference at 112 TFLOPs FP16 peak and 250 W rated TDP. The 1080 Ti contrast in Figure 2 uses a matched Pascal subset with no tensor cores (11.3 TFLOPs FP32 peak). Baselines are official checkpoints for parameters, manual instrumentation and `fvcore` for FLOPs, and single-$\mu$ transfer for hardware efficiency.

**Models and configurations.** The hardware efficiency sweep covers 83 configurations over six denoising backbones and four variant families: DiT-XL/2 at native $256^2$ and $512^2$ training resolutions, PixArt-$\alpha$, PixArt-$\Sigma$, SD3 Medium, and CogVideoX-2B (Peebles & Xie, 2023; Chen et al., 2024b;a; Esser et al., 2024; Yang et al., 2025). Image models span $256^2$ to $1024^2$ with $B \in \{1, 4, 8\}$. CogVideoX-2B spans $256^2$, $480^2$, $720^2$ and $f \in \{6, 13, 25, 49\}$ at $B = 1$. All sweeps use 20 denoising steps.

**Implementation details.** Section 4.4 uses PUE = 1.20, grid carbon intensity 390 $gCO_2eq/kWh$, and V100 chip embodied carbon $C_{\text{chip}} = 9.78$ $kgCO_2eq$ from the LLMCarbon chip-area estimate, amortized over five years. Table 7 uses BGR predicted $\mu$ from Section 4.3. Prediction uncertainty is roughly 10%, dominated by the 7.8% $\mu$ MAPE, with FLOP error under 1%. Rated TDP makes operational carbon conservative by 5 to 15% on the V100 corpus.

**Findings overview.** Hardware efficiency on V100 varies $6.4\times$ across the full configuration sweep, ruling out a single transferred constant for DiT inference. Within production configurations (resolution $\geq 512^2$), one saturation model fits across architectures within 7.8% pooled MAPE and bounds leave-one-architecture-out error at 12%. Predicted operational carbon agrees with measured ground truth from wall-clock runtime and NVML GPU power within 10.7% MAPE in the production regime. A representative PixArt-$\alpha$ workload varies $45\times$ across regions, reaching 559 kg $CO_2eq$ per day at one million daily requests under the US-average profile. Sections 4.1 and 4.2 confirm the parameter and FLOP models behind these findings.

## 4.1. Parameter Model Verification

*Table 3.* Parameter model verification against five reference models from official HuggingFace checkpoints. Actual counts exclude VAE, text encoder, and tokenizer.

| Model | Predicted | Actual | Error |
|---|---|---|---|
| DiT-XL/2 | 668.9 M | 675 M | 0.91% |
| PixArt-$\alpha$ | 594.5 M | 611 M | 2.69% |
| SD3 Medium | 2.039 B | 2.028 B | 0.54% |
| SD3.5 Large | 8.091 B | 8.057 B | 0.42% |
| CogVideoX-2B | 1.681 B | 1.694 B | 0.77% |

The leading-order approximation reproduces parameter counts within 3% across the evaluated backbones. Table 3 compares predicted counts from Equation (1) with actual counts from official HuggingFace checkpoints. Errors range from 0.42% (SD3.5 Large) to 2.69% (PixArt-$\alpha$) across five models that span four architecture classes.

The PixArt-$\alpha$ residual is consistent with omitted global terms: roughly 16 M of 611 M parameters split between the shared adaLN-Single projection ($\sim 8$ M) and upstream T5 text projection ($\sim 5$ M). Both scale as $\mathcal{O}(d^2)$ globally rather than $\mathcal{O}(N \cdot d^2)$ and are intentionally dropped by the leading-order form. This precision is small relative to downstream uncertainty: measured V100 efficiency $\mu$ varies by $6.4\times$ across configurations (Section 4.3).

## 4.2. FLOP Model Validation

The analytical FLOP model agrees with manual operator instrumentation within 0.5% on the dominant denoising transformer block. We instrumented PyTorch forward hooks on every `Linear`, `Conv2d`, `bmm`, `baddbmm`, and `F.scaled_dot_product_attention` call, then ran PixArt-$\alpha$ at $256^2$ and $512^2$ under SDPA and eager attention. The standard profiler `fvcore` undercounts by 3.8% to 13.7% under SDPA because it cannot trace fused attention kernels; the gap scales quadratically with resolution and matches the hidden $4\ell^2 d$ term in Table 2. The one-shot text encoder and VAE terms are not independently profiled, so the 0.5% agreement validates the main denoising path rather than the complete FLOP decomposition.

The denoising loop accounts for 98% of total FLOPs at typical configurations. At PixArt-$\alpha$ $512^2$ with 50 steps, text encoder and VAE decoder contribute the remaining 2%; at DiT-XL/2 $256^2$ with 20 steps, the VAE share rises to 3%. Because the denoising path dominates, residual error in one-shot terms has little effect on typical carbon estimates.

## 4.3. Hardware Efficiency Characterization

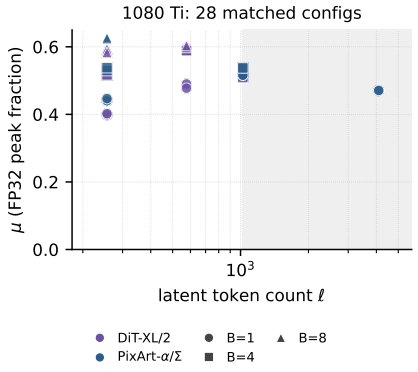


*Figure 2.* Matched 1080 Ti subset (DiT-XL/2 and PixArt-$\alpha$). Gray marks production configurations (resolution $\geq 512^2$); white marks lower resolutions. Without tensor cores, $\mu$ stays in a $1.5\times$ band versus V100's $6.4\times$ full-regime spread in Figure 3.

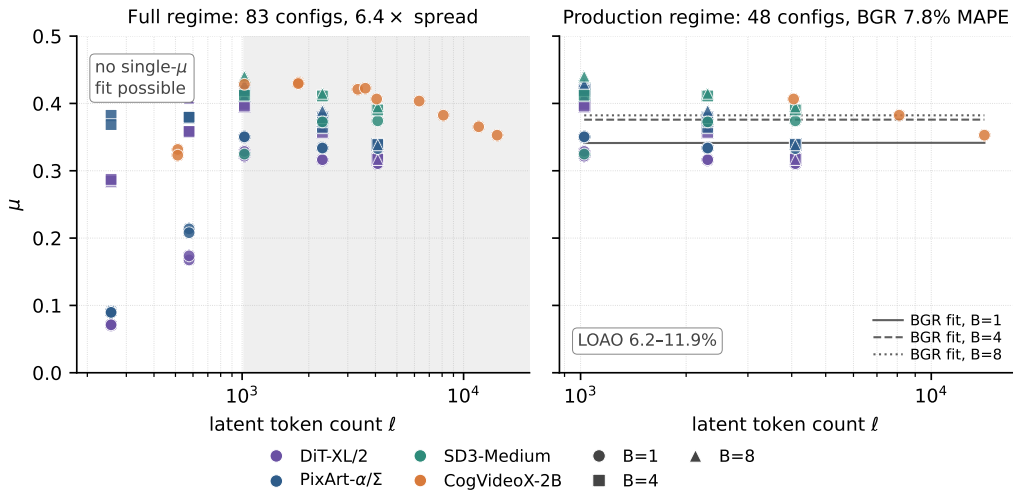


*Figure 3.* The dual finding visualized. Left: full configuration sweep, where the gray background marks production configurations (resolution $\geq 512^2$) and the white region marks lower resolutions; across all 83 configurations, $\mu$ varies $6.4\times$ and no single transferred constant predicts the range. Right: production configurations only, where $\mu$ converges across architectures and the BGR saturation model fits within 7.8% MAPE pooled across all six architectures, with leave-one-architecture-out cross-validation predicting held-out architectures within 6.2–11.9% MAPE. The regime split is the central empirical contribution of this work.

Figure 2 shows the matched 1080 Ti hardware contrast, with gray marking production configurations (resolution $\geq 512^2$) and white marking lower resolutions. Figure 3 shows the V100 sweep: the left panel keeps both regions to expose the full-regime spread, while the right panel keeps only the production region for the BGR fit. Together, the figures visualize the same hardware efficiency sweep from complementary angles: hardware contrast and regime separation. On V100, full-regime $\mu$ varies $6.4\times$, from 0.070 at $256^2$ $B = 1$ (DiT-XL/2 50-step step-independence repeat; the primary 20-step grid in Table 5 starts at 0.071) to 0.448 at $256^2$ $B = 8$. Any single transferred value can produce timing errors of several factors. On the Pascal 1080 Ti, which has an 11.3 TFLOPs FP32 peak, the same 28 configurations occupy only approximately 0.40 to 0.60 ($1.5\times$). This contrast matches the hardware difference: the V100's $10\times$ higher FP16 peak means small workloads leave more throughput unused, widening the efficiency range.

Paired architectures are consistent within each family. PixArt-$\alpha$ and PixArt-$\Sigma$ match across all 15 paired measured runs (max $\Delta\mu = 0.017$ at $768^2$ $B = 8$, mean $\Delta\mu = 0.005$). DiT-XL/2 at $256^2$ and $512^2$ training resolutions matches across 16 paired measured runs (max $\Delta\mu = 0.015$ at $768^2$ $B = 4$, mean $\Delta\mu = 0.004$), making transfer within each family the measured calibration point.

The full-regime spread is driven by resolutions below the production regime. Across all 83 configurations, a pooled BGR saturation model reaches 21.8% MAPE. DiT to PixArt transfer reaches 18.1% MAPE and PixArt to DiT 26.9%. Errors concentrate at $256^2$ and $384^2$, while open DiT systems target $1024^2$ scale generation: PixArt-$\alpha$ provides $1024 \times 1024$ checkpoints and resolution binning based on 1024, and Stable Diffusion 3 defaults to $1024 \times 1024$ for best results. We use $512^2$ as a conservative lower bound and define production as resolution $\geq 512^2$.

Production configurations converge across architectures. Across 48 production configurations, $\mu$ ranges from 0.310 to 0.440 with mean 0.365 across all six models and four variant families. At $512^2$ with any batch size, all five image and dual-stream models agree within 6% (DiT 0.399, PixArt-$\alpha$ 0.421, SD3 Medium 0.412 at $B = 4$). SD3 Medium runs 10 to 20% higher than DiT and PixArt at resolutions $\geq 768^2$ with $B = 1$, consistent with MMDiT's denser per-block computation.

Video DiTs occupy a narrower regime: CogVideoX-2B ranges from 0.323 to 0.431 across 12 V100 configurations from $256^2$ to $720^2$ and 6 to 49 frames, a $1.33\times$ spread. Even its smallest workload ($256^2$, 6 frames, $\ell = 512$) has a longer sequence than DiT-XL/2's smallest ($256^2$ $B = 1$, $\ell = 256$), keeping CogVideoX above the short sequence regime where small image workloads fall below $\mu = 0.10$.

The BGR saturation model from Equation (3) fits the production regime within 7.8% MAPE. Joint fitting across all six architectures on 48 production configurations yields residuals of 6.1 to 9.9%. Leave-one-architecture-out cross-validation fits five architectures and predicts the sixth at 6.2 to 11.9% MAPE (Table 4). The batch-discrete alternative noted in Section 3.3 also reaches 7.8% MAPE with one additional parameter.

*Table 4.* Leave-one-architecture-out cross-validation. The BGR saturation model is fit on production configurations (resolution $\geq 512^2$) from five architectures and used to predict the sixth. Every held-out architecture is predicted within 12% MAPE.

| Held-out architecture | Train $n$ | Test $n$ | LOAO MAPE |
|---|---|---|---|
| DiT-XL/2 | 39 | 9 | 10.6% |
| DiT-XL/2-512 | 39 | 9 | 9.2% |
| PixArt-$\alpha$ | 39 | 9 | 6.5% |
| PixArt-$\Sigma$ | 39 | 9 | 6.2% |
| SD3-Medium | 39 | 9 | 8.8% |
| CogVideoX-2B | 45 | 3 | 11.9% |

Combining these observations yields the dual finding in Figure 3: $\mu$ varies $6.4\times$ across the full regime and a pooled fit gives 21.8% MAPE, but a production-scale fit predicts held-out architectures within 12% MAPE. For production carbon prediction, a single calibration of $\mu$ for each hardware type on representative configurations suffices. Characterization below the production regime still needs calibration within each family. All measurements and the BGR fit apply to Tesla V100-PCIE-16GB within the measured $(B, \ell)$ range. Other accelerators require recalibration.

*Table 5.* Hardware efficiency $\mu$ across the 83-run V100 corpus. The table displays the 78 unique 20-step resolution/batch/frame cells; the remaining 5 entries are 50-step step-independence repeats not shown here. Image rows report $B = 1, 4, 8$ at each resolution. CogVideoX-2B reports $B = 1$ across frame counts.

| Model | Resolution | $\mu$ B=1 | $\mu$ B=4 | $\mu$ B=8 |
|---|---|---|---|---|
| DiT-XL/2 | $256^2$ | 0.071 | 0.284 | 0.417 |
| | $384^2$ | 0.168 | 0.358 | 0.407 |
| | $512^2$ | 0.322 | 0.399 | 0.411 |
| | $768^2$ | 0.316 | 0.357 | 0.364 |
| | $1024^2$ | 0.310 | 0.317 | 0.317 |
| DiT-XL/2-512 | $256^2$ | 0.071 | 0.287 | 0.415 |
| | $384^2$ | 0.174 | 0.358 | 0.408 |
| | $512^2$ | 0.329 | 0.396 | 0.412 |
| | $768^2$ | 0.317 | 0.372 | 0.378 |
| | $1024^2$ | 0.311 | 0.317 | 0.317 |
| PixArt-$\alpha$ | $256^2$ | 0.091 | 0.382 | 0.448 |
| | $384^2$ | 0.214 | 0.380 | — |
| | $512^2$ | 0.350 | 0.421 | 0.432 |
| | $768^2$ | 0.334 | 0.365 | 0.373 |
| | $1024^2$ | 0.335 | 0.340 | 0.339 |
| PixArt-$\Sigma$ | $256^2$ | 0.090 | 0.369 | 0.445 |
| | $384^2$ | 0.208 | 0.379 | — |
| | $512^2$ | 0.350 | 0.419 | 0.429 |
| | $768^2$ | 0.334 | 0.379 | 0.390 |
| | $1024^2$ | 0.333 | 0.339 | 0.338 |
| SD3-Medium | $512^2$ | 0.325 | 0.412 | 0.440 |
| | $768^2$ | 0.373 | 0.411 | 0.415 |
| | $1024^2$ | 0.374 | 0.390 | 0.395 |

CogVideoX-2B (B=1)

| Model | Res. | $f$=6 | $f$=13 | $f$=25 | $f$=49 |
|---|---|---|---|---|---|
| CogVideoX-2B | $256^2$ | 0.323 | 0.428 | 0.429 | 0.421 |
| | $480^2$ | 0.431 | 0.423 | 0.404 | 0.365 |
| | $720^2$ | 0.407 | 0.383 | 0.353 | — |

*Table 6.* Datacenter sensitivity for PixArt-$\alpha$ $1024^2$ batch 1 50 denoising steps. Embodied carbon is 0.0011 g across all profiles and is omitted.

| Datacenter | PUE | Grid ($gCO_2eq$/kWh) | Op. (g) | Total (g) |
|---|---|---|---|---|
| US Average | 1.20 | 390 | 0.558 | 0.559 |
| France Nuclear | 1.10 | 50 | 0.066 | 0.067 |
| Germany | 1.15 | 350 | 0.480 | 0.481 |
| Norway Hydro | 1.05 | 20 | 0.025 | 0.026 |
| India Coal | 1.40 | 700 | 1.169 | 1.170 |
| Azure US West | 1.18 | 250 | 0.352 | 0.353 |
| GCP Iowa | 1.10 | 410 | 0.538 | 0.539 |

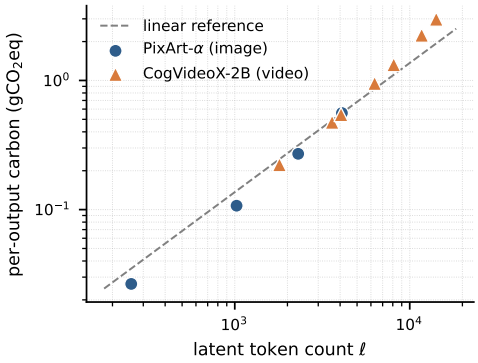


*Figure 4.* Per-output carbon scaling with latent token count $\ell$ for image (PixArt-$\alpha$, circles) and video (CogVideoX-2B, triangles) workloads on V100 under the US average datacenter profile. Image carbon scales approximately linearly with $\ell$ below the production regime and bends superlinear above $\ell \approx 2000$. Video workloads approximately align with the image trend. Production points cluster near 0.15 mg $CO_2$eq per latent token.

### 4.4. Carbon Predictions

Table 6 isolates regional effects for PixArt-$\alpha$ $1024^2$ $B = 1$: carbon ranges from 0.026 g (Norway hydro) to 1.170 g (India coal), a $45\times$ swing. This dominates the 9% reduction per output from batching PixArt-$\alpha$ $1024^2$ from $B = 1$ to $B = 4$, making deployment location the largest measured factor when model, resolution, and sampling configuration are fixed.

Table 7 reports lifecycle carbon for representative workloads spanning four backbone families. Image generations at $512^2$ batch 1 produce under 0.16 g $CO_2$eq per output across all models. PixArt-$\alpha$ carbon rises from 0.107 g at $512^2$ to 0.559 g at $1024^2$, a $5.2\times$ increase for $4\times$ more pixels, consistent with the quadratic attention term in Section 3.2. SD3-Medium $1024^2$ $B = 4$ reaches 0.658 g per output, while CogVideoX-2B $720^2$ at 13 latent frames reaches 1.325 g per output. At one million PixArt-$\alpha$ $1024^2$ images per day, the US-average profile gives 559 kg $CO_2$eq.

Figure 4 plots carbon per output against latent token count $\ell$. Carbon grows approximately linearly below the production regime and bends upward above $\ell \approx 2000$. PixArt $1024^2$ runs near 0.14 mg $CO_2$eq per latent token, CogVideoX $480^2$

*Table 7.* End-to-end DiTCarbon predictions on V100-PCIE under the US average datacenter profile (PUE = 1.20, grid intensity = 390 $gCO_2eq/kWh$). All values use BGR predicted hardware efficiency from Section 4.3. In the production regime, the BGR fit saturates in $\ell$ for fixed $B$, so $B = 1$ rows share $\mu = 0.341$ and $B = 4$ rows share $\mu = 0.376$ after rounding. Measured per configuration values are reported in Table 5.

| Model | Config | FLOPs | $\mu$ | Time (s) | Energy (Wh) | Op. (g) | Emb. (g) | Total (g) | $CO_2$/output (g) |
|---|---|---|---|---|---|---|---|---|---|
| DiT-XL/2 | $512^2$ B=1 50st | $1.06\times10^{14}$ | 0.341 | 2.78 | 0.193 | 0.090 | 0.0002 | 0.090 | 0.090 |
| DiT-XL/2 | $1024^2$ B=1 50st | $5.87\times10^{14}$ | 0.341 | 15.34 | 1.065 | 0.499 | 0.0010 | 0.500 | 0.500 |
| PixArt-$\alpha$ | $512^2$ B=1 50st | $1.26\times10^{14}$ | 0.341 | 3.29 | 0.229 | 0.107 | 0.0002 | 0.107 | 0.107 |
| PixArt-$\alpha$ | $1024^2$ B=1 50st | $6.57\times10^{14}$ | 0.341 | 17.18 | 1.193 | 0.558 | 0.0011 | 0.559 | 0.559 |
| PixArt-$\alpha$ | $1024^2$ B=4 50st | $2.63\times10^{15}$ | 0.376 | 62.40 | 4.333 | 2.028 | 0.0039 | 2.032 | 0.508 |
| SD3-Medium | $1024^2$ B=1 50st | $8.50\times10^{14}$ | 0.341 | 22.24 | 1.544 | 0.723 | 0.0014 | 0.724 | 0.724 |
| SD3-Medium | $1024^2$ B=4 50st | $3.40\times10^{15}$ | 0.376 | 80.79 | 5.611 | 2.626 | 0.0050 | 2.631 | 0.658 |
| CogVideoX-2B | $480^2$ B=1 20st f=13 | $5.57\times10^{14}$ | 0.341 | 14.57 | 1.012 | 0.474 | 0.0009 | 0.475 | 0.475 |
| CogVideoX-2B | $720^2$ B=1 20st f=13 | $1.56\times10^{15}$ | 0.341 | 40.70 | 2.826 | 1.323 | 0.0025 | 1.325 | 1.325 |

at 49 frames runs near 0.19 mg/token, and production points cluster near 0.15 mg $CO_2$eq per latent token.

Operational carbon dominates the lifecycle estimate. For PixArt-$\alpha$ $1024^2$ $B = 1$, operational carbon is 0.558 g while amortized chip embodied is 0.0011 g, a $507\times$ ratio. Across all workloads in Table 7, embodied never exceeds 0.5% of total under the US grid and reaches 4.2% under Norway hydro. Carbon reductions in the near term therefore depend mainly on grid intensity, PUE, batching, and runtime efficiency rather than on hardware manufacturing.

Existing predictive carbon tools target settings outside single-GPU DiT inference: LLMCarbon (Faiz et al., 2024) calibrates hardware efficiency for autoregressive transformer training and inference on multi-GPU systems, while diffusion energy work (Li et al., 2024; Delavande et al., 2025) measures specific models on specific hardware. No prior tool produces predictive lifecycle carbon for the same DiT setting, so we validate end-to-end against operational carbon computed from measured wall-clock runtime and NVML GPU power, the ground truth that runtime profilers such as CodeCarbon (Schmidt et al., 2021) produce. Predicted operational carbon agrees with measured carbon within 10.7% MAPE in the production regime (15.4% across the full 83-configuration corpus), averaging $1.08\times$ measured due to the conservative TDP-based power estimate.

## 5. Discussion

The $45\times$ regional carbon variance in Section 4 does not make workload prediction marginal but rather defines the decision hierarchy. Regional factors determine relative differences, but absolute carbon and tradeoffs across configurations still require the workload energy term. DiTCarbon provides that term before workloads run, making the $45\times$ spread comparable across region, batching, resolution, and model choices. Many generation requests tolerate seconds to minutes of scheduling delay, even though real-time use, on-device generation, and jobs pinned to a region do not. For flexible requests, a scheduler that routes jobs to regions with lower carbon intensity when capacity allows could reduce operational emissions. Realized savings depend on regional grid intensity differences, fleet capacity, and quality-of-service constraints we do not model.

The current scope has three limitations. Hardware efficiency measurements are V100 only: A100, H100, and consumer GPUs require calibration sweeps. The framework targets single-GPU inference. Tensor parallelism, replica batching, sharding, and training are out of scope. The BGR fit applies within resolution $\geq 512^2$ and the measured step range. Extreme batch and sequence length settings remain uncharacterized.

These limitations suggest four extensions: calibration across hardware, modeling across multiple GPUs, a scheduler prototype for carbon-aware routing, and training-time carbon estimates for DiT architectures.

## 6. Conclusion

DiTCarbon predicts lifecycle carbon for DiT inference from architecture and generation specifications before the workload runs. It covers class-conditional DiT, text-conditional PixArt, dual-stream MMDiT, and video DiT using unified parameter and FLOP models. Across 83 V100 configurations, hardware efficiency spans $6.4\times$; in the production regime, a pooled saturation fit reaches 7.8% MAPE with leave-one-architecture-out error bounded at 12%, and predicted operational carbon agrees with measured ground truth within 10.7% MAPE. Combined with the $45\times$ regional carbon variance observed for a representative deployment, the dominant factor in per-output emissions shifts from architecture choice to deployment location, giving operators and policymakers a way to act before DiT services scale further.

The main practical use is planning before launch. A target accelerator still needs hardware-efficiency calibration, but after calibration DiTCarbon can compare model, resolution, batching, and region choices without executing every candidate workload. This moves carbon accounting from post-hoc reporting to a design step for DiT services as image and video generation moves into consumer products.

## Impact Statement

DiT image and video services are moving into consumer products, where high per-request inference cost makes carbon emissions a deployment concern. For a representative PixArt-$\alpha$ $1024^2$ deployment, carbon per output varies $45\times$ across regions. At one million daily requests, the US-average profile extrapolates to 559 kg $CO_2$eq per day. DiTCarbon provides the evidence base for decisions at this deployment scale: lifecycle carbon estimates from architecture specifications, generation settings, and efficiency profiles calibrated on hardware before workloads are deployed. Predictions agree with measured operational carbon within 10.7% MAPE in the production regime, letting operators compare regional placement, batching, resolution, and architecture choices in measurable carbon terms. This shifts deployment decisions from runtime measurement to predictive planning that can reduce emissions when operators act on the estimates. Several caveats apply: estimates calibrated on V100 should not be applied to other hardware without recalibration, roughly 10% uncertainty limits absolute carbon disclosure claims, and per-request savings could be erased if predictive planning enables deployment growth that outpaces the efficiency gains.

## References


Chen, J., Ge, C., Xie, E., Wu, Y., Yao, L., Ren, X., Wang, Z., Luo, P., Lu, H., and Li, Z. Pixart-$\sigma$: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024a.

Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., and Li, Z. PixArt-$\alpha$: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International Conference on Learning Representations (ICLR)*, 2024b. Spotlight.

Delavande, J., Pierrard, R., and Luccioni, A. S. Video killed the energy budget: Characterizing the latency and power regimes of open text-to-video models. *arXiv preprint arXiv:2509.19222*, 2025.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Lacey, K., Goodwin, A., Marek, Y., and Rombach, R. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *PMLR*, pp. 12606–12633, 2024.

Faiz, A., Kaneda, S., Wang, R., Osi, R., Sharma, P., Chen, F., and Jiang, L. LLMCarbon: Modeling the end-to-end carbon footprint of large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., de Las Casas, D., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., van den Driessche, G., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Rae, J. W., Vinyals, O., and Sifre, L. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Lacoste, A., Luccioni, A., Schmidt, V., and Dandres, T. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019.

Li, B., Jiang, Y., and Tiwari, D. Carbon in motion: Characterizing Open-Sora on the sustainability of generative AI for video generation. In *Proceedings of the 3rd Workshop on Sustainable Computer Systems (HotCarbon '24)*, Santa Cruz, CA, USA, 2024.

Luccioni, A. S., Viguier, S., and Ligozat, A.-L. Estimating the carbon footprint of BLOOM, a 176B parameter language model. *Journal of Machine Learning Research*, 24 (253):1–15, 2023.

Luccioni, A. S., Jernite, Y., and Strubell, E. Power hungry processing: Watts driving the cost of AI deployment? In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT '24)*, Rio de Janeiro, Brazil, 2024.

Patterson, D., Gonzalez, J., Le, Q., Liang, C., Munguia, L.-M., Rothchild, D., So, D., Texier, M., and Dean, J. Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*, 2021.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Schmidt, V., Goyal, K., Joshi, A., Feld, B., Conell, L., Laskaris, N., Blank, D., Wilson, J., Friedler, S., and Luccioni, S. CodeCarbon: Estimate and track carbon emissions from machine learning computing, 2021.

Strubell, E., Ganesh, A., and McCallum, A. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3645–3650, 2019.

Wan-AI. Wan2.1-t2v-14b model card. https://huggingface.co/Wan-AI/Wan2.1-T2V-14B, 2025.

Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D., Zhang, Y., Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., and Tang, J. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations (ICLR)*, 2025.