DTP: Delta-Guided Two Stage Pruning for Mamba-based Multimodal Large Language Models

Seong Yeol Park, Min Jung Kwon, XIANGHUA PIAO, Yeong Hyeon Gu

Motivation

- Efficient decoding in Mamba-based MLLMs, but **high computational cost** remains in the **prefill stage**
- **Excessive and redundant visual tokens** significantly increase computation
- Existing pruning methods are **Transformer-based**, making them unsuitable for **Mamba-based MLLMs**
  - ➢ **Need for an efficient, Mamba-specific token pruning method**

Our Contribution

- Propose a **Mamba-specific** visual token pruning method (DTP)
- Utilize **Δ_t-based token importance** without any additional training
- Introduce **two-stage pruning** (early selective pruning + late complete pruning)
- Achieve **~50% computation reduction** and **~35% latency reduction** with minimal performance drop

Methods

- Method for estimating token importance

$$\overline{A_t} = \exp(\Delta_t A), \quad \overline{B_t} = (\Delta_t A)^{-1}(\exp(\Delta_t A) - I)\Delta_t B$$

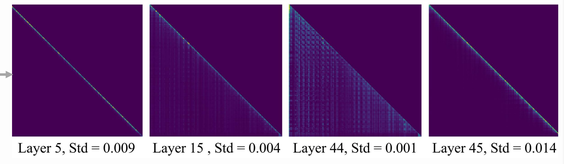
$$h_t = \overline{A_t} h_{t-1} + \overline{B_t} x_t, \quad y_t = C_t h_t$$

$$s_j = \frac{1}{D}\sum_{d=1}^{D} \Delta_{j,d}$$

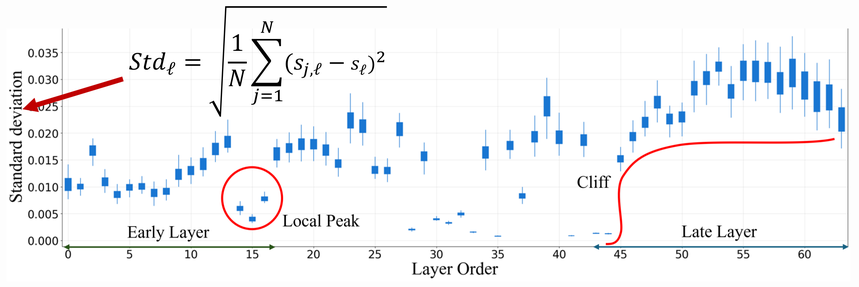
Δ_t: Input-dependent parameter & SSM parameter control

- Method for identifying pruning layers

$$y_t = \sum_{j=1}^{t} K_{t,j} x_j, \qquad K_{t,j} = C_t\left(\prod_{k=j+1}^{t} \overline{A_k}\right)\overline{B_j}$$



Visualization of Attention-like Patterns in Mamba



Layer-wise standard deviation of token-importance

$$\boldsymbol{k_{early}} = \underset{\ell}{argmin}\, Std_\ell, \quad where\ 0 \le \ell \le 0.25L \qquad \boldsymbol{k_{late}} = \underset{\ell}{argmax} |Std_{\ell+1} - Std_\ell| + 1, \quad where\ 0.7L \le \ell < L - 1$$

Experiments

- Main Results

Method	FLOPs	FLOPs ratio	GQA	VQAv2	TextVQA	POPE	VSR	VizWiz	Avg
Baseline (Cobra)	2.01	100%	62.3	77.8	58.2	88.4	58.4	49.7	65.8
FastV (k=2, r=0.7)	1.45	72%	62.1 (−0.2)	77.4 (−0.4)	56.9 (−1.3)	87.7 (−0.7)	58.0 (−0.4)	49.8 (+0.1)	65.3 (−0.5)
VTW (k=45)	1.43	71%	62.1 (−0.2)	77.7 (−0.1)	58.2 (+0.0)	88.3 (−0.1)	58.5 (+0.1)	49.5 (−0.2)	65.7 (−0.1)
DART (r=0.66)	1.38	69%	62.0 (−0.3)	77.1 (−0.7)	57.0 (−1.2)	87.4 (−1.0)	58.2 (−0.2)	49.7 (+0.0)	65.2 (−0.6)
Ours (r=0.9)	1.35	67%	62.0 (−0.3)	77.7 (−0.1)	57.9 (−0.3)	88.3 (−0.1)	58.9 (+0.5)	49.7 (+0.0)	65.8 (+0.0)
FastV (k=2, r=0.5)	1.06	53%	61.7 (−0.6)	76.8 (−1.0)	55.0 (−3.2)	87.4 (−1.0)	57.3 (−1.1)	50.1 (+0.4)	64.7 (−1.1)
VTW (k=32)	1.04	52%	47.1 (−15.2)	54.1 (−23.7)	42.6 (−15.6)	74.1 (−14.3)	57.9 (−0.5)	48.5 (−1.2)	54.0 (−11.8)
DART (r=0.44)	0.96	48%	61.2 (−1.1)	76.1 (−1.7)	55.1 (−3.1)	86.3 (−2.1)	57.7 (−0.7)	49.5 (−0.2)	64.3 (−1.5)
Ours (r=0.5)	0.97	48%	61.4 (−0.9)	77.1 (−0.7)	56.1 (−2.1)	87.3 (−1.1)	57.9 (−0.5)	49.6 (−0.1)	64.9 (−0.9)

- Ablation study on token importance estimation

Parameter	Cobra GQA	Cobra TextVQA	Cobra Vizwiz	RoboMamba GQA	RoboMamba POPE	RoboMamba OKVQA
y_t	61.1	48.0	49.0	54.5	83.8	63.8
B_t	60.2	47.4	49.2	54.2	82.9	62.6
C_t	58.9	44.9	49.6	53.0	81.6	62.0
Δ_t	61.4	56.1	49.6	54.9	84.4	63.8

※ Both Cobra and Robomamba are Mamba-based Multimodal LLMs ※

- Efficiency Analysis

Method	FLOPs	Prefill Mean (ms)	Decode Mean (ms)	Total Latency	GPU Memory
Cobra	2.01	98.04	5.04	16m 05s	8.8 GB
FastV (k=2, r=0.5)	1.06	60.64	4.94	10m 24s	8.5 GB
VTW (k=32)	1.04	67.31	4.99	11m 25s	8.3 GB
DART (r=0.44)	0.96	76.45	4.95	12m 44s	8.5 GB
DTP (r=0.5)	0.97	61.54	5.02	10m 35s	8.3 GB

Conclusion & Future work

- DTP effectively prunes visual tokens in Mamba-based MLLMs using Δ_t-based importance
- Future Work: Extending DTP to diverse architectures