AAAI-25 / IAAI-25 / EAAI-25
FEBRUARY 25 – MARCH 4, 2025 | PHILADELPHIA, USA

# Qua²SeDiMo: Quantifiable Quantization Sensitivity of Diffusion Models

Keith G. Mills[1,2], Mohammad Salameh[2], Ruichen Chen[1], Negar Hassanpour[2], Wei Lu[3] and Di Niu[1]

[1]Dept. ECE, University of Alberta [2]Huawei Technologies Canada Co., Ltd [3]Huawei Kirin Solution, Shanghai, China

UNIVERSITY OF ALBERTA

HUAWEI

ALBERTA INNOVATES

# Motivation

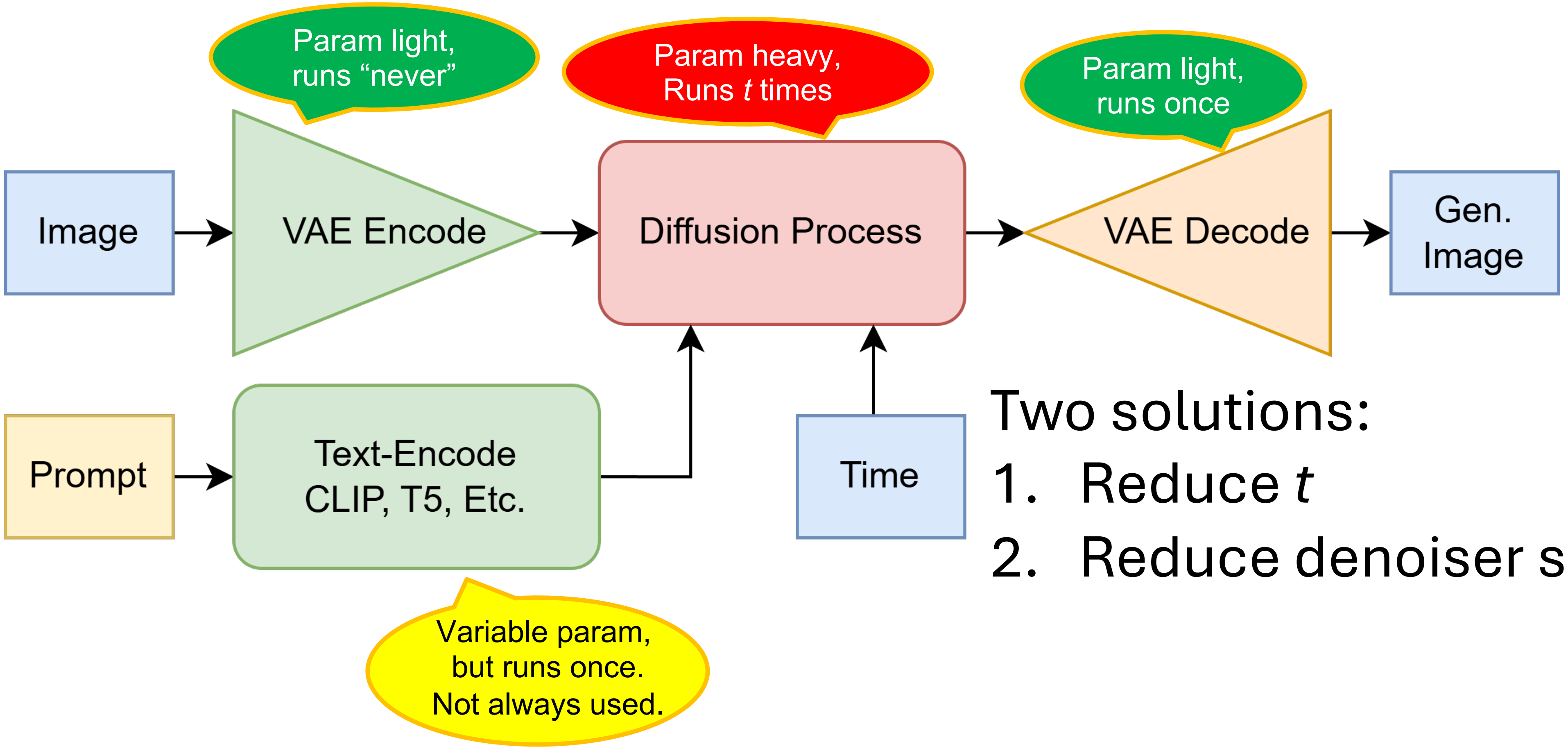
- Diffusion Models generate great visual content!
  - Examples: SDXL, PixArt, Hunyuan, etc.

PixArt-Σ

Prompt:
“A western-style medieval dragon with large white wings spread wide”

HunYuan-DiT

# Problem

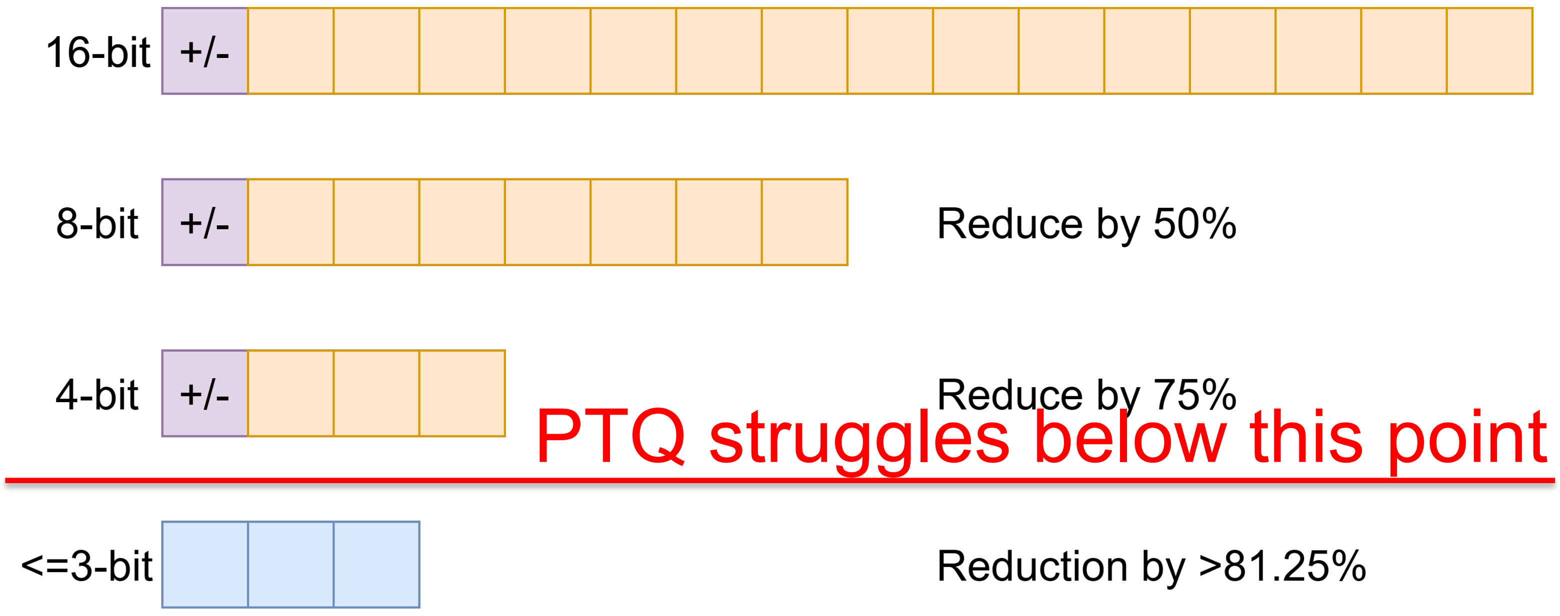



Two solutions:

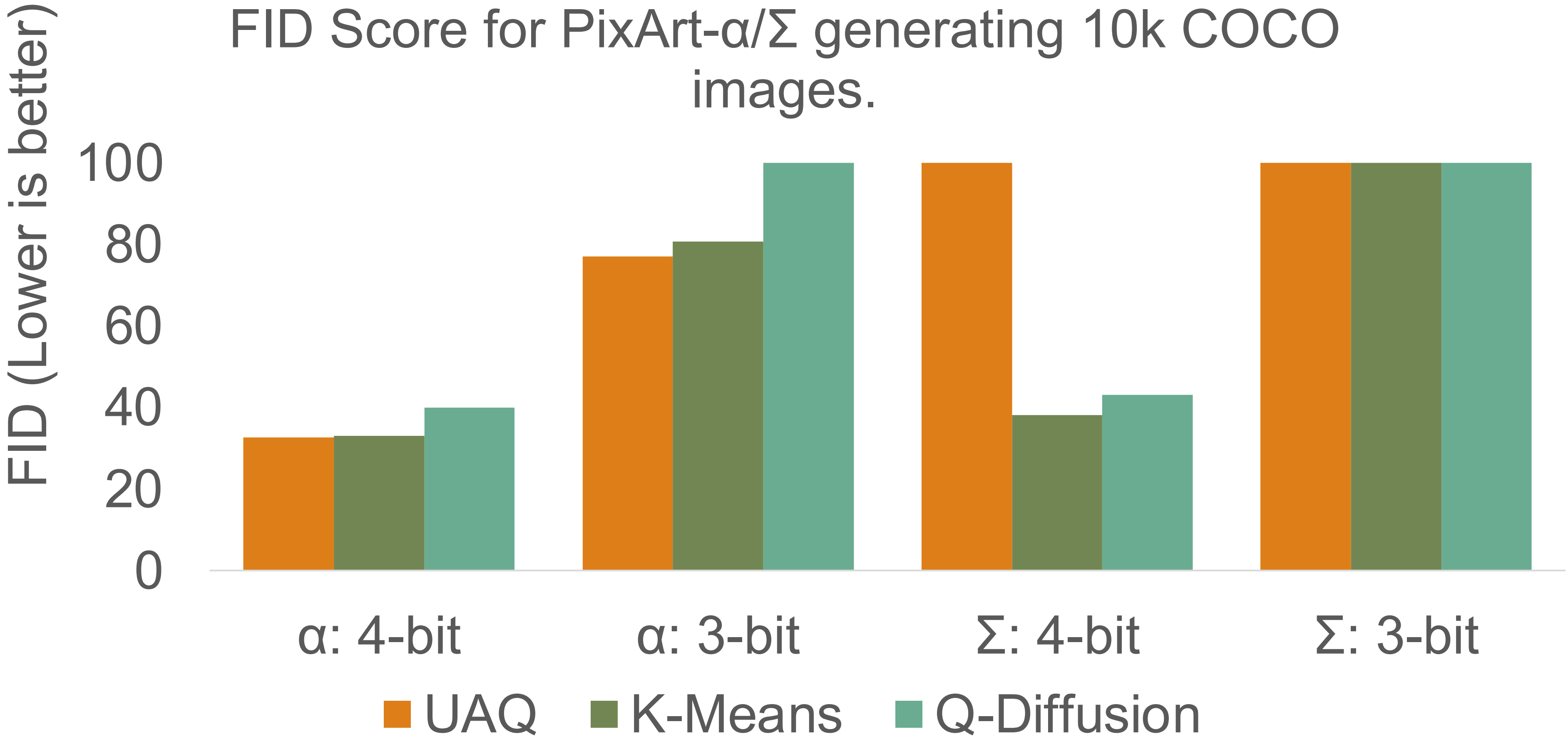
1. Reduce *t*
2. Reduce denoiser size

# Quantization

Reduces bit precision of weights/activations.
Quantization-Aware Training (QAT) (costly)
Post-Training Quantization (PTQ) (feasible)



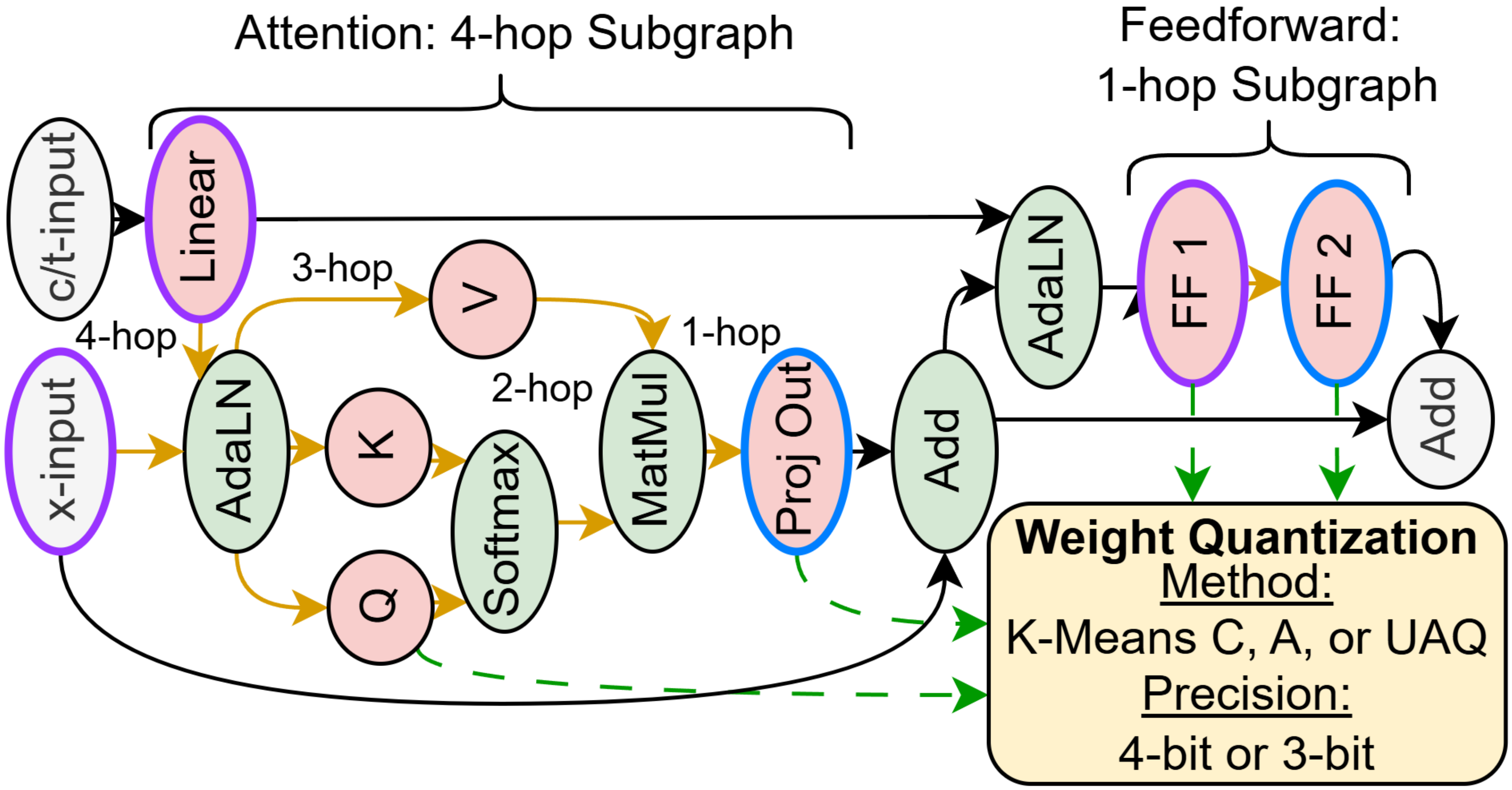
# 4-bit and 3-bit PTQ



# Why? Sensitivity Hypothesis

- Not all weights cause 3-bit performance loss.
- Assert that some *sensitive* weights are the culprit.
    - Motivates mixed-precision approach!
- “Sensitive”?
    - Individual weights? Too granular.
    - Weight categories? E.g., time-embed vs. caption-embed.
    - Weights in specific transformer blocks, like first/last?
- How to find sensitive weights?

# Our Solution



# Predictor with Hop-Level Ranking Loss

Preliminary: Graphs and GNNs

- $(arch, perf) = (G_1, y_1)$
- Learn $y'_1 = GNN(G_1)$

**Building Optimal Neural Architectures using Interpretable Knowledge**

CVPR'24

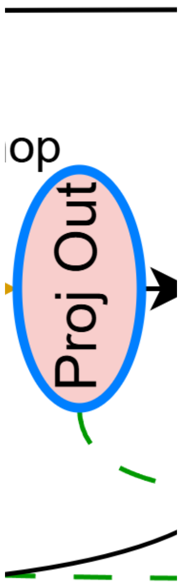

Keith G. Mills[1,2] Fred X. Han[2] Mohammad Salameh[2] Shengyao Lu[1]
Chunhua Zhou[3] Jiao He[3] Fengyu Sun[3] Di Niu[1]
[1]Dept. ECE, University of Alberta [2]Huawei Technologies Canada [3]Huawei Kirin Solution, China
{kgmills, shengyao, dniu}@ualberta.ca sunfengyu@hisilicon.com
{fred.xuefei.han1, mohammad.salameh, zhouchunhua, hejiao4}@huawei.com

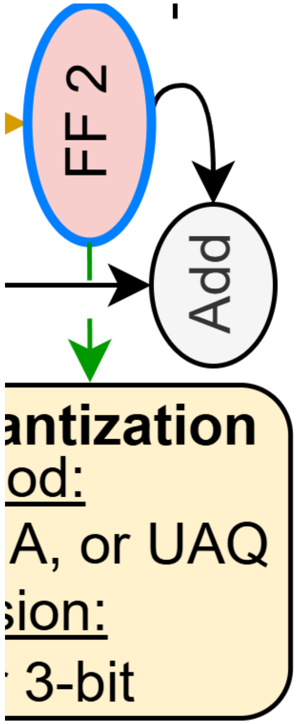

Intermediate workings: Node and Graph Embeddings

- $GNN(G) = MLP(h_G^m);\ h_G^m = \frac{1}{|V_G|}\sum_{v\in V_G} h_v^m$
- *m* is hop-level => $h_v^m$ represents an *entire* subgraph/module!

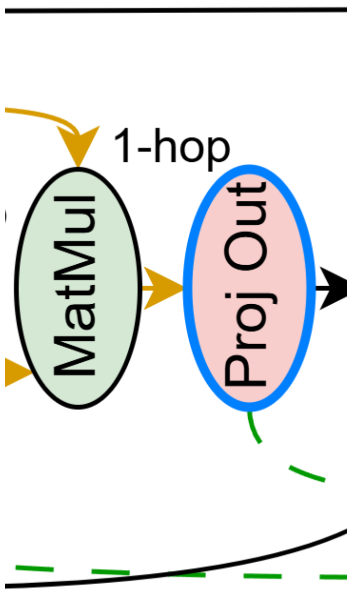
Key learning constraint: if $y_1 > y_2$, then $\left\|h_{G_1}\right\|_1 > \left\|h_{G_2}\right\|_1$

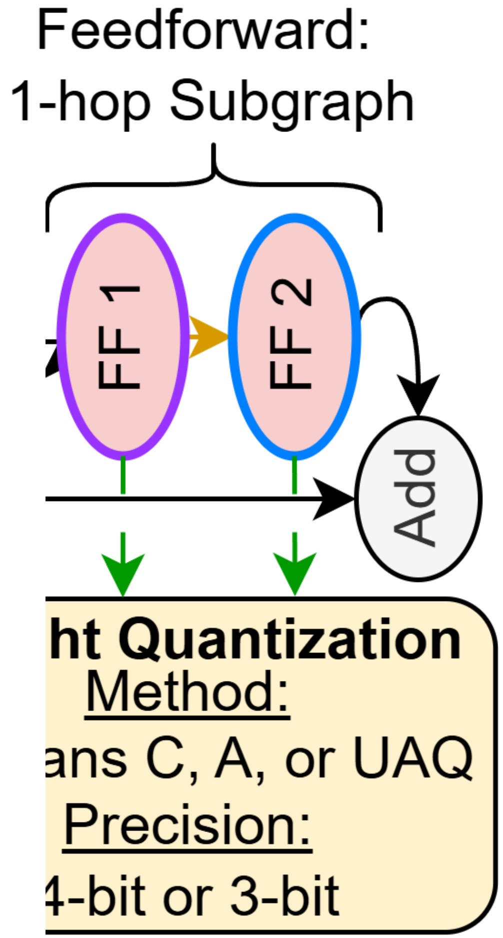
# Visual Example



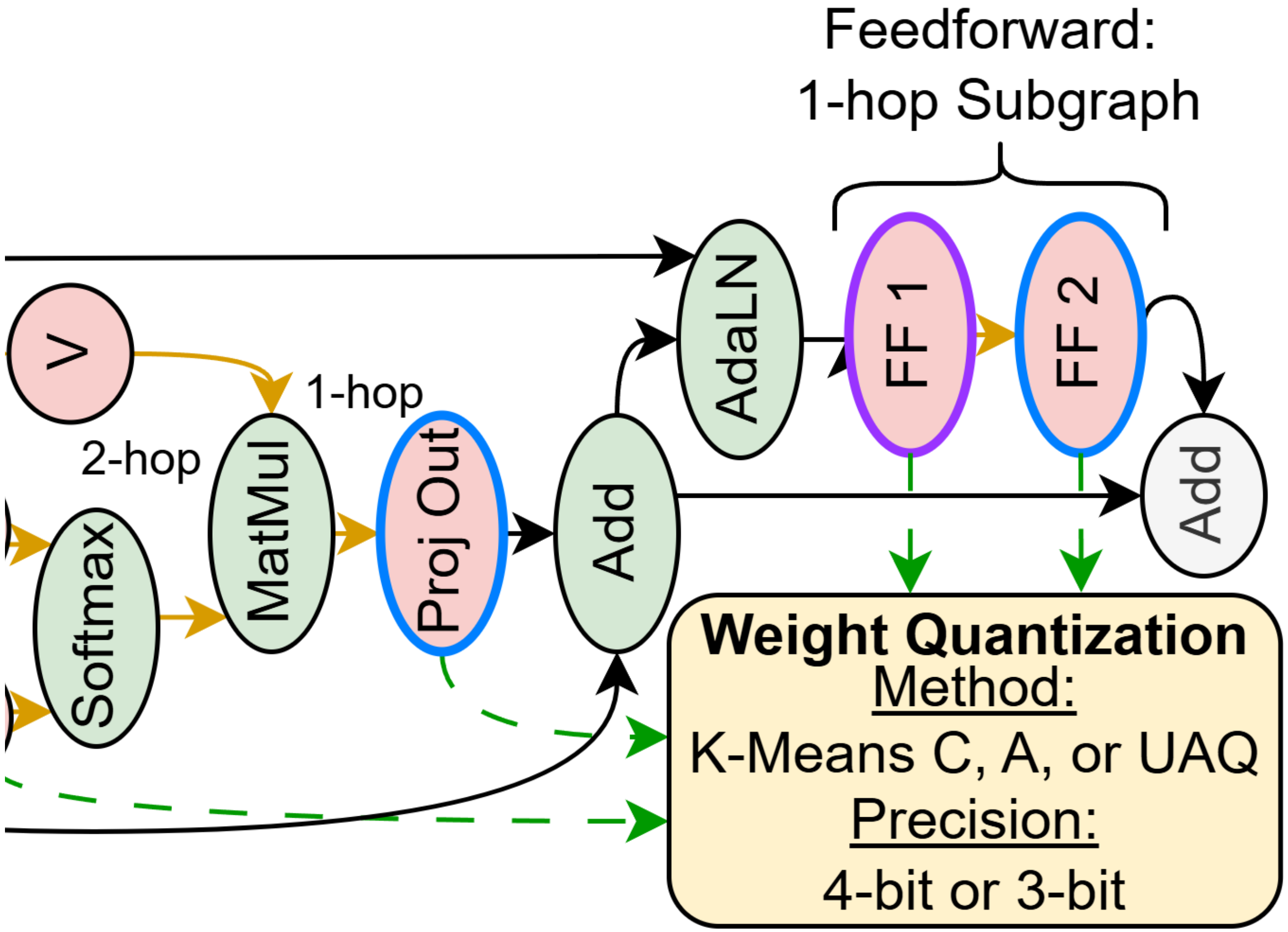



# Visual Example

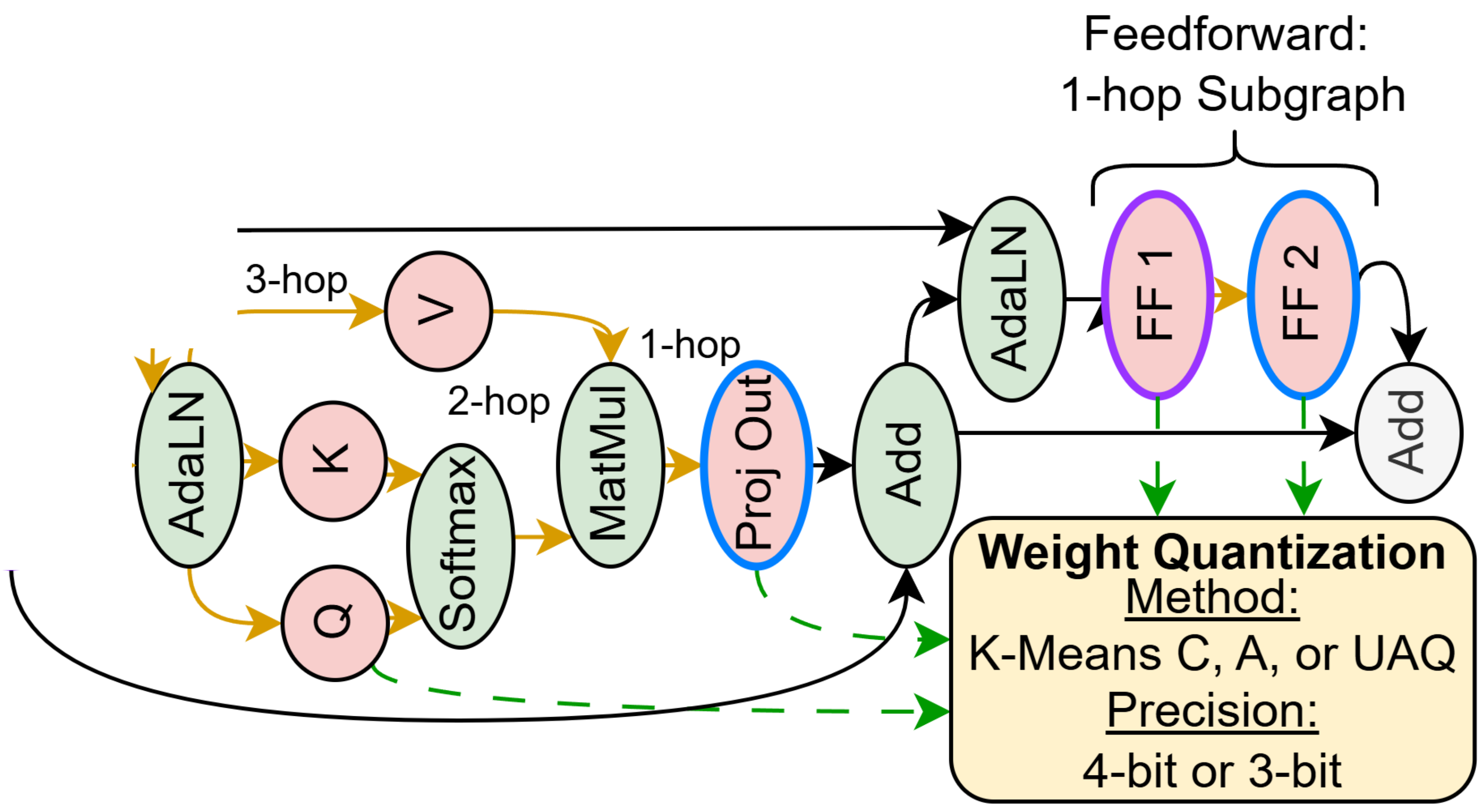


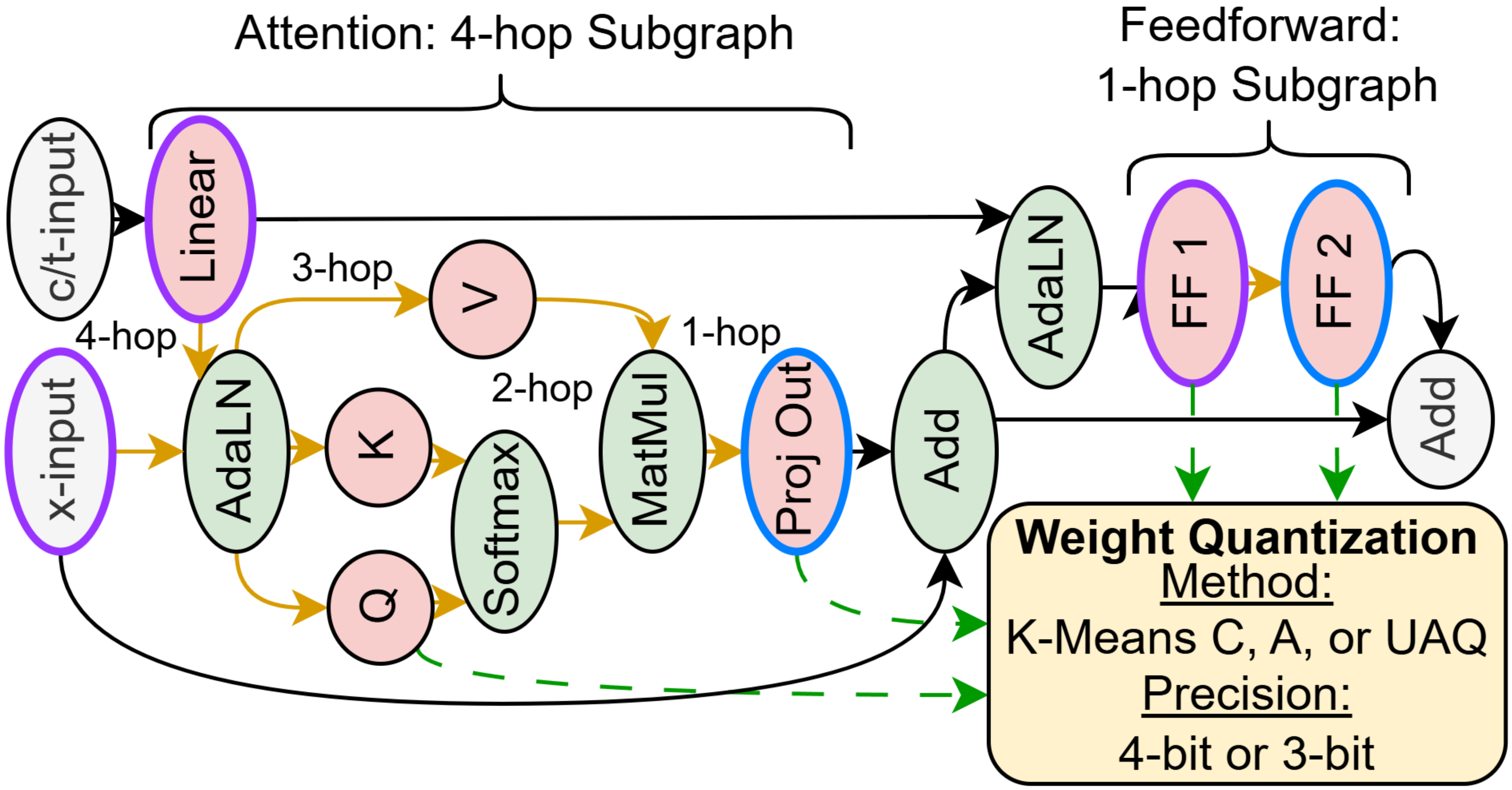
# Visual Example



# Visual Example



# Visual Example



# Predictor with Hop-Level Ranking Loss

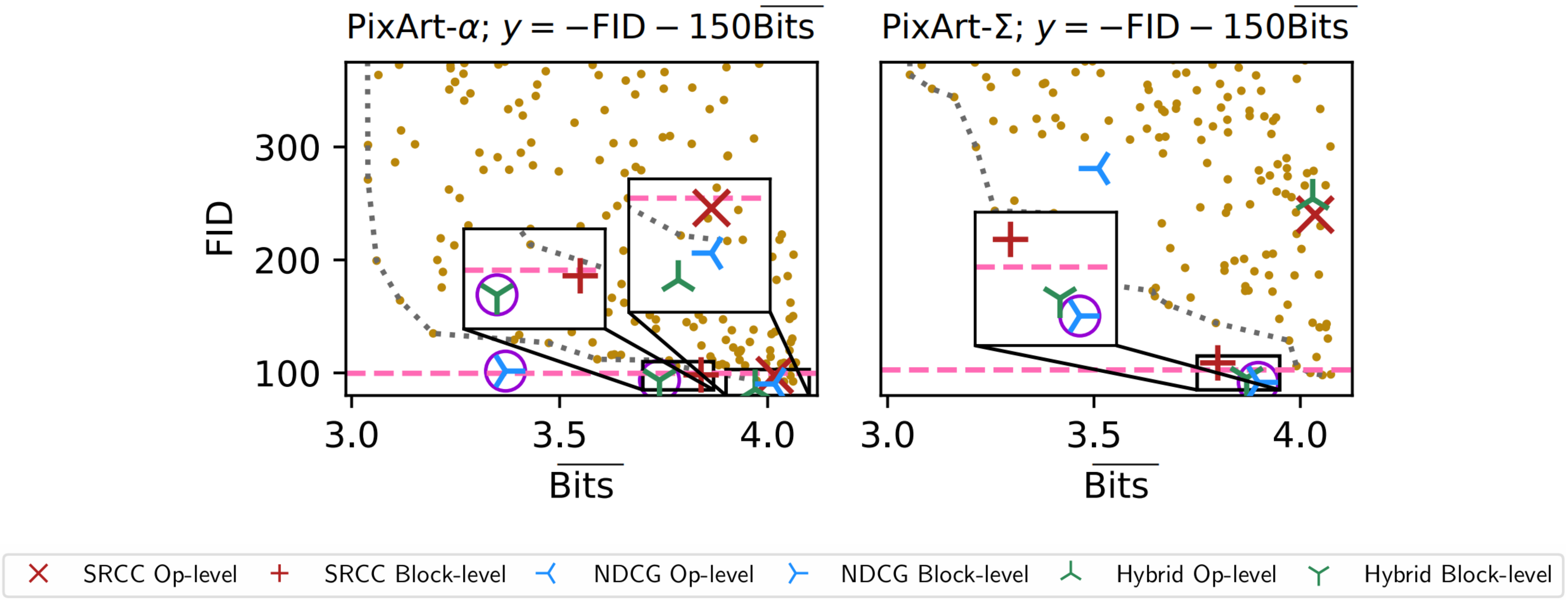
Preliminary: Graphs and GNNs

- $(arch, perf) = (G_1, y_1)$
- Learn $y'_1 = GNN(G_1)$

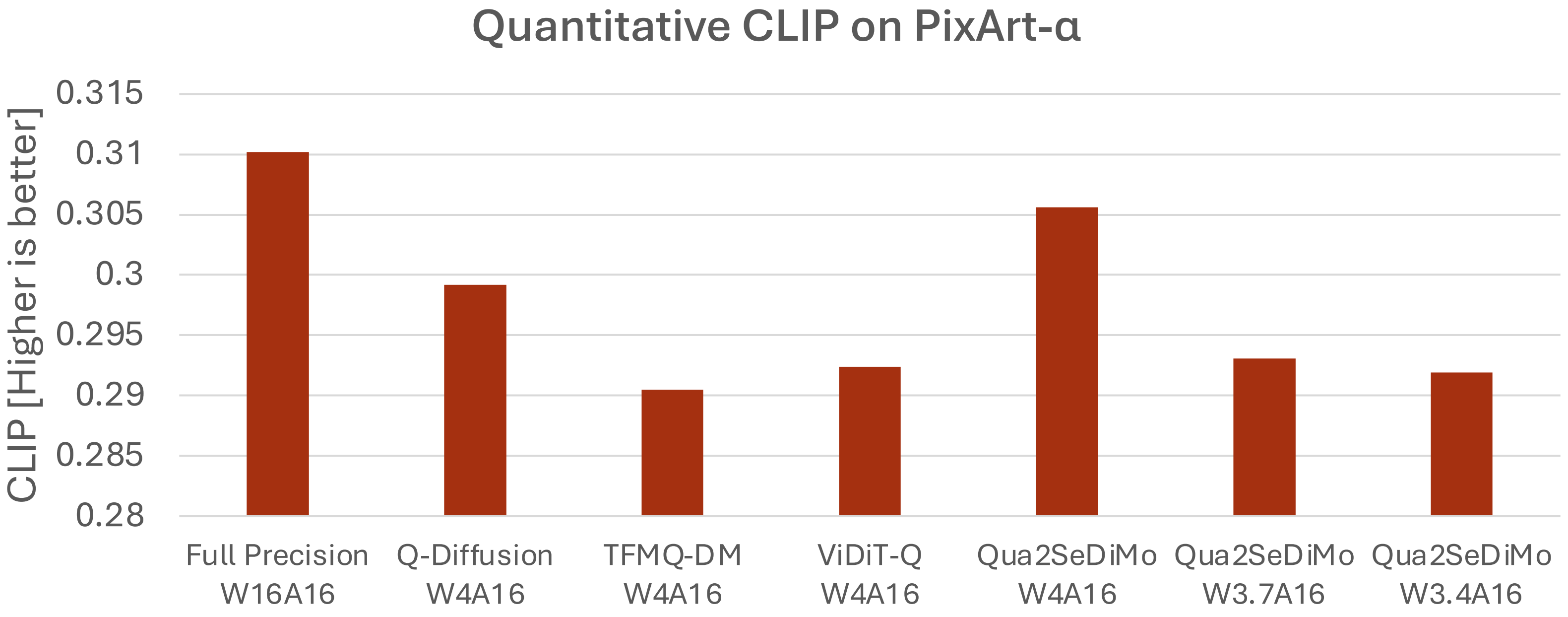
Optimize $L_{orig}(y, y') + \frac{1}{M+1} \sum_{m=0}^{M} L_{rank}(y, \|h_G^m\|_1)$

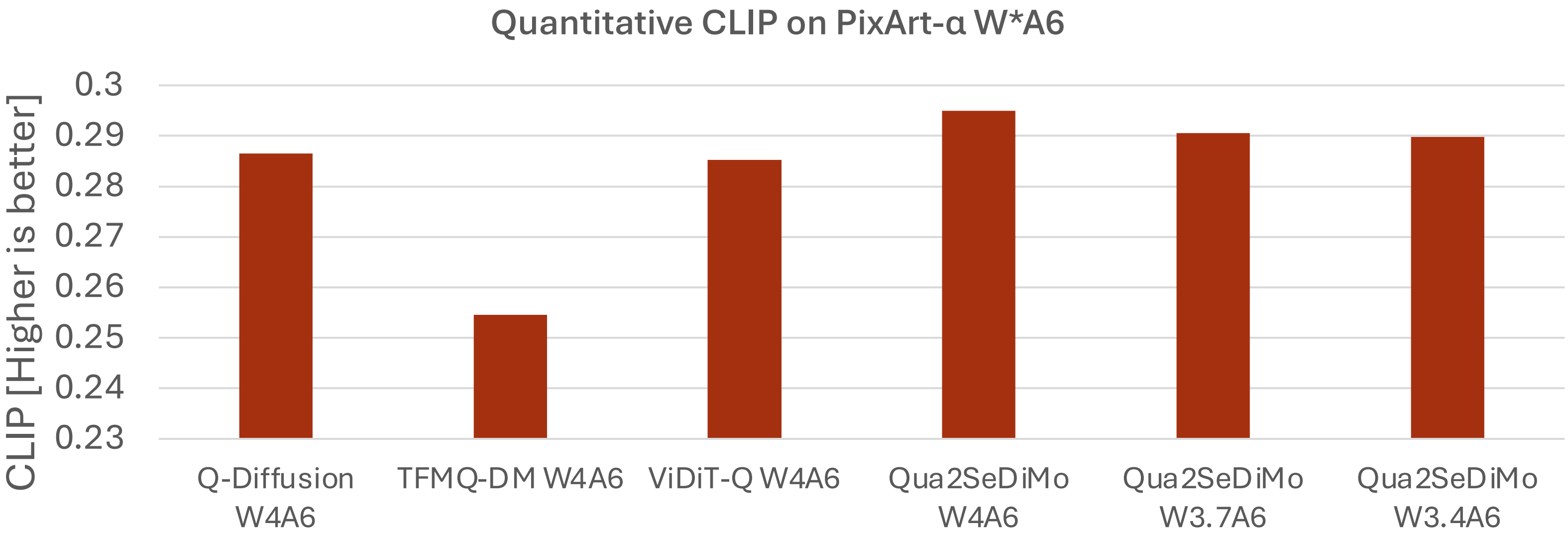
- $L_{orig}$ is traditional predictor loss, like MSE
- $L_{rank}$ is SRCC, LambdaRank, or both.

# Pareto Frontier Results

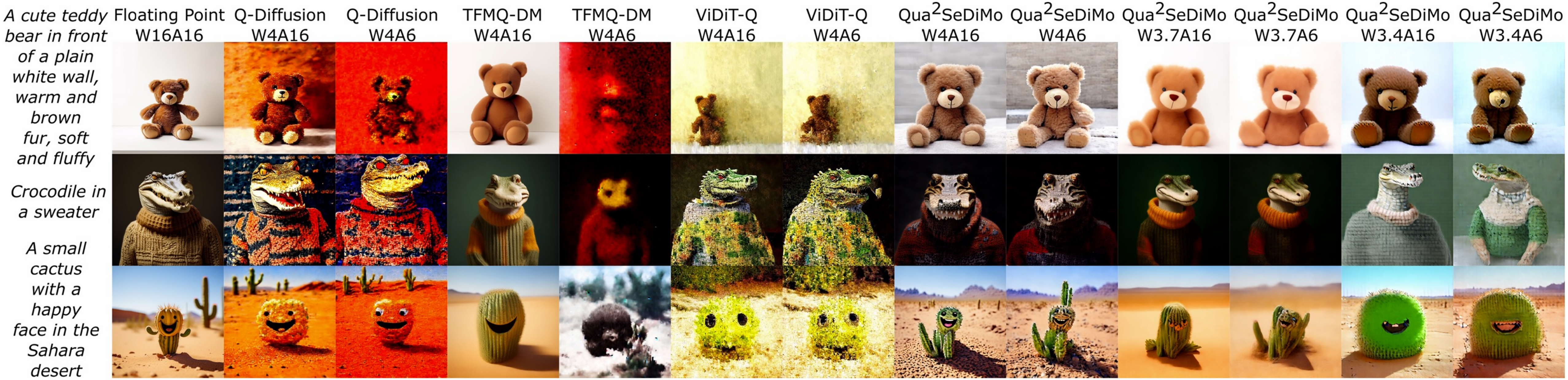


# Pareto Frontier Results

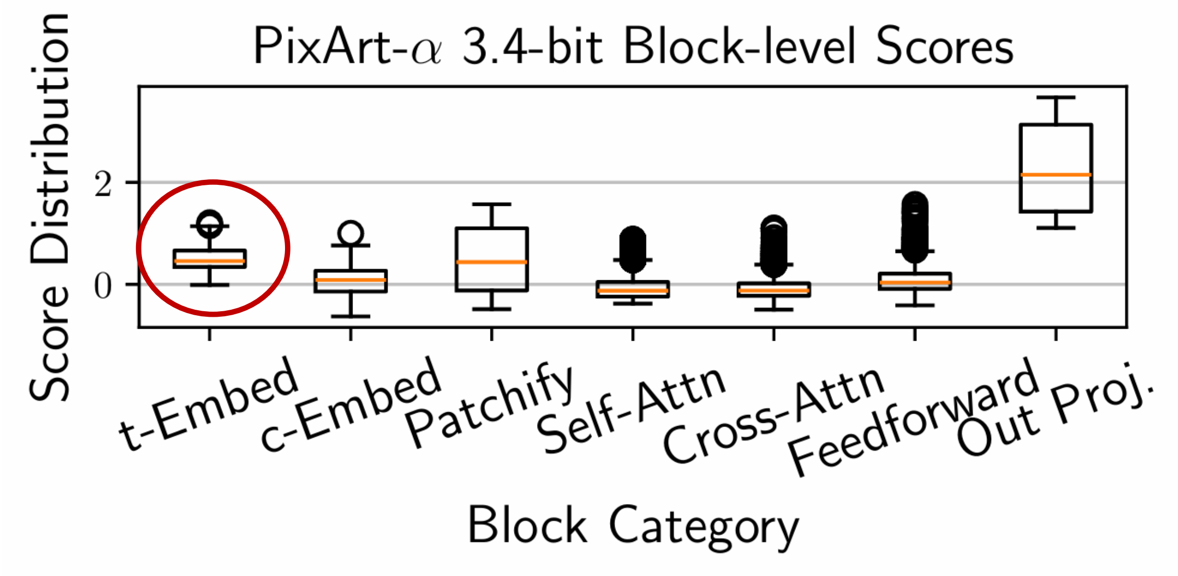


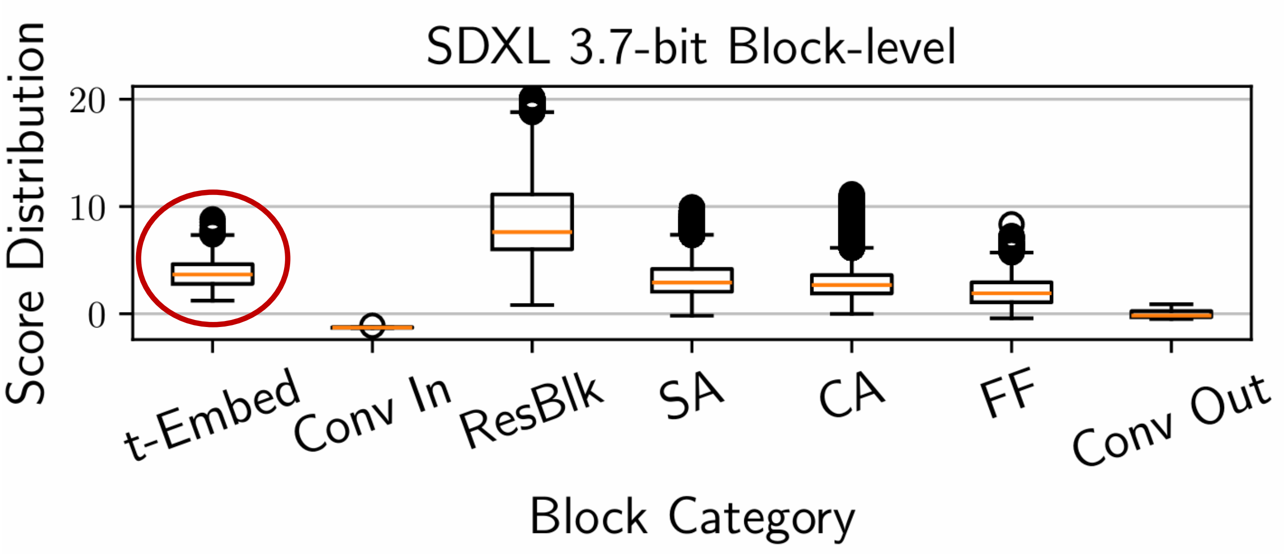



# Qualitative Visual Results



# Sample Insights



AAAI-25 / IAAI-25 / EAAI-25

FEBRUARY 25 – MARCH 4, 2025 | PHILADELPHIA, USA

# Qua$^2$SeDiMo:
# Quantifiable Quantization Sensitivity of Diffusion Models

Thank you for watching 'till the end!

See you in Philly!

UNIVERSITY OF ALBERTA

HUAWEI

ALBERTA INNOVATES