Zero-shot Quantization: A Comprehensive Survey

Minjun Kim*, Jaehyeon Choi*, Jongkeun Lee, Wonjin Cho, and U Kang†

Seoul National University, Seoul, South Korea

*Equal Contribution, †Correspond to: ukang@snu.ac.kr

Overview

- We survey **Zero-shot Quantization** (ZSQ), a data-free model compression paradigm
  - ZSQ faces three key challenges: knowledge transfer, synthetic-real discrepancy, and task adaptability
- We categorize and review ZSQ methods in three main groups
  - Synthesis-free, generator-based, and noise-optimization
- We discuss current limitations and future directions
  - Improving synthetic dataset, theory, problem setting, and evaluation remain open research questions

Outline

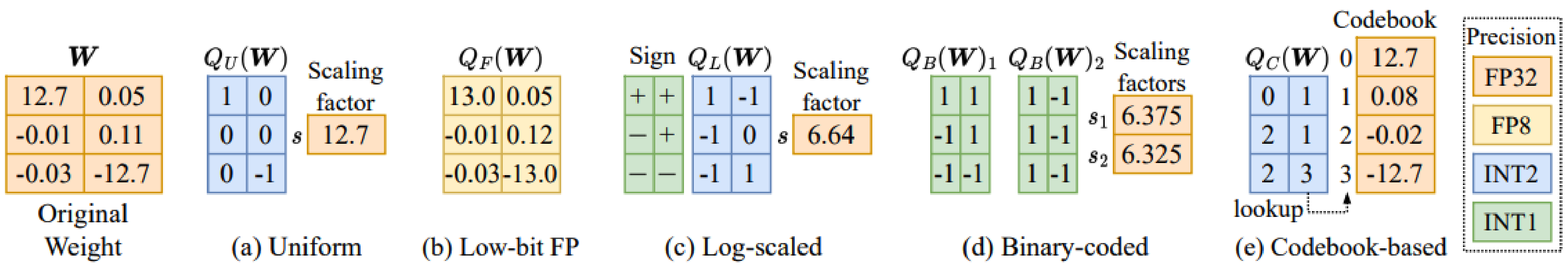
- **Introduction**
- Problem Formulation
- Categorization
- ZSQ Algorithms
- Future Research Directions
- Conclusion

Model Compression

- **Task:** Deploying neural networks on resource-constrained edge devices is challenging
- Various model compression techniques:
  - **Quantization**
  - Pruning
  - Knowledge distillation
  - Low-rank approximation
  - Parameter sharing
  - Efficient architecture design
  - and more…

Quantization

- Quantization methods represent a full-precision model with lower-bit formats
  - High compression and acceleration rate with minimal performance degradation
  - e.g., 32-bit model ➔ 4-bit quantization: 8× compression

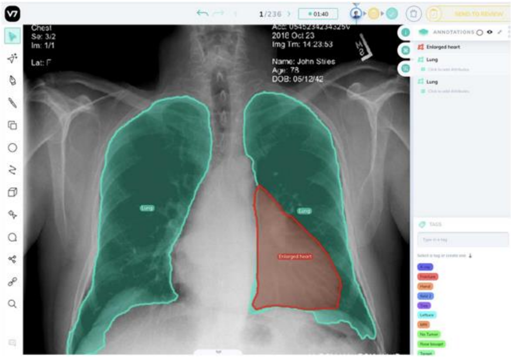


S. Park et al., "A Comprehensive Survey of Compression Algorithms for Language Models", arXiv:2401.15347

Zero-shot Quantization

- Zero-shot Quantization (ZSQ) achieves quantization without requiring any real data
  - **Limitation of existing methods.** the dependence on training data

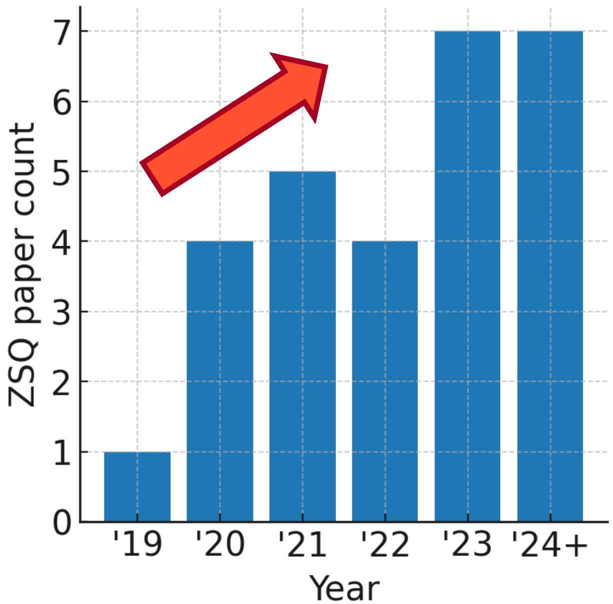
- Privacy or policy issues may block access to data
  - e.g., medical records, confidential business information



R. Kundu, “The Essential Guide to Zero-Shot Learning”, V7 Blog, Jan 6, 2022

Survey on ZSQ

- 25+ paper in major venues since DFQ [ICCV 2019]
  - Rapid growth in research
  - **Limitation.** Existing surveys focus on broader topics
    - e.g., model compression or network quantization



Survey on ZSQ

- We conduct the first in-depth survey on ZSQ
  - **Formulation.** We formulate the ZSQ problem and explore three critical challenges
  - **Categorization.** We categorize ZSQ algorithms based on their data generation strategies
  - **Analysis.** We analyze current ZSQ algorithms, highlighting their motivations, ideas, and key findings
  - **Discussion.** We outline future research questions to guide research toward impactful advancements

Outline

- Introduction
- **Problem Formulation**
- Categorization
- ZSQ Algorithms
- Future Research Directions
- Conclusion

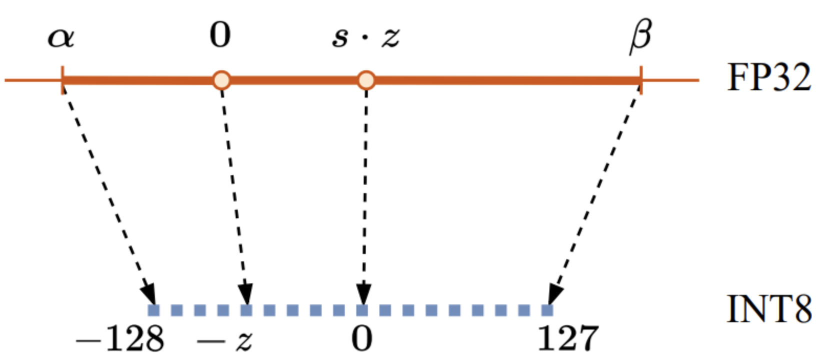
Preliminaries

Network Quantization

- Min-max Uniform Quantization (Input: $\mathbf{W}, B$ ➔ Output: $\mathbf{W}_q$)

$$\boldsymbol{W}_q = \left\lfloor \frac{\mathbf{W}}{s} - z + \frac{1}{2} \right\rfloor, \quad s = \frac{\beta - \alpha}{2^B - 1}, \quad z = \frac{\alpha}{s} + 2^{B-1}$$

  - $\mathbf{W}$: weight matrix of the full precision model
  - $\mathbf{W}^q$: B-bit quantized matrix of $\mathbf{W}$
  - B: quantization bits
  - s: scaling factor
  - z: integer offset
  - α: minimum value in $\boldsymbol{W}$
  - β: maximum value in $\boldsymbol{W}$



S. Park et al., "A Comprehensive Survey of Compression Algorithms for Language Models", arXiv:2401.15347

Preliminaries
QAT and PTQ

- Quantization methods are classified into two settings by their need of additional fine-tuning
  - **QAT (Quantization-Aware Training).** First quantize the model, then fine-tune the weight parameters
    - Rely on min-max quantization
  - **PTQ (Post-Training Quantization).** No additional training required
    - e.g., adaptive rounding, block reconstruction, random dropping

Problem Definition

Zero-shot Quantization

- **Given**
  - A model θ trained on a task $\mathcal{T}$
  - Quantization bits B
- **Generate**
  - a quantized model θ_q within the B-bit limit for maximum accuracy on $\mathcal{T}$ **without the use of real data**

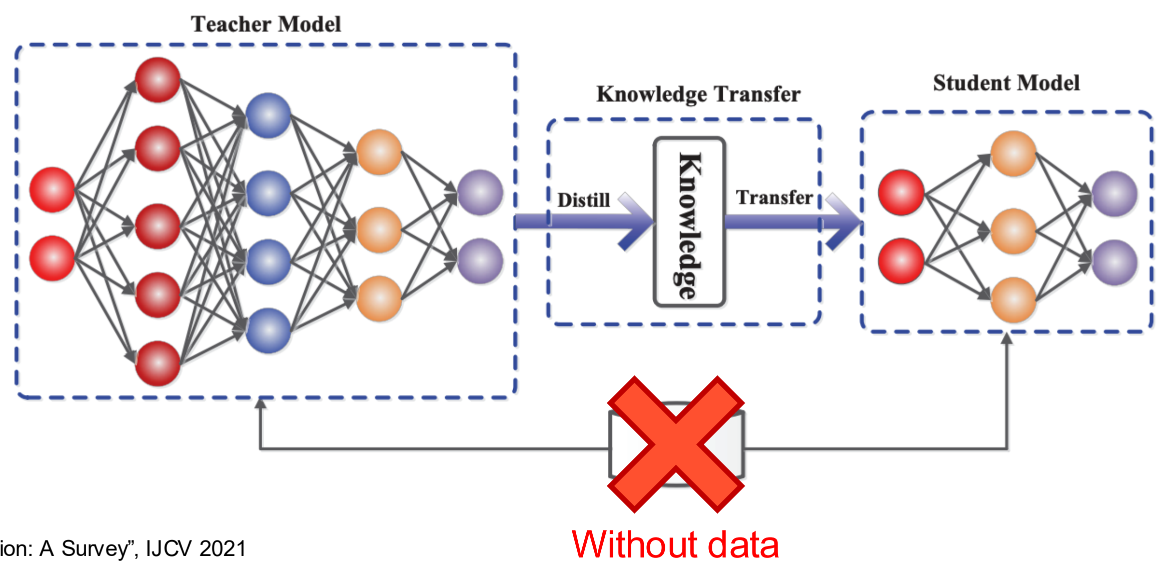
Main Challenges of ZSQ

- ZSQ algorithms should overcome key challenges that arise due to the absence of real data
  - 1. Knowledge transfer from the pre-trained model
  - 2. Discrepancy between real and synthetic datasets
  - 3. Diversity of the problem setting

Main Challenges of ZSQ

Knowledge transfer from the pre-trained model

- How do we transfer knowledge without real data?
  - Quantized model must preserve original behaviors
  - **Challenge.** No real data for alignment or calibration
  - **Solution Direction.** Adapt *synthetic data*, distillation losses, or architectural constraints to mimic the original

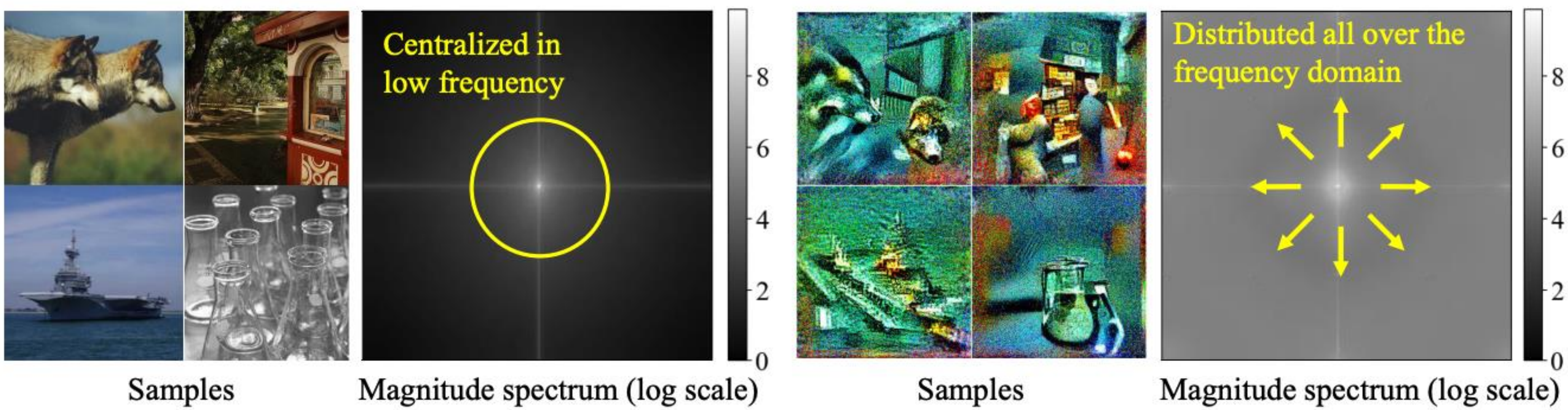


J. Gou et al., "Knowledge Distillation: A Survey", IJCV 2021

Main Challenges of ZSQ

Discrepancy between real and synthetic datasets

- Synthetic data doesn't match real data distributions
  - **Challenge.** Models quantized with synthetic data may underperform on real-world tasks
  - **Solution Direction**. Improving the quality of synthetic data or dataset reduces performance degradation
    - e.g., noise in image, intra-class heterogeneity



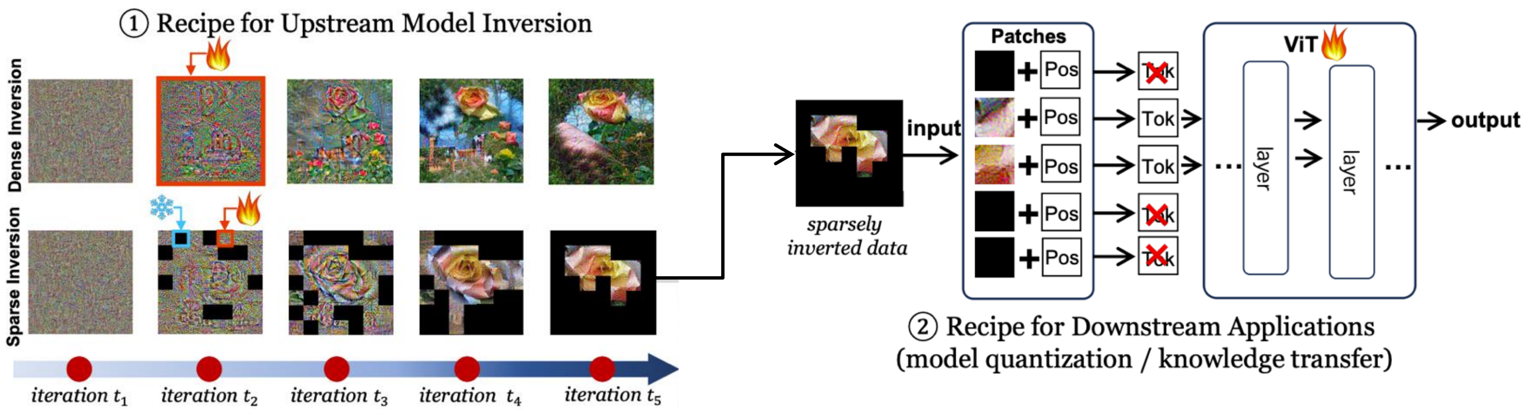
(a) ImageNet dataset (b) Synthetic dataset (TexQ)

M. Kim et al., "SynQ: Accurate Zero-shot Quantization by Synthesis-aware Fine-tuning", ICLR 2025

Main Challenges of ZSQ

Diversity of the problem setting

- ZSQ should generalize to various architectures, tasks, and quantization bit-widths
  - **Challenge.** Some algorithms work only for specific settings
  - **Solution Direction.** Develop universal frameworks or adaptable techniques
    - e.g., ViT-specific method due to patch-wise operation



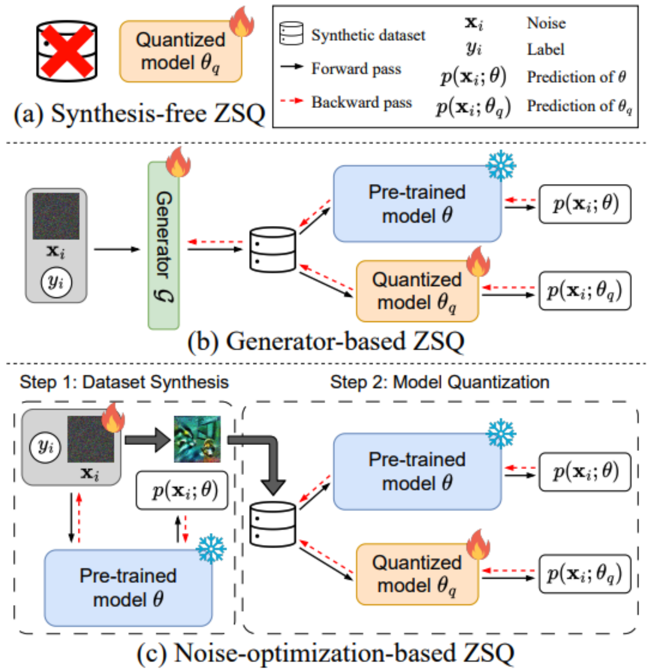
Z. Hu et al., "Sparse Model Inversion: Efficient Inversion of Vision Transformers for Data-Free Applications", ICML 2024

Outline

- Introduction
- Problem Formulation
- **Categorization**
- ZSQ Algorithms
- Future Research Directions
- Conclusion

Taxonomy

- We categorize ZSQ algorithms based on their data generation approach as:
  - **Synthesis-free ZSQ**
    - Quantize models without generating any synthetic data
  - **Generator-based ZSQ**
    - Train an additional generator $\mathcal{G}$ to produce synthetic data
  - **Noise-optimization-based ZSQ**
    - Directly optimize noise inputs to make synthetic data



Taxonomy

- We summarize the key features of ZSQ methods
  - 1. Data Generation Approach

	Method	Training Requirement	Scope of Contribution	Architecture	# Images	Accuracy (FP = 71.47)	
						W4A4	W3A3
Synthesis-free	DFQ [2019]	PTQ	Q	CNN	0	55.78	-
	SQuant [2022]	PTQ	Q	CNN	0	66.14	25.74
	UDFC [2023]	PTQ	Q	CNN	0	63.49	-
Generator-based	GDFQ [2020]	QAT	S, Q	CNN	1.28M	60.60	20.23
	ZAQ [2021]	QAT	S, Q	CNN	1.28M	52.64	-
	ARC [2021]	QAT	S, Q	CNN	1.28M	61.32	23.37
	Qimera [2021]	QAT	S, Q	CNN	1.28M	63.84	1.17
	ARC + AIT [2022]	QAT	Q	CNN	1.28M	65.73	-
	AdaSG [2023b]	QAT	S, Q	CNN	1.28M	66.50	37.04
	AdaDFQ [2023a]	QAT	S, Q	CNN	1.28M	66.53	38.10
	Causal-DFQ [2023]	QAT	S, Q	CNN	1.28M	68.11	-
	RIS [2024]	QAT	S	CNN	1.28M	67.75	-
	GenQ [2024b]	PTQ / QAT	S	CNN	1K§	69.77§	-
Noise-optimization	DeepInversion [2020]	QAT	S	CNN	32	70.27*	64.28†
	IntraQ [2022]	QAT	S, Q	CNN	5.12K	66.47	45.51
	HAST [2023]	QAT	S, Q	CNN	5.12K	66.91	51.15
	TexQ [2023]	QAT	S, Q	CNN	5.12K	67.73	50.28
	PLF [2024]	QAT	Q	CNN	5.12K	67.02	-
	SynQ [2025b]	QAT	Q	CNN / ViT	5.12K	67.90	52.02
	ZeroQ [2020]	PTQ	S, Q	CNN	1K	26.04	-
	KW [2020]	PTQ	S, Q	CNN	1K	69.08	-
	DSG [2021]	PTQ	S	CNN	1K	34.53	-
	MixMix [2021b]	PTQ / QAT	S	CNN	1K§	69.46§	-
	PSAQ-ViT [2022]	PTQ	S	ViT	32	71.56*	65.57†
	Genie [2023b]	PTQ	S, Q	CNN	1K	69.66	66.89
	SADAG [2024]	PTQ	S, Q	CNN	1K	69.72	67.10
	SMI [2024]	PTQ	S	ViT	32	70.13*	64.04†
	CLAMP-ViT [2024]	PTQ	S, Q	ViT	32	72.17*	69.93†

Taxonomy

- We summarize the key features of ZSQ methods
  - 2. Training Requirement

PTQ

QAT

Method	Training Requirement	Scope of Contribution	Architecture	# Images	Accuracy (FP = 71.47)	
					W4A4	W3A3
DFQ [2019]	PTQ	Q	CNN	0	55.78	-
SQuant [2022]	PTQ	Q	CNN	0	66.14	25.74
UDFC [2023]	PTQ	Q	CNN	0	63.49	-
GDFQ [2020]	QAT	S, Q	CNN	1.28M	60.60	20.23
ZAQ [2021]	QAT	S, Q	CNN	1.28M	52.64	-
ARC [2021]	QAT	S, Q	CNN	1.28M	61.32	23.37
Qimera [2021]	QAT	S, Q	CNN	1.28M	63.84	1.17
ARC + AIT [2022]	QAT	Q	CNN	1.28M	65.73	-
AdaSG [2023b]	QAT	S, Q	CNN	1.28M	66.50	37.04
AdaDFQ [2023a]	QAT	S, Q	CNN	1.28M	66.53	38.10
Causal-DFQ [2023]	QAT	S, Q	CNN	1.28M	68.11	-
RIS [2024]	QAT	S	CNN	1.28M	67.75	-
GenQ [2024b]	PTQ / QAT	S	CNN	1K§	69.77§	-
DeepInversion [2020]	QAT	S	CNN	32	70.27*	64.28†
IntraQ [2022]	QAT	S, Q	CNN	5.12K	66.47	45.51
HAST [2023]	QAT	S, Q	CNN	5.12K	66.91	51.15
TexQ [2023]	QAT	S, Q	CNN	5.12K	67.73	50.28
PLF [2024]	QAT	Q	CNN	5.12K	67.02	-
SynQ [2025b]	QAT	Q	CNN / ViT	5.12K	67.90	52.02
ZeroQ [2020]	PTQ	S, Q	CNN	1K	26.04	-
KW [2020]	PTQ	S, Q	CNN	1K	69.08	-
DSG [2021]	PTQ	S	CNN	1K	34.53	-
MixMix [2021b]	PTQ / QAT	S	CNN	1K§	69.46§	-
PSAQ-ViT [2022]	PTQ	S	ViT	32	71.56*	65.57†
Genie [2023b]	PTQ	S, Q	CNN	1K	69.66	66.89
SADAG [2024]	PTQ	S, Q	CNN	1K	69.72	67.10
SMI [2024]	PTQ	S	ViT	32	70.13*	64.04†
CLAMP-ViT [2024]	PTQ	S, Q	ViT	32	72.17*	69.93†

Taxonomy

- We summarize the key features of ZSQ methods
  - 3. Scope of Contribution

S: Data Synthesis

Q: Network Quantization

Method	Training Requirement	Scope of Contribution	Architecture	# Images	Accuracy (FP = 71.47)	
					W4A4	W3A3
DFQ [2019]	PTQ	Q	CNN	0	55.78	-
SQuant [2022]	PTQ	Q	CNN	0	66.14	25.74
UDFC [2023]	PTQ	Q	CNN	0	63.49	-
GDFQ [2020]	QAT	S, Q	CNN	1.28M	60.60	20.23
ZAQ [2021]	QAT	S, Q	CNN	1.28M	52.64	-
ARC [2021]	QAT	S, Q	CNN	1.28M	61.32	23.37
Qimera [2021]	QAT	S, Q	CNN	1.28M	63.84	1.17
ARC + AIT [2022]	QAT	Q	CNN	1.28M	65.73	-
AdaSG [2023b]	QAT	S, Q	CNN	1.28M	66.50	37.04
AdaDFQ [2023a]	QAT	S, Q	CNN	1.28M	66.53	38.10
Causal-DFQ [2023]	QAT	S, Q	CNN	1.28M	68.11	-
RIS [2024]	QAT	S	CNN	1.28M	67.75	-
GenQ [2024b]	PTQ / QAT	S	CNN	1K§	69.77§	-
DeepInversion [2020]	QAT	S	CNN	32	70.27*	64.28†
IntraQ [2022]	QAT	S, Q	CNN	5.12K	66.47	45.51
HAST [2023]	QAT	S, Q	CNN	5.12K	66.91	51.15
TexQ [2023]	QAT	S, Q	CNN	5.12K	67.73	50.28
PLF [2024]	QAT	Q	CNN	5.12K	67.02	-
SynQ [2025b]	QAT	Q	CNN / ViT	5.12K	67.90	52.02
ZeroQ [2020]	PTQ	S, Q	CNN	1K	26.04	-
KW [2020]	PTQ	S, Q	CNN	1K	69.08	-
DSG [2021]	PTQ	S	CNN	1K	34.53	-
MixMix [2021b]	PTQ / QAT	S	CNN	1K§	69.46§	-
PSAQ-ViT [2022]	PTQ	S	ViT	32	71.56*	65.57†
Genie [2023b]	PTQ	S, Q	CNN	1K	69.66	66.89
SADAG [2024]	PTQ	S, Q	CNN	1K	69.72	67.10
SMI [2024]	PTQ	S	ViT	32	70.13*	64.04†
CLAMP-ViT [2024]	PTQ	S, Q	ViT	32	72.17*	69.93†

Taxonomy

- We summarize the key features of ZSQ methods
  - 4. Architecture of the Target Network

CNN

ViT

Method	Training Requirement	Scope of Contribution	Architecture	# Images	Accuracy (FP = 71.47)	
					W4A4	W3A3
DFQ [2019]	PTQ	Q	CNN	0	55.78	-
SQuant [2022]	PTQ	Q	CNN	0	66.14	25.74
UDFC [2023]	PTQ	Q	CNN	0	63.49	-
GDFQ [2020]	QAT	S, Q	CNN	1.28M	60.60	20.23
ZAQ [2021]	QAT	S, Q	CNN	1.28M	52.64	-
ARC [2021]	QAT	S, Q	CNN	1.28M	61.32	23.37
Qimera [2021]	QAT	S, Q	CNN	1.28M	63.84	1.17
ARC + AIT [2022]	QAT	Q	CNN	1.28M	65.73	-
AdaSG [2023b]	QAT	S, Q	CNN	1.28M	66.50	37.04
AdaDFQ [2023a]	QAT	S, Q	CNN	1.28M	66.53	38.10
Causal-DFQ [2023]	QAT	S, Q	CNN	1.28M	68.11	-
RIS [2024]	QAT	S	CNN	1.28M	67.75	-
GenQ [2024b]	PTQ / QAT	S	CNN	1K§	69.77§	-
DeepInversion [2020]	QAT	S	CNN	32	70.27*	64.28†
IntraQ [2022]	QAT	S, Q	CNN	5.12K	66.47	45.51
HAST [2023]	QAT	S, Q	CNN	5.12K	66.91	51.15
TexQ [2023]	QAT	S, Q	CNN	5.12K	67.73	50.28
PLF [2024]	QAT	Q	CNN	5.12K	67.02	-
SynQ [2025b]	QAT	Q	CNN / ViT	5.12K	67.90	52.02
ZeroQ [2020]	PTQ	S, Q	CNN	1K	26.04	-
KW [2020]	PTQ	S, Q	CNN	1K	69.08	-
DSG [2021]	PTQ	S	CNN	1K	34.53	-
MixMix [2021b]	PTQ / QAT	S	CNN	1K§	69.46§	-
PSAQ-ViT [2022]	PTQ	S	ViT	32	71.56*	65.57†
Genie [2023b]	PTQ	S, Q	CNN	1K	69.66	66.89
SADAG [2024]	PTQ	S, Q	CNN	1K	69.72	67.10
SMI [2024]	PTQ	S	ViT	32	70.13*	64.04†
CLAMP-ViT [2024]	PTQ	S, Q	ViT	32	72.17*	69.93†

Taxonomy

- We summarize the key features of ZSQ methods
  - 5. Performance with the Number of Synthetic Images

Method	Training Requirement	Scope of Contribution	Architecture	# Images	Accuracy (FP = 71.47)	
					W4A4	W3A3
DFQ [2019]	PTQ	Q	CNN	0	55.78	-
SQuant [2022]	PTQ	Q	CNN	0	66.14	25.74
UDFC [2023]	PTQ	Q	CNN	0	63.49	-
GDFQ [2020]	QAT	S, Q	CNN	1.28M	60.60	20.23
ZAQ [2021]	QAT	S, Q	CNN	1.28M	52.64	-
ARC [2021]	QAT	S, Q	CNN	1.28M	61.32	23.37
Qimera [2021]	QAT	S, Q	CNN	1.28M	63.84	1.17
ARC + AIT [2022]	QAT	Q	CNN	1.28M	65.73	-
AdaSG [2023b]	QAT	S, Q	CNN	1.28M	66.50	37.04
AdaDFQ [2023a]	QAT	S, Q	CNN	1.28M	66.53	38.10
Causal-DFQ [2023]	QAT	S, Q	CNN	1.28M	68.11	-
RIS [2024]	QAT	S	CNN	1.28M	67.75	-
GenQ [2024b]	PTQ / QAT	S	CNN	1K§	69.77§	-
DeepInversion [2020]	QAT	S	CNN	32	70.27*	64.28†
IntraQ [2022]	QAT	S, Q	CNN	5.12K	66.47	45.51
HAST [2023]	QAT	S, Q	CNN	5.12K	66.91	51.15
TexQ [2023]	QAT	S, Q	CNN	5.12K	67.73	50.28
PLF [2024]	QAT	Q	CNN	5.12K	67.02	-
SynQ [2025b]	QAT	Q	CNN / ViT	5.12K	67.90	52.02
ZeroQ [2020]	PTQ	S, Q	CNN	1K	26.04	-
KW [2020]	PTQ	S, Q	CNN	1K	69.08	-
DSG [2021]	PTQ	S	CNN	1K	34.53	-
MixMix [2021b]	PTQ / QAT	S	CNN	1K§	69.46§	-
PSAQ-ViT [2022]	PTQ	S	ViT	32	71.56*	65.57†
Genie [2023b]	PTQ	S, Q	CNN	1K	69.66	66.89
SADAG [2024]	PTQ	S, Q	CNN	1K	69.72	67.10
SMI [2024]	PTQ	S	ViT	32	70.13*	64.04†
CLAMP-ViT [2024]	PTQ	S, Q	ViT	32	72.17*	69.93†

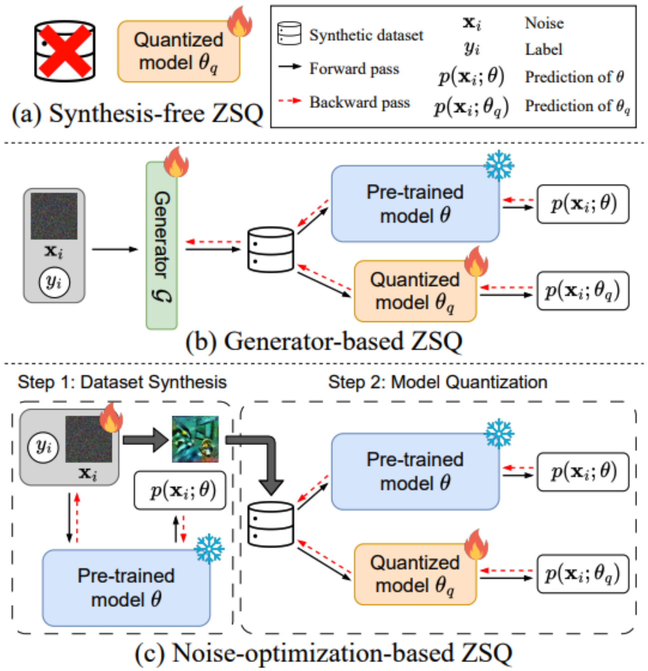
Classification accuracy of a ResNet-18 model trained on ImageNet
* W8A8 on CIFAR-100
† W8A8/W4A8 of DeiT-T

Outline

- Introduction
- Problem Formulation
- Categorization
- **ZSQ Algorithms**
- Future Research Directions
- Conclusion

Taxonomy
Revisited

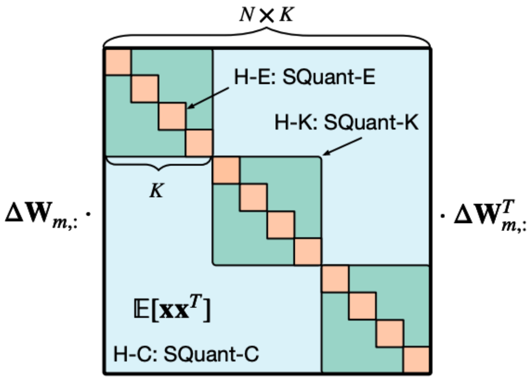
- We categorize ZSQ algorithms based on their data generation approach as:
  - **Synthesis-free ZSQ**
    - Quantize models without generating any synthetic data
  - **Generator-based ZSQ**
    - Train an additional generator $\mathcal{G}$ to produce synthetic data
  - **Noise-optimization-based ZSQ**
    - Directly optimize noise inputs to make synthetic data



ZSQ Algorithms

Synthesis-free ZSQ

- Synthesis-free ZSQ methods compress a pre-trained model without generating any synthetic data
  - They leverage structural properties or theoretical foundations to mitigate performance degradation
  - **Representative method**. SQuant [ICLR 2022]
    - Evaluating the quantization error with the Hessian of each layer
    - Diagonal Hessian approximation for efficient computation



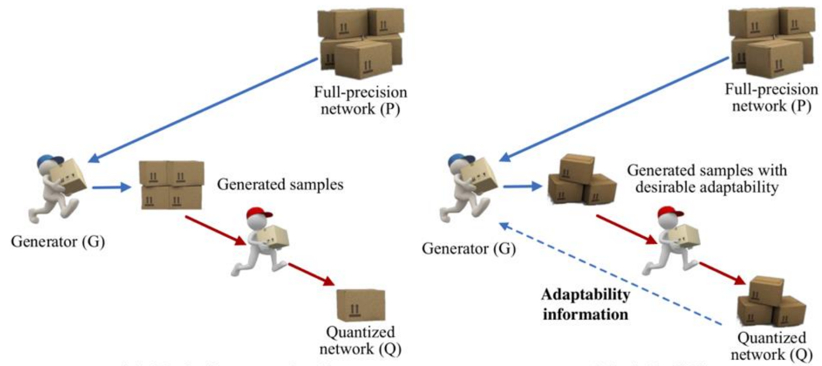
C. Guo et al., "SQuant: On-the-Fly Data-Free Quantization via Diagonal Hessian Approximation", ICLR 2022

ZSQ Algorithms

Generator-based ZSQ

- Generator-based ZSQ employs an independent generator model $\mathcal{G}$ to produce synthetic datasets
  - Generally, they train a GAN-based generator from scratch
  - **Representative method**. AdaSG [AAAI 2023]
    - Reformulating ZSQ into a zero-sum game between the generator $\mathcal{G}$ and the quantized model θ_q on reward $\mathcal{R}(\cdot)$
    - Adversarial sample generation

$$\min_{\theta_q} \max_{\mathcal{G}} \mathcal{R}\left(\mathcal{G}, \theta_q\right)$$



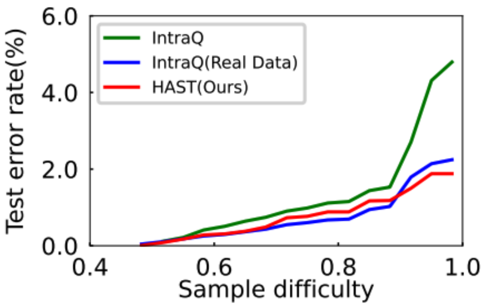
(a) Existing methods (b) AdaSG

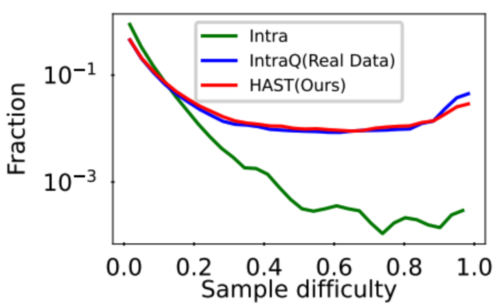
B. Qian et al., "Rethinking Data-Free Quantization as a Zero-Sum Game", AAAI 2023

ZSQ Algorithms

Noise-optimization-based ZSQ

- Noise-optimization-based ZSQ directly optimizes noise to generate the dataset from iterative updates
  - They universally follow a two-step scheme:
    - 1. Dataset synthesis → 2. Model quantization
  - **Representative method**. HAST [CVPR 2023]
    - Previous methods perform poorly on difficult images, since their synthetic datasets lack challenging samples
    - Produce more samples difficult for both original / quantized models





H. Li et al., "Hard Sample Matters a Lot in Zero-Shot Quantization", CVPR 2023

Outline

- Introduction
- Problem Formulation
- Categorization
- ZSQ Algorithms
- **Future Research Directions**
- Conclusion

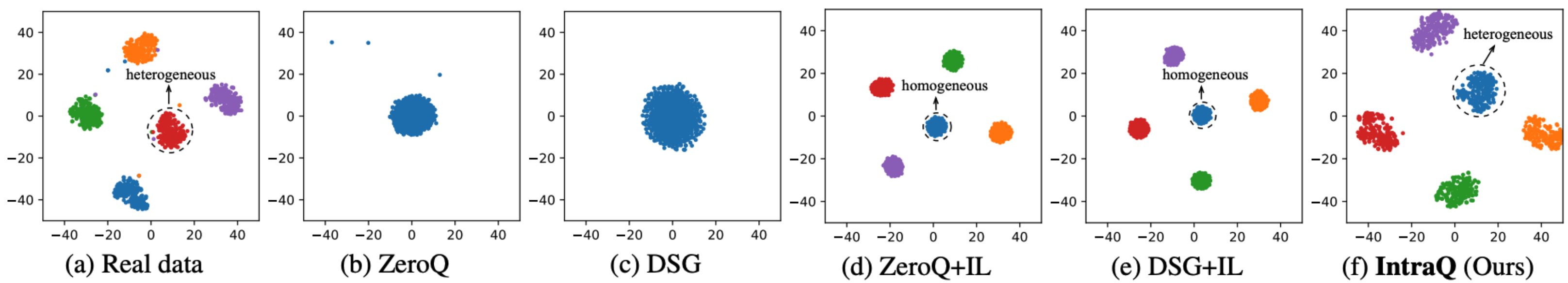
Future Directions

- Research questions remain open for exploration
  - Synthetic datasets
    - 1. More principled analysis on synthetic datasets
    - 2. Faster generation of synthetic datasets
  - Theory
    - 3. Theoretical exploration of ZSQ
  - Problem setting
    - 4. Broader application to various tasks and domains
    - 5. Diverse problem settings
    - 6. Combining other model compression techniques
  - Evaluation
    - 7. Evaluating practical impact on real-world scenarios

Future Directions

Synthetic Datasets

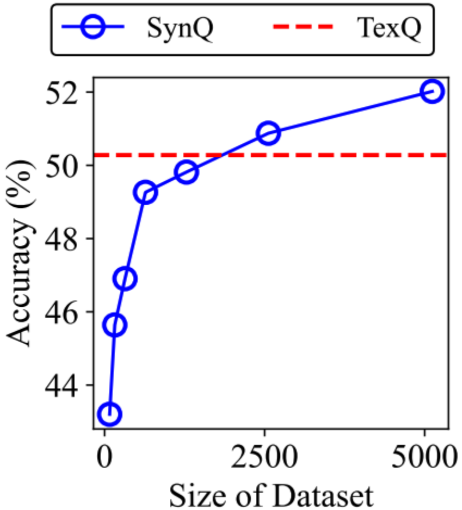
- 1. More principled analysis on synthetic datasets
  - Most studies fix individual features instead of investigating their root causes
  - Deeper analysis may yield fundamental improvements



Y. Zhong et al., "IntraQ: Learning Synthetic Images with Intra-Class Heterogeneity for Zero-Shot Network Quantization", CVPR 2022

Future Directions
Synthetic Datasets

- 2. Faster generation of synthetic datasets
  - Increasing the size of synthetic datasets enhances the performance of quantized models
  - How can we reduce the generation time?
    - 1 to 4 GPU hours required to generate 5k 224×224 images

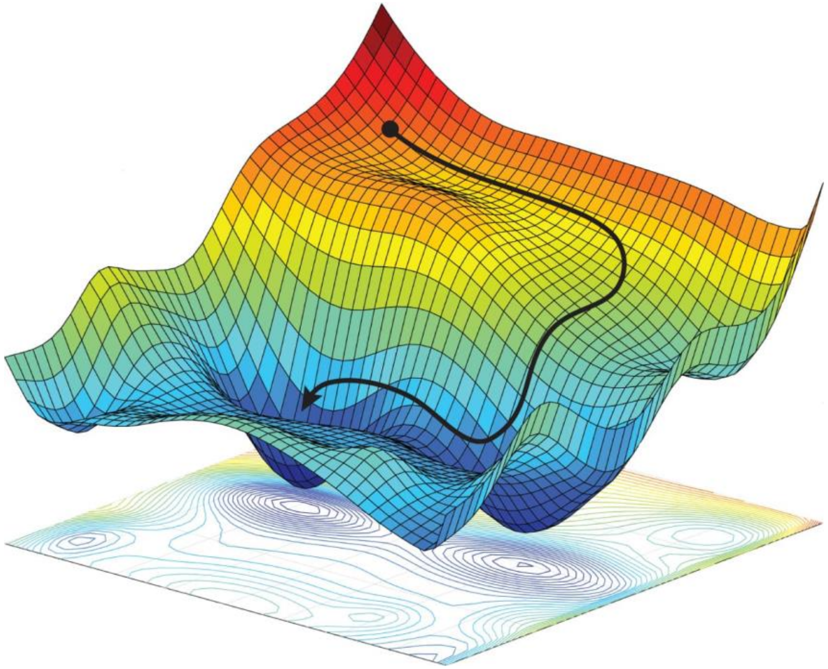


M. Kim et al., “SynQ: Accurate Zero-shot Quantization by Synthesis-aware Fine-tuning”, ICLR 2025

Future Directions

Theory

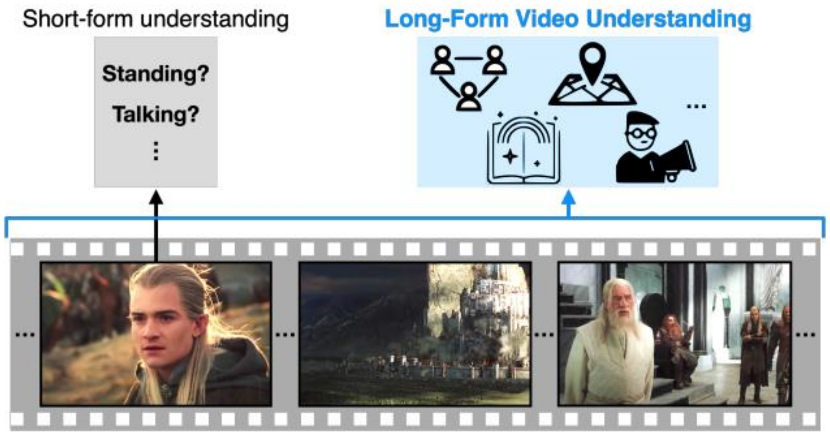
- 3. Theoretical exploration of ZSQ
  - ZSQ lacks formal understanding such as convergence guarantees or error bounds
    - Mathematical principles would guide towards robust algorithms

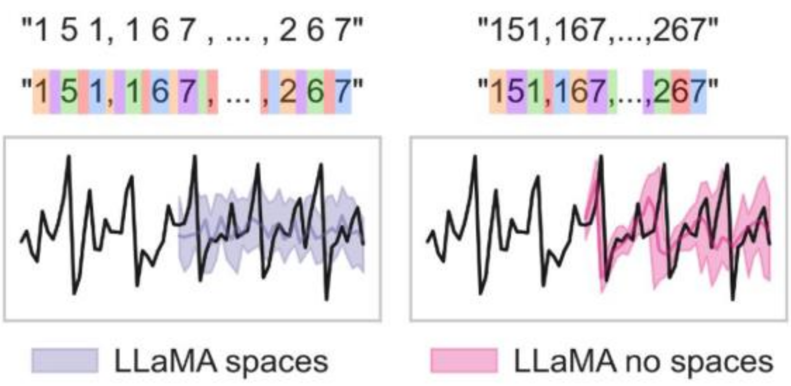


Future Directions

Problem Setting

- 4. Broader application to various tasks and domains
  - Most research sets task $\mathcal{T}$ as *image classification*, with a few work on *object detection*
  - Extending research to various tasks is crucial
    - Other vision tasks
    - Language, multi-variate, graph domains

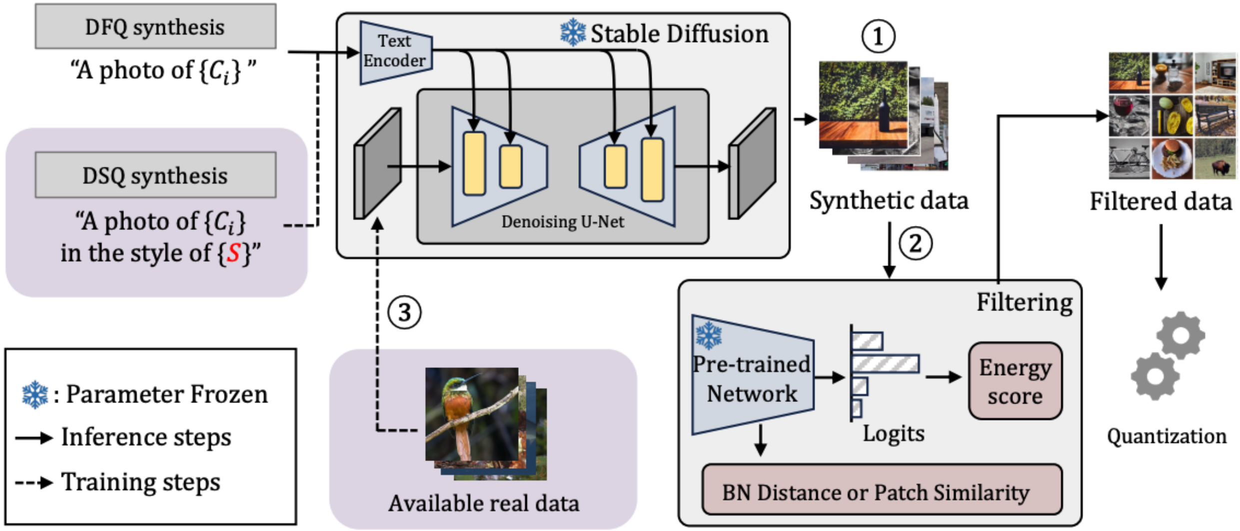




C. Wu et al., "Towards Long-Form Video Understanding", CVPR 2021
N. Gruver et al., "Large Language Models Are Zero-Shot Time Series Forecasters", NeurIPS 2023

Future Directions
Problem Setting

- 5. Diverse problem settings
  - Extending ZSQ to real-time quantization and edge-device deployments
    - e.g., few-instance quantization (1 to 10 real images), leveraging a pre-trained diffusion model for dataset synthesis

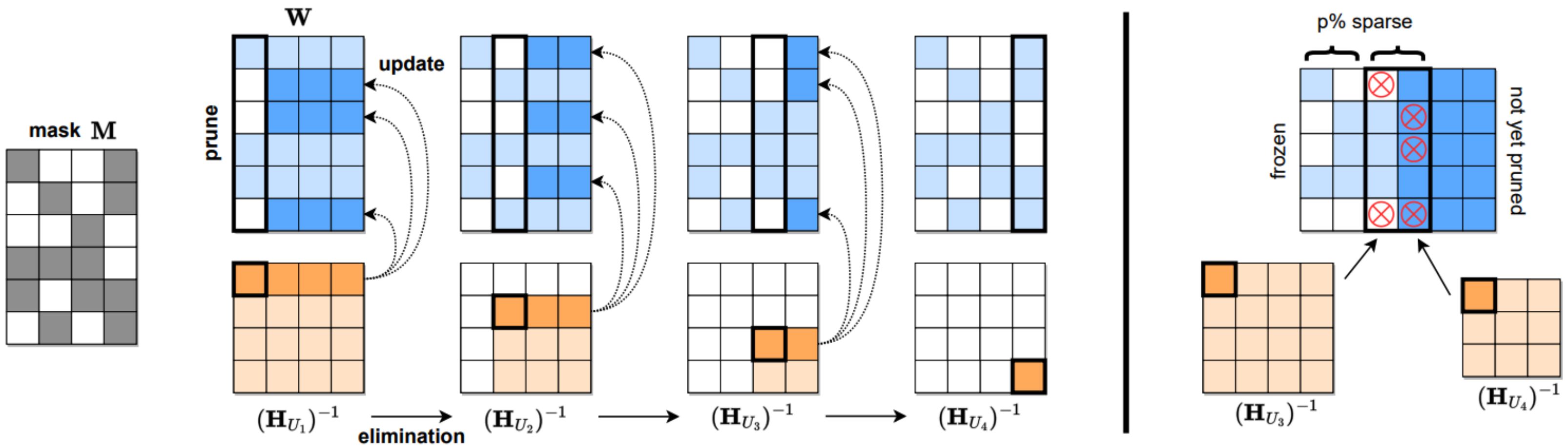


Y. Li et al., "GenQ: Quantization in Low Data Regimes with Generative Synthetic Data", ECCV 2024

Future Directions

Problem Setting

- 6. Combining other model compression techniques
  - Current ZSQ algorithms achieve competitive results in 4-bit regime, but struggle in 3-bit or lower-bits
  - Integrating with other methods would help to achieve a higher compression rate while maintaining accuracy
    - e.g., pruning, weight sharing, low-rank approximation

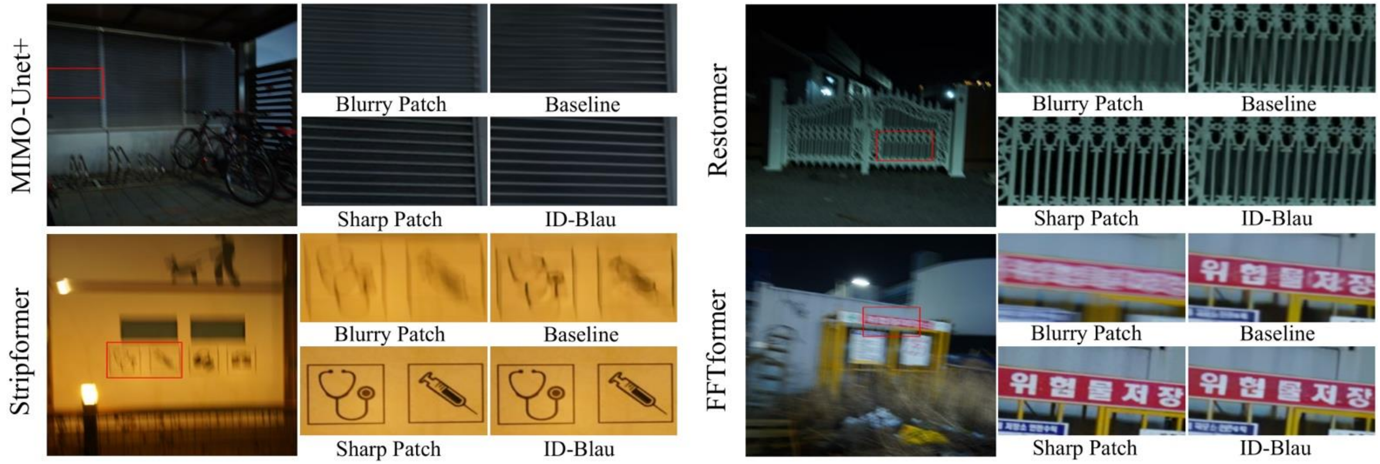


E. Frantar et al., "SparseGPT: Massive Language Models Can Be Accurately Pruned in One-Shot", ICML 2023

Future Directions

Evaluation

- 7. Evaluating practical impact on real-world scenarios
  - The importance of ZSQ lies in its applications for handling real-world scenarios with limited data
  - However, current ZSQ methods present experimental results solely on benchmark datasets and models



J.-H. Wu et al., “ID-Blau: Image Deblurring by Implicit Diffusion-based reBLurring AUgmentation”, CVPR 2024

Outline

- Introduction
- Problem Formulation
- Categorization
- ZSQ Algorithms
- Future Research Directions
- **Conclusion**

Conclusion

- We provide a comprehensive survey of ZSQ
  - ZSQ enables model compression without access to real data
- **Main Challenges**
  - **Knowledge transfer from the pre-trained model**
  - **Discrepancy between real and synthetic datasets**
  - **Diversity of the problem setting**
- Future work aims to improve synthetic data, theory, problem setting, and practical evaluation

Thank you !

Minjun Kim (minjun.kim@snu.ac.kr)

Paper   **GitHub**