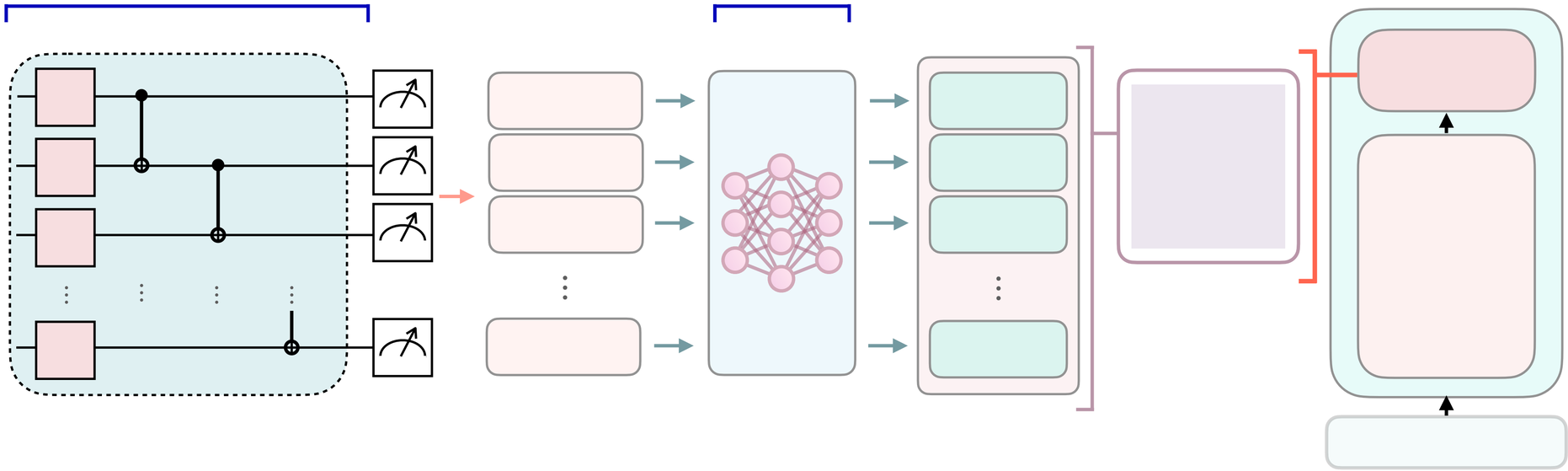


A Quantum Circuit-Based Compression Perspective for Parameter-Efficient Learning

Chen-Yu Liu[1,2], Chao-Han Huck Yang[3], Hsi-Sheng Goan[1], Min-Hsiu Hsieh[2]

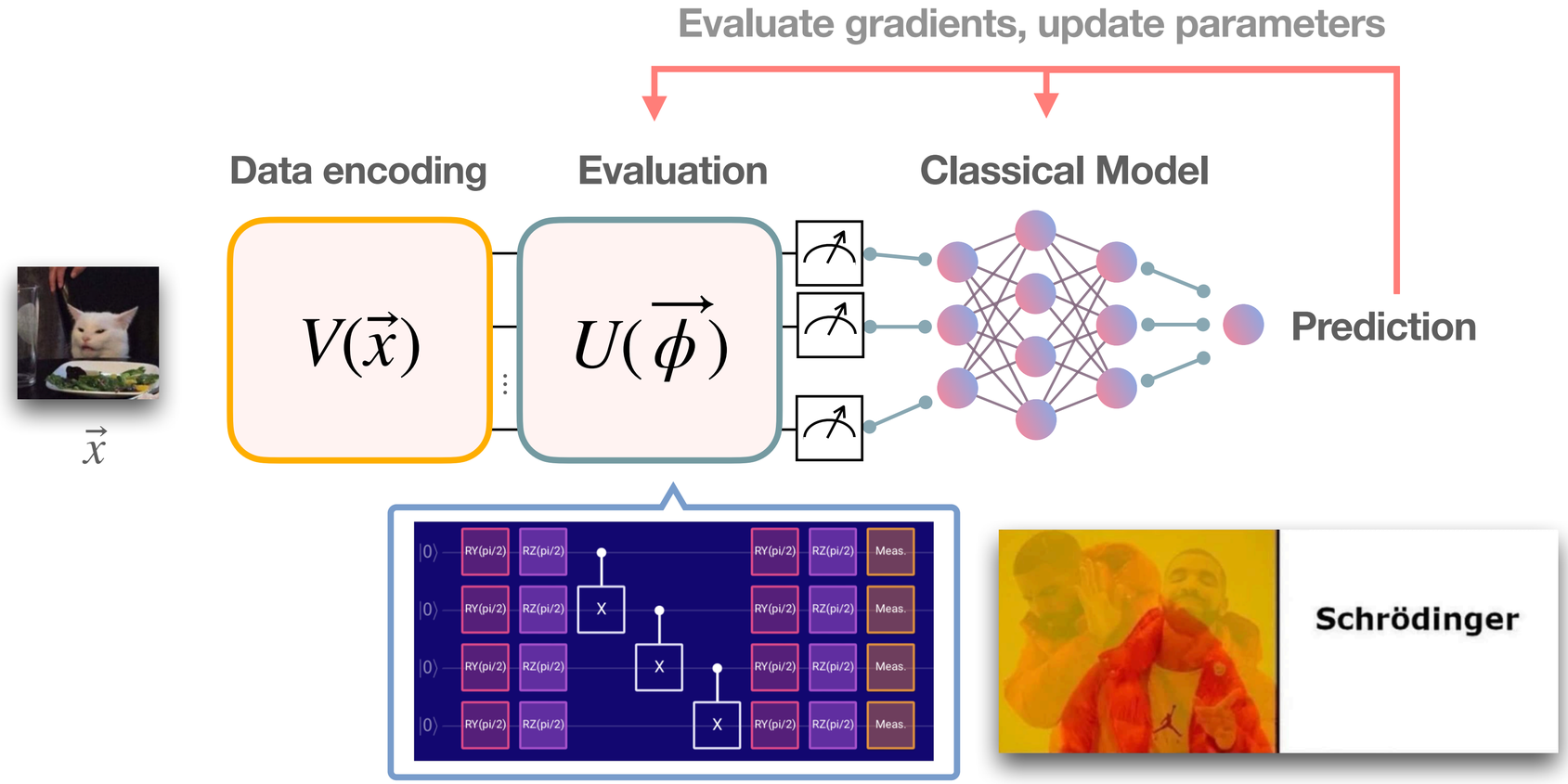
[1]National Taiwan University

[2] Hon Hai Quantum Computing Research Center

[3]Georgia Institute of Technology

Hybrid Quantum-classical architecture

- Effectiveness/challenge of data encoding
- Quantum hardware requirement during inference



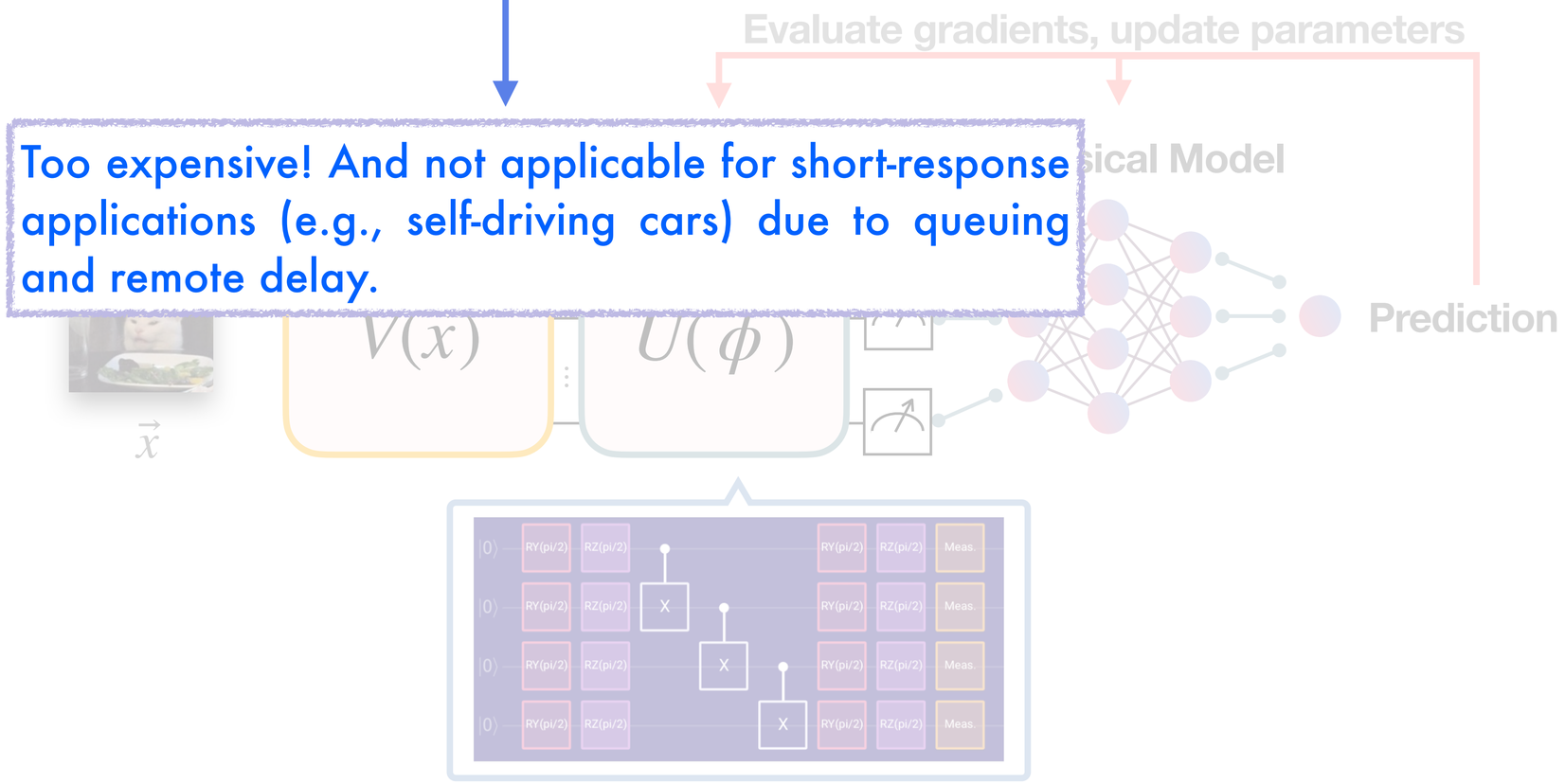
Quantum Machine Learning

Hybrid Quantum-classical architecture

- Effectiveness/challenge of data encoding
- Quantum hardware requirement during inference

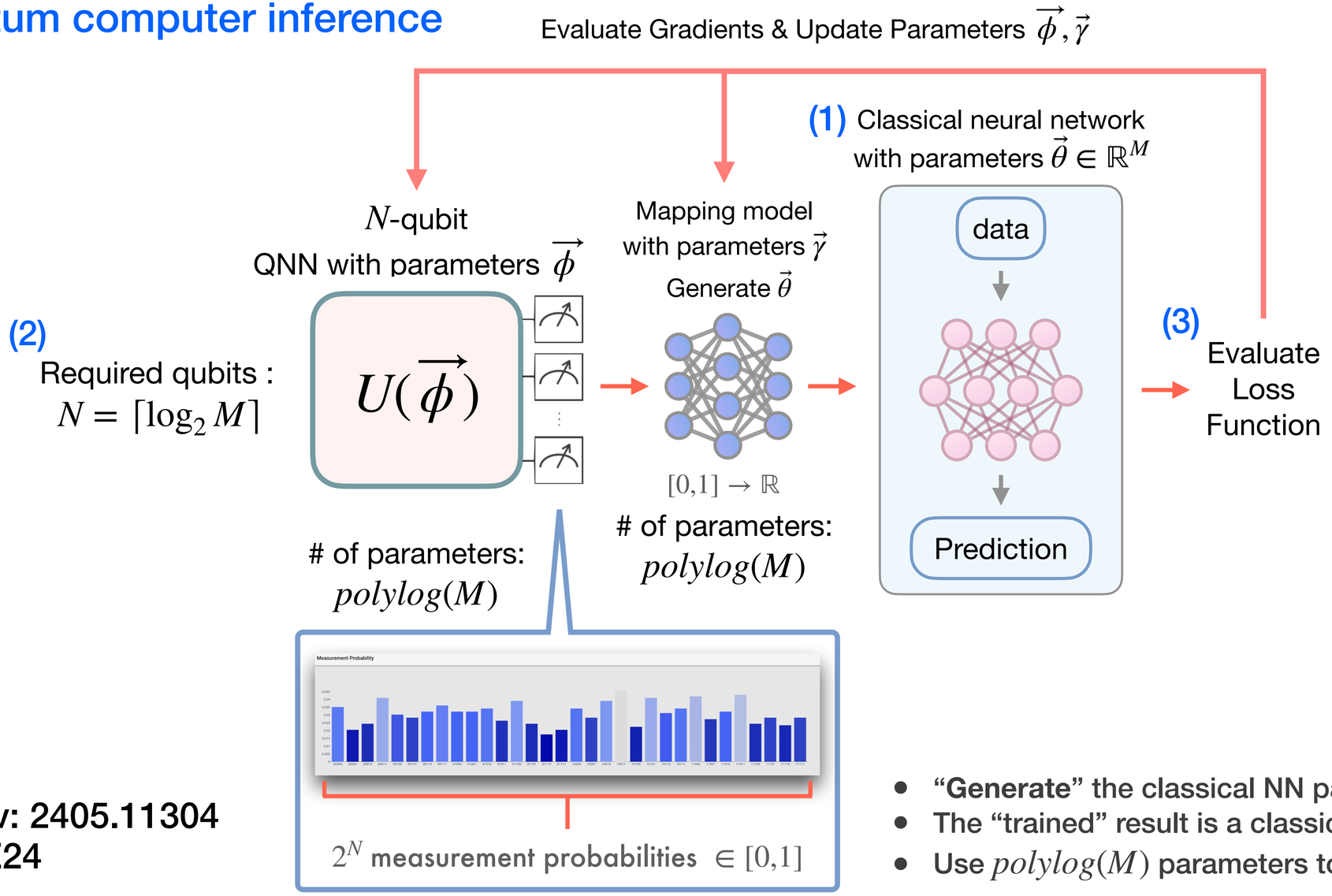
Qubit count, circuit depth, and normalization to rotational angles, …

Too expensive! And not applicable for short-response applications (e.g., self-driving cars) due to queuing and remote delay.



Quantum-Train

Beyond the data-encoding circuit and quantum computer inference



Ex.
$N = 10,\ 2^N = 1024$
~ 100 training parameters to control QNN

- "**Generate**" the classical NN parameters by Quantum NN
- The "trained" result is a classical NN
- Use $polylog(M)$ parameters to train M parameters

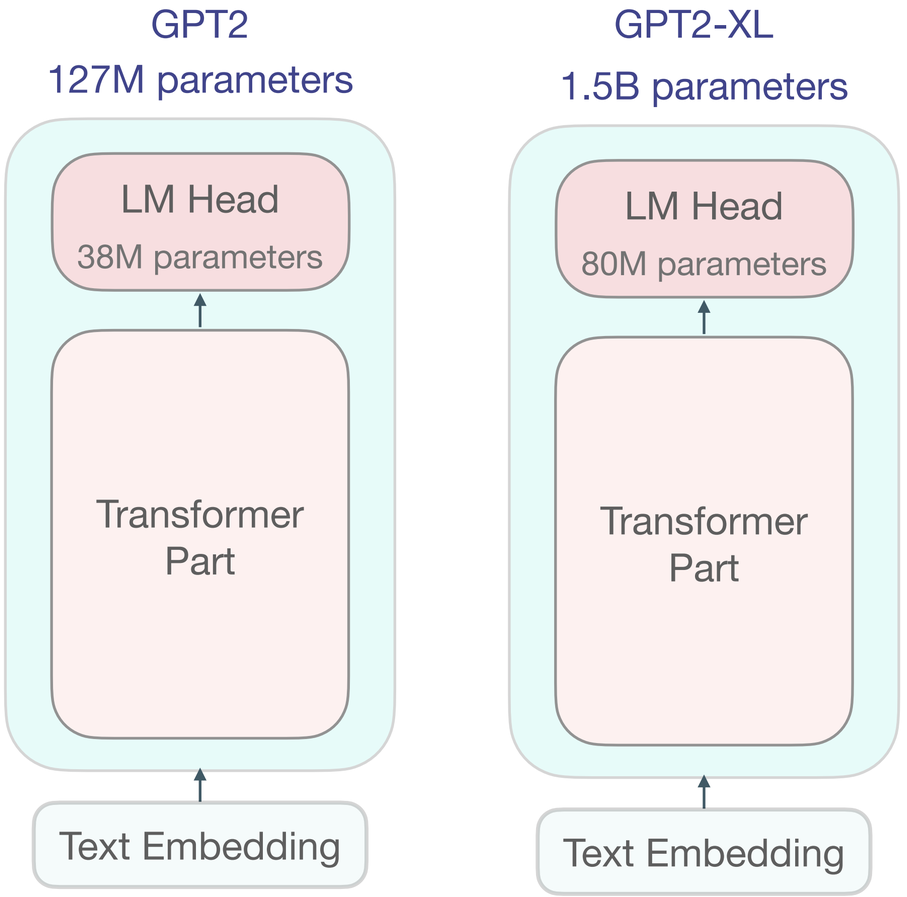
Liu, et. al. arXiv: 2405.11304
Liu, et. al. QCE24

Large Language Models (LLMs)

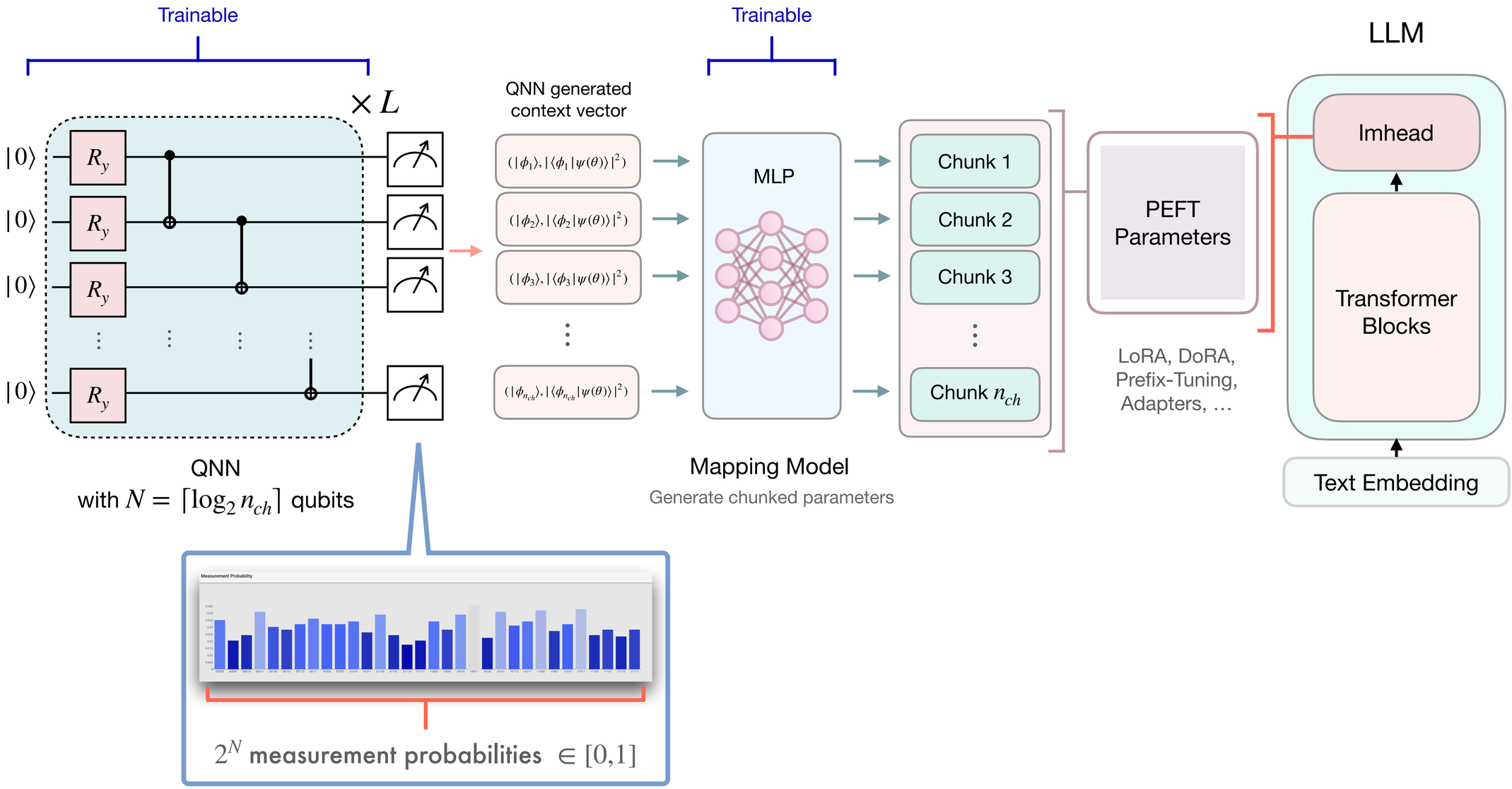
Transformer-based LLMs

- Pre-trained on massive datasets for tasks like Q&A, summarization, and translation.
- Highly flexible but **challenging** to train and fine-tune due to billions of parameters.

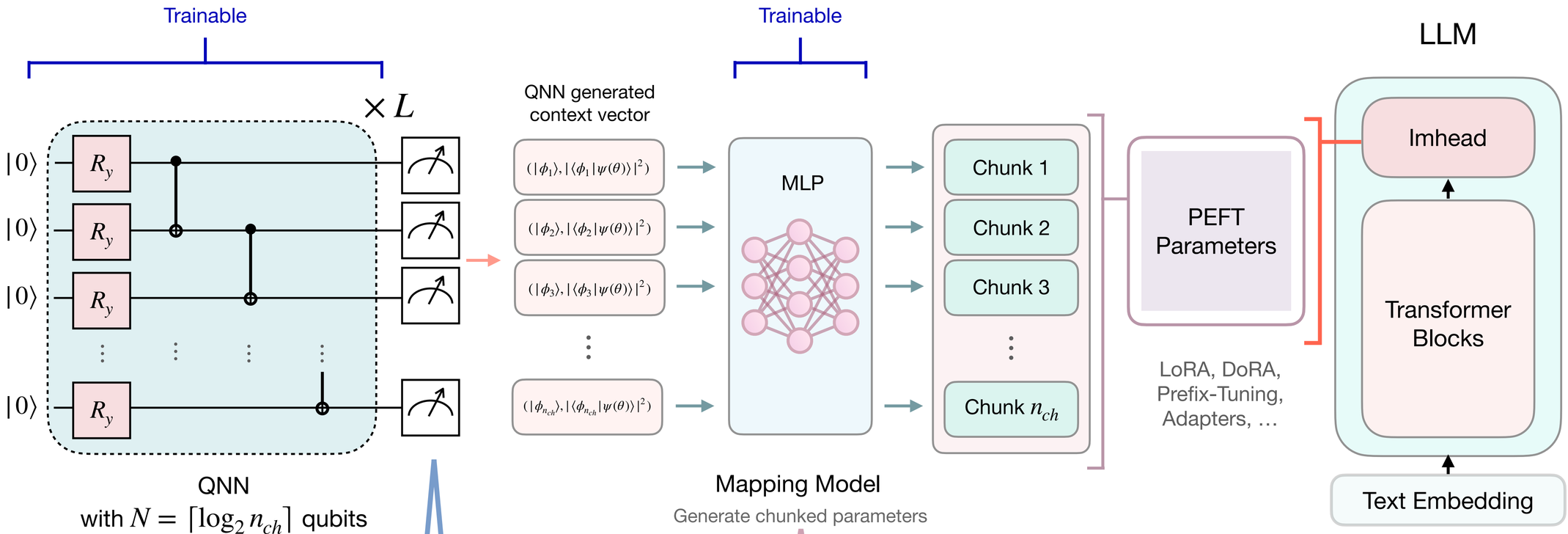
OpenAI	GPT3	175B parameters
	GPT4	1.76T parameters
Meta	Llama-3	8B parameters
	Llama-3	70B parameters
Google	Gemma	2B parameters
	Gemma	7B parameters

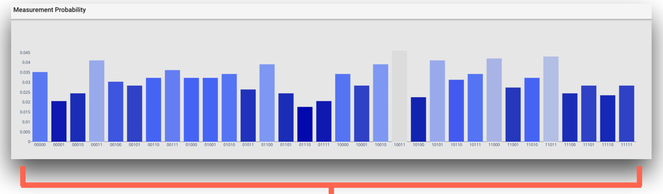


Quantum Parameter Adaptation



Quantum Parameter Adaptation





2^N measurement probabilities $\in [0,1]$

Input: basis information & prob.
Output: A chunk (batch) of parameters

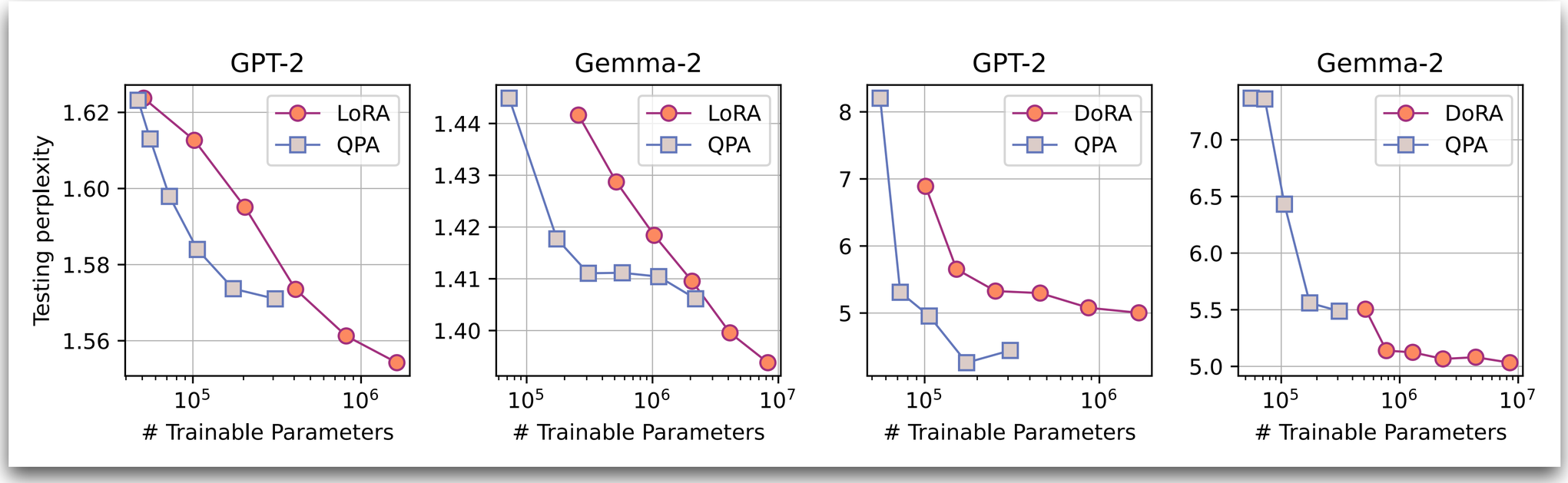
The qubit count is reduced from

$$\lceil \log_2 M \rceil \text{ to } \lceil \log_2 \lceil \frac{M}{n_{mlp}} \rceil \rceil = \lceil \log_2 n_{ch} \rceil$$

Batched Parameter Generation
Liu, et. al.
ICTC 2024 Best AI paper award

Quantum Parameter Adaptation

- Low-rank adaptation methods with QPA.

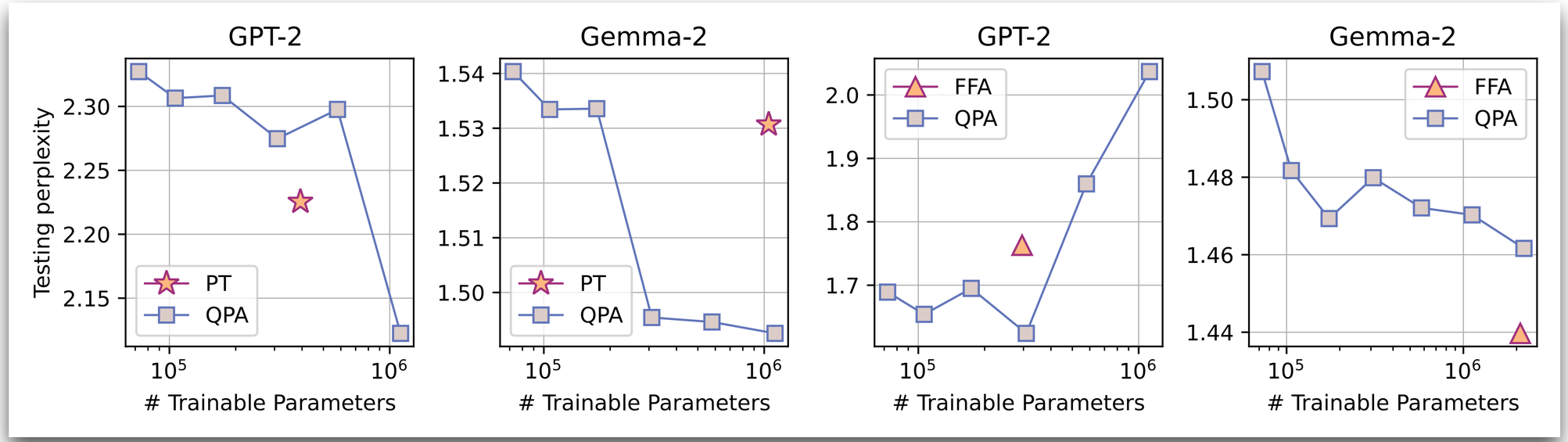


- Testing perplexity of GPT-2 (80M) and Gemma-2 (2B) models compared to the number of trainable parameters for LoRA, DoRA, and QPA on the WikiText-2 dataset.

* On ideal simulator
* All experiments were conducted on NVIDIA V100S and NVIDIA H100 GPUs.

Quantum Parameter Adaptation

- QPA on Prefix-Tuning and Feed-forward adapter.

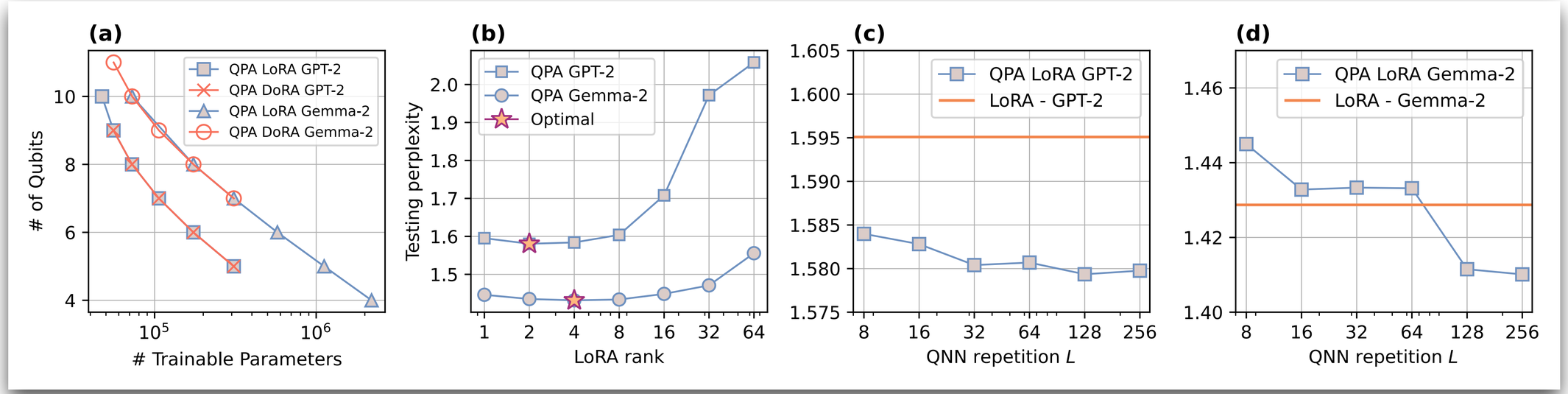


- Testing perplexity of GPT-2 and Gemma-2 models compared to the number of trainable parameters for prefix-tuning (PT), feed-forward adapter (FFA), and QPA on the WikiText-2 dataset.

- On ideal simulator
- All experiments were conducted on NVIDIA V100S and NVIDIA H100 GPUs.

Quantum Parameter Adaptation

- Effects of QPA settings



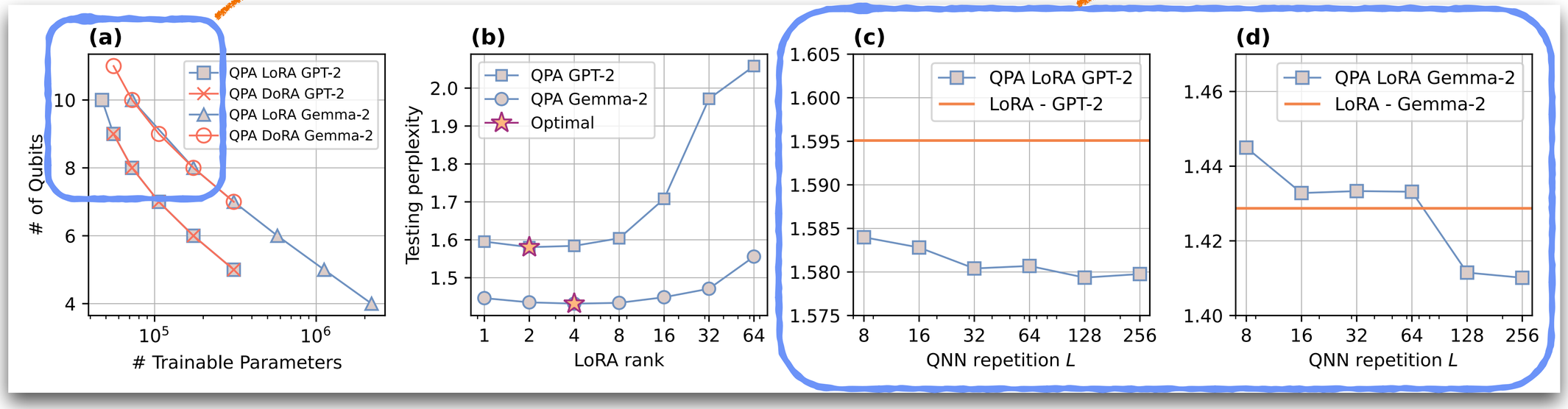
- **(a)** Qubit usage versus the number of trainable parameters for QPA applied to LoRA and DoRA on GPT-2 and Gemma-2 models. **(b)** The relationship between testing perplexity and LoRA rank for QPA applied to GPT-2 and Gemma-2. **(c)** and **(d)** Testing perplexity depending on the QNN repetition L for QPA applied to LoRA on GPT-2 and Gemma-2.

* On ideal simulator

Quantum Parameter Adaptation

Reduced qubit count due to batched parameter generation (arXiv:2409.02763), Low-rank adaptation, …

More expressive circuits have better QPA performance

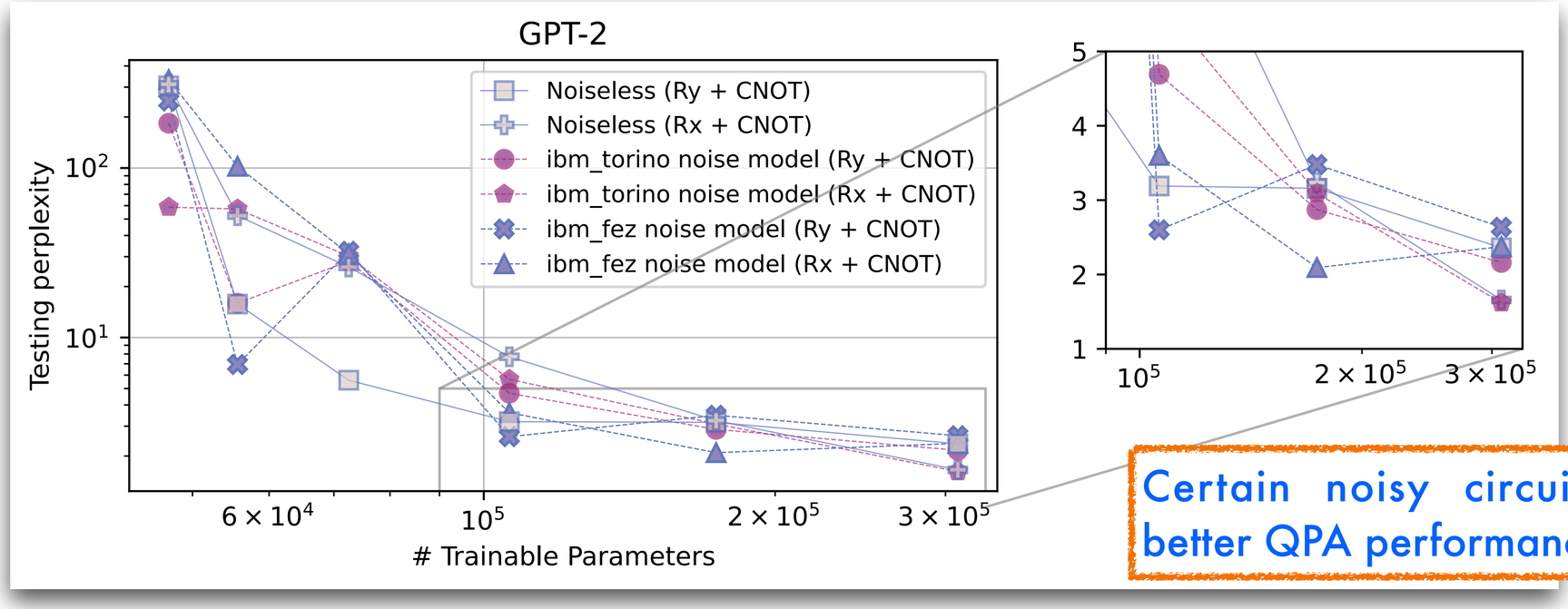


- **(a)** Qubit usage versus the number of trainable parameters for QPA applied to LoRA and DoRA on GPT-2 and Gemma-2 models. **(b)** The relationship between testing perplexity and LoRA rank for QPA applied to GPT-2 and Gemma-2. **(c)** and **(d)** Testing perplexity depending on the QNN repetition L for QPA applied to LoRA on GPT-2 and Gemma-2.

- On ideal simulator

Quantum Parameter Adaptation

- Effects of quantum noise model from IBM quantum computers



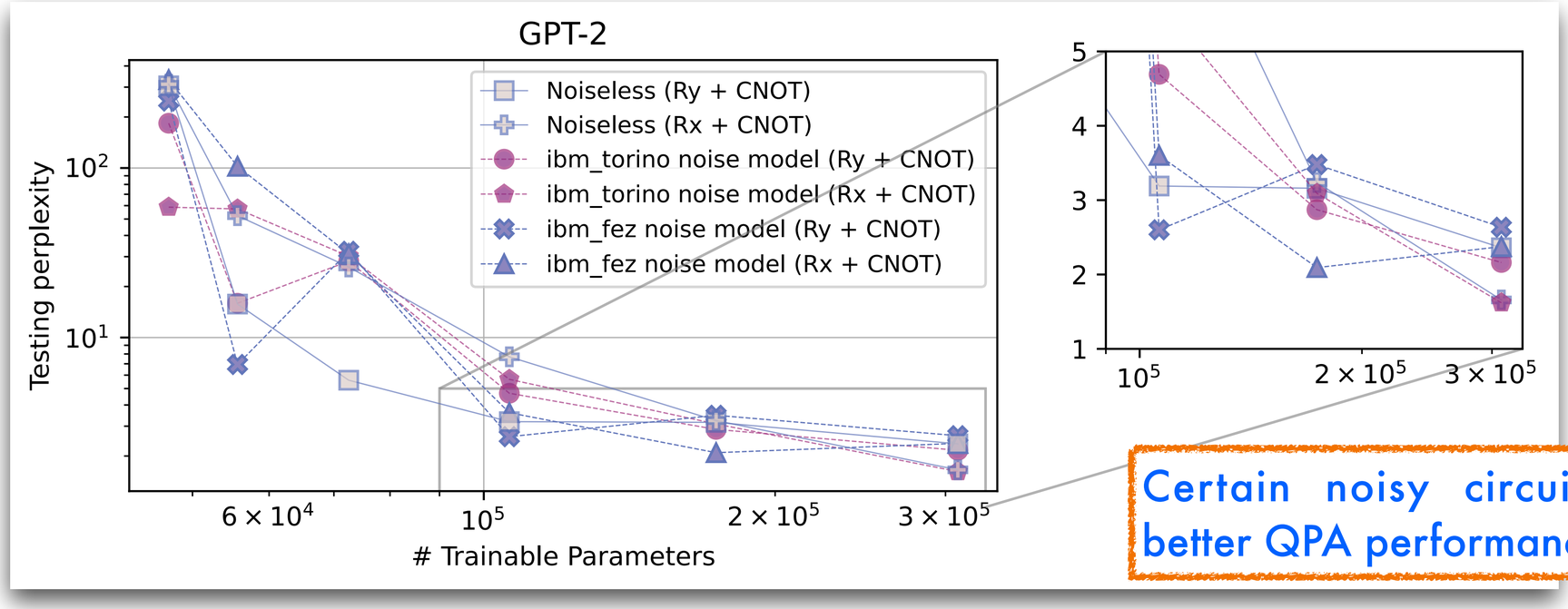
Certain noisy circuits exhibit better QPA performance.

- Testing perplexity of GPT-2 versus the number of trainable parameters on different noise setting with RY + CNOT and RX+ CNOT ansatz. The comparison includes LoRA and QPA applied at LoRA rank (r = 4), where n_{shot} is fixed at $n_{shot} = 20 \times 2^N$.

- On noisy simulator

Quantum Parameter Adaptation

- Effects of quantum noise model from IBM quantum computers



Certain noisy circuits exhibit better QPA performance.

NoisyTune: A Little Noise Can Help You Finetune Pretrained Language Models Better

Chuhan Wu[†] **Fangzhao Wu**[‡*] **Tao Qi**[†] **Yongfeng Huang**[†]

[†]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

[‡]Microsoft Research Asia, Beijing 100080, China

(ACL2022)

Summary

Generate PEFT parameters by QNN and MLP for LLMs

- The **first example** of using QML to fine-tune classical LLMs with improved performance.
- Further reduction of training parameters based on classical PEFT methods.
- No issues with data encoding.
- No quantum hardware required during inference.

On-going & Future work

- A generative method to mitigate the issue of finite measurement shots.
- Characterizing the effect of real quantum computer noise on QPA.