GEORGE MASON UNIVERSITY

Characterizing and Enhancing the Resilience of Generative Large Language Models against Soft Errors

Lishan Yang

Assistant Professor

Department of Computer Science, George Mason University

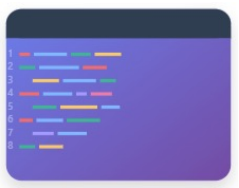
In collaboration with Yu Sun (GMU), Zhu Zhu (GMU), Cherish Mulpuru (GMU), Zachary Coalson (OSU), Shiyang Chen (Rutgers), Hang Liu (Rutgers), Zhao Zhang (Rutgers), Sanghyun Hong (OSU), Roberto Gioiosa (PNNL), Bo Fang (UTA)

Reliability Issue of LLMs

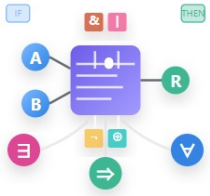
LLM: Large Language Model

Question Answering

Translation



Coding



Logic Reasoning

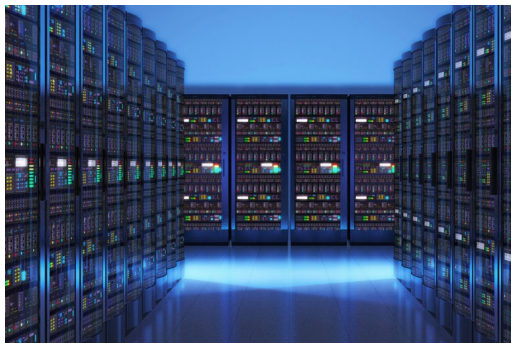
ChatGPT

Claude

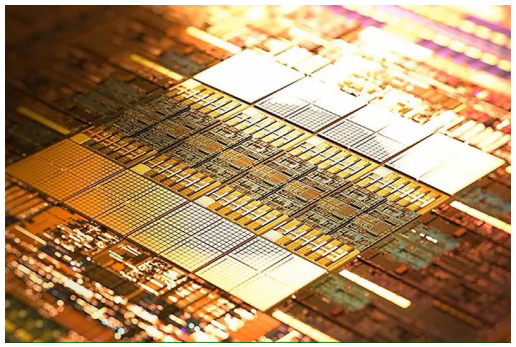
ChatGPT daily:

> 1 billion requests

> 100 billion words



Large scale

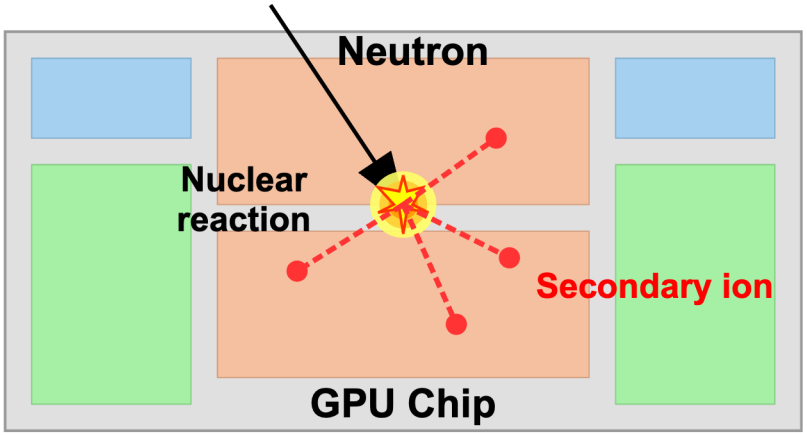


Dense chip

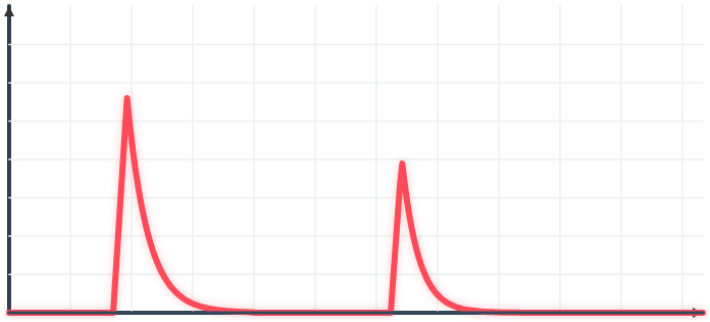
Zhou, Zixuan, et al. "A survey on efficient inference for large language models." *arXiv preprint arXiv:2404.14294* (2024).

Soft Errors

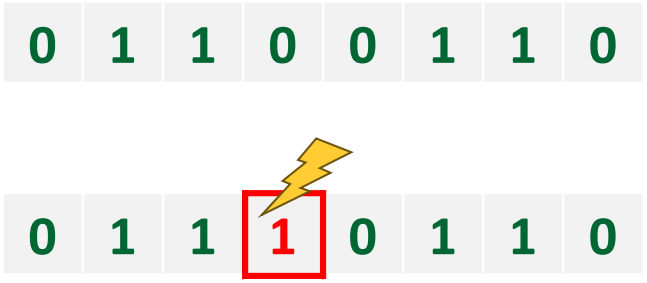
Cosmic ray



Neutrons hit silicon die



Current pulses



Bit flips

Output Type: Subtly Wrong vs Distorted

Question: Joe's bag of Halloween candy has 25 chocolate bars and 80 candied apples. Each chocolate bar weighs twice as each candied apple. If each chocolate bar weighs 40g, how much does Joe's bag of candy weigh?
Reference: 2600
Baseline: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of chocolate bars: 25 bars * 40g / bar = 1000g. Then find the total weight of the candied apples: 80 apples * 20g / apple = 1600g. Then add the weight of chocolate bars and candied apples to find the total weight: 1000g + 1600g = 2600g. Therefore, the answer is 2600.

Output Type: Subtly Wrong vs Distorted

Question: Joe's bag of Halloween candy has 25 chocolate bars and 80 candied apples. Each chocolate bar weighs twice as each candied apple. If each chocolate bar weighs 40g, how much does Joe's bag of candy weigh?
Reference: 2600
Baseline: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of chocolate bars: 25 bars * 40g / bar = 1000g. Then find the total weight of the candied apples: 80 apples * 20g / apple = 1600g. Then add the weight of chocolate bars and candied apples to find the total weight: 1000g + 1600g = 2600g. Therefore, the answer is 2600.
Subtly wrong output: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of the apple chocolate bars: 5 bars * 40g / bar = 200g. Then find the total weight of the apples: 80 apples * 20g / apple = 1600g. Then add the weight to find the total weight: 200g + 1600g = 1800g. The answer is 1800.

Output Type: Subtly Wrong vs Distorted

Question: Joe's bag of Halloween candy has 25 chocolate bars and 80 candied apples. Each chocolate bar weighs twice as each candied apple. If each chocolate bar weighs 40g, how much does Joe's bag of candy weigh?

Reference: 2600

Baseline: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of chocolate bars: 25 bars * 40g / bar = 1000g. Then find the total weight of the candied apples: 80 apples * 20g / apple = 1600g. Then add the weight of chocolate bars and candied apples to find the total weight: 1000g + 1600g = 2600g. Therefore, the answer is 2600.

Subtly wrong output: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of the apple chocolate bars: 5 bars * 40g / bar = 200g. Then find the total weight of the apples: 80 apples * 20g / apple = 1600g. Then add the weight to find the total weight: 200g + 1600g = 1800g. The answer is 1800.

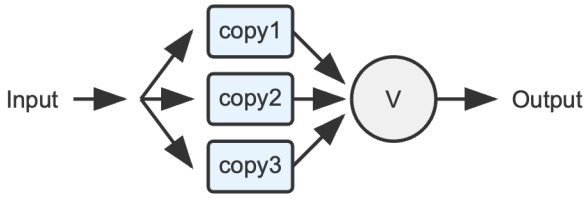
Distorted output: !!
!!

Existing Protection

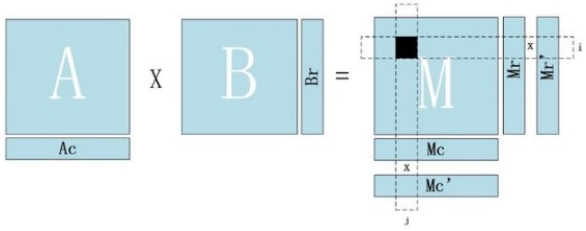
TMR: Triple Modular Redundancy

ABFT: Algorithm-Based Fault Tolerance

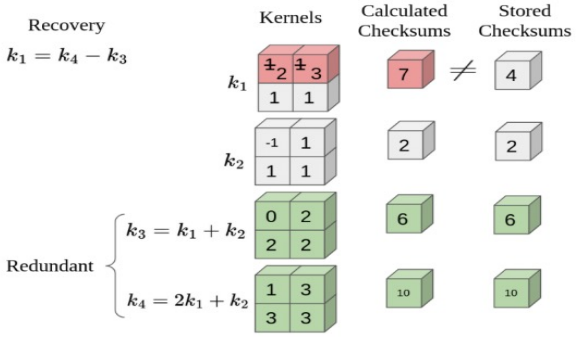
TMR



ABFT



Structural Coding

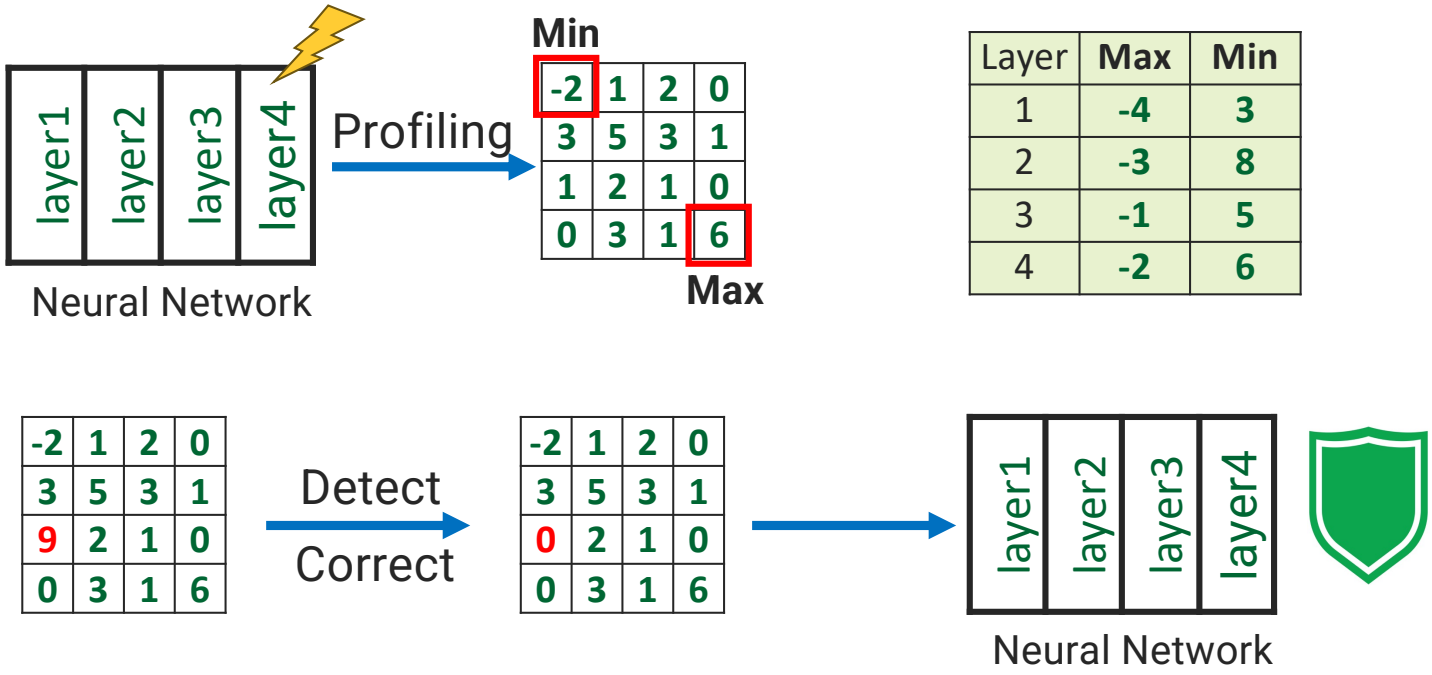


High overhead for LLMs

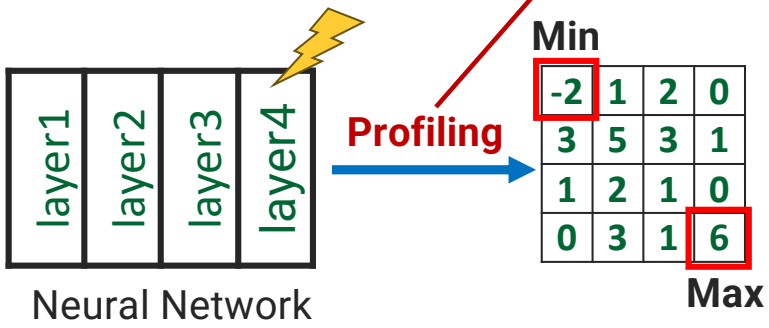
Asgari Khoshouyeh, Ali, et al. "Structural coding: A low-cost scheme to protect cnns from large-granularity memory faults." SC '23

Chen, Zitao, Guanpeng Li, and Karthik Pattabiraman. "A low-cost fault corrector for deep neural networks through range restriction." DSN '21

Range Restriction

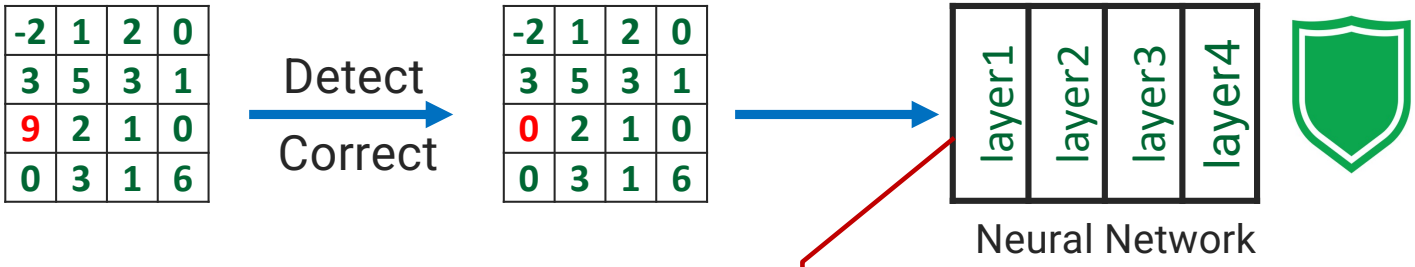


Range Restriction



Heavy Overhead

Layer	Max	Min
1	-4	3
2	-3	8
3	-1	5
4	-2	6



Which Layers?

FT2: First-Token-Inspired Online Fault Tolerance on Critical Layers for Generative Large Language Models

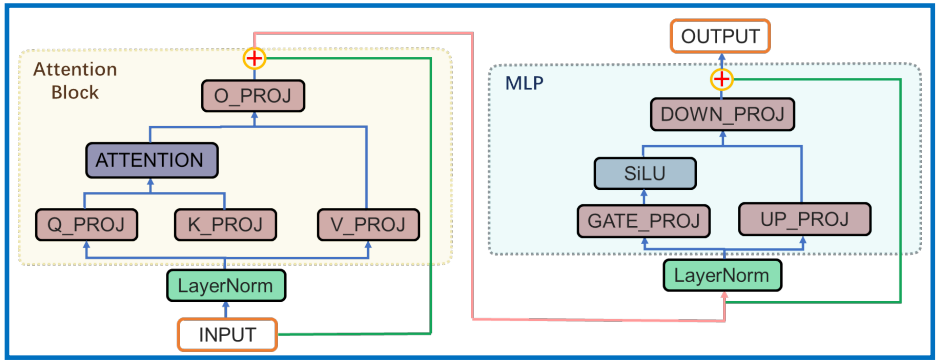
HPDC 2025

Yu Sun[§], Zhu Zhu[§], Cherish Mulpuru[§], Roberto Gioiosa*,
Zhao Zhang[‡], Bo Fang*, and **Lishan Yang**[§]

[§]George Mason University
[‡]Rutgers University
*Pacific Northwest National Lab

FT2: Identify Critical Layers

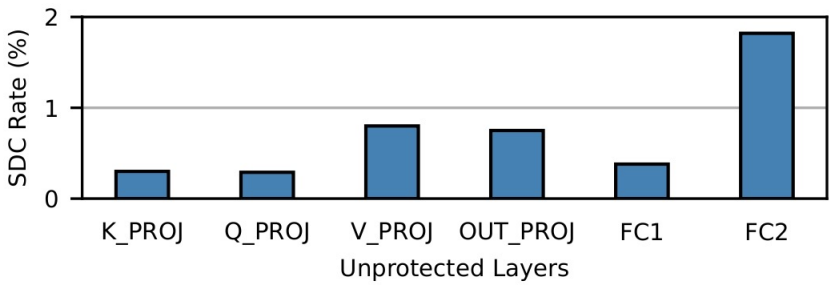
Critical layers: Poor reliability if not specifically protected



× **N**

Protect all → High overhead

- Fault injection experiments



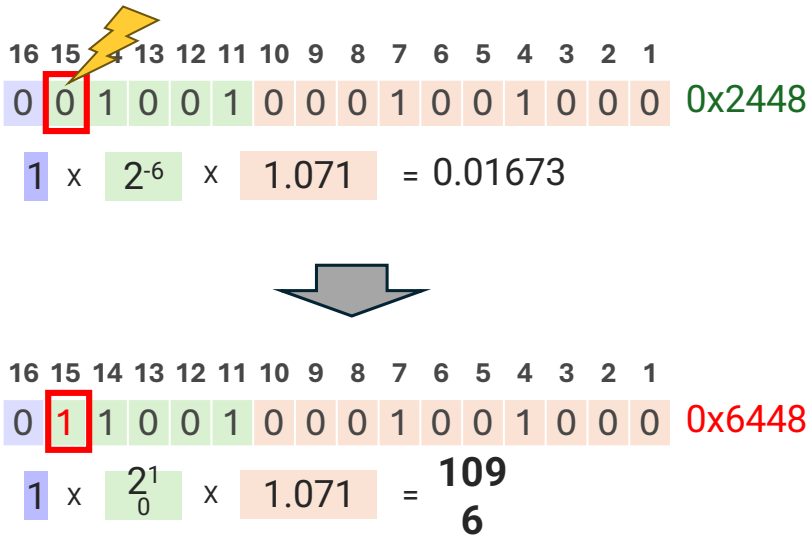
Model: GPTJ-6B
Dataset: SQuAD2.0
Fault model: 1-bit

Which layers are critical?
Why critical?

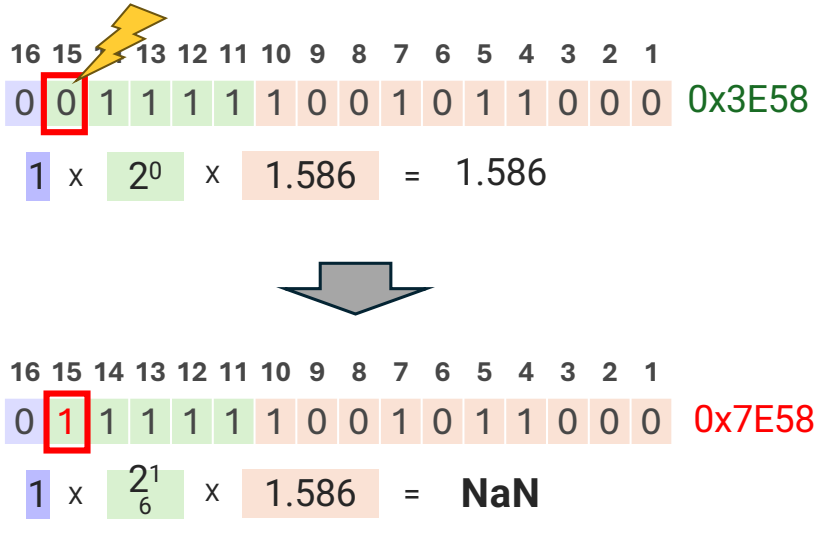
FT2: Identify Critical Layers

There are two types of abnormal values caused by bit flips

① Extreme value



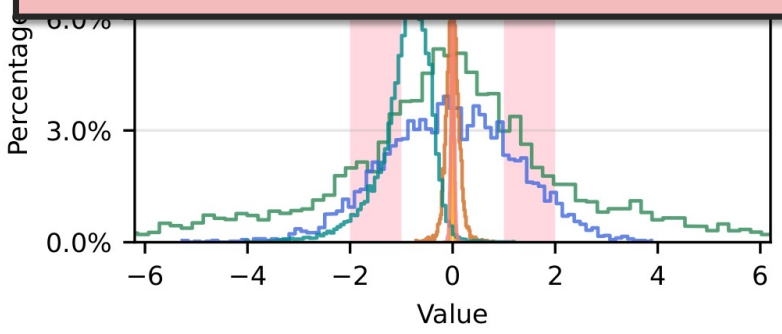
② Not a number (NaN)

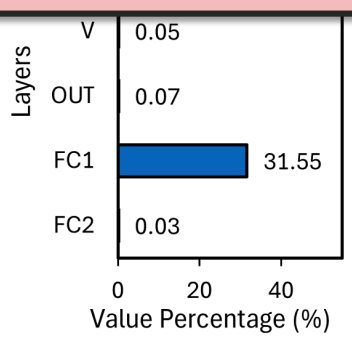


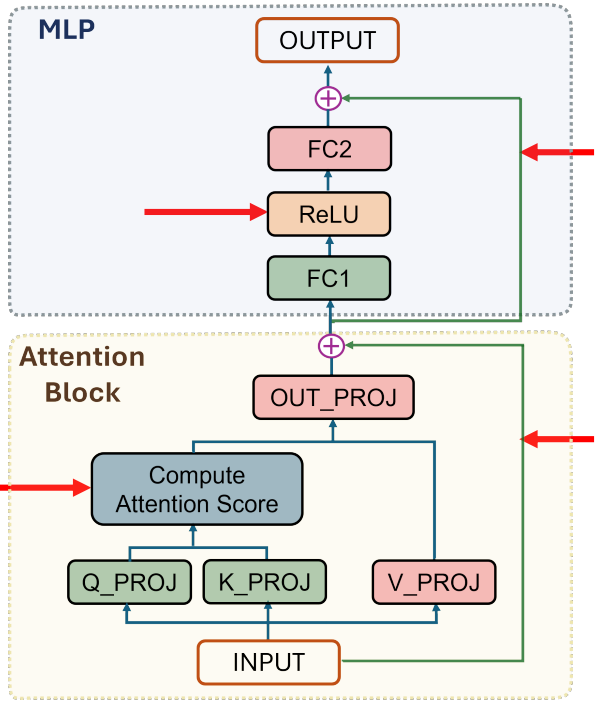
FT2: Identify Critical Layers

- Different neuron value distributions
  - Scaling operations

A heuristic: *a layer is critical if no scaling operation or activation layer is behind.*

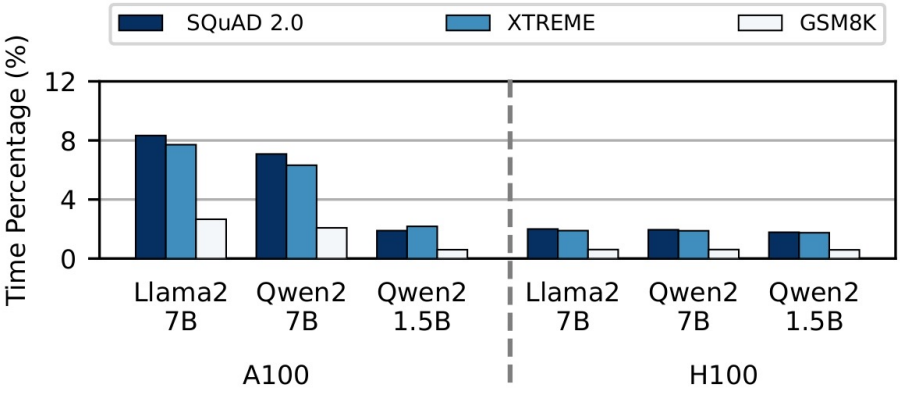




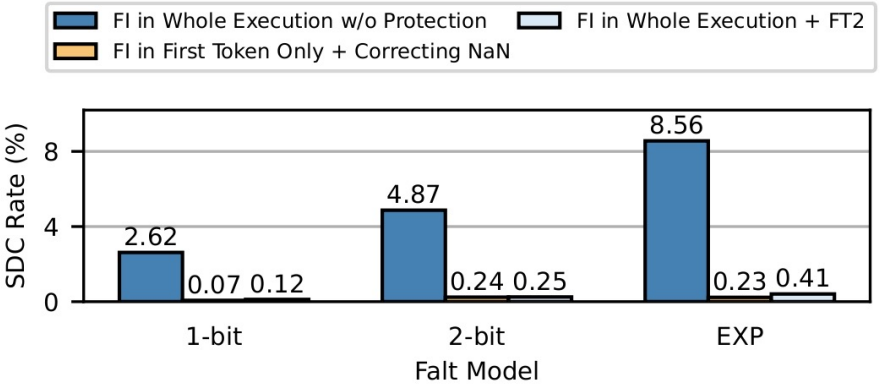


FT2: Obtain Bounds Online

- Obtain bounds from the first token generation
  - Input → ***Longer***
  - Information → ***More***
  - Bounds → ***More accurate***
- The impact of not protecting this process is negligible



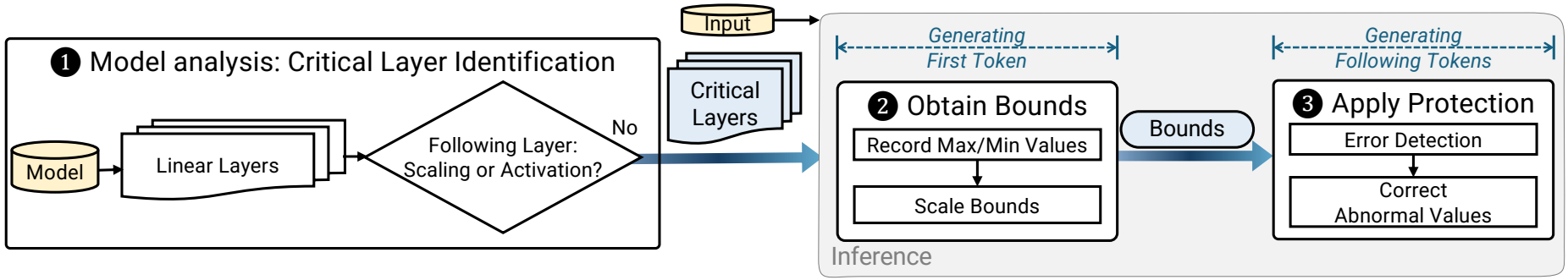
The percentage of execution time is low



The resilience is high

FT2: First-Token-Inspired Online Fault Tolerance on Critical Layers

Our Methodology: FT2



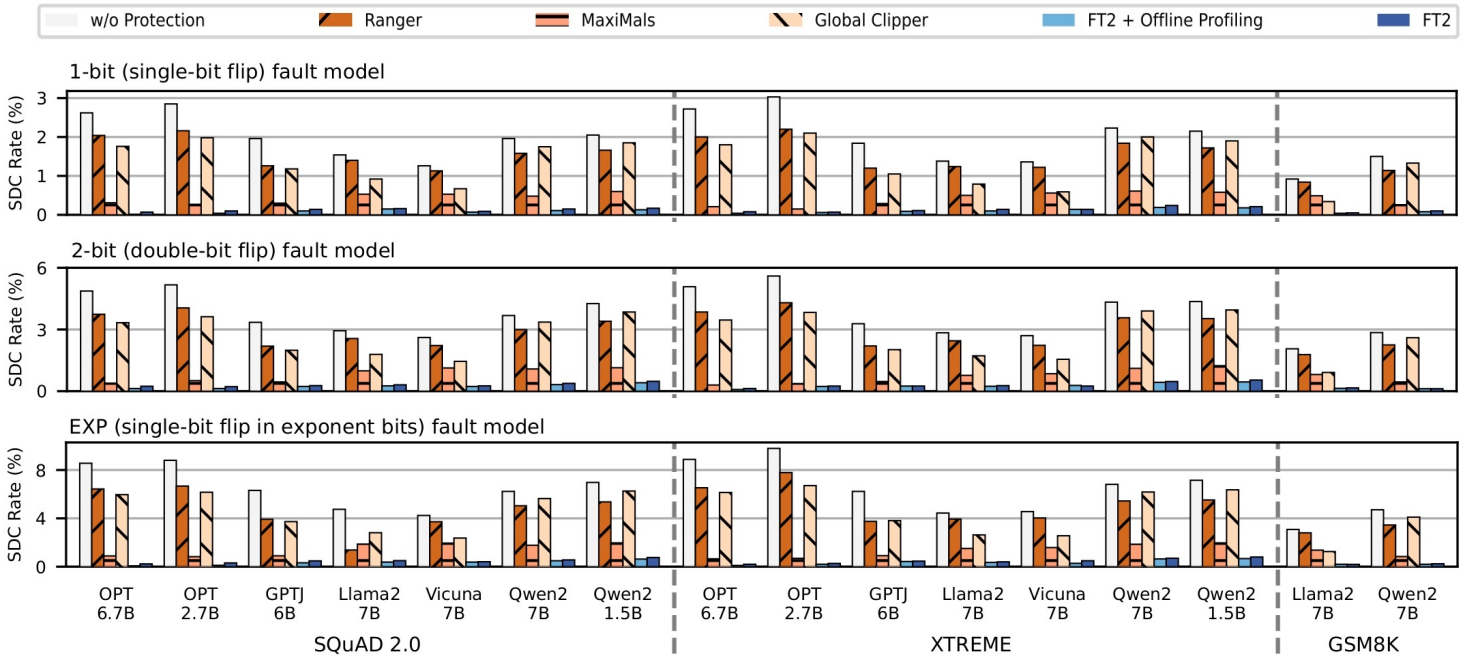
- *Better Protection*
- *No profiling required*

Evaluation: Experimental Set-up

Model Name	# of Parameters	Task Type
OPT-6.7B	6.66B	QA
OPT-2.7B	2.65B	QA
GPTJ-6B	6.05B	QA
Llama2-7B	6.74B	QA/MATH
Vicuna-7B	6.74B	QA
Qwen2-7B	7.62B	QA/MATH
Qwen2-1.5B	1.54B	QA

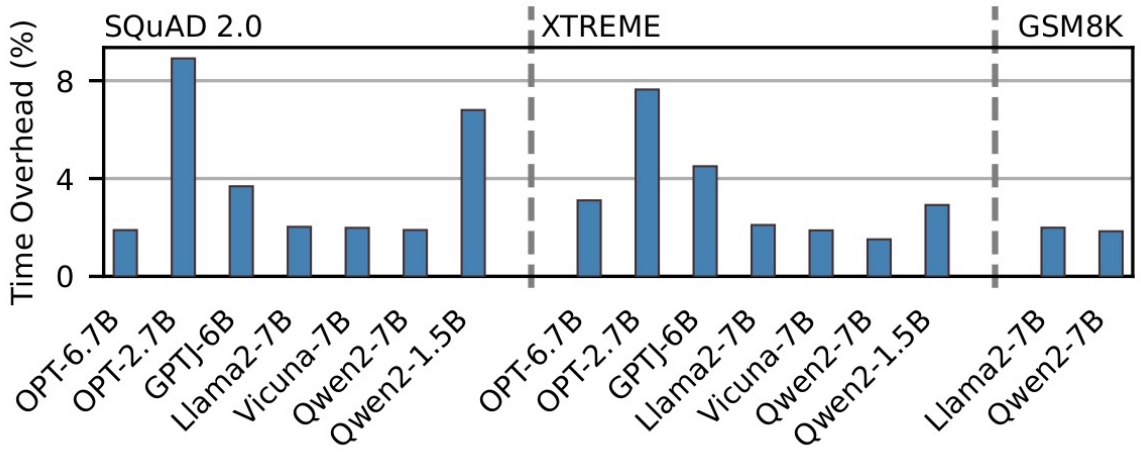
- ***7 models covering 2 architectures***
- ***3 datasets from 2 tasks***
- ***Datatype***: FP16 and FP32
- ***2 GPU configurations***: NVDIA A100 and H100 GPU
- ***3 fault models***: 1-bit, 2-bit and EXP

Overall Evaluation



- *FT2 outperforms all baselines among all models, datasets, and fault models*
- *The average SDC rate reduction is 92.92%*

Evaluation: Overhead



- *FT2 introduces 3.42% runtime overhead on average*
- *Memory overhead is negligible (288 - 512 Bytes, <0.2% for all models)*

FT2 Conclusion

- LLMs suffer from soft errors
  - Leads to SDC → Harmful and hard to detect
- Ranger has limitations applying to LLMs
  - Insufficient protection
  - Require bound profiling
- Our method: FT2
  - Identify and protect critical layers → high efficiency and low overhead
  - Obtain bounds during first token generation → no offline profiling
- Achieve 92.92% SDC rate reduction
- Only 3.42% overhead on average
- Code: https://github.com/pipijing13/FT2-LLM-inference-protection

What's Next…

- Different fault types?
- Other LLM types?
- New features?

GEORGE MASON UNIVERSITY

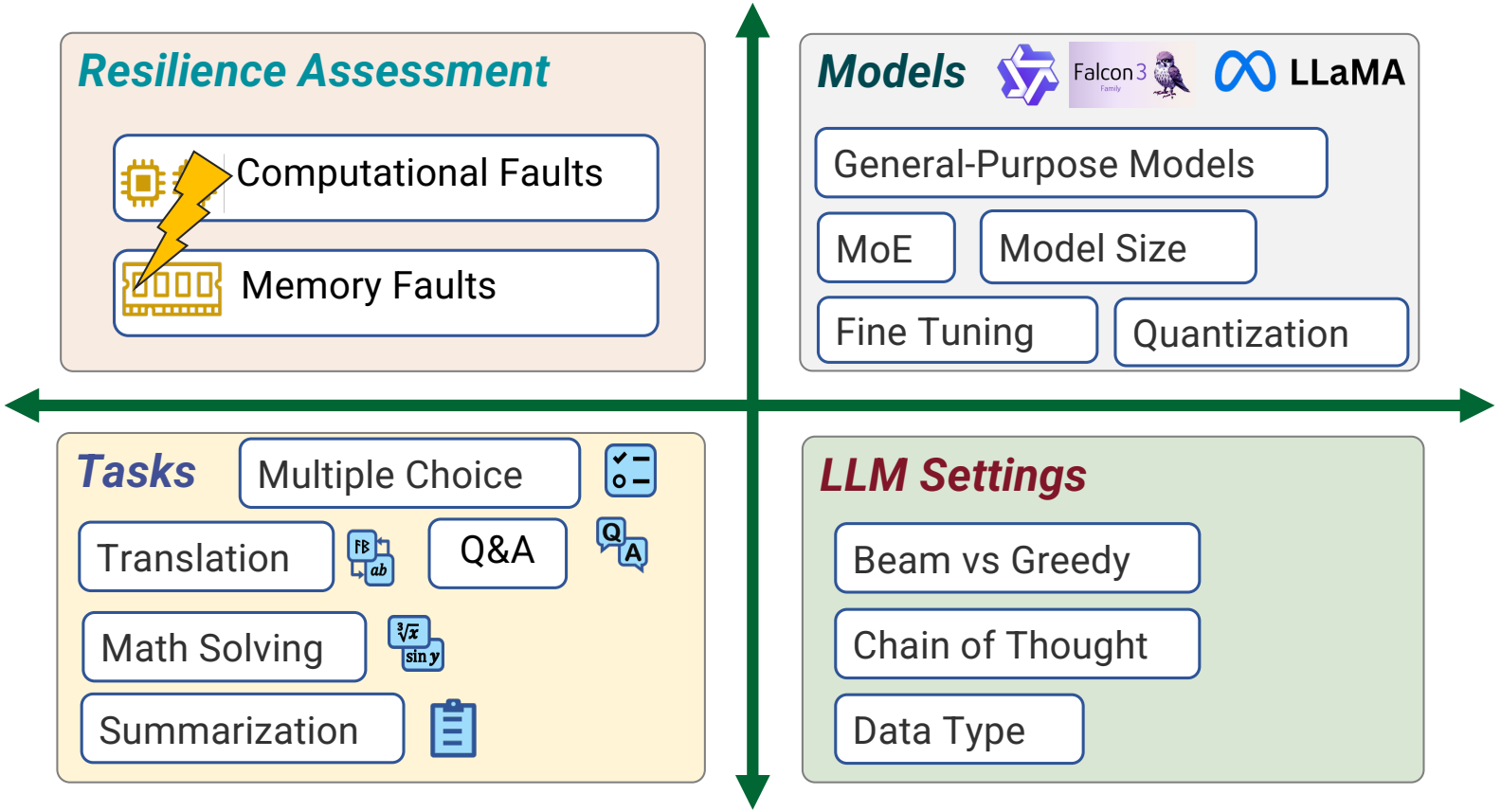
Demystifying the Resilience of Large Language Model Inference: An End-to-End Perspective

SC 2025

Yu Sun[§], Zachary Coalson[†], Shiyang Chen[‡], Hang Liu[‡],
Zhao Zhang[‡], Sanghyun Hong[†], Bo Fang[*], and **Lishan Yang**[§]

[§]George Mason University
[†]Oregon State University
[‡]Rutgers University
[*]Pacific Northwest National Lab

End-to-end Characterization



Experimental Set-up

- ***10 models from 4 families***
- ***9 datasets from 5 tasks***
- ***Datatype***
  - BF16
  - FP16
  - FP32
- ***3 fault models***
  - 1-bit computational fault
  - 2-bit computational fault
  - 2-bit memory fault
- ***2 GPU configurations***
  - NVIDIA A100
  - NVIDIA H100

Tasks	Datasets	Metrics	Test models
General model understanding and reasoning	MMLU AI2_ARC TruthfulQA WinoGrande HellaSwag	Accuracy	Llama3.1-8B Qwen2.5-7B Falcon3-7B
Math	GSM8k	Accuracy	Qwen2.5-7B Falcon3-7B
Translation	WMT16	BLEU chrF++	Llama2-7B Qwen2.5-7B ALMA-7B
Summarization	XLSum	Rouge-1 Rouge-L	Llama3.1-8B Qwen2.5-7B Summarizer
Question answering	SQuAD v2	Exact Match F1 score	Llama3.1-8B Qwen2.5-7B Falcon3-7B

Lessons Learned

An end-to-end large-scale characterization of LLM inference resilience

- For HPC designers:
  - Memory faults are more problematic
- For LLM providers:
  - Different error propagation → differences in reliability
  - Generative tasks are more vulnerable
- For practitioners
  - Using MoE models and CoT increase reliability
  - Quantized models are more resilient

Thank you

GEORGE MASON UNIVERSITY

Characterizing and Enhancing the Resilience of Generative Large Language Models against Soft Errors

Lishan Yang

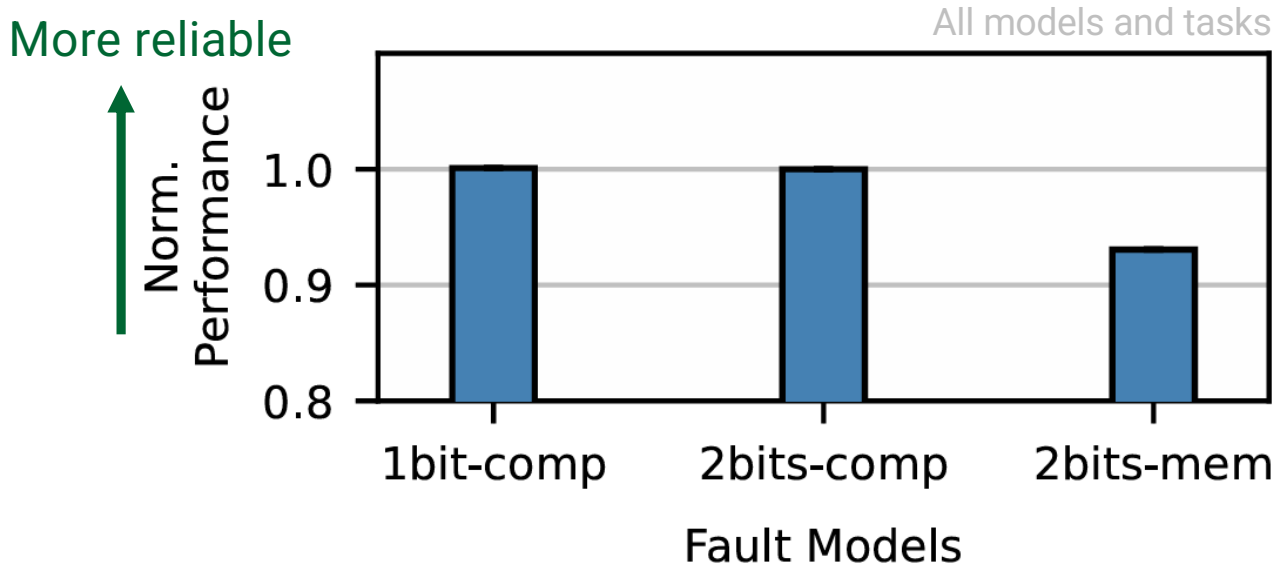
Assistant Professor

Department of Computer Science, George Mason University

In collaboration with Yu Sun (GMU), Zhu Zhu (GMU), Cherish Mulpuru (GMU), Zachary Coalson (OSU), Shiyang Chen (Rutgers), Hang Liu (Rutgers), Zhao Zhang (Rutgers), Sanghyun Hong (OSU), Roberto Gioiosa (PNNL), Bo Fang (UTA)

Resilience of Different Fault Models

LLMs are significantly more vulnerable to memory faults.



Why?

Error Propagation

Block1
up_proj

-2	1	2	0
3	2	3	1
1	2	1	0
0	3	1	3

Input

×

1	-1	0	2
-2	1	-1	1
0	0	1	0
-2	1	1	-1

Weights

=

-4	3	1	-3
-3	0	2	7
-3	1	-1	4
-12	6	1	0

Output

Block1
down_proj

-4	3	1	-3
-3	0	2	7
-3	1	-1	4
-12	6	1	0

Input

×

-1	0	1	0
-2	1	-1	0
0	0	1	1
-2	1	1	-1

Weights

=

4	0	-9	4
-11	7	6	-5
-7	5	-1	-5
0	6	-17	1

Output

Error Propagation

Block1
up_proj

-2	1	2	0
3	2	3	1
1	2	1	0
0	3	1	3

Input

×

1	NaN	0	2
-2	1	-1	1
0	0	1	0
-2	1	1	-1

Weights

=

-4	NaN	1	-3
-3	NaN	2	7
-3	NaN	-1	4
-12	NaN	1	0

Output

Block1
down_proj
(Memory)

-4	NaN	1	-3
-3	NaN	2	7
-3	NaN	-1	4
-12	NaN	1	0

Input

×

-1	0	1	0
-2	1	-1	0
0	0	1	1
-2	1	1	-1

Weights

=

NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN

Output

Propagate to the whole tensor

Error Propagation

Block1 up_proj

-2	1	2	0
3	2	3	1
1	2	1	0
0	3	1	3

Input

×

1	-1	0	2
-2	1	-1	1
0	0	1	0
-2	1	1	-1

Weights

=

NaN	3	1	-3
-3	0	2	7
-3	1	-1	4
-12	6	1	0

Output

Block1 down_proj (Memory)

-4	NaN	1	-3
-3	NaN	2	7
-3	NaN	-1	4
-12	NaN	1	0

Input

×

-1	0	1	0
-2	1	-1	0
0	0	1	1
-2	1	1	-1

Weights

=

NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN
NaN	NaN	NaN	NaN

Output

Block1 down_proj (Computational)

NaN	3	1	-3
-3	0	2	7
-3	1	-1	4
-12	6	1	0

Input

×

-1	0	1	0
-2	1	-1	0
0	0	1	1
-2	1	1	-1

Weights

=

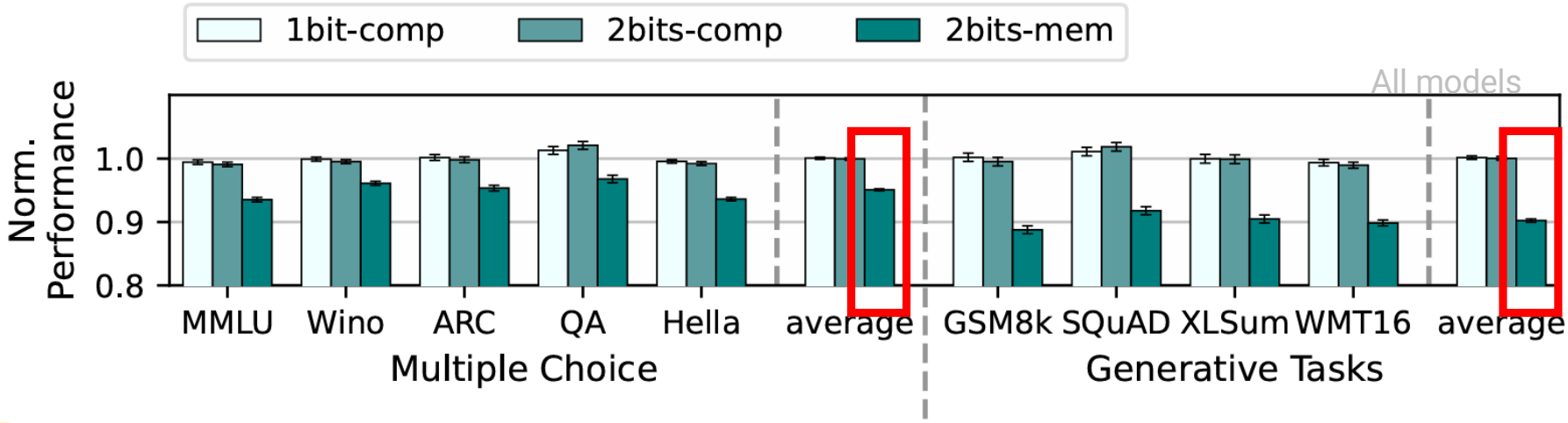
NaN	NaN	NaN	NaN
-11	7	6	-5
-7	5	-1	-5
0	6	-17	1

Output

Limited in only one row

Resilience of Different Tasks

LLMs are more vulnerable on generative tasks under memory faults.



Why?

Resilience of Different Tasks

Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?
Reference: 14

Resilience of Different Tasks

Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?	
Reference: 14	
Baseline: Let's break it down step by step: 1. Samantha is currently 31 years old.	**FI:** Let's break it down step by step: 1. Samantha is currently 31 years old.

Resilience of Different Tasks

Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?	
Reference: 14	
Baseline: Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond is 6 years older than Samantha, so Raymond is 31 **+** 6 = 37 years old.	**FI:** Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond was born 6 years before Samantha, so Raymond is 31 **-** 6 = 25 years old.

Resilience of Different Tasks

GEORGE MASON UNIVERSITY

Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?	
Reference: 14	
Baseline: Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond is 6 years older than Samantha, so Raymond is 31 + 6 = 37 years old. 3. Raymond had his son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract the age Raymond was when he had his son from Raymond's current age:	**FI:** Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond was born 6 years before Samantha, so Raymond is 31 - 6 = 25 years old. 3. Raymond had a son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract the age Raymond was when his son was born from Raymond's current age:

Resilience of Different Tasks

GEORGE MASON UNIVERSITY

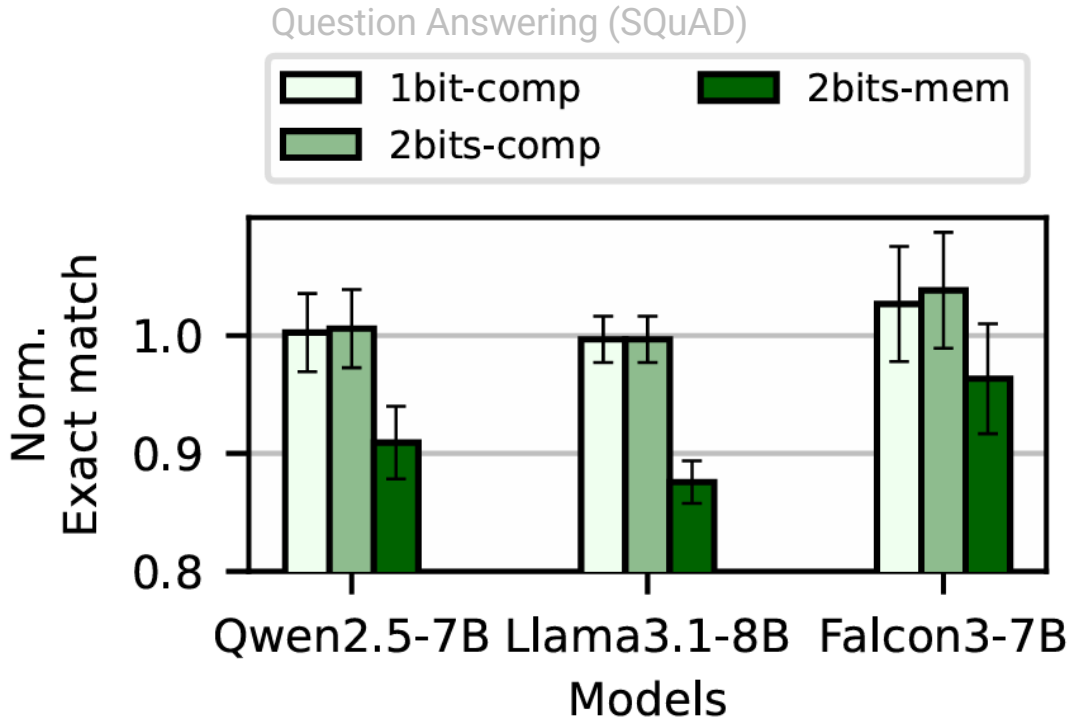
Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?	
Reference: 14	
Baseline: Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond is 6 years older than Samantha, so Raymond is 31 + 6 = 37 years old. 3. Raymond had his son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract the age Raymond was when he had his son from Raymond's current age: 4. 37 (Raymond's current age) - 23 (age when Raymond had his son) = 14 years ago.	**FI:** Let's break it down step by step: 1. Samantha is currently 31 years old. 2. Raymond was born 6 years before Samantha, so Raymond is 31 - 6 = 25 years old. 3. Raymond had a son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract age Raymond was when his son was born from Raymond's current age: 25 (Raymond's current age) - 23 (age when Raymond had his son) = 2

Resilience of Different Tasks

<table>
<tr><td colspan="2">Question: Raymond and Samantha are cousins. Raymond was born 6 years before Samantha. Raymond had a son at the age of 23. If Samantha is now 31, how many years ago was Raymond's son born?</td></tr>
<tr><td colspan="2">Reference: 14</td></tr>
<tr><td>Baseline: Let's break it down step by step:
1. Samantha is currently 31 years old.
2. Raymond is 6 years older than Samantha, so Raymond is 31 + 6 = 37 years old.
3. Raymond had his son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract the age Raymond was when he had his son from Raymond's current age:
4. 37 (Raymond's current age) - 23 (age when Raymond had his son) = 14 years ago.
So, Raymond's son was born 14 years ago.</td><td>FI: Let's break it down step by step:
1. Samantha is currently 31 years old.
2. Raymond was born 6 years before Samantha, so Raymond is 31 - 6 = 25 years old.
3. Raymond had a son when he was 23 years old.To find out how many years ago Raymond's son was born, we need to subtract age Raymond was when his son was born from Raymond's current age:
25 (Raymond's current age) - 23 (age when Raymond had his son) = 2
So, Raymond's son was born 2 years ago.</td></tr>
</table>

Resilience of Different Models

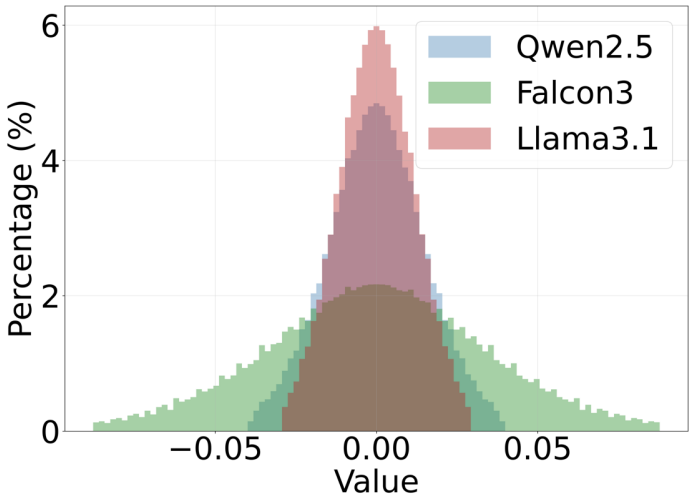
LLMs with similar architecture → resilience varies



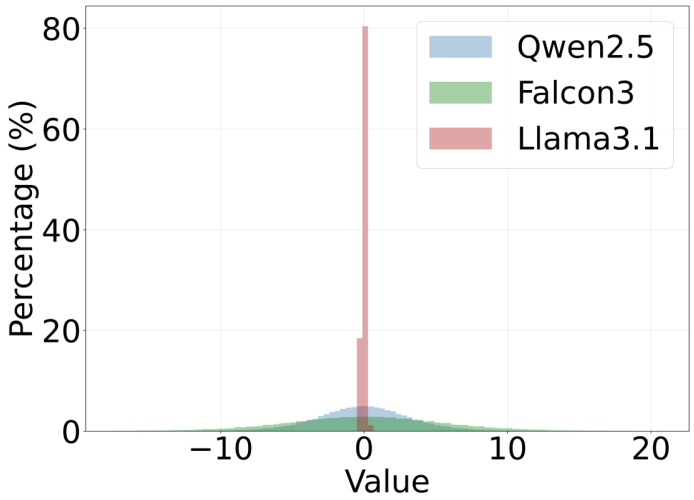
Resilience of Different Models

Different training data/method → different value distributions.

Question Answering (SQuAD)



Weights distribution



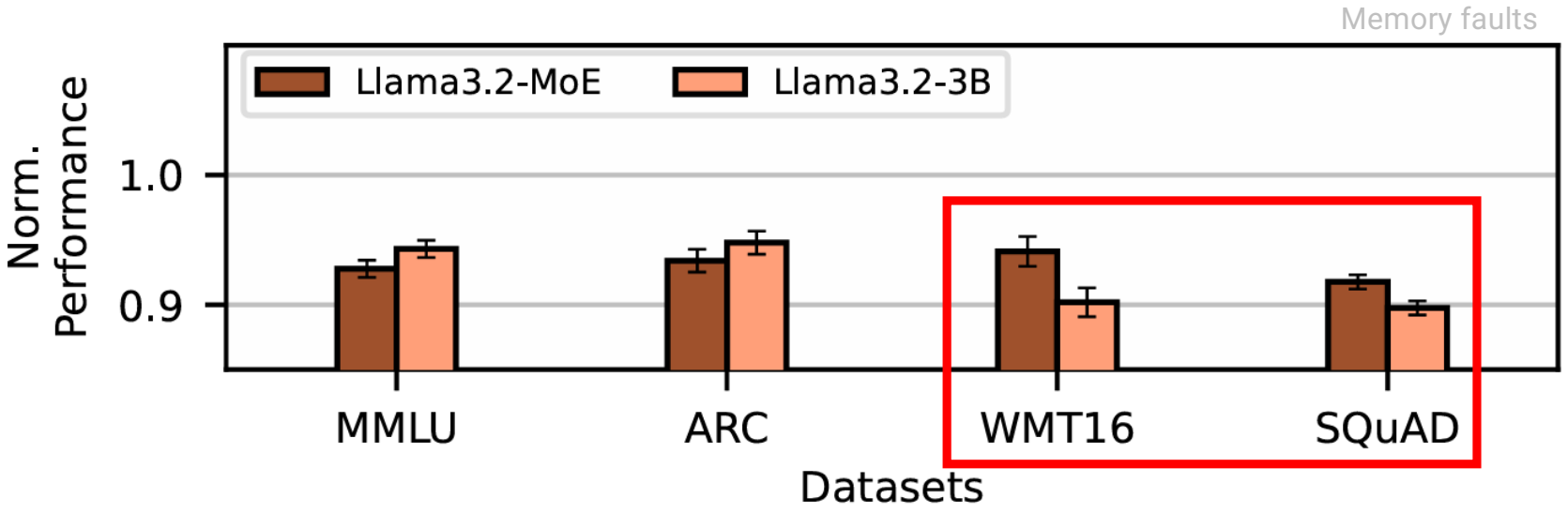
Neurons distribution

Resilience: Beam Search vs Greedy Search

Using beam search is more resilient to computational faults.

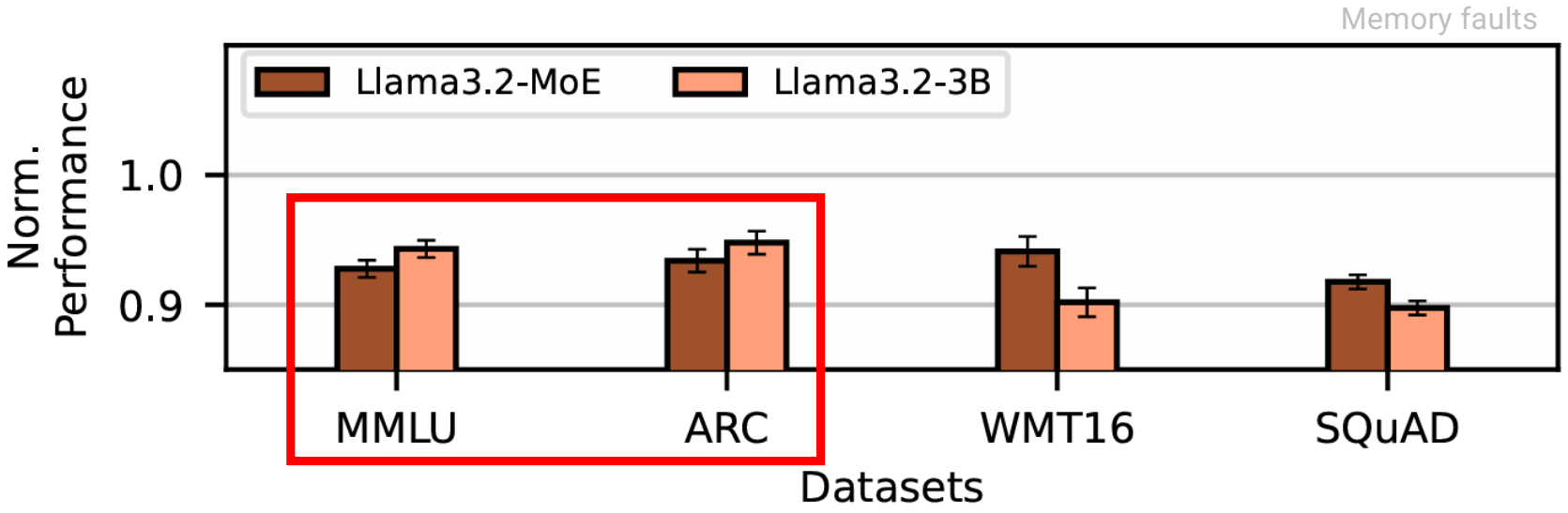
Resilience of Mixture of Experts (MoE)

MoE models are more reliable on generative tasks.



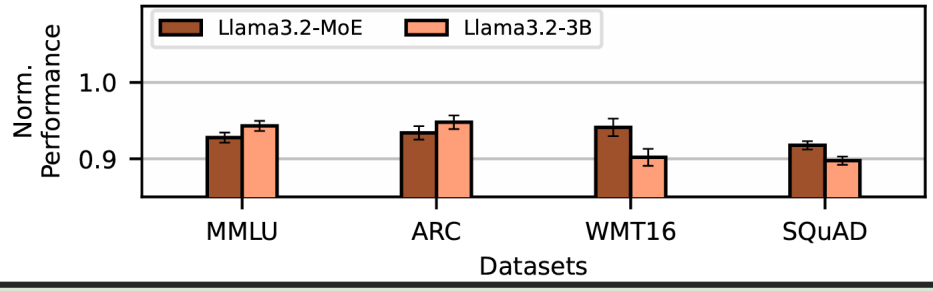
Resilience of Mixture of Experts (MoE)

MoE models are more vulnerable on multiple-choice tasks.

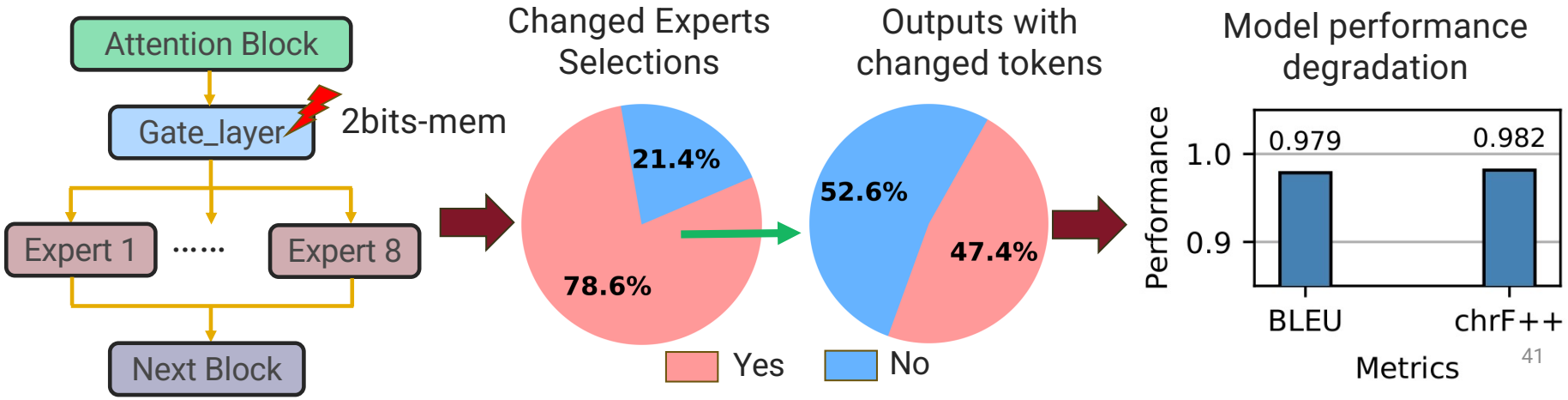


Why?

Resilience of Mixture of Experts (MoE)

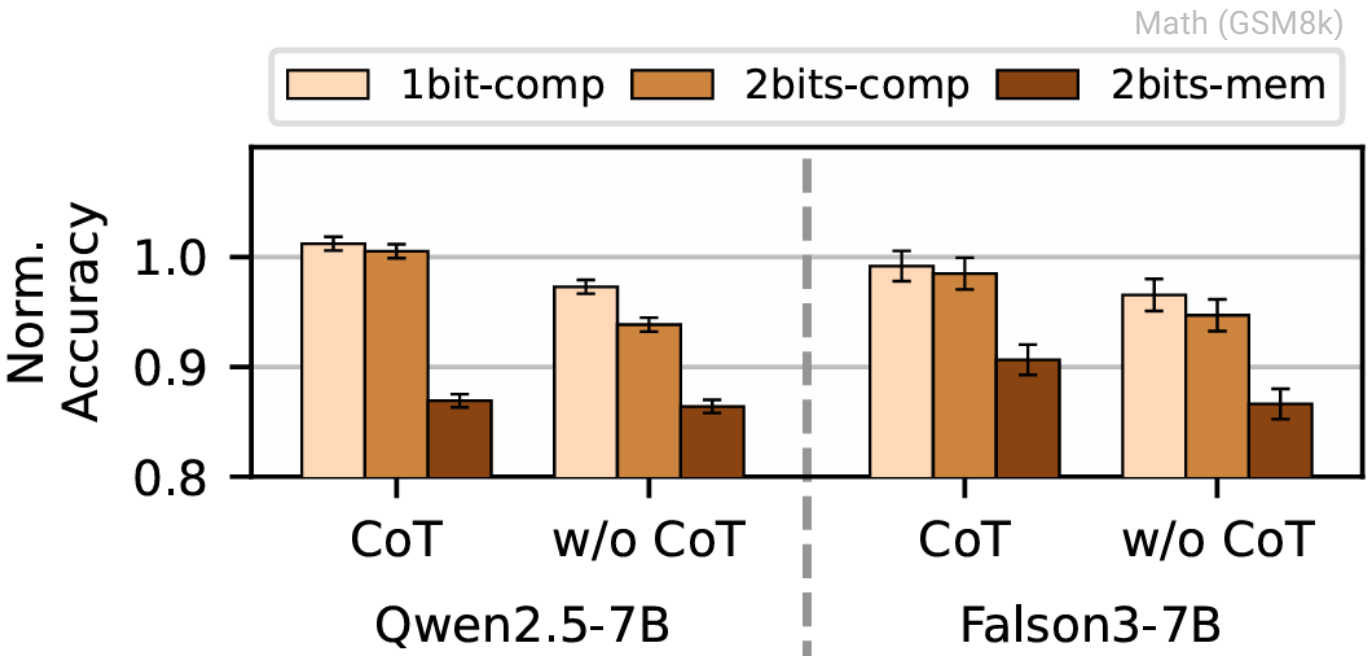


Errors in gate layers → expert change → performance degradation



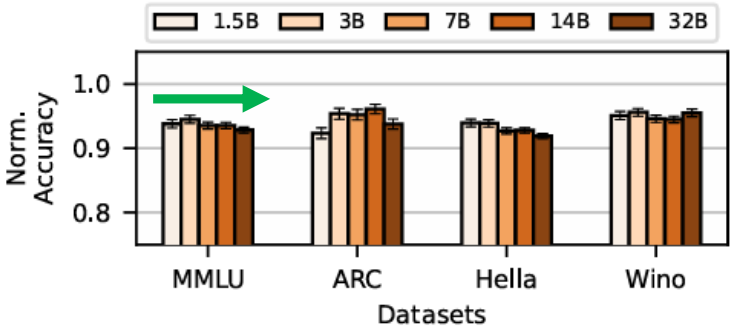
Resilience of Chain of Thoughts (CoT)

Using CoT for math tasks improves the reliability.



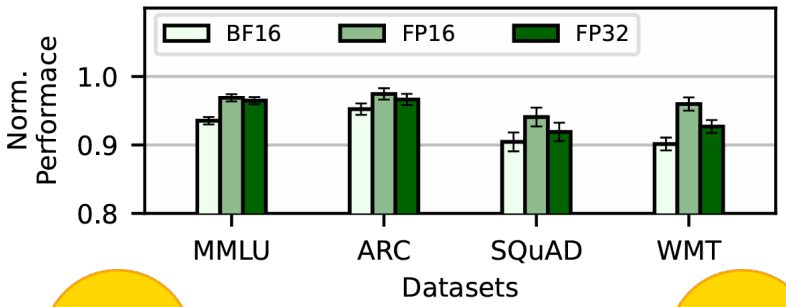
Other Insights

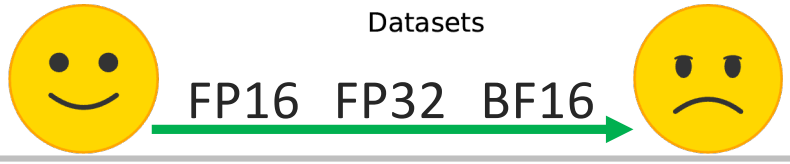
Memory faults



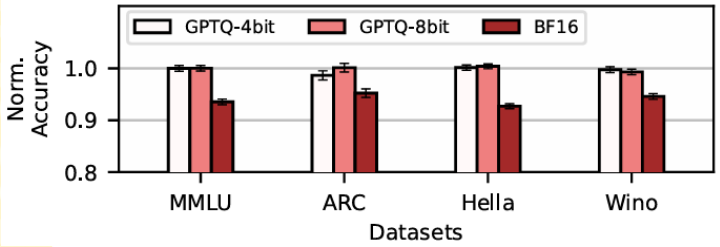
Model size: **Minimal** effect

MMLU

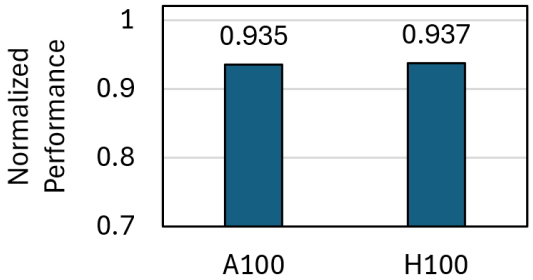




FP16 FP32 BF16



Quantized model



NVIDIA A100 ≈ NVIDIA H100

Output Type: Subtly Wrong vs Distorted

Question: Joe's bag of Halloween candy has 25 chocolate bars and 80 candied apples. Each chocolate bar weighs twice as each candied apple. If each chocolate bar weighs 40g, how much does Joe's bag of candy weigh?

Reference: 2600

Baseline: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of chocolate bars: 25 bars * 40g / bar = 1000g. Then find the total weight of the candied apples: 80 apples * 20g / apple = 1600g. Then add the weight of chocolate bars and candied apples to find the total weight: 1000g + 1600g = 2600g. Therefore, the answer is 2600.

Subtly wrong output: First find the weight of each candied apple: 40g / 2 = 20g. Then find the total weight of the apple chocolate bars: 5 bars * 40g / bar = 200g. Then find the total weight of the apples: 80 apples * 20g / apple = 1600g. Then add the weight to find the total weight: 200g + 1600g = 1800g. The answer is 1800.

Distorted output: !!!