An Efficient Checkpointing System for Large Machine Learning Model Training

Wubiao Xu, Xin Huang, Weiping Zhang, Shiman Meng, Guoyuan Jia

Luanzheng Guo, Kento Sato

Nanchang Hangkong University

Kobe University

Pacific Northwest National Laboratory

RIKEN Center for Computational Science

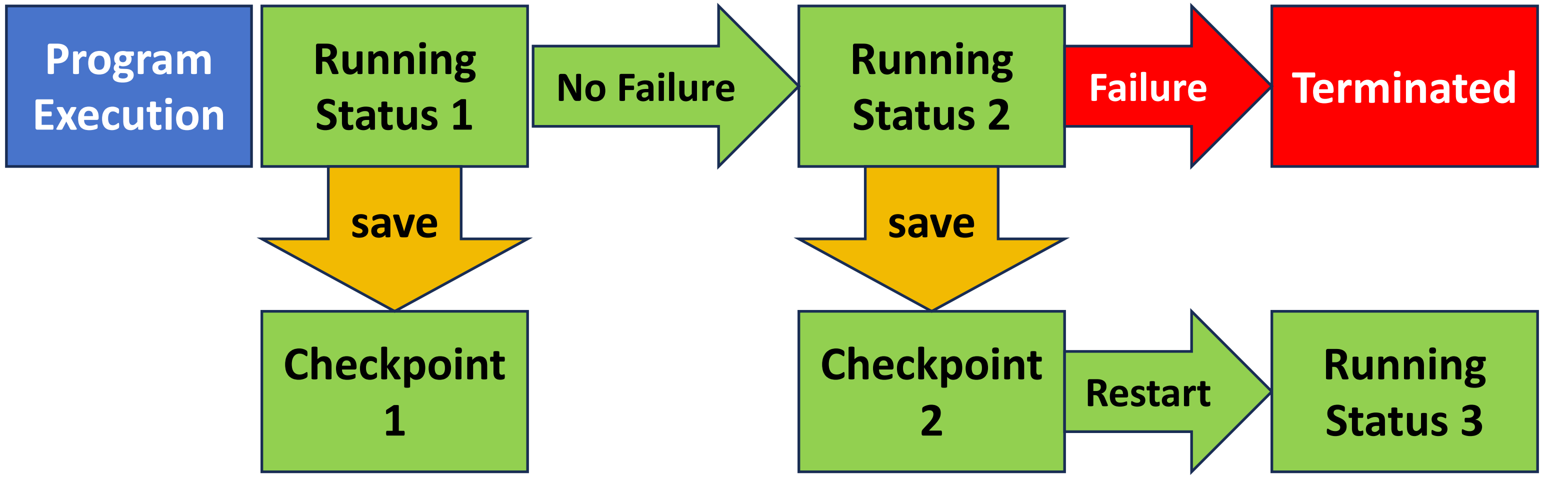
Outline

- Introduction
- Problem Definition
- Optimization Strategies
- Evaluation
- Conclusion

Checkpointing

- **What is Checkpointing?**
  - ➢Checkpointing is a technique that saves the state of program execution; upon a system failure, training can resume from the latest checkpoint

- **Why checkpointing needs improvement for large machine learning (ML) models?**
  - ➢As ML model sizes increase, frequently checkpointing can lead to significant performance and storage overhead



Problem definition

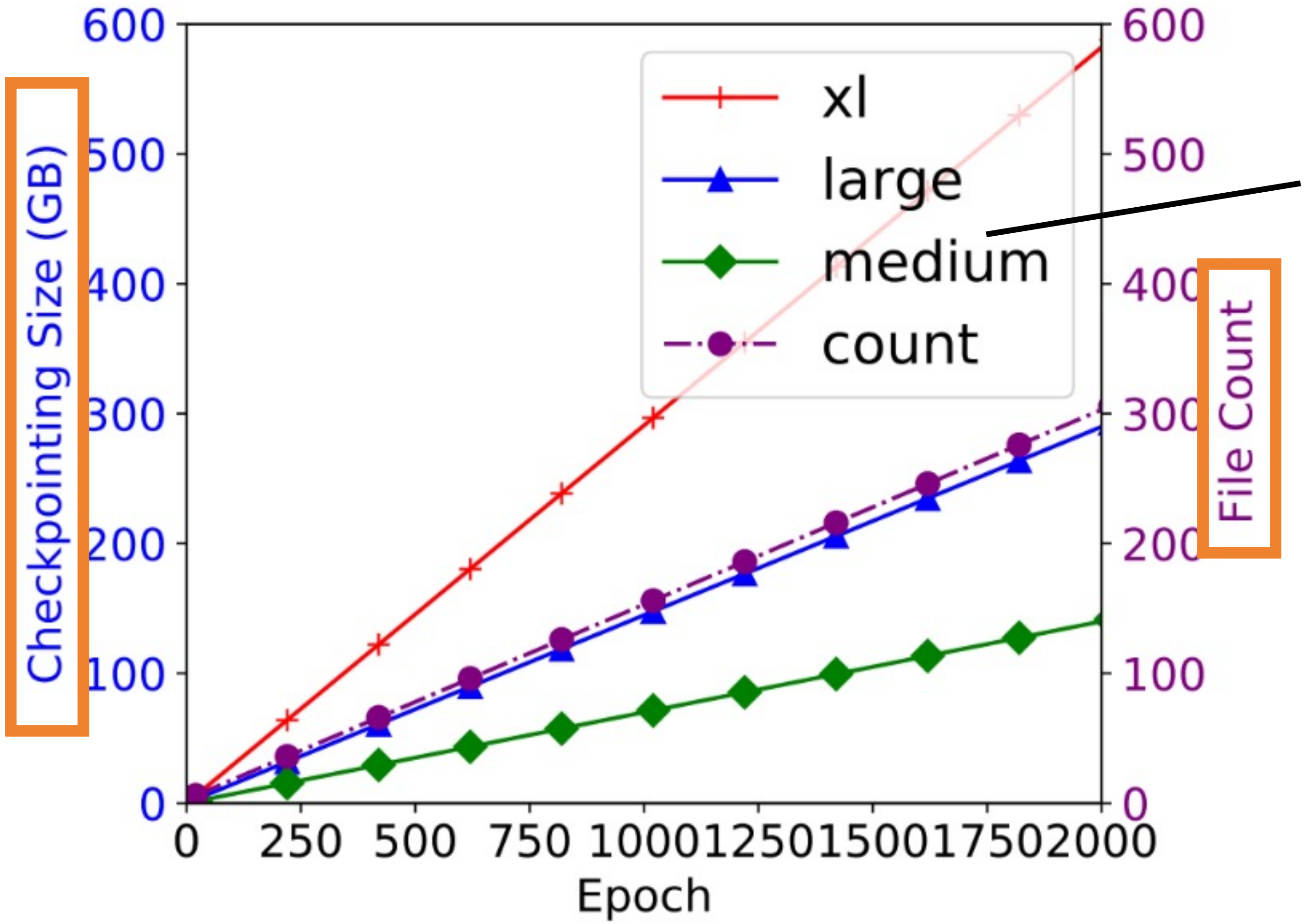
- The growing ML model size results in much larger checkpoint sizes

Model	Model Size	Checkpoint Size
GPT2-large	774M	2.9GB
GPT2-xl	1.5B	5.9GB
Vicuna	7B	13.4GB
OpenOrca	70B	14.5GB
LLama-2	70B	140GB
GLM	130B	70GB
BLOOM	176B	329GB
GPT3	175B	700GB

- Large language models: model size and checkpoint size

Problem definition (cont.)

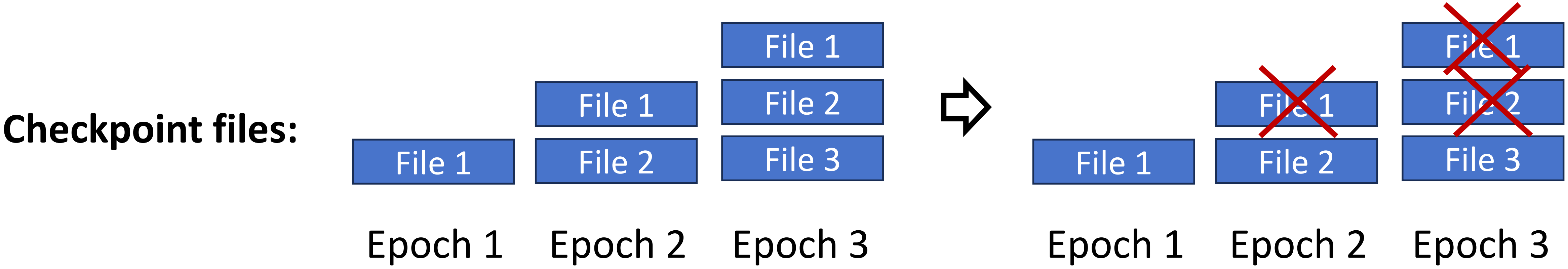
- We observed an increase in both the number of cumulative checkpoint files and the storage consumed by checkpoints



GPT2 variants:
GPT2-xl;
GPT2-large;
GPT2-medium

Optimization: *Periodic cleaning*

- Periodically and asynchronously delete outdated checkpoints while keeping the latest ones



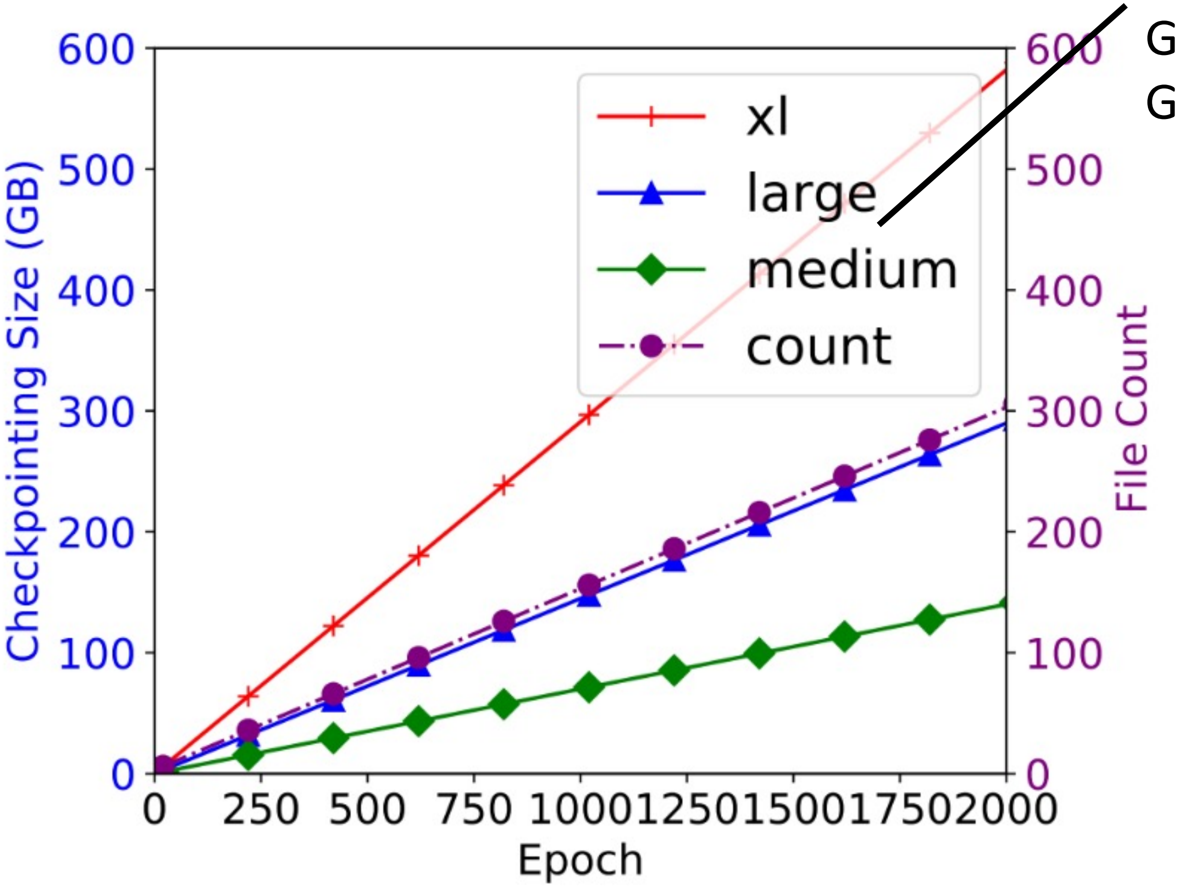
Optimization: *Staging*

- We observe that existing ML training writes checkpoints to parallel file systems
  - However, I/O bandwidth on parallel file system is much lower than local storage
  - E.g., I/O bandwidth is about 23 MB/s on NFS while 4.1 GB/s on local SSD

- Optimization:
  - Write checkpoints to the local file system like SSD
  - Spawn a separate process to move the outdated checkpoints to parallel file systems
  - The staging is independent from the training process

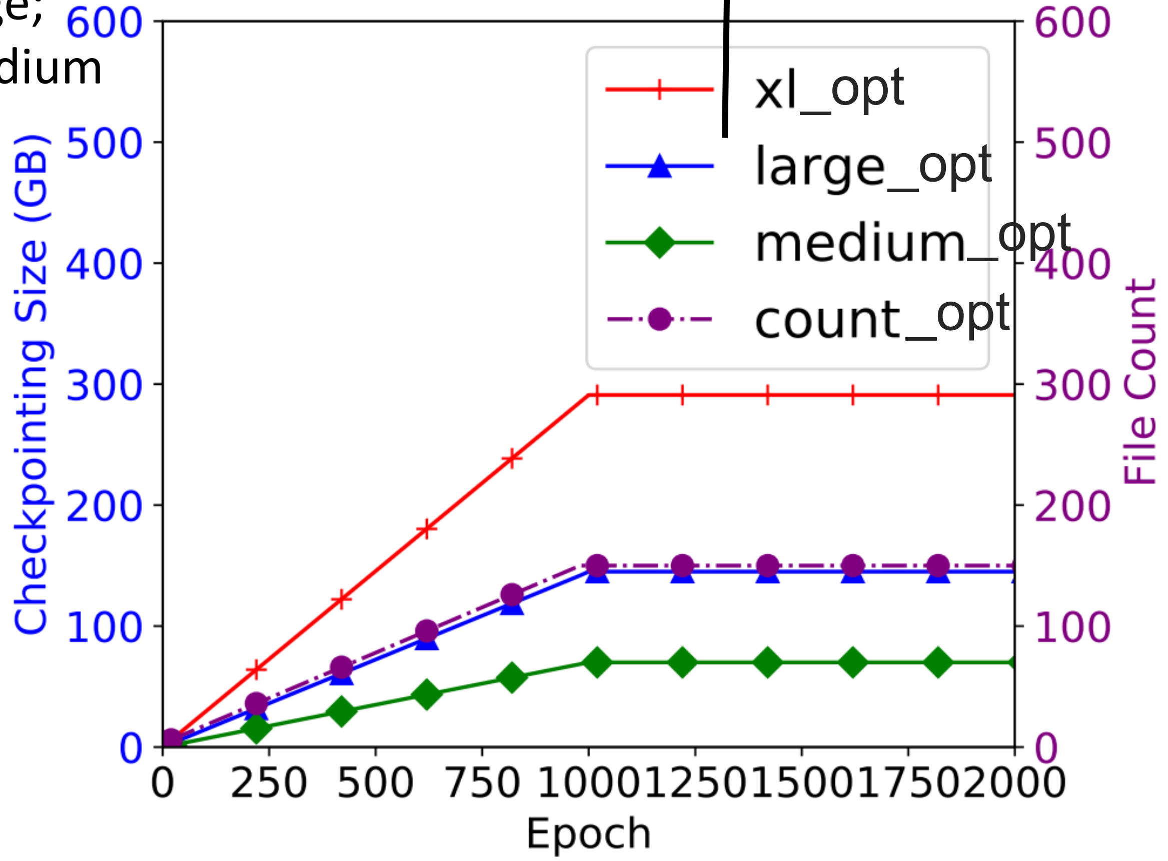
Evaluation: *Periodic cleaning*

GPT2 variants:
GPT2-xl;
GPT2-large;
GPT2-medium

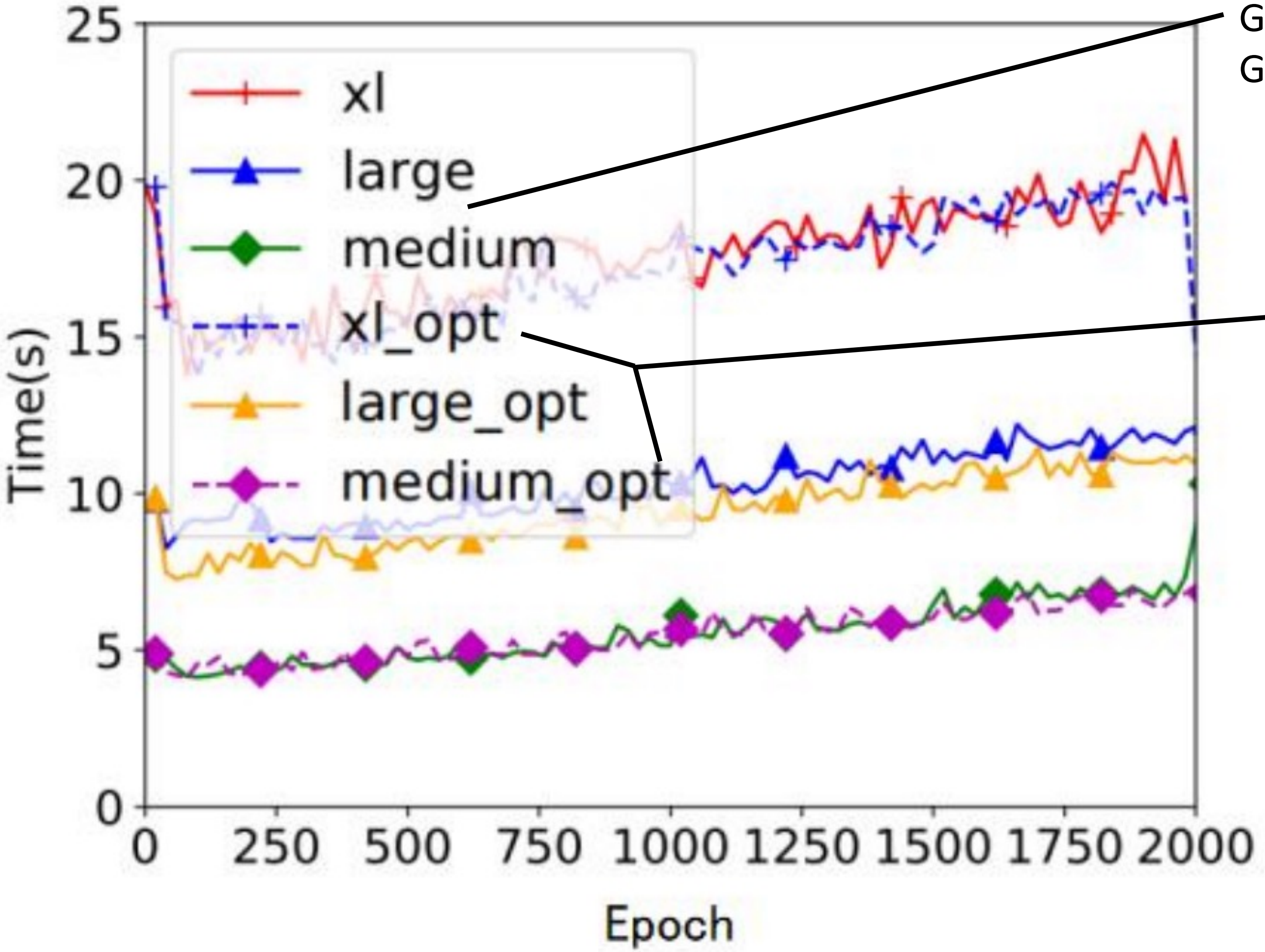


⇨

With periodic cleaning



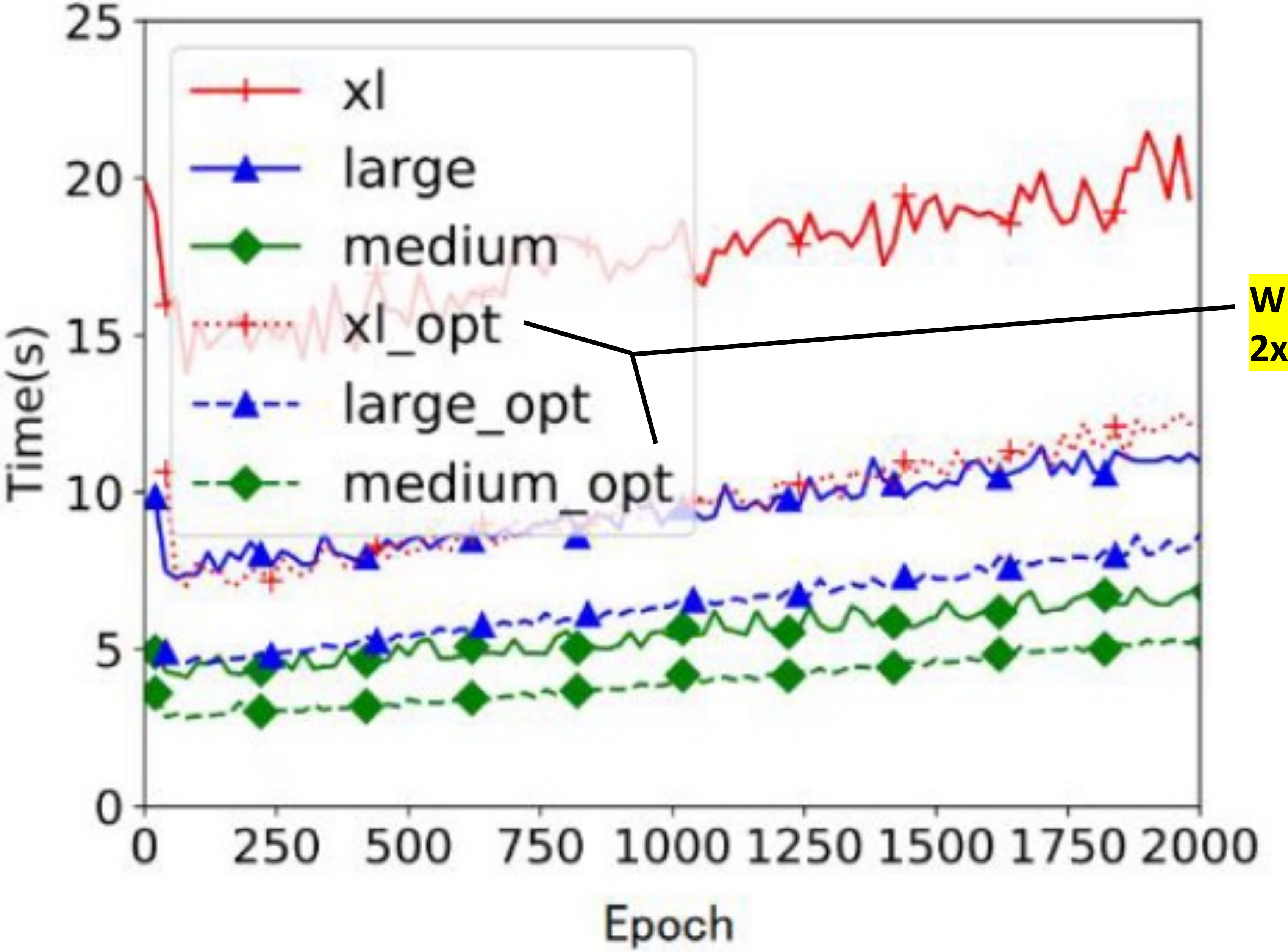
Evaluation: *Periodic cleaning (Cont.)*



GPT2 variants:
GPT2-xl;
GPT2-large;
GPT2-medium

With periodic cleaning

Evaluation: *Staging*



With staging: 2x speedup

Conclusion

- We characterize the checkpointing with respect to storage and performance in large ML model training
- We propose two checkpointing optimization strategies for large ML models
- We verify the effectiveness and reliability of the proposed optimizations with GPT-2 variants

Thank you

for your attention