Conference Paper Title*

*Note: Sub-titles are not captured in Xplore and should not be used

1st André Lucena Ribas Ferreira
University of Minho
Braga, Portugal
pg52672@uminho.pt

1st Carlos Eduardo da Silva Machado
University of Minho
Braga, Portugal
pg52675@uminho.pt

1st Gonçalo Manuel Maia de Sousa
University of Minho
Braga, Portugal
pg52682@uminho.pt

***Abstract*—This document is a model and instructions for LaTeX. This and the IEEEtran.cls file define the components of your paper [title, text, heads, etc.]. *CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

- Machine learning has increased in popularity
  - image classification
  - natural language processing
- studies have tried to analyze I/O patterns in DL Workflows (source)
- I/O Characterization has served to produce diverse assumptions about DL I/O behavior, like high overhead for random reads, which are commonly used to justify various types of optimization explorations ([1], [2], [3]). Application-specific storage characterization is required to find unique and worthwhile solutions for newer performance problems.
- very few get down to kernel level
- eBPF are ...
- we seek to provide a tool to Characterize DL workloads using eBPF's

II. Background

- DL involves iterating multiple times (epochs) through a dataset
- Dl is a subset of machine learning algorithms based on neural networks
- for accuracy, all data is read exactly once per one epoch and is done in random reads(I/O intensive)
- passing it through all the layers to calculate a loss (forward pass)
- use calculated loss to update the learnable parameters of the network (backpropagation)
- SGD is an optimizer for loss function minimization widely used for its lower computation compared to working through the whole dataset
- increasing batch-size in the last years, from the usual range of 32-256 [2]
- DL is usually I/O-bound [need source], due to the use of accelerators (GPU), size of the data and random reads
- pytorch is a DL framework with some particularities (Dataloader, ...)
- Tensorflow is also a DL framework, with other particularities...
- Imagenet
- Distributed DNN training (data parallelism)
- checkpointing involves saving the model state
- in pytorch its done explicitly with torch.save() and in official workloads is done in-between epochs
- eBPF's

III. Related work

- papers que usam darshan/tf-darshan, como per-file statistics, para caracterizar padrões [4] [5]
- MLPerf Storage/tese de um aluno da Oana
- DIO e tools de observabilidade que usam eBPF e outras (related work do DIO), LD PRELOAD, captura de de I/O request por intrumentação do código fonte.
- Utilizar a descrição do I/O pipeline de Tensorflow workloads para benchmark [6]
- Caracterizar o I/O de LMDB, que é análoga à do PyTorch e do Tensorflow, a **database** usada como base do Caffe, baseada em *mmap* e numa *B+-tree*. [1]
- Comparar o overhead das leituras no treino inteiro com e sem shuffling. [7]
- Que métricas analisam principalmente?
  - I/O skew por processo;
  - tempo de leitura por processo;
  - bandwidth do read por cada I/O block size;
  - número de context switches;
  - latência com/sem shuffle
- O que falta fazer?
  - Análise empírica dos padrões de I/O como parte do processo de treino ao longo do tempo
  - Análise da cache como interveniente no processo de I/O
  - Testes de Rede (para modelos distribuídos)
  - Analisar PyTorch com o nível de detalhe que analisaram outras
  - Analisar kernel-level I/O calls

IV. Design

- high level description of the system

V. Evaluation Methodology

- dstat, nvidia-smi to get cost of using the tool
- python parser and plots
- grafana dashboard to get data

VI. Evaluation Results

VII. Conclusion

References

[1] S. Pumma, M. Si, W.-C. Feng, and P. Balaji, "Scalable deep learning via i/o analysis and optimization," *ACM Trans. Parallel Comput.*, vol. 6, no. 2, Jul. 2019. [Online]. Available: https://doi.org/10.1145/3331526

[2] Y. Zhu, W. Yu, B. Jiao, K. Mohror, A. Moody, and F. Chowdhury, "Efficient user-level storage disaggregation for deep learning," in *2019 IEEE International Conference on Cluster Computing (CLUSTER)*, 2019, pp. 1–12.

[3] F. Chowdhury, Y. Zhu, T. Heer, S. Paredes, A. Moody, R. Goldstone, K. Mohror, and W. Yu, "I/o characterization and performance evaluation of beegfs for deep learning," in *Proceedings of the 48th International Conference on Parallel Processing*, ser. ICPP '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3337821.3337902

[4] S. W. D. Chien, A. Podobas, I. B. Peng, and S. Markidis, "tf-darshan: Understanding fine-grained i/o performance in machine learning workloads," in *2020 IEEE International Conference on Cluster Computing (CLUSTER)*, 2020, pp. 359–370.

[5] T. Wang, S. Byna, G. K. Lockwood, S. Snyder, P. Carns, S. Kim, and N. J. Wright, "A zoom-in analysis of i/o logs to detect root causes of i/o performance bottlenecks," in *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, 2019, pp. 102–111.

[6] S. W. D. Chien, S. Markidis, C. P. Sishtla, L. Santos, P. Herman, S. Narasimhamurthy, and E. Laure, "Characterizing deep-learning i/o workloads in tensorflow," in *2018 IEEE/ACM 3rd International Workshop on Parallel Data Storage & Data Intensive Scalable Computing Systems (PDSW-DISCS)*, 2018, pp. 54–63.

[7] F. Chowdhury, J. Liu, Q. Koziol, T. Kurth, S. Farrell, S. Byna, and W. Yu, "Initial characterization of i/o in large-scale deep learning applications," in *3rd Joint International Workshop on Parallel Data Storage and Data Intensive Scalable Computing Systems (PDSW-DISCS'18) at the International Conference on High Performance Computing, Networking, Storage and Analysis*, 2018.