Is CPU and be replace in GPU?

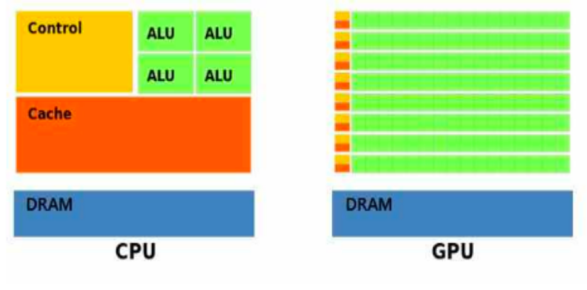
Daeyeop Kim
Luddy School of Informatics, Computing, and Engineering Indiana University

Abstract

Since the machine learning is getting more important in these days. Even GPU programming gets normalized, the level of the GPU programming is the big obstacle in these days. And it lead to the lots of new package or library with GPU even for CPU machine learning program.
Although CPUs and GPUs each support parallel processing, there are cases where the CPU is more efficient when using the CPU and the GPU, depending on the nature or environment of the application. So, I used BigRed 200 which have GPU(NVIDIA Tesla A100 GPU)model and CPU (AMD Epyc7713 processor(64 -core)) model to check the GPU and CPU difference by comparing the time for the transposing with different size of the matrixes. And check why CPU cannot be replace on GPU with Deep learning with comparing the time difference with different strategies by transposing different size of the matrix (dataset).

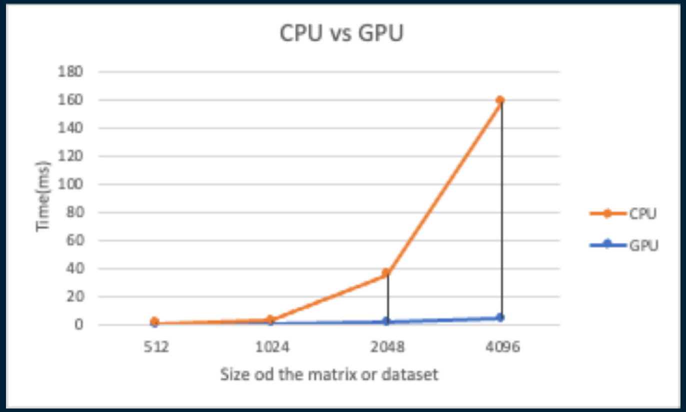
Introduction

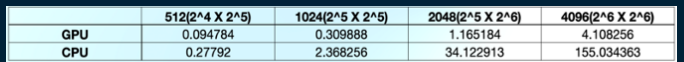
GPU(Graphics Processing Unit): It is designed for parallelizable problem.
The GPU is a processor that is made up of many smaller and more specialized cores. By working together, the cores deliver massive performance when a processing task can be divided up and processed across many cores.
A CPU is a central processing unit, a top-tier unit that uses computer resources, including input/output devices, memory devices, and computing devices, and serves as the brain of a computer. In addition to data processing, the algorithm analyzed in the program determines the next behavior, prioritizes and switches tasks divided for multitasking, and directs the computer, including managing virtual memory.
Although CPUs and GPUs each support parallel processing, there are cases where the CPU is more efficient when using the CPU and the GPU, depending on the nature or environment of the application. So from this reserach project, which processor is more efficiency with utilizing one of the deep learning process, Transposing dataset, to decide CPU can replace on GPU.
GPU, CPU
The picture shows the structure of GPU and CPU. As can be seen from this figure, GPU has an overwhelmingly high ratio of ALU chips compared to CPU, which makes it specialized in parallel computation.
Relation between CPU and GPU is more like Employee(GPU) and Employer (CPU)



Method

For my reserach experiment...
By Transposing the matrix for comparing the CPU(AMD EPYC 7742 processors) and GPU(NVIDIA A100) work process time and efficiency, From this project, we checked 4 times with 4 different size(512(2^4 X 2^5), 1024 (2^5 X 2^5), 2048(2^5 X 2^6), 4096 (2^6 X 2^6)) of the matrix transpose. With different size matrixes it shows that GPU is more efficiency on the transposing with the naive method.
For the GPU transposing model, I used shmem and optimal model to show GPU programming can even more be efficiency with different method with same experiment (Transposing matrix with different set of matrix)
CPU vs GPU
The matrix transpose application is an application that calculates a value by multiplying a matrix of a size input by a user. A total of four sizes of matrices were performed, and each operation was performed 4 times to record the results of CPU and GPU.
For this line graph, There were huge difference in the time with transpose the matrix with same strategies(Naive)
Time (ms)





	512(2^4 X 2^5)	1024(2^5 X 2^5)	2048(2^5 X 2^6)	4096(2^6 X 2^6)
GPU	0.094784	0.309888	1.165184	4.108256
CPU	0.27792	2.368256	34.122913	155.034363

Why I choose to do Deep learning techique
Why I choose to use the transpose data to compare
For most deep learning methods use neural network architectures, which is why deep learning models are often referred to as deep neural networks.
Deep learning models are trained by using large sets of labeled data and neural network architectures that learn features directly from the data without the need for manual feature extraction.
And the Transposing the matrix is the one of the way or step that feature the dataset for the deep learning.

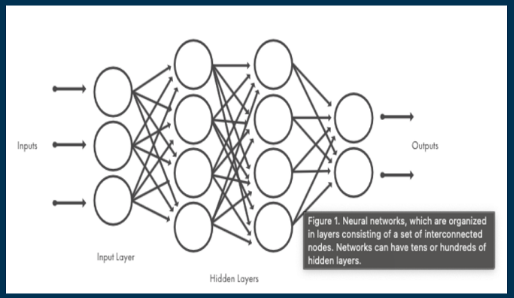
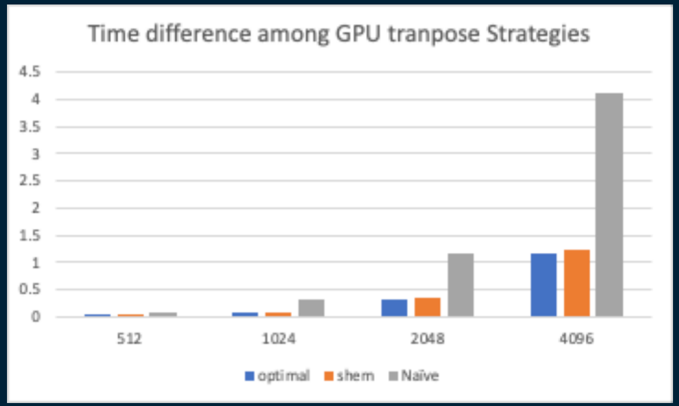
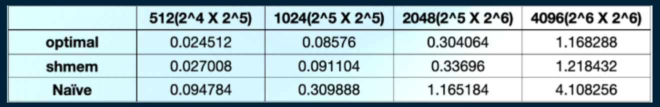


Figure 1. Neural networks, which are organized in layers consisting of a set of interconnected nodes. Networks can have tens or hundreds of hidden layers.

With various Model for Transpose
Shmem, optimal, Naive with different size of the data set.
As this graph shows, There are even more optimal and reduce the time for the transpose the program.
SHMEM is a family of parallel programming libraries, providing one-sided parallel-processing interfaces for low-latency distributed-memory supercomputers.
The SHMEM acronym was subsequently reverse engineered to mean "Symmetric Hierarchical MEMory"





	512(2^4 X 2^5)	1024(2^5 X 2^5)	2048(2^5 X 2^6)	4096(2^6 X 2^6)
optimal	0.024512	0.08576	0.304064	1.168288
shmem	0.027008	0.091104	0.33696	1.218432
Naïve	0.094784	0.309888	1.165184	4.108256

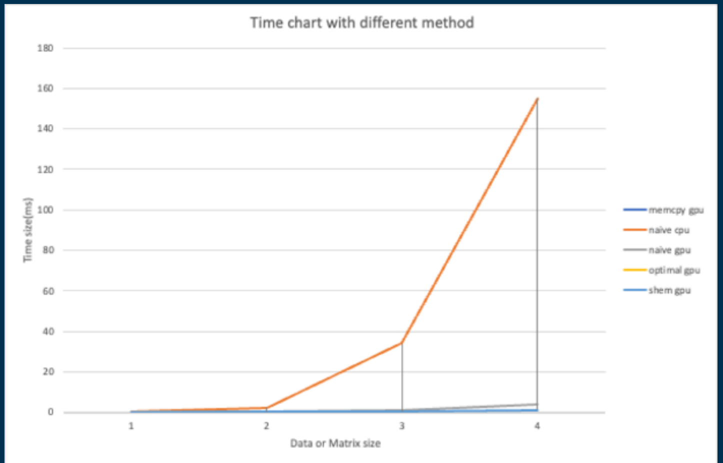
References

any Documentation or research for my project ...
https://www.intel.com/content/www/us/en/products/docs/processors/cpu-vs-gpu.html (For GPU Definition and CPU definition)
https://www.mathworks.com/discovery/deep-learning.html(why I transpose matrix can be show the difference between GPU and CPU)
https://en.wikipedia.org/wiki/In-place_matrix_transposition (Matrix Transpose iwth naive)
https://kb.iu.edu/d/brcc (BigRed200 info)
https://en.wikipedia.org/wiki/SHMEM (SHEM method)

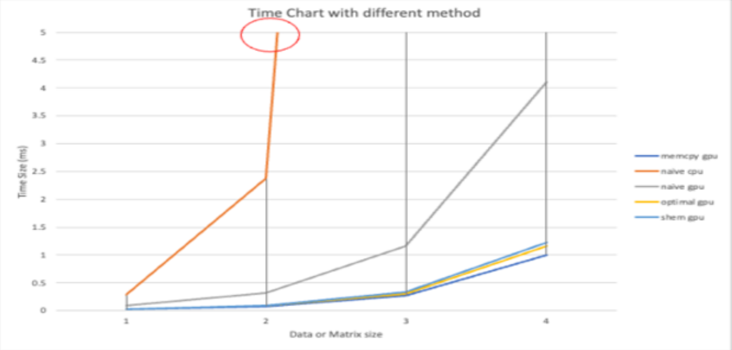
Conclusion

Result

GPU vs CPU
This figure is for showing the how the process time difference with the CPU and GPU with different size of the matrix or dataset.
And it shows how GPU programming can reduce the time for the same experiment at the same time.



As this figure shows, There are huge difference with CPU and GPU in transposing the different size of the matrix.
This even shows in GPU programming can make more optimizing the time on transpose matrix.



From my Research Project.
In the research, when I experiment the transposing the matrix, check the time with CPU and GPU. For efficiency on CPU, and GPU, as research data shows we can find out CPU and GPU there are huge gap on running time. Since Deep- learning treats the big data set for training, CPU cannot be replace on GPU on the efficiency and memory especially on the time.
For further research, considering the CPU and GPU working process, I need to think about the temperature check with CPU and GPU working process to check the other way of the efficiency.