CSE 585 Recap

Mosharaf Chowdhury

Resource-Centric View

- **Datacenter as a Computer**
  - Scale-out architecture but has all the same components as a single machine
  - High parallelism, diverse workloads, heterogeneous resources, failures, and communication-driven performance

- **GPU as a Computer**
  - Scale-up architecture but has all the same components as a single machine
  - High parallelism, diverse workloads, heterogeneous resources, ~~failures~~, and communication-driven performance (at least for LLMs)

Common "Systems" Techniques

Technique	Performance/Efficiency	Availability/Resilience
Replication & Erasure coding	X	X
Sharding/partitioning	X	X
Scheduling & Load balancing	X	
Health & Integrity checks		X
Compression & Quantization	X	
Centralized controller	X	
Canaries		X
Speculation & Redundant execution	X	

Workload-Centric View

- GenAI Basics
- Pre-Training
- Post-Training
- Inference
- Grounding
- GenAI (for) Systems
- Power and Energy
- Ethical Considerations

GenAI Basics

Transformers

- Quadratic self-attention forms the backbone of modern LLMs
- Compute-bound in training, but memory (bandwidth)-bound during inference

Transformer Alternatives and Extensions

- MoEs increase model capacity by introducing sparsity, but introduces many challenges related to expert balancing
- SSMs approximate transformers using linear scaling in seq. length, while their memory doesn't grow linearly during inference (unlike transformers)

GenAI Basics

Beyond-Text Models

- MLLMs extends transformers beyond text comprehension by putting encoder-projector pairs to convert different modality inputs to LLM embedding space
  - Encoders and LLMs are often frozen during training
- Diffusion process is used to generate images by repeated noise prediction and removal
  - While diffusion models started with UNet at their core, more recently they are being replaced by Diffusion Transformers (DiT) for better scalability
  - Image/video generation workloads are primarily compute bound (and can potentially be communication-bound)

Pre-Training

Megratron-LM

- Introduces 3D parallelism: data, tensor, and pipeline
- Manually optimized parallelism degrees of freedom based on computation and communication requirements

Oobleck

- Planning hybrid/3D parallelism while considering up to **f** failures
- Automated planning using pipeline templates

Pre-Training

FSDP

- Trades of communication to save memory requirement to enable data-parallel training of large models
- Uses combinations of sharding and replication

Tutel

- Dynamically adapts parallelism and pipelining to address expert imbalance issues
- Considers data, model, and expert parallelisms

Pre-Training

ZeRO-Infinity

- Characterizes memory needed for model states, activation states, and working memory and bandwidth needed using arithmetic intensity
- Overall goal is to train large model in memory-constrained scenarios

Activation Recomputation

- Selective recomputation to recompute only a selected set of layers instead of recomputing all layers

Pre-Training

FastFlow

- Improve input data pipelines by balancing mismatch between GPU throughput and the rest

Rail-Only

- Network architecture that matches communication patterns of large model training instead of assuming all-to-all bandwidth requirement

Post-Training

LoRA

- Freezes model weights and injects trainable low-rank decomposition matrices into each layer of the model architecture
- Reduces resource requirement by orders of magnitude, while empirically providing good fine-tuning performance

Sparse Upcycling

- Efficiently trains MoE models from pre-trained dense models

Inference

vLLM

- Fine-grained KV-cache memory allocation
- Support for (virtual) virtual memory and paging in software
- Non-contiguous virtual memory

vAttention

- vLLM using hardware support
- Contiguous virtual memory

Inference

FlashAttention-2

- Tight mapping of model architectures to underlying hardware memory hierarchy

SpecInfer

- Speculative LLM inference using tree-based speculation
- In the best case, it can produce a bunch of tokens while still produces the next token when speculation doesn't work out

Inference

Splitwise

- Statically split the two phases of LLM inference (prefill and decoding) and assign resources accordingly

Llumnix

- Dynamically migrates requests around across servers to use all available resources

Inference

Andes

- Consider the impact of all tokens instead of any specific token (first, last, 90th, or 99th token)
- Introduces the notion of QoE for LLMs
- Preemption within the same request

VTC

- Fairness between multiple requests similar to packet-level fairness in networking
- Preempts at request boundaries to avoid preemption overhead

Inference

dLoRA

- Efficiently serves multiple LoRAs on the same base model
- Dynamically merges/unmerges LoRA adapters with the base model within replicas
- Migrates LoRA adapters across replicas to balance load

MoLE

- Combines multiple LoRAs to create a single MoE
- Hierarchical weight control through learnable gating functions within each layer of trained LoRAs

Inference

AWQ

- Varying levels of quantization for weights of different importance, leading to heterogeneous data types
- Inference-time scaling allows everything to run in FP16

LLM in a Flash

- Tight mapping between model and hardware to take advantage of flash memory characteristics
- Focuses only on FFN/MLP layers and tries to increase ReLU to reduce data transfers

Grounding

Self-RAG

- Introduces a lot of meta tokens during token generation to think, retrieve, critique what's data tokens are being generated
- There is a critic model and a generator model that are both based on the same core dataset with augmentation for the latter

Rummy

- Better uses GPU compute and memory to implement an efficient vector database

Applications

Parrot

- Static DAG of LLMs
- Application-level DAG allows for optimizations in terms of packing and scheduling

RCACoPilot

- Clustering + RAG + LLM etc. to find the root cause of incidents

AI Energy

Perseus

- Reduces internal energy bloat due to differences between critical and non-critical paths of computation
- Reduces external energy bloat due to stragglers and external events like power capping, thermal throttling, etc.

DynamoLLM

- Maintain model replicas with diverse energy-performance tradeoffs
- Elastically scale the number of replicas and route requests accordingly with an aim to reduce overall energy consumption

Ethics

- We must carefully consider MANY aspects when evaluating AI
  - Societal
  - Environmental
  - Financial
  - Etc.

- There is no right answer, but less wrong ones

Papers We Dislike

Final Step

- **Poster presentation on December 5 at Tishman during class hours**
  - Email us final project titles by 10AM tomorrow (Wednesday, December 4)
  - Poster logistics are on piazza (https://piazza.com/class/lyogt3gsjc625o/post/49)

- **Research paper**
  - The key component of the course
  - Should be written and organized like the papers you've read
  - As if you'd submit it to a workshop with ~3 more months of work or to a conference after ~6 more months of work
  - How to Write a Great Research Paper by Simon Peyton Jones
  - Eight pages tops using the MLSys conference's template
  - Due on or before 6:00PM EST on December 16

Thank You!

Please complete the course evaluation