CS 744: MARIUS

Shivaram Venkataraman

Fall 2022

ADMINISTRIVIA

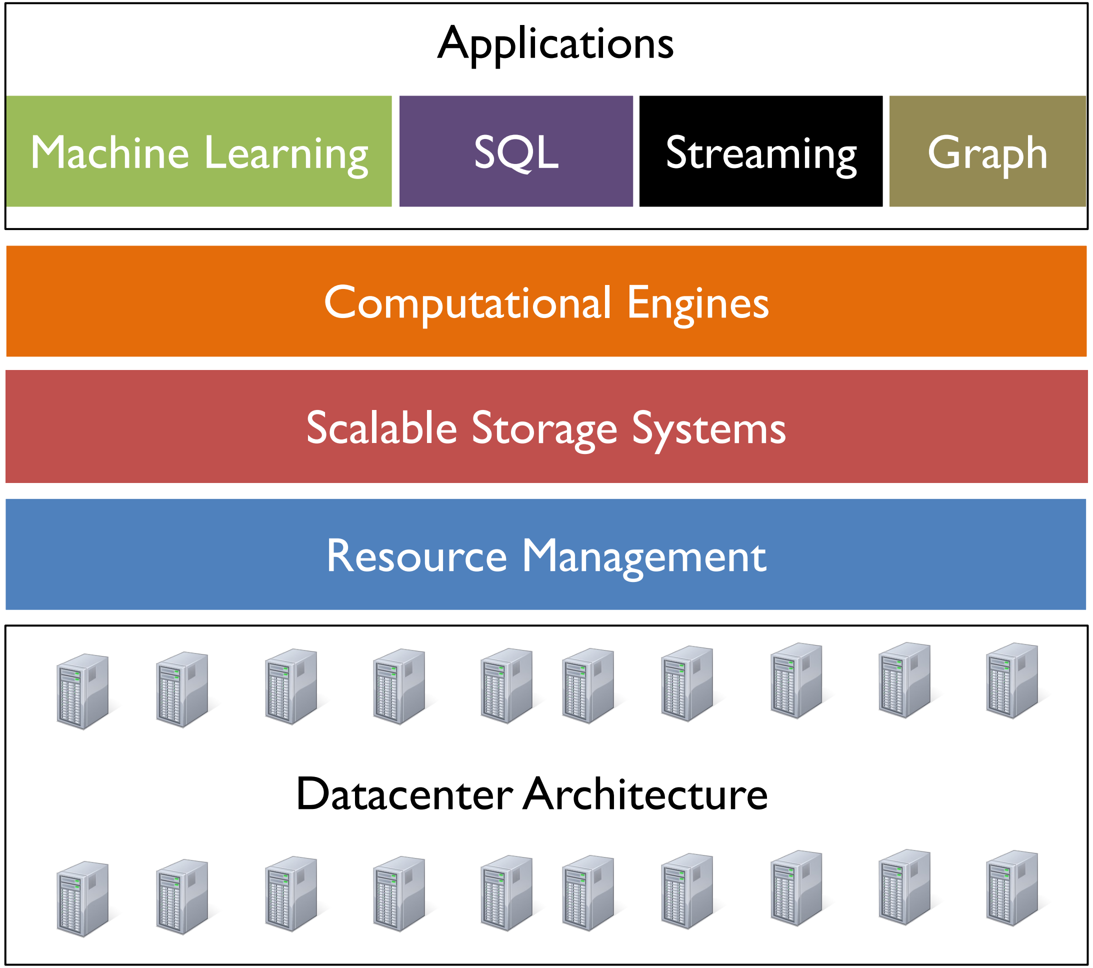
- Midterm grades out!
- Regrade requests: In-person (strongly preferred)
  - Thu: After class, Roger's OH
  - Mon: Shivaram's OH, Roger's OH
  - Tue: After class

- Course Project: Check in by Nov 23th

PROJECT CHECK-INS

One page document that includes the following

- What have you done so far
- Any challenges that you have faced so far
- Your timeline (from now till end of the semester)
- Things you need help from the course staff
- Any other comments/remarks



Applications

Machine Learning

SQL

Streaming

Graph

Computational Engines

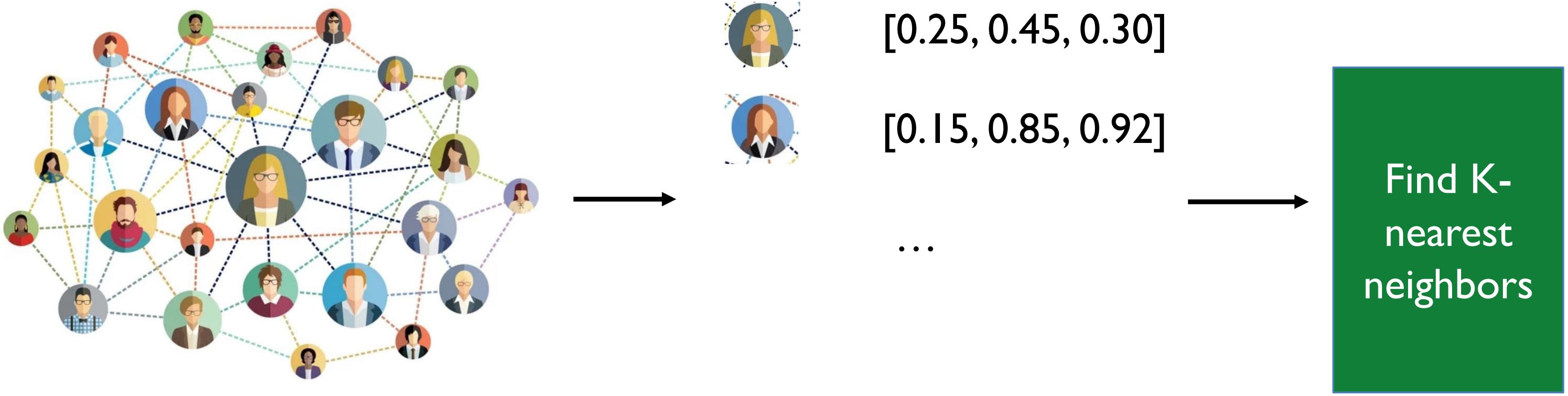
Scalable Storage Systems

Resource Management

Datacenter Architecture

EXAMPLE: LINK PREDICTION

Task: Predict potential connections in a social network



[0.25, 0.45, 0.30]

[0.15, 0.85, 0.92]

…

Find K-nearest neighbors

BACKGROUND: GRAPH EMBEDDING MODELS

Score function

Capture structure of the graph given source, destination embedding

Loss function

Maximize score for edges in graph

Minimize for others (negative edges)

$$\mathcal{L} = \sum_{e \in G} \sum_{e' \in S'_e} \max(f(e) - f(e') + \lambda, 0))$$

TRAINING ALGORITHM

SGD/AdaGrad optimizer

Sample positive, negative edges

Access source, dest embeddings for each edge in batch

```
for i in range(num_batches)
    B = getBatchEdges(i)
    E = getEmbeddingParams(B)
    G = computeGrad(E, B)
    updateEmbeddingParams(G)
```

CHALLENGE: LARGE GRAPHS

Large graphs → Large model sizes

Example

3 Billion vertices, d = 400

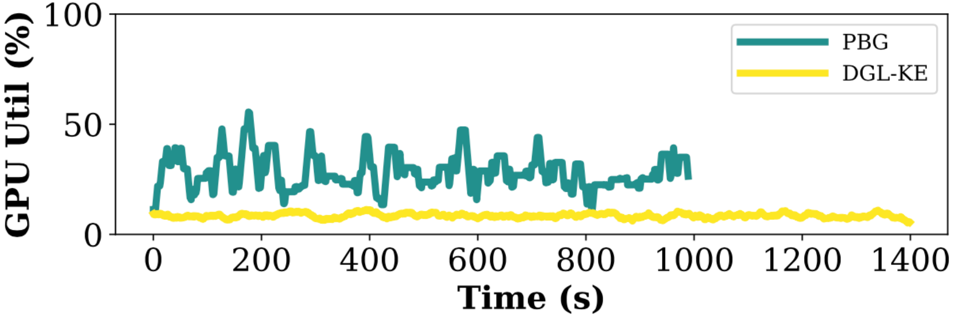
Model size = 3 billion * 400 * 4 = 4.8 TB!

Need to scale beyond GPU memory, CPU memory!

CHALLENGE: DATA MOVEMENT

DGL-KE: Sample edges, embeddings from CPU memory

Pytorch-BigGraph: Partition embeddings so that one partition fits on GPU memory. Load sequentially



One epoch on the Freebase86m knowledge graph

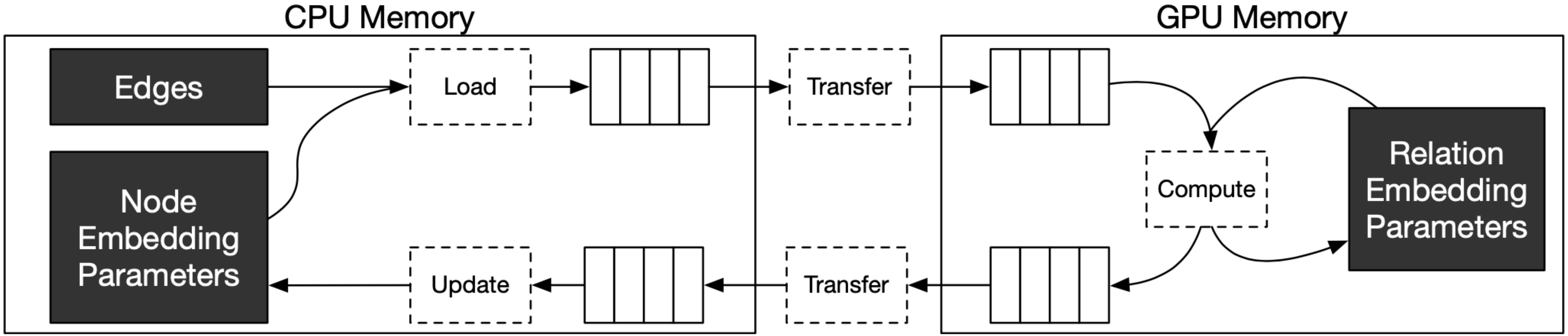
MARIUS

I/O efficient system for learning graph embeddings

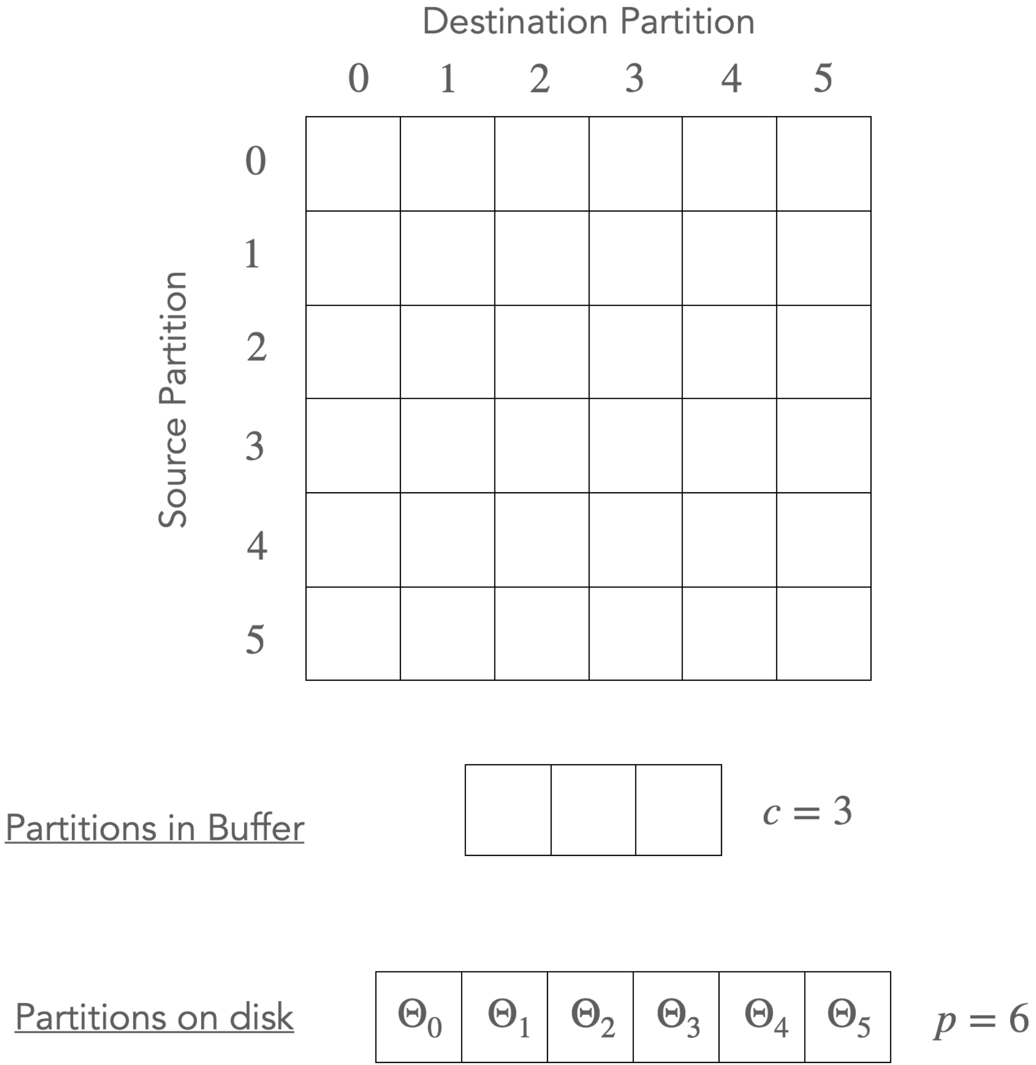
Marius Design

- Pipelined training
- Partition ordering

PIPELINED TRAINING



OUT OF MEMORY TRAINING



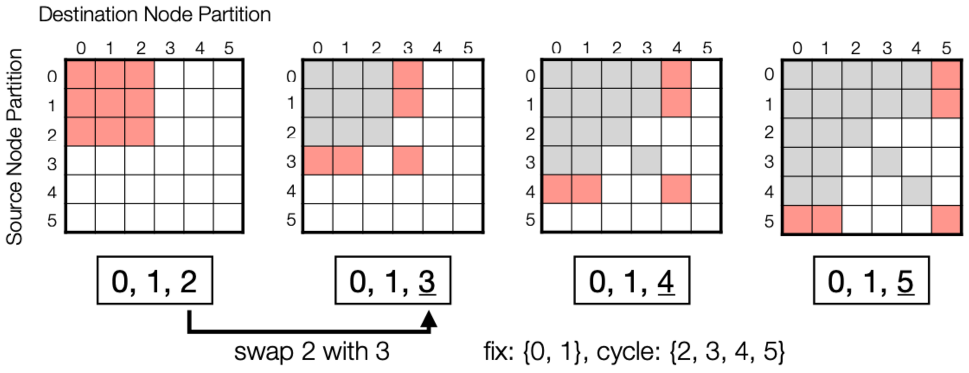
Key idea: Maintain a *cache* of partitions in CPU memory

Questions

Order of partition traversal?

How to perform eviction?

BETA ORDERING



Initialize cache with c partitions

Swap in partition that leads to highest number of unseen pairs

Achieved by fixing c-1 partitions and swap remaining in any order

SUMMARY

Graph Embeddings: Learn embeddings from graph data for ML

Marius: Efficient single-machine training

Pipelining to use CPU, GPU

Partition buffer, BETA ordering

DISCUSSION

https://forms.gle/uNAKsPsZp56Cc1Vz9

How does the partitioning scheme used in this paper differ from partitioning schemes used in PowerGraph and why?

System	Deployment	Epoch Time (s)	Per Epoch Cost ($)
Marius	1-GPU	727	**.61**
DGL-KE	2-GPUs	1068	1.81
DGL-KE	4-GPUs	542	1.84
DGL-KE	8-GPUs	277	1.88
DGL-KE	Distributed	1622	2.22
PBG	1-GPU	3060	2.6
PBG	2-GPUs	1400	2.38
PBG	4-GPUs	515	1.75
PBG	8-GPUs	419	2.84
PBG	Distributed	1474	2.02

What are some shortcomings of Marius? What could the authors do to further improve the system?

NEXT STEPS

Next class: Distributed GNNs

Project check-ins by Nov 23th