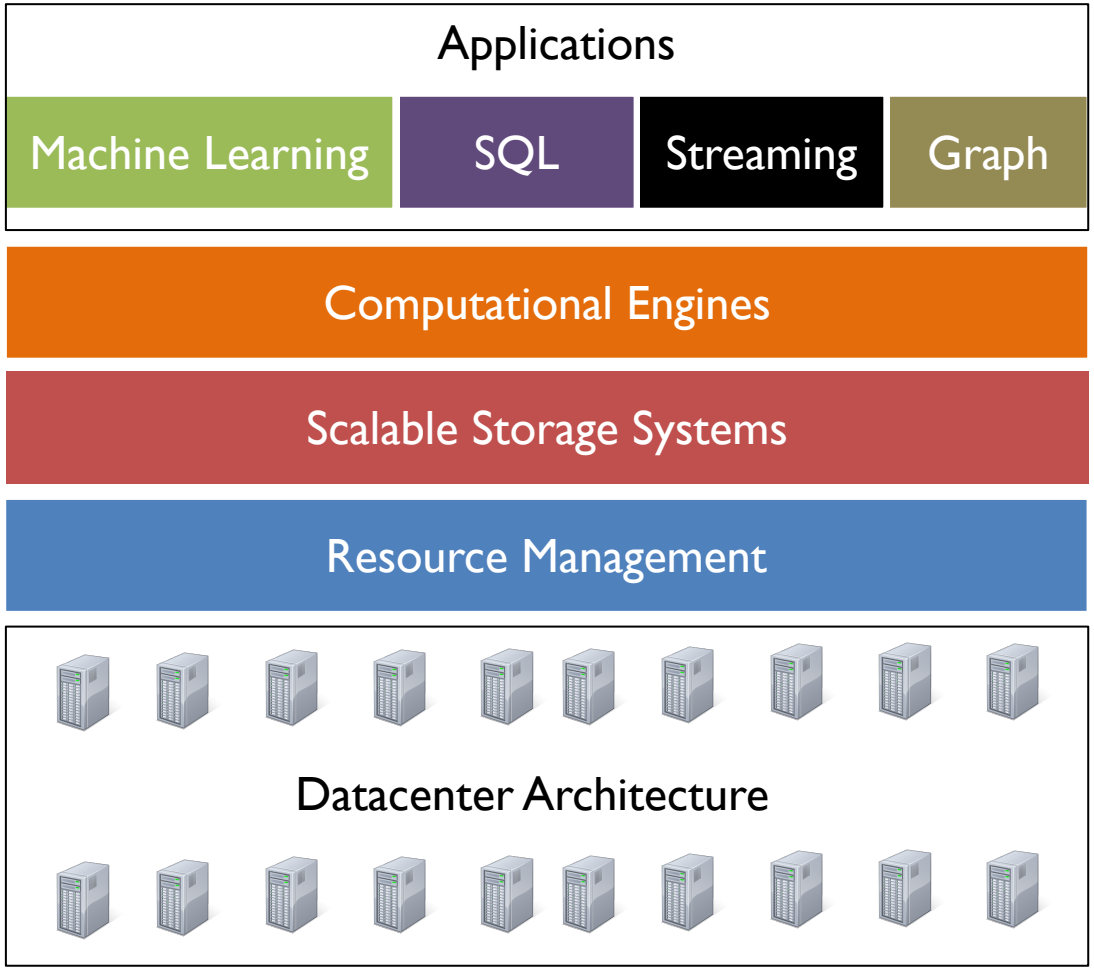
# CS 744: MARIUS

Shivaram Venkataraman

Fall 2021

# ADMINISTRIVIA

- Midterm grades today!
- Course Project: Check in by Nov 30th


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

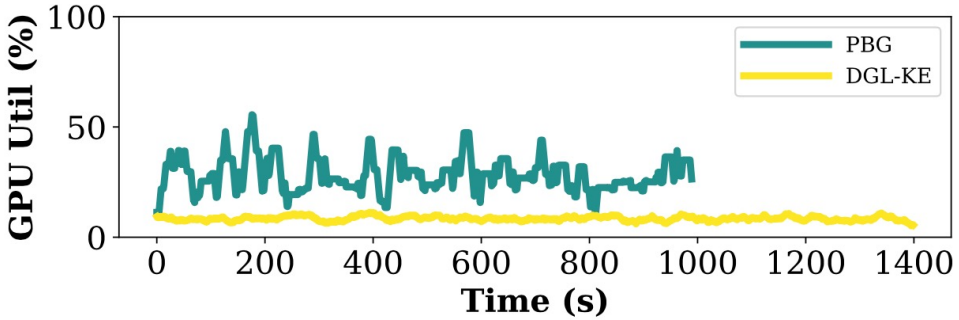
Model size = 3 billion * 400 * 4 = 4.8 TB!

Need to scale beyond GPU memory, CPU memory!

# CHALLENGE: DATA MOVEMENT

(a) Sample edges, embeddings from CPU memory (DGL-KE)

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




One epoch on the Freebase86m knowledge graph

**Data movement overheads → low GPU util**

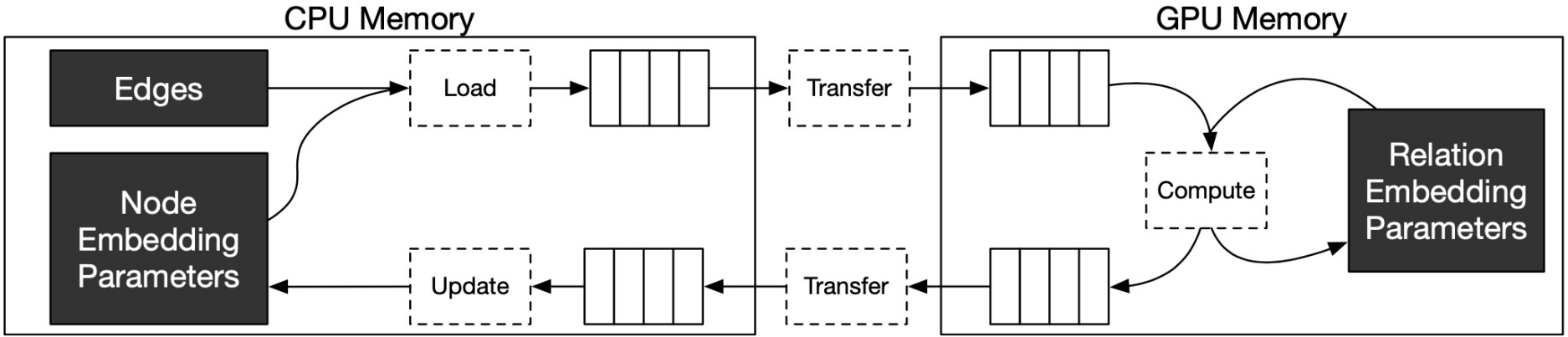
# MARIUS

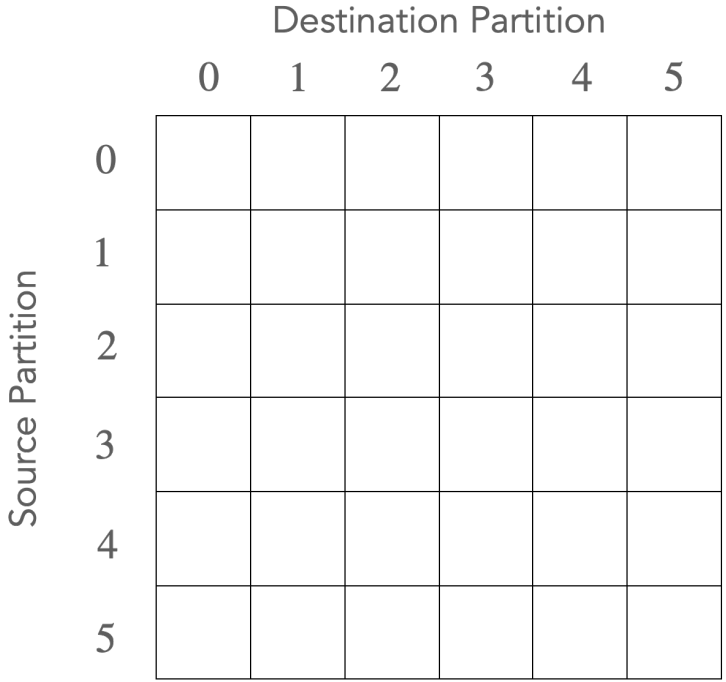
I/O efficient system for learning graph embeddings

Marius Design

- Pipelined training
- Partition ordering

# PIPELINED TRAINING





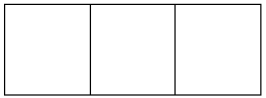

# OUT OF MEMORY TRAINING

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

Questions

- Order of partition traversal?
- How to perform eviction?

Partitions in Buffer

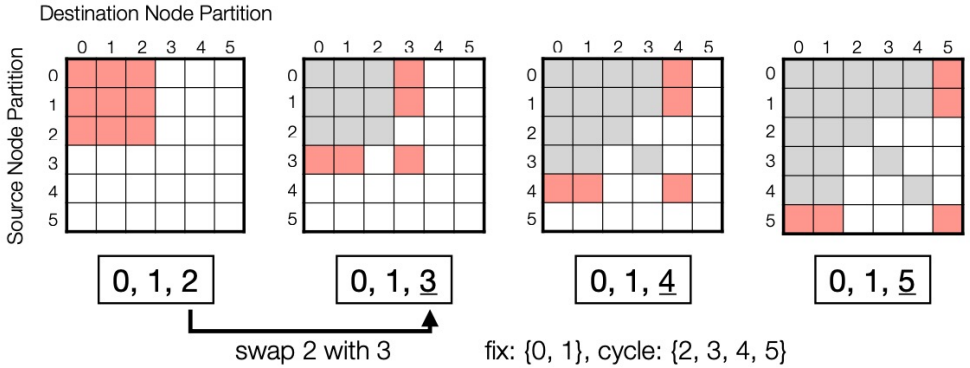


$c = 3$

Partitions on disk


$p = 6$

# ELIMINTATION 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/LtoT8nEmw3oLvXuo9

If you were going to repeat the COST analysis for knowledge graph embedding training, what would you expect to find and why?

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

# NEXT STEPS

Next class: New module!

Project check-ins by Nov 30th