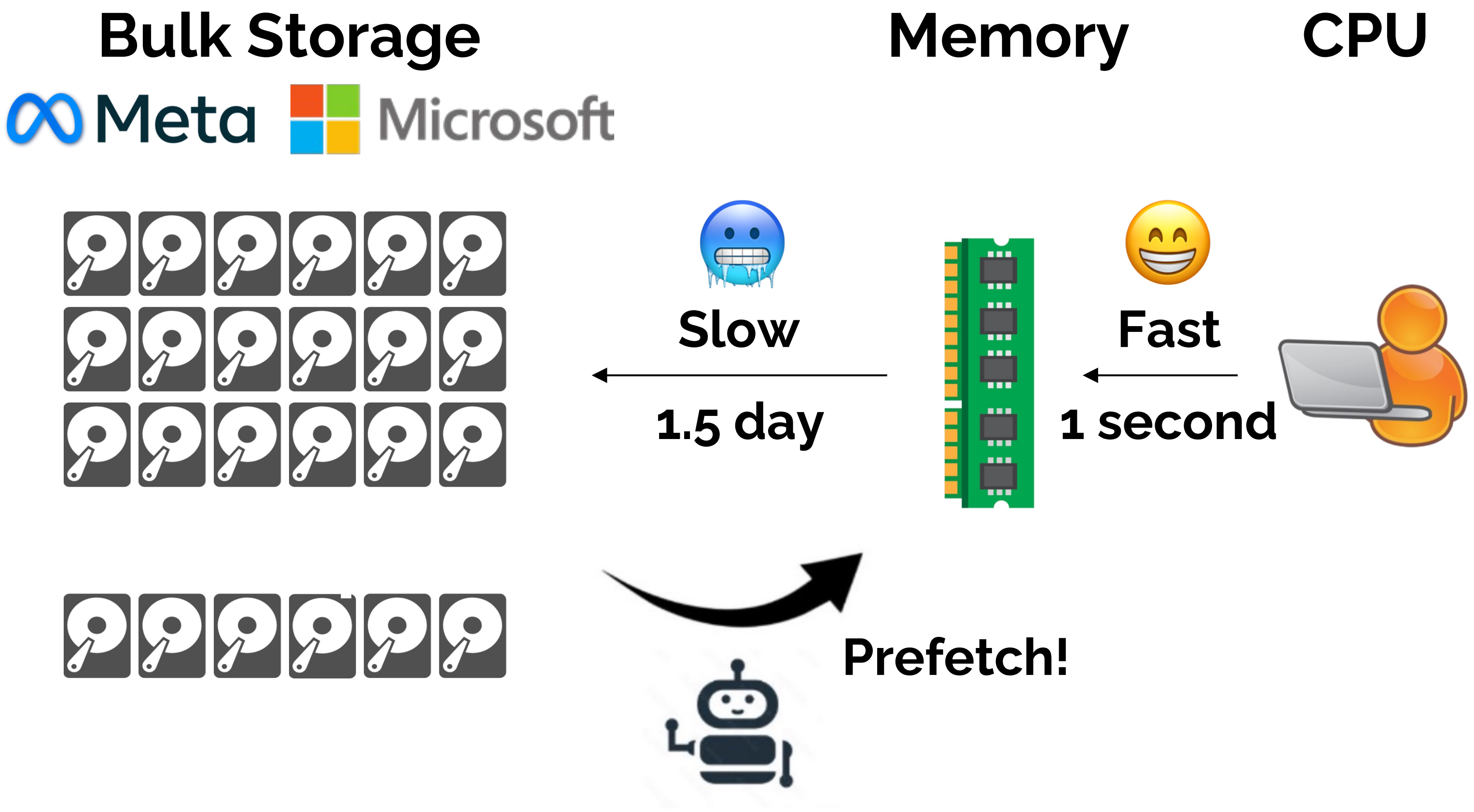
Leveraging AI for Faster Storage Access: a Graph-Neural-Network-Based Prefetcher

Faradawn Yang

Advisor: Haryadi S. Gunawi

THE UNIVERSITY OF CHICAGO

Backend storage is slow



Bulk Storage

Meta Microsoft

Memory

CPU

Slow

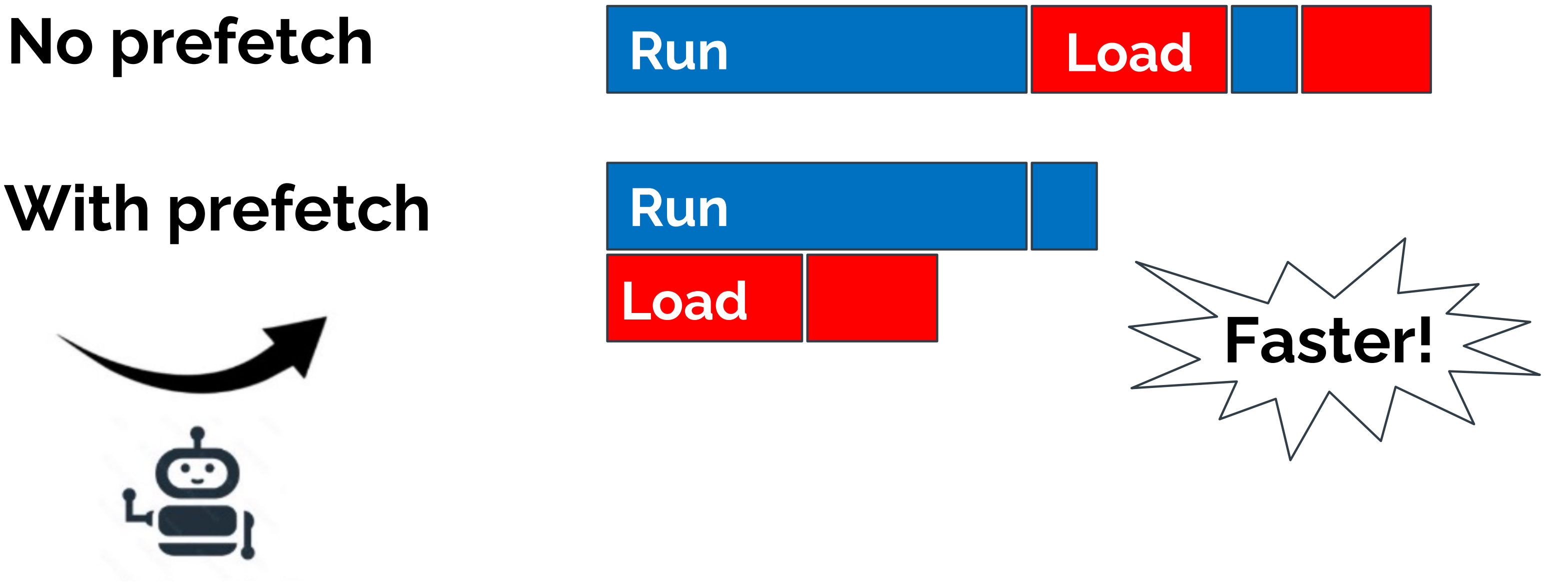
1.5 day

Fast

1 second

Prefetch!

Prefetching improves access speed



No prefetch

Run

Load

With prefetch

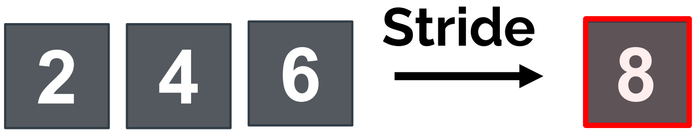
Run

Load

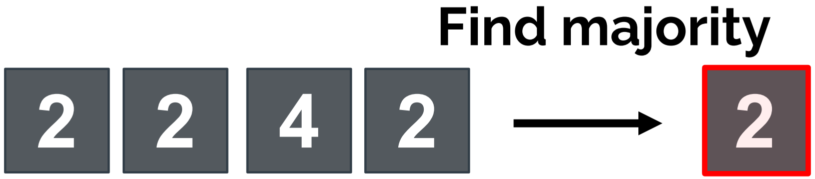
Faster!

Previous work

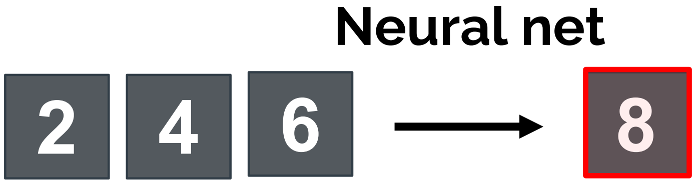
['92] Stride



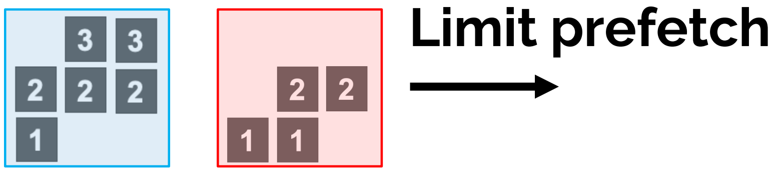
['20] Leap



['21] LSTM

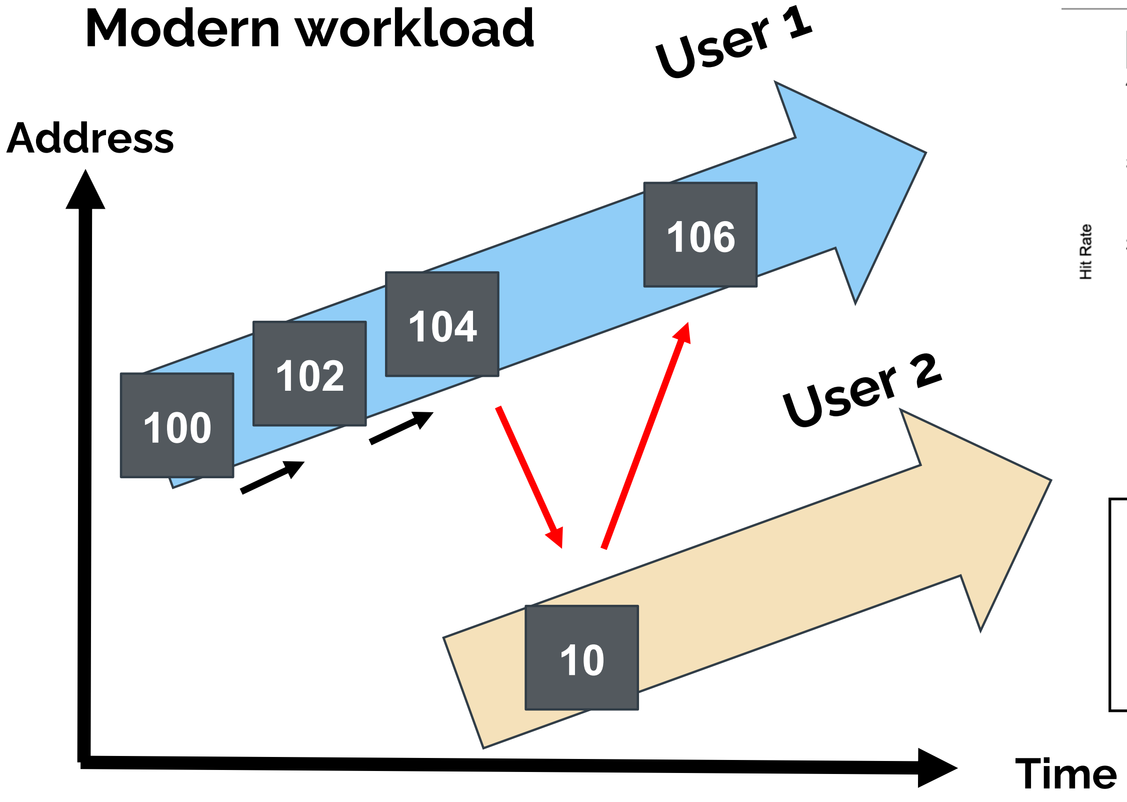


['24] Baleen

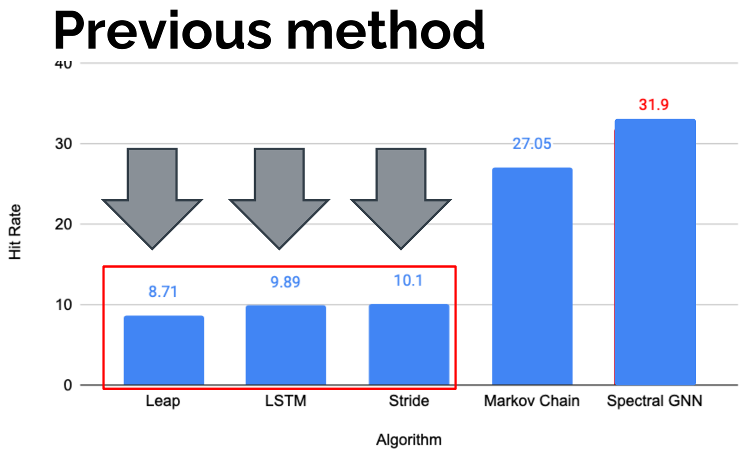


Current problem

Modern workload



Average hit rate on Alibaba, Tencent, and Microsoft traces

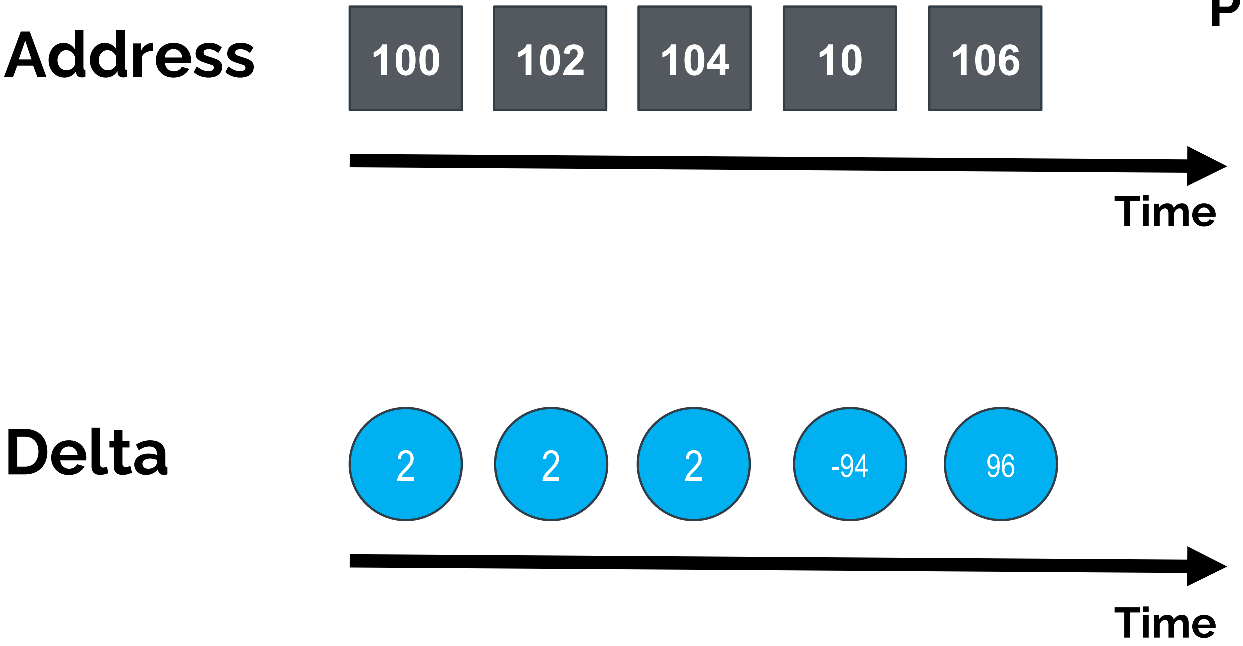


1) Not prone against short-term irregularities.

2) Can’t see a global picture.

Methodology

Step 1: Convert to deltas



Address

100 102 104 10 106

Time

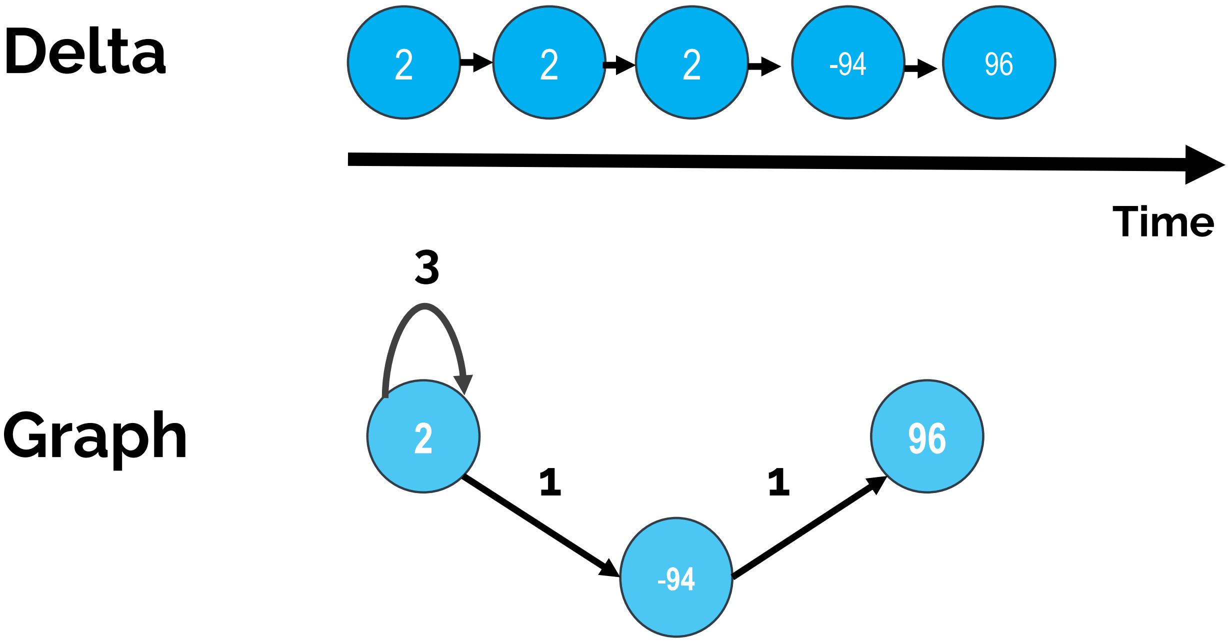
Delta

2 2 2 -94 96

Time

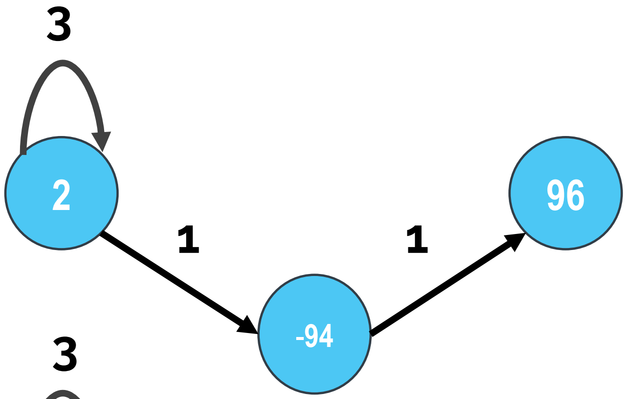
Past 32 accesses.

Step 2: Build a graph



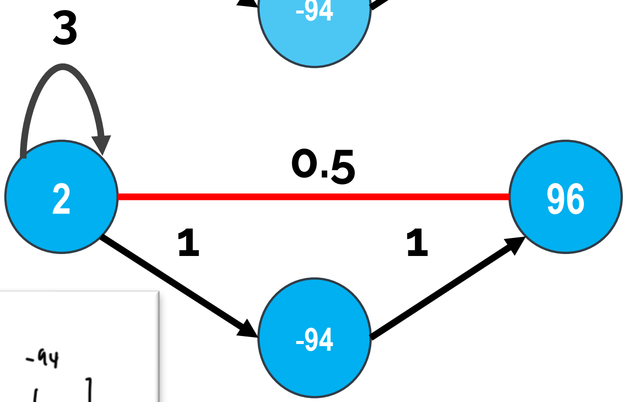
Step 3: Add full connections

Graph



Directed edges capture temporal information.

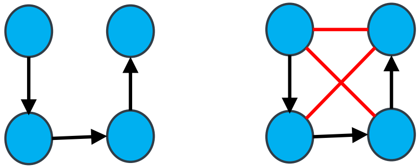
Fully-connected Graph



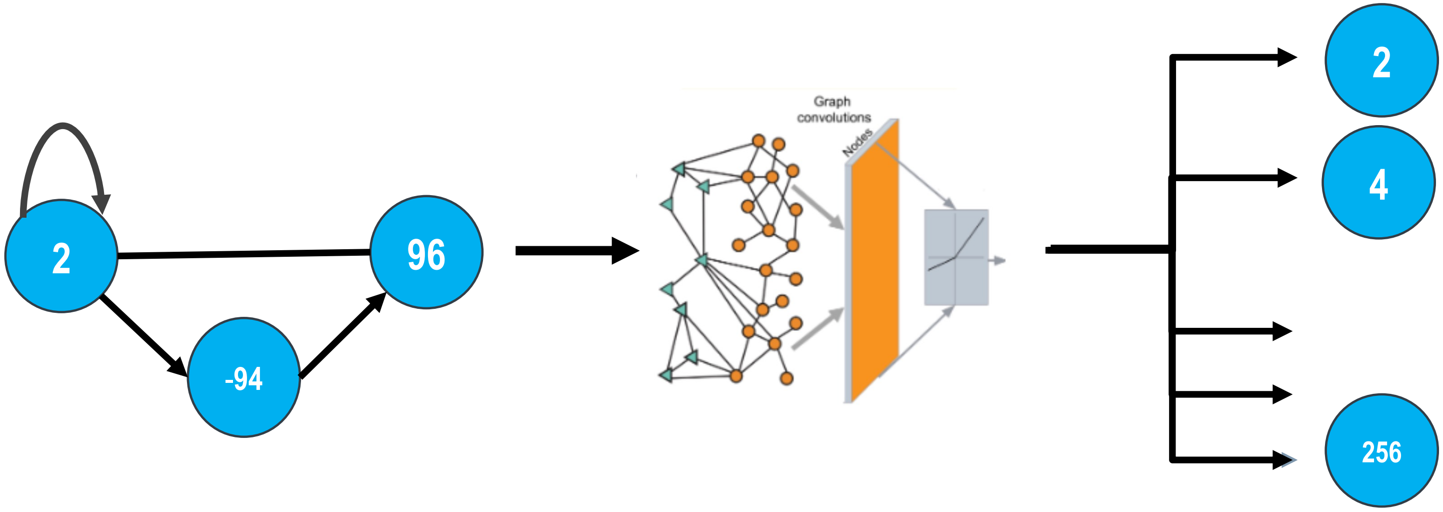
$$\begin{array}{c|ccc} & 2 & 96 & -94 \\ \hline 2 & 3 & 0.5 & 1 \\ 96 & 0.5 & 0 & 0 \\ -94 & 0 & 1 & 0 \end{array}$$

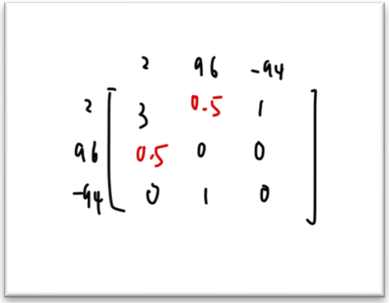
Fully-connected edges reveal spatial pattern.

Another example



Step 4: Feed into ML

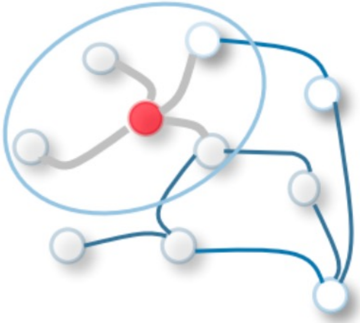




$$\begin{array}{c|ccc} & 2 & 96 & -94 \\ \hline 2 & 3 & 0.5 & 1 \\ 96 & 0.5 & 0 & 0 \\ -94 & 0 & 1 & 0 \end{array}$$

1000 delta addresses.

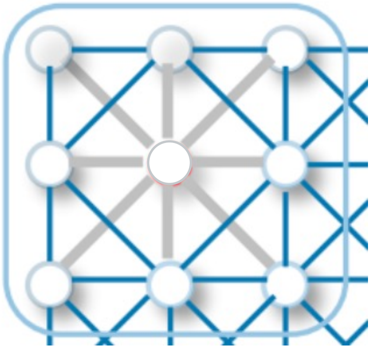
What ML model



[SGDP ‘23]

Message passing

Local patterns.
Many iterations.
😞



[Ours]

Spectral network

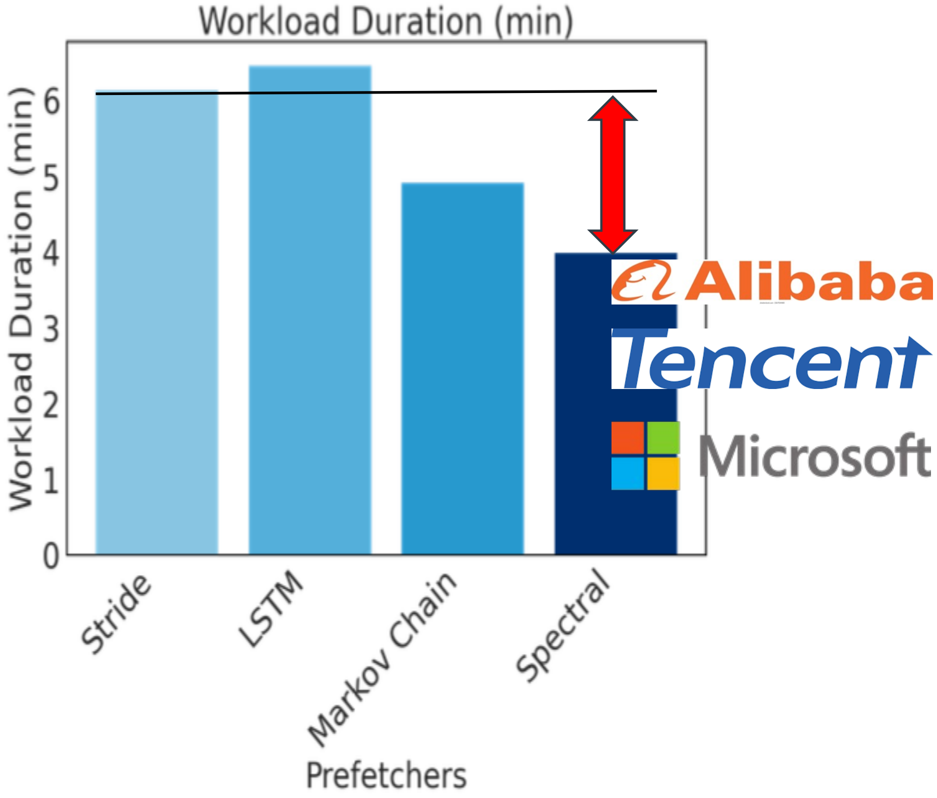
Global structure.
One pass.
😄

Evaluations

Faster access speed

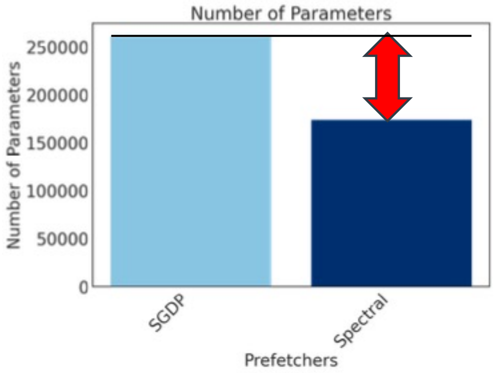
- Setup: python simulator.
- Respected arrival time of requests

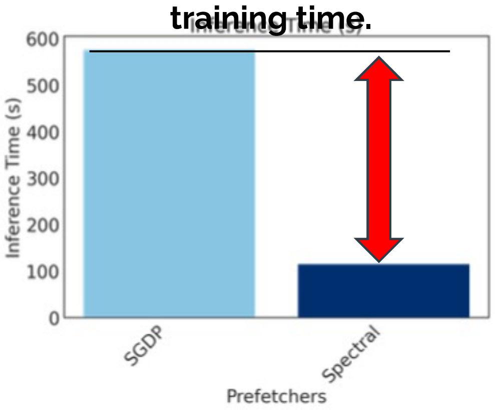
Saves 33% access time!



Saves memory and time

- Similar hit rate, but much smaller model and faster training and inference than state-of-the-art, [SGDP '23].



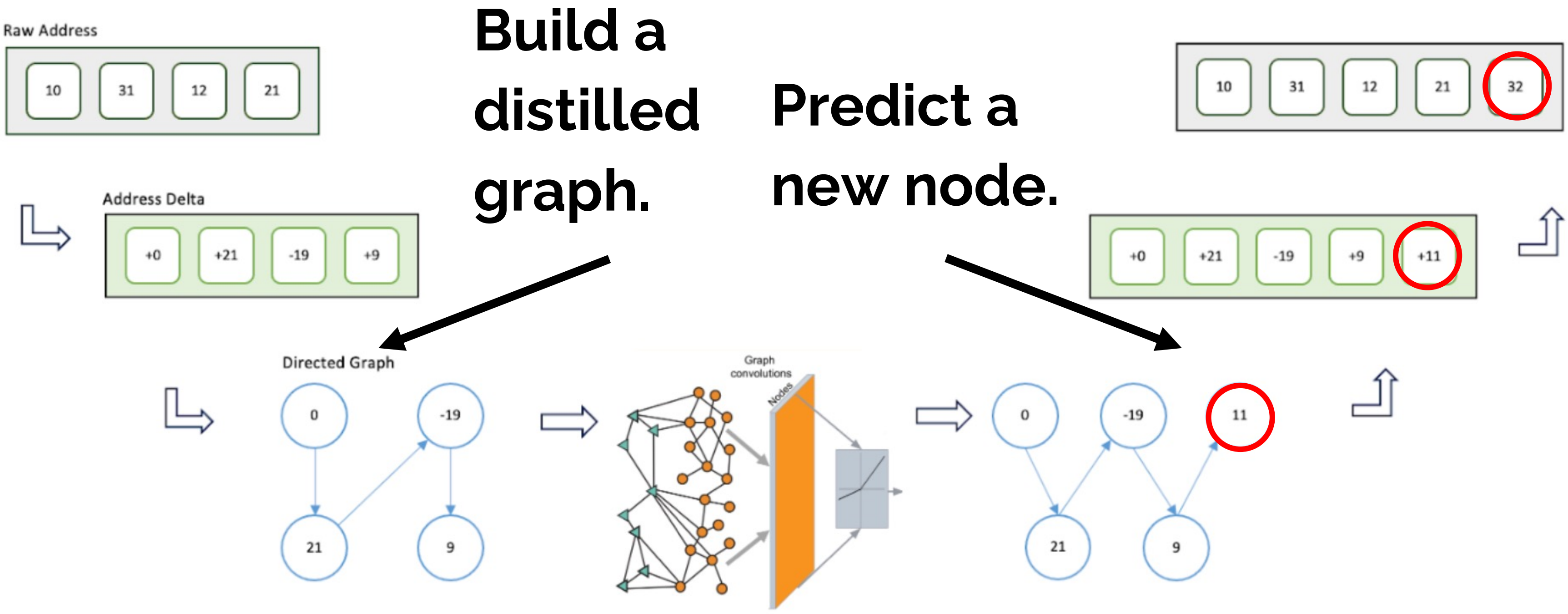


Saves 33.4% model size.

Saves 79% training time.

Conclusion

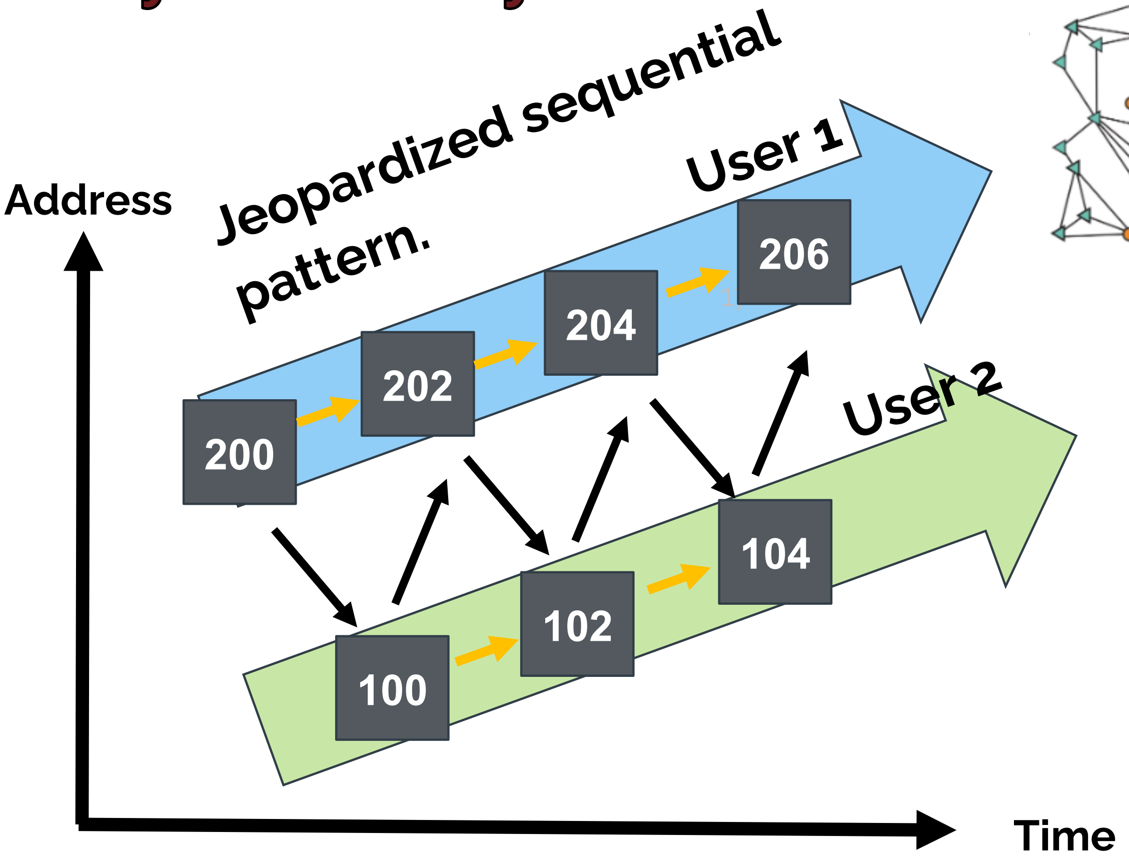
Entire pipeline

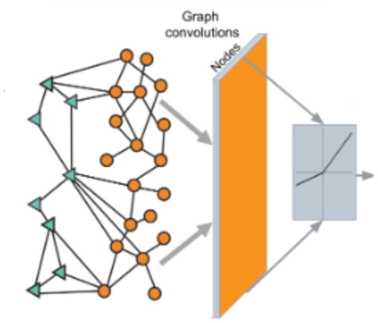


Build a distilled graph.

Predict a new node.

Key takeaway





Use graph to detect a global pattern.

Thank you!

Leveraging AI for Faster Storage Access: a Graph-Neural-Network-Based Prefetcher

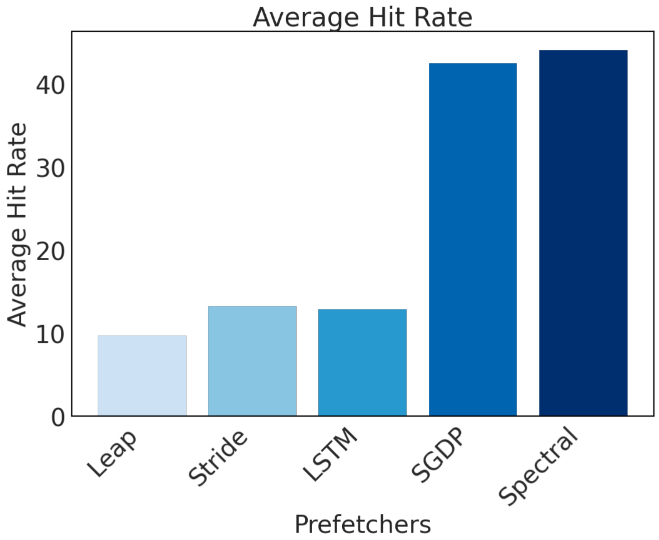
Backup Slides

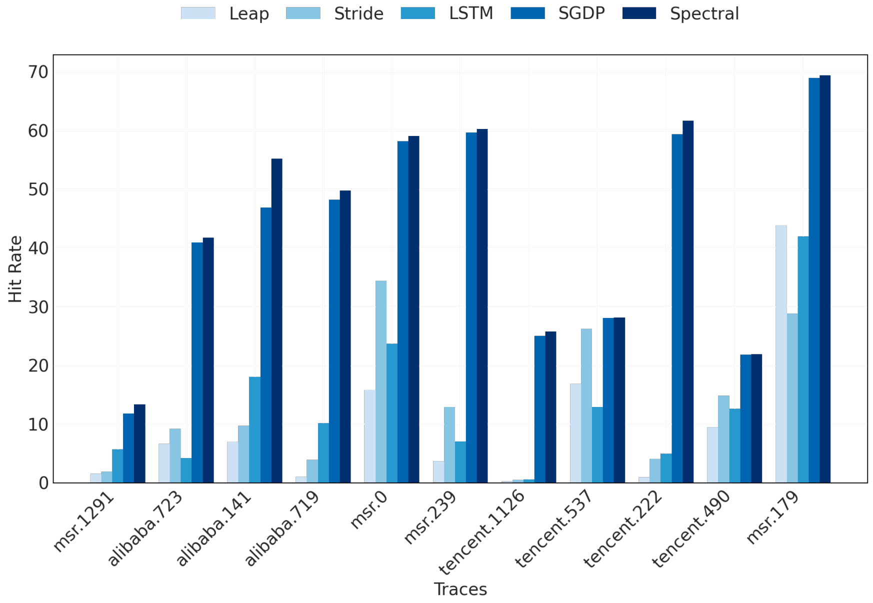
Is graph building too costly

- Only putting 32 integers into a matrix.
- Main Bottleneck is disk access (20ms), then inference (2ms), then graph building (1ms).
- Spectral < SGDP < LSTM < transformer.

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Higher hit rate





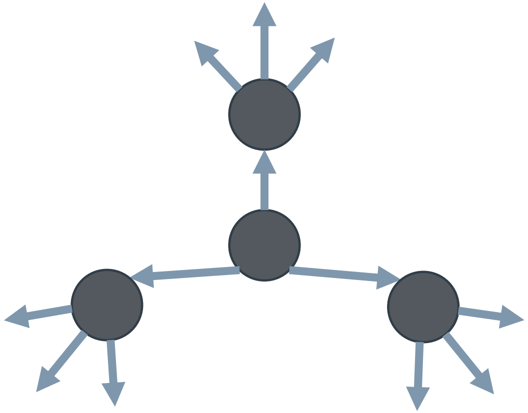
Why spectral method

Message passing

$$m_v^{(t+1)} = \sum_{u \in \mathcal{N}(v)} M^{(t)}\left(h_v^{(t)}, h_u^{(t)}, e_{uv}\right)$$

$$h_v^{(t+1)} = U^{(t)}\left(h_v^{(t)}, m_v^{(t+1)}\right)$$

Many iterations



Spectral method

$$H^{(2)} = \sigma_2\left(\hat{A}\sigma_1\left(\hat{A}XW^{(0)}\right)W^{(1)}\right)$$

One go