SeLeP: Learning Based Semantic Prefetching for Exploratory Database Workloads

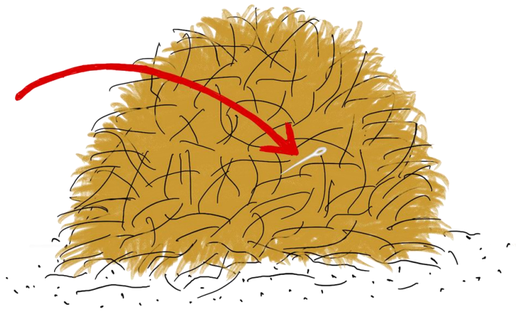
Farzaneh Zirak, Farhana Choudhury, and Renata Borovica-Gajic

School of Computing and Information Systems

August 2024

Data Exploration

Data Exploration — searching for hidden (unknown) information



The fastest-growing black hole of past 9 billion years has just been discovered.

Now, an international team of astronomy researchers say a student has accidentally discovered this supermassive black hole that gobbles up the equivalent of one Earth every minutes. This black hole is also the biggest. "Researchers have been looking for these kinds of objects since 1960s", and "**somehow, this one seemed to have *escaped* all our previous efforts to find it**". said lead author Christopher Onken, an astronomer at the Australian National University…

- June 2022

Not always sure what we are looking for (until we find it!)

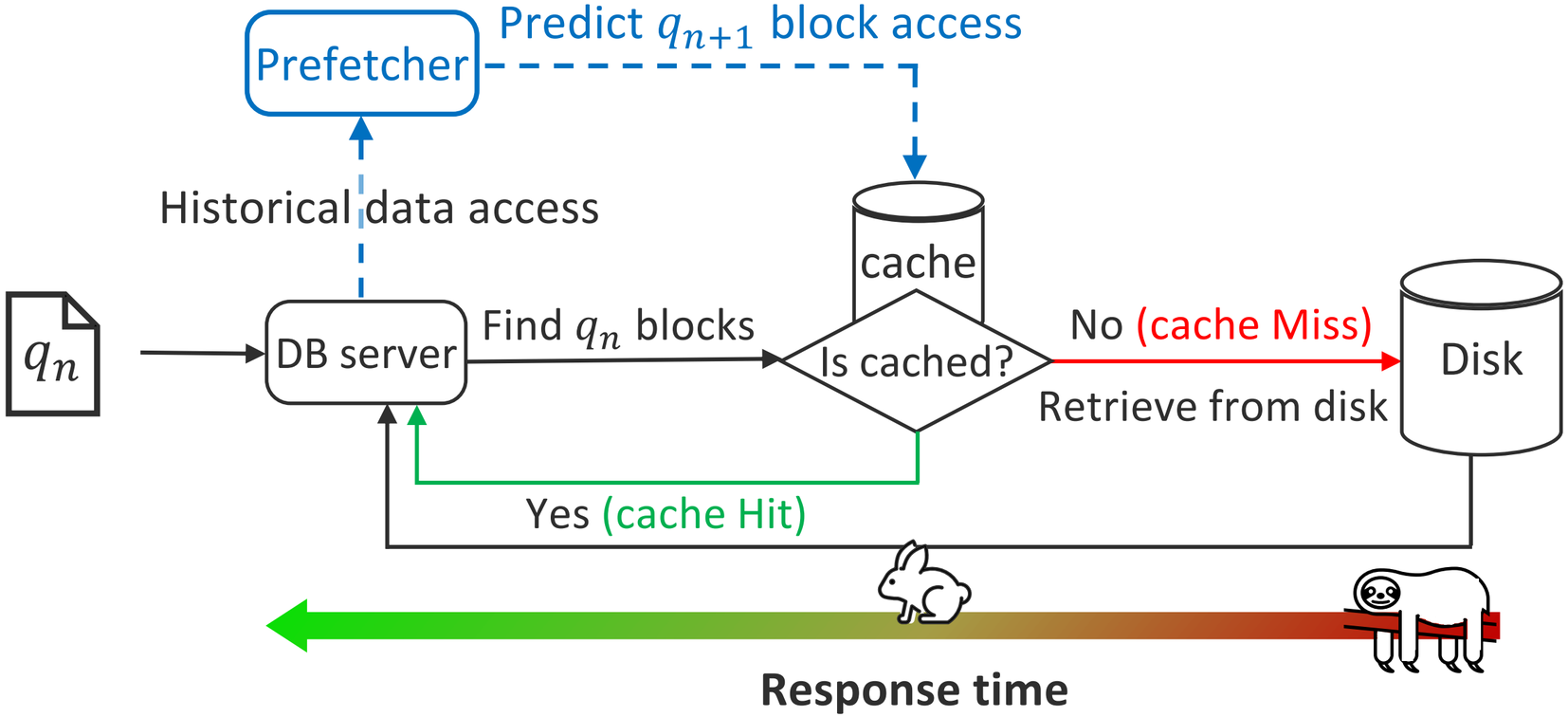
Unsuccessful exploration

Exploratory tools need high interactivity and adaptivity

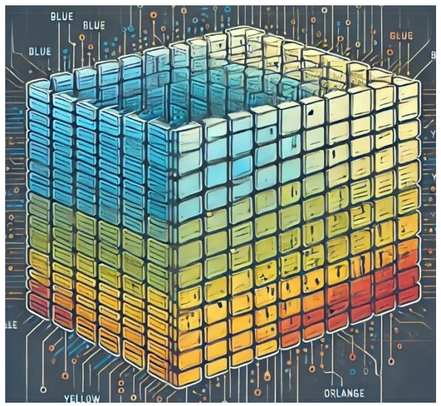
- **Predict and cache blocks likely to be accessed soon to reduce query execution time (user wait time)**

Prefetching in the Current Landscape

- ☒ **Not suitable for SQL workloads**
- ☒ **Work with logical block address—No data semantics**
- ☒ **Not adaptive**



Motivation

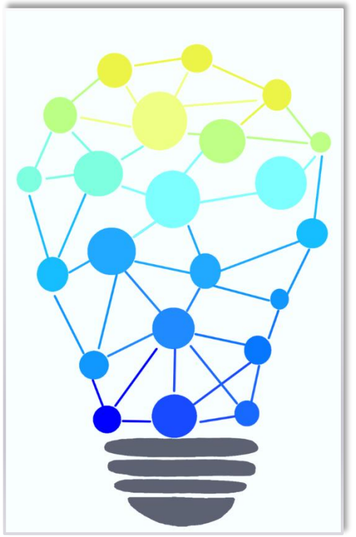


Data semantics are important

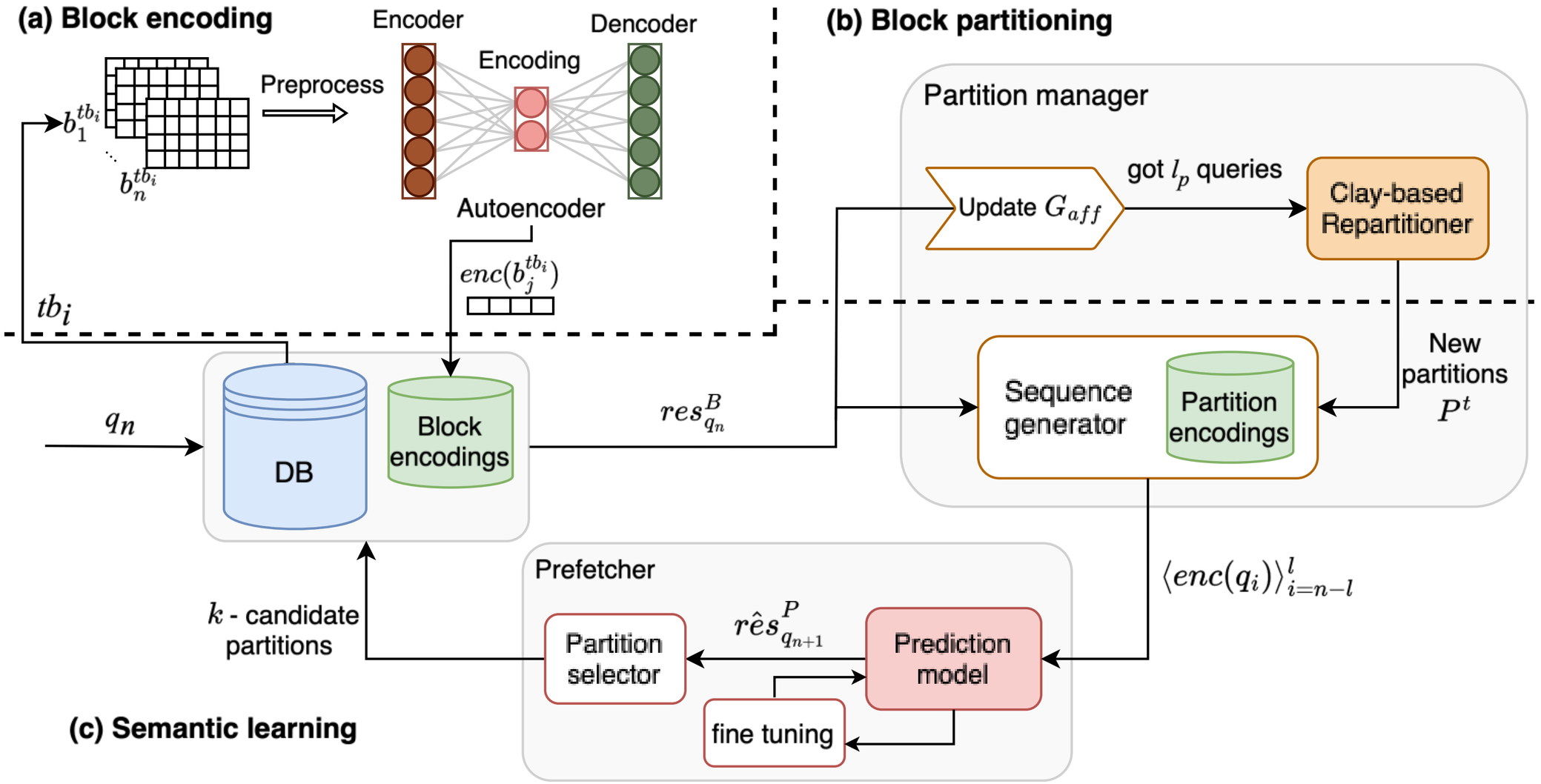
There usually is an inter-dependency among values stored in the data blocks accessed together

Prefetching → **time series forecasting**

Results observed from the queries in the previous time steps form the upcoming queries

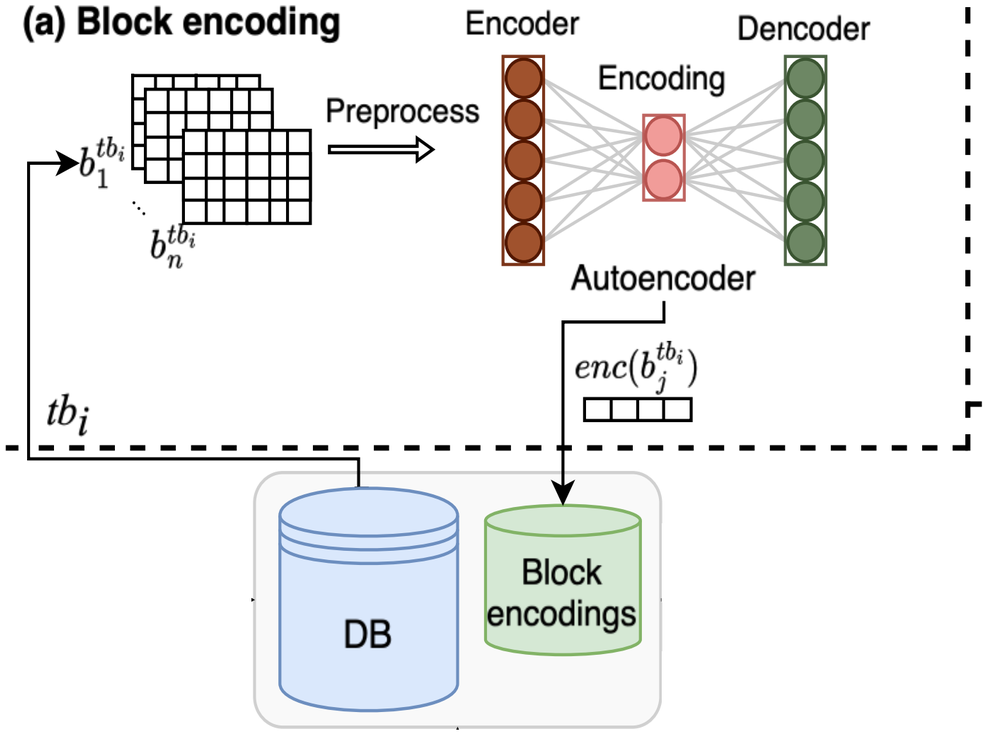


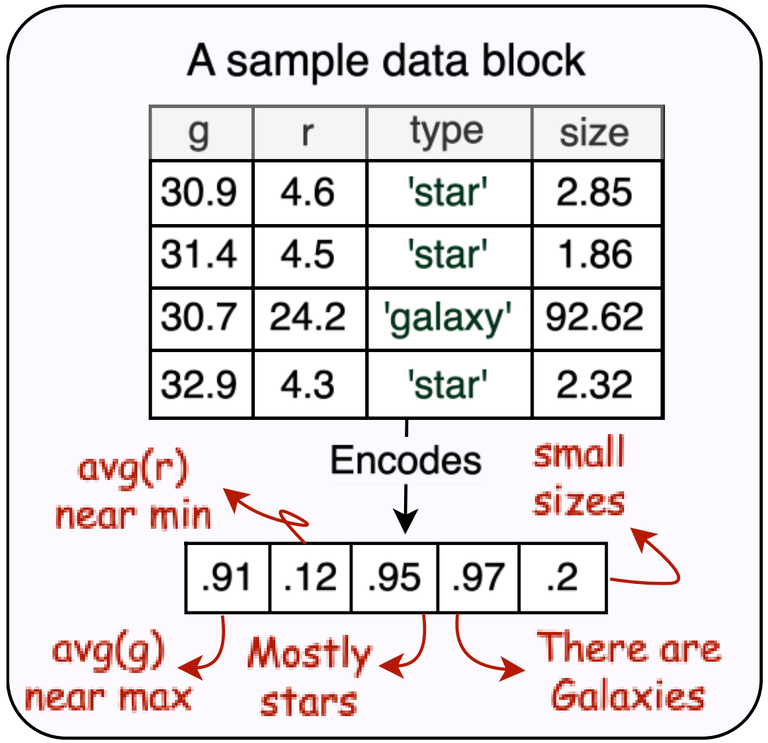
SeLeP Overview



Block Encoding

- Block can contain hundreds of values
- Need a concise block representation which captures the distinctive characteristics of the data
  - ➢ Encode blocks into vectors and aggregate them to form query encodings





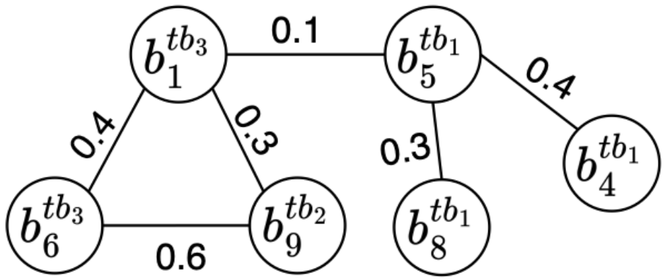
Block Partitioning

- Classification problem: Having the sequence of last l query encodings, predict and fetch blocks will be accessed next

Large dataset → Substantial number of labels

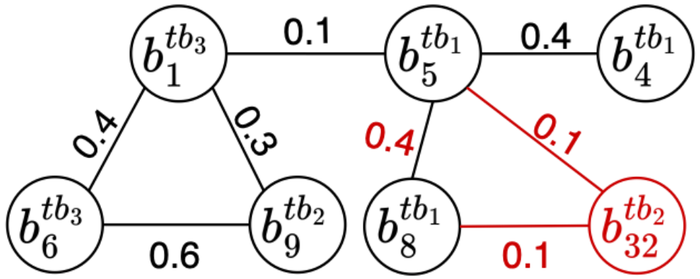
- Group blocks frequently accessed together into partitions

Previous G_{aff} :



$res_{q_n}^{B}$: $b_5^{tb_1}$ $b_{32}^{tb_2}$ $b_8^{tb_2}$

Updated G_{aff} with l_p = 10 :



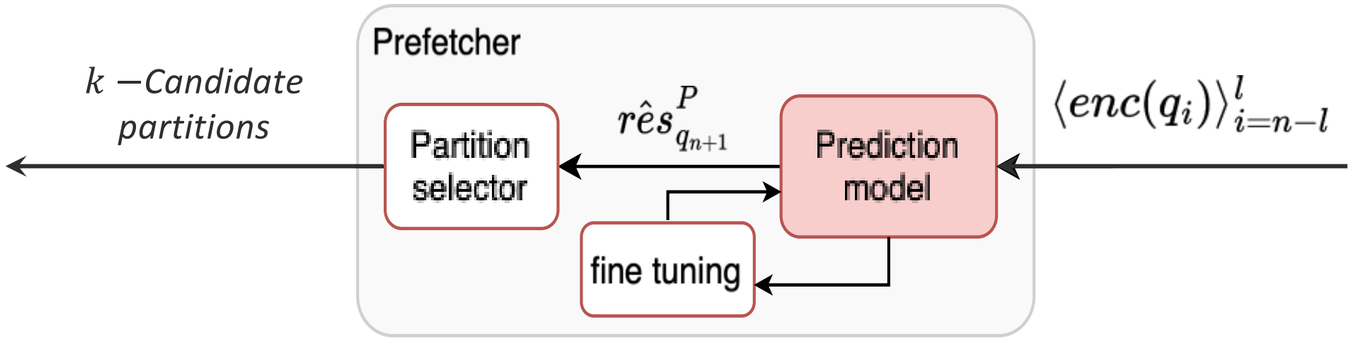
tb_1	0	0	0	0	0
tb_2	.89	.21	.79	.68	.24
tb_3	0	0	0	0	0
tb_4	.4	.13	.33	.19	.81

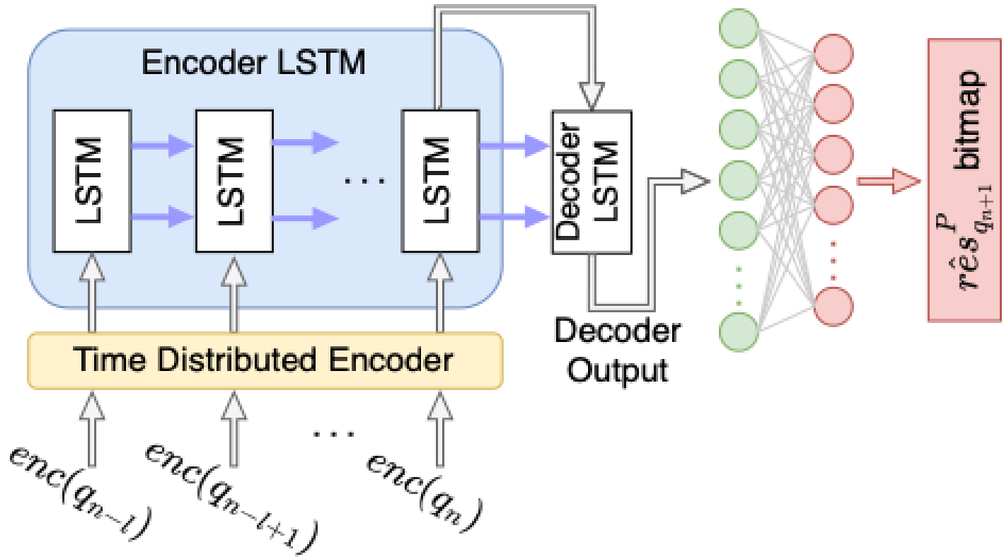
Partition encodings

Graph partitioning on affinity graph

Semantic Learning

Learn partition access pattern from sequence of query encodings and fine tune the model with new workloads

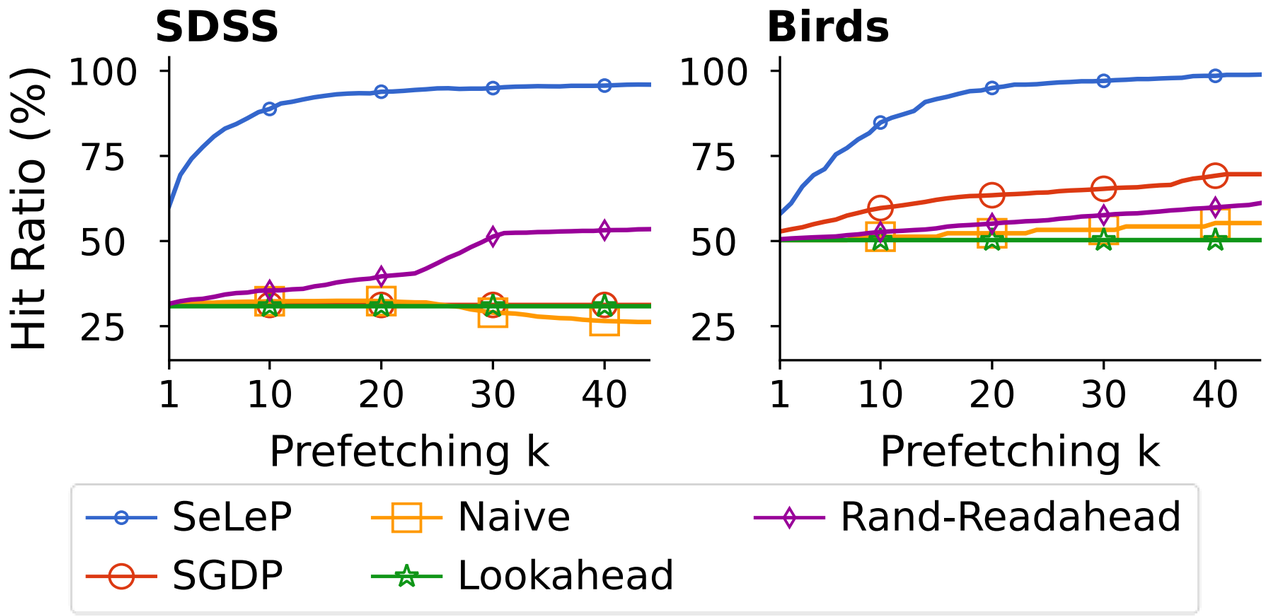




SeLeP Hit Ratio - SQL Workloads

Setting: 16GB SDSS DR7, 8GB Birds, prefetch size = k ×128 block, 4GB cache

Queries: SQL workloads



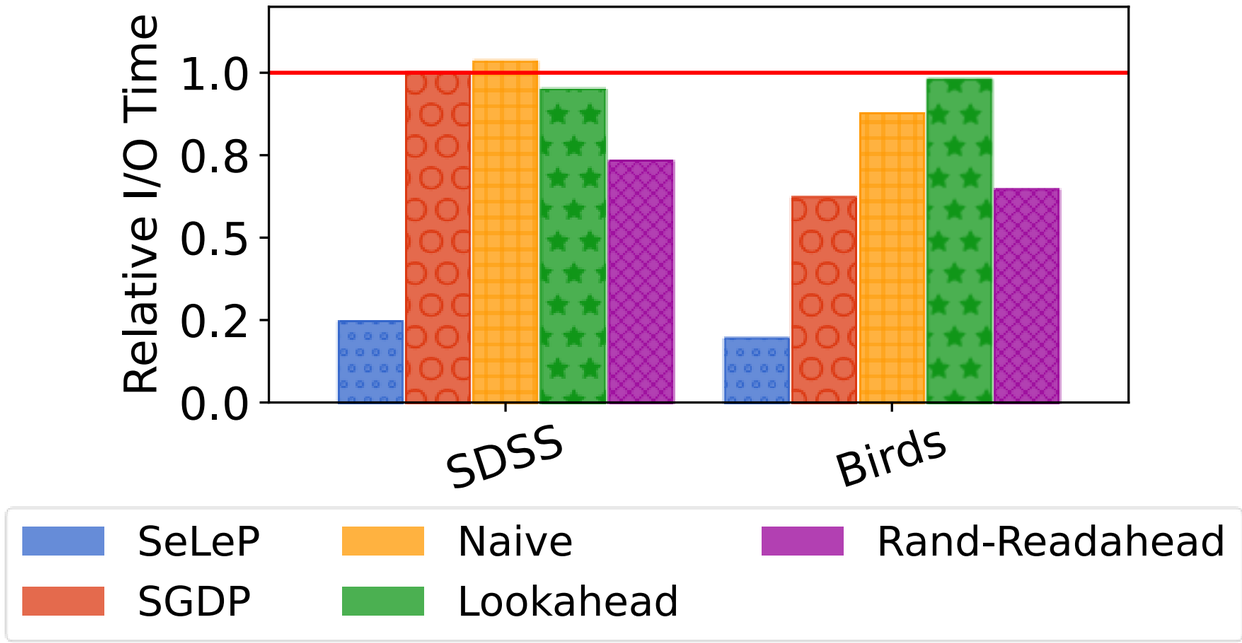
$$\text{Hit ratio} = \frac{\#\text{hits}}{\#\text{hits}+\#\text{misses}}$$

- **95%** hit ratio on **average**
- Up to **40% improvement** compared to the traditional and state-of-the-art

SeLeP I/O time - SQL Workloads

Setting: 16GB SDSS DR7, 8GB Birds, prefetch size = k ×128 block, 4GB cache

Queries: SQL workloads

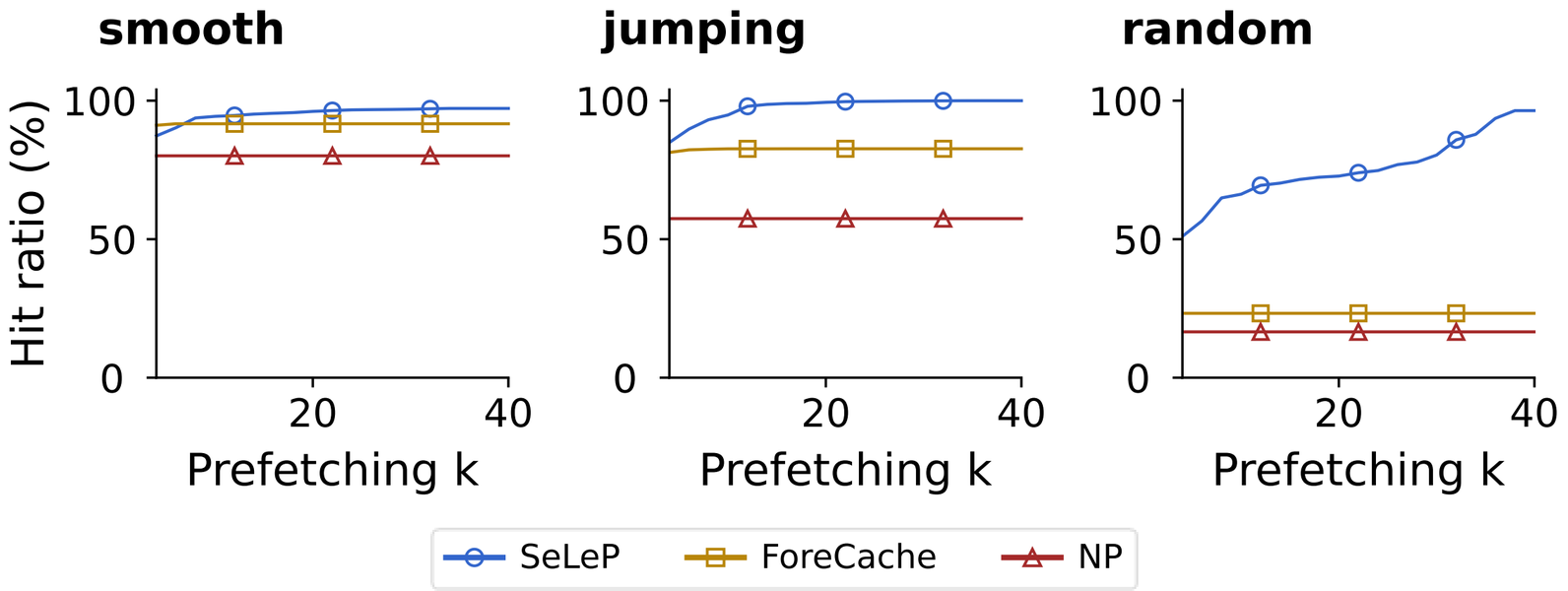


- **80%** I/O time reduction on **average**
- Up to **45% improvement** compared to the traditional and state-of-the-art

SeLeP - Navigational Workloads

Setting: 8GB SDSS DR7, prefetch size = k ×64 block, 2GB cache

Queries: Navigational workloads

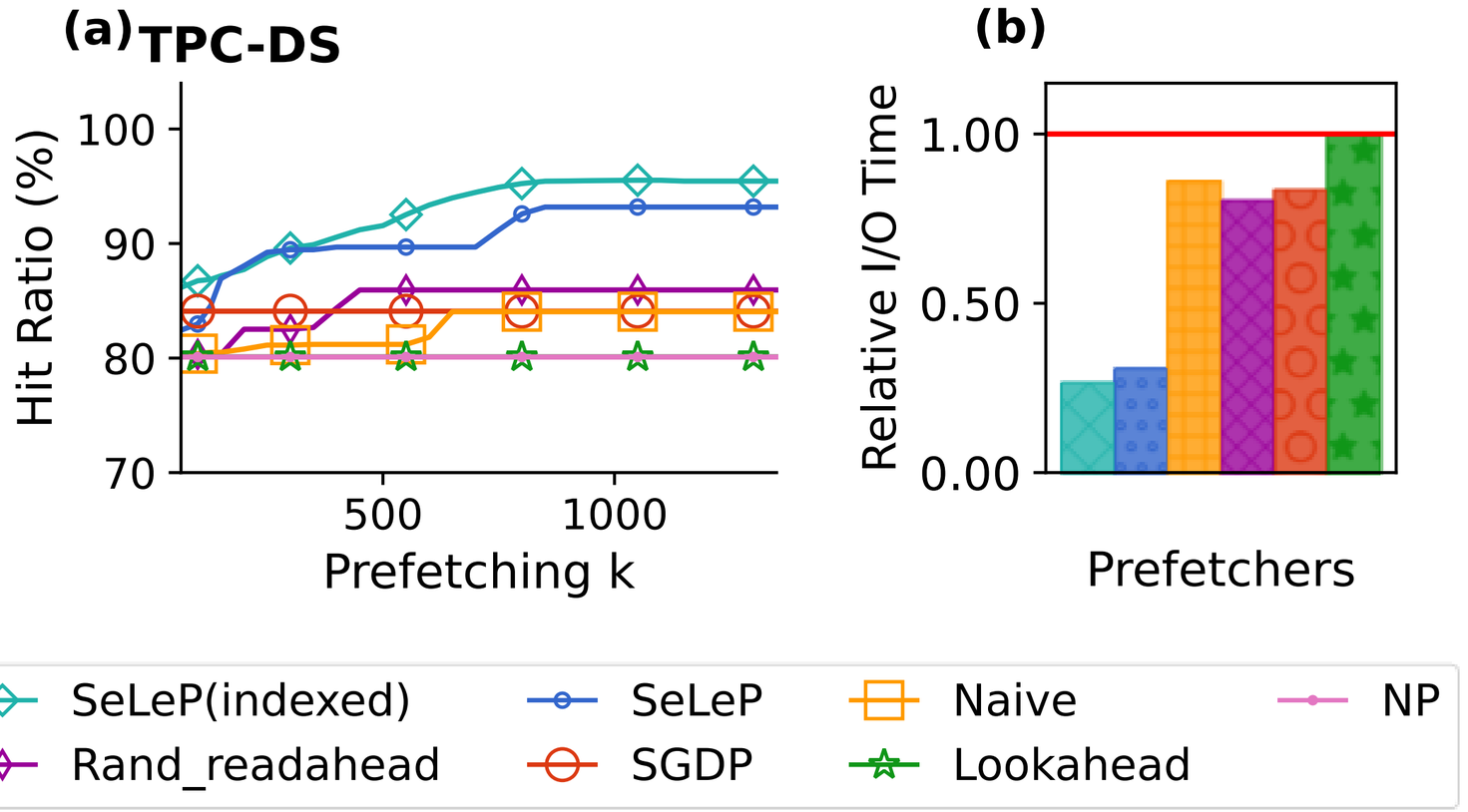


- **96%** hit ratio on **average**
- **93%** I/O time reduction on average. Up to **80%** improvement over state-of-the-art

Leilani Battle, Remco Chang, and Michael Stonebraker. 2016. Dynamic prefetching of data tiles for interactive visualization. In Proceedings of the 2016 International Conference on Management of Data. 1363–1375.

SeLeP - Non-Exploratory Workloads

Setting: TPC-DS SF10, prefetch size = $k \times 128$ block, 4GB cache

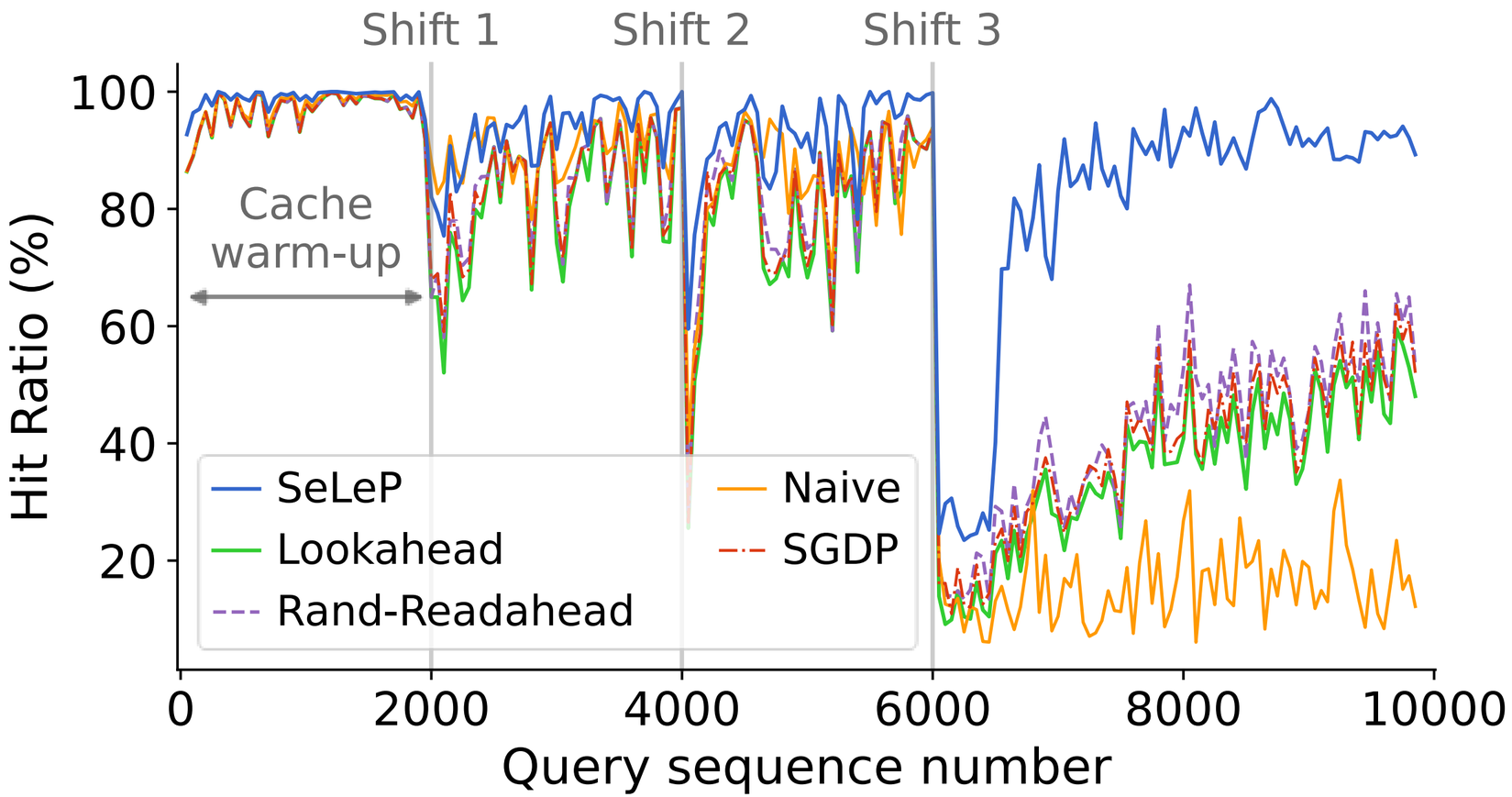


R Malinga Perera, Bastian Oetomo, Benjamin IP Rubinstein, and Renata Borovica- Gajic. 2022. HMAB: self-driving hierarchy of bandits for integrated physical database design tuning. Proceedings of the VLDB Endowment 16, 2 (2022), 216– 229.

SeLeP Adaptivity

Setting: 16GB SDSS DR7, prefetch size = $k \times 128$ block, 4GB cache

Queries: Shifts at sequence number = {2000, 4000, 6000} with novel query templates and access to unseen data



Summary

- Prefetching can substantially reduce I/O time, but
  - The existing prefetchers cannot perform well in SQL-based and jumping navigational workloads
  - They also ignore data semantics in their access prediction

- SeLeP can benefit all types of exploratory workloads using data semantics
  - Encodes blocks into vectors and **extract their semantics** using AutoEncoders
  - **Dynamically partitions blocks** frequently accessed together
  - **Learns partition access pattern** using LSTM model
  - **Improves hit ratio** up to **40%** and **reduces I/O time** up to **45%** compared to traditional and state-of-the-art prefetchers

Special Thanks

Farhana Choudhury

Renata Borovica-Gajic

+ YOU

Questions?

VLDB2024 GUANGZHOU

Full paper & github link

THE UNIVERSITY OF MELBOURNE