Efficient Index Learning via Model Reuse and Fine-tuning

Guanli Liu, Jianzhong Qi, Lars Kulik, Kazuya Soga, Renata Borovica-Gajic, Benjamin I. P. Rubinstein

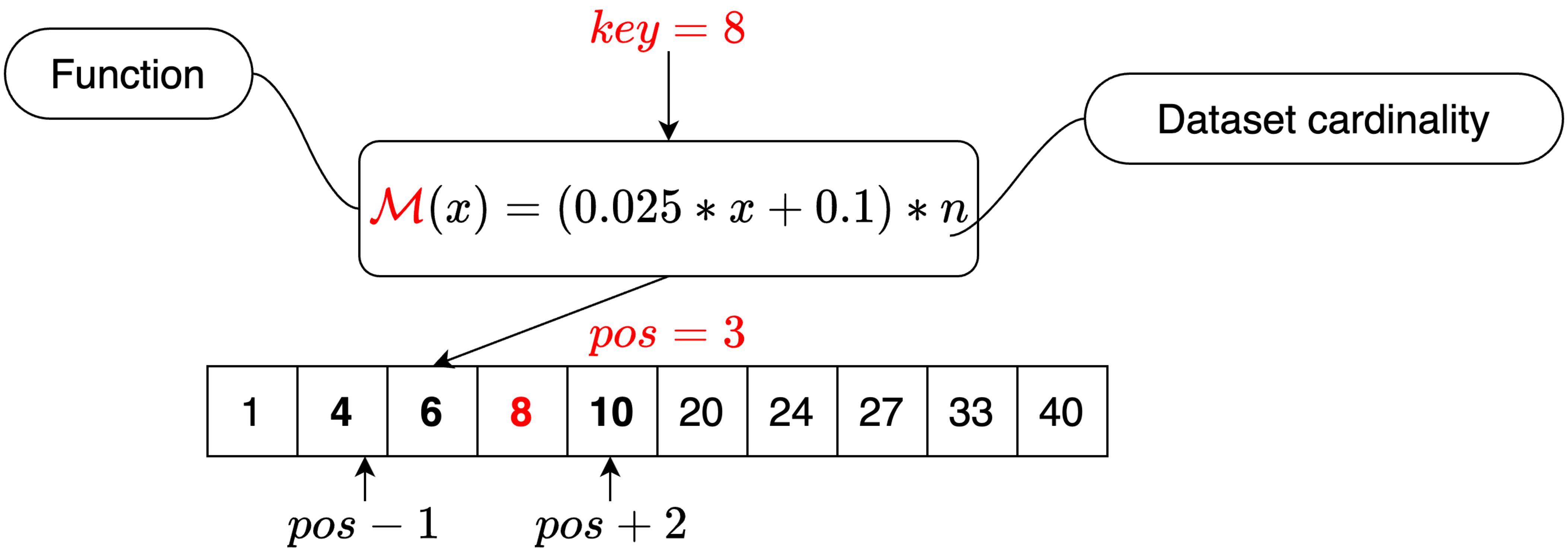
The University of Melbourne

Outline

- **Background**
  - Learned Index
  - Related Works
  - Challenges
- **Methods**
  - Our Goals
  - Solutions
  - Overview
- **Experiments**
- **Conclusions**

Background

- **Learned Index**
  – A function that maps a search key to the storage address



Background

- **Related Works**
  - Regression based

    RMI [1]: Tree structure + Linear regression/NN models

    ALEX [2]: Updatable based on RMI

  - Interpolation based

    PGM-index [3]: Piecewise linear models

    RadixSpline [4]: Spline points + a radix table

Background

- **Challenge: the index learning cost is high**
  - Regression based

    Multiple iterations in model training

    One-pass way cannot learn well
  - Interpolation based

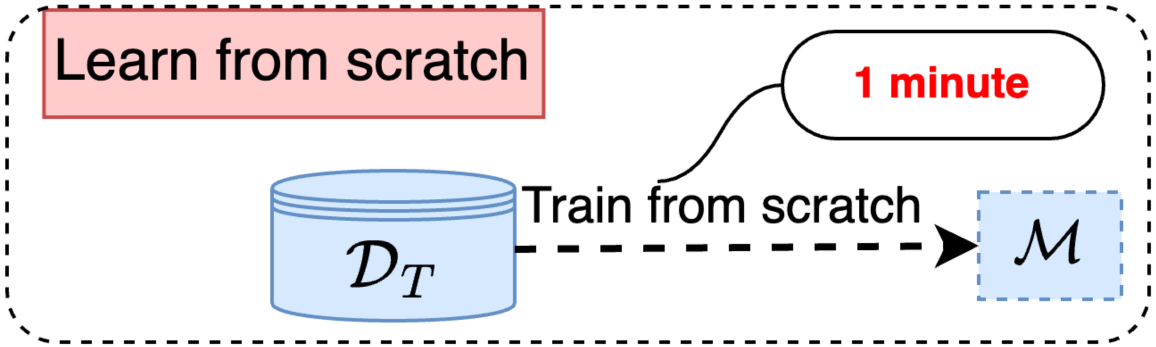
    The one-pass way needs nested loops

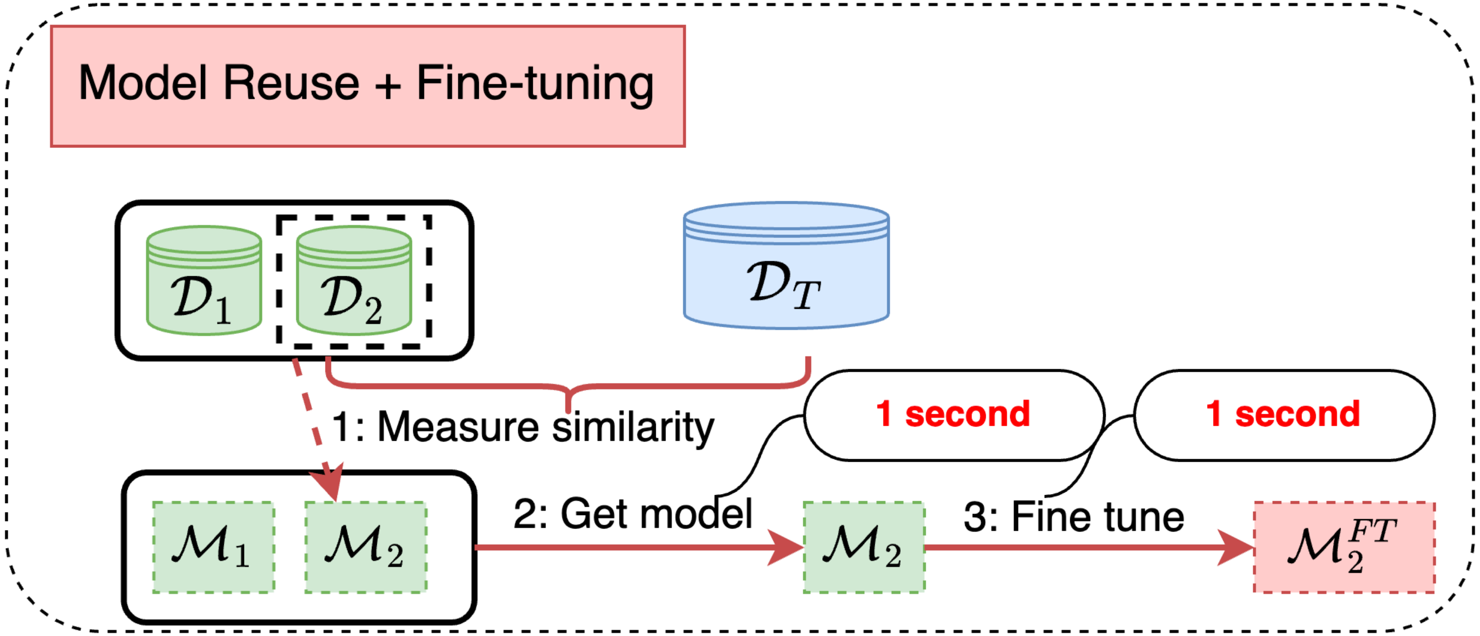
Methods

- **Our goals**
  - Reduce the build cost
  - Maintain the query efficiency
  - Keep the index structure and index size

Methods

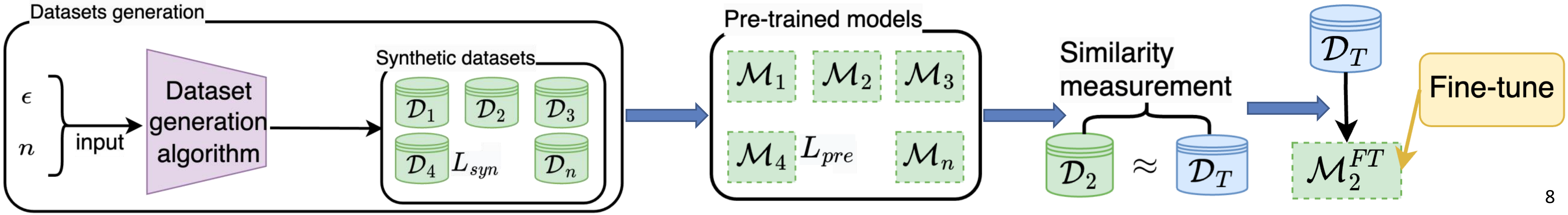
- **Solutions**
  - Use pre-trained models
  - Fine-tuning





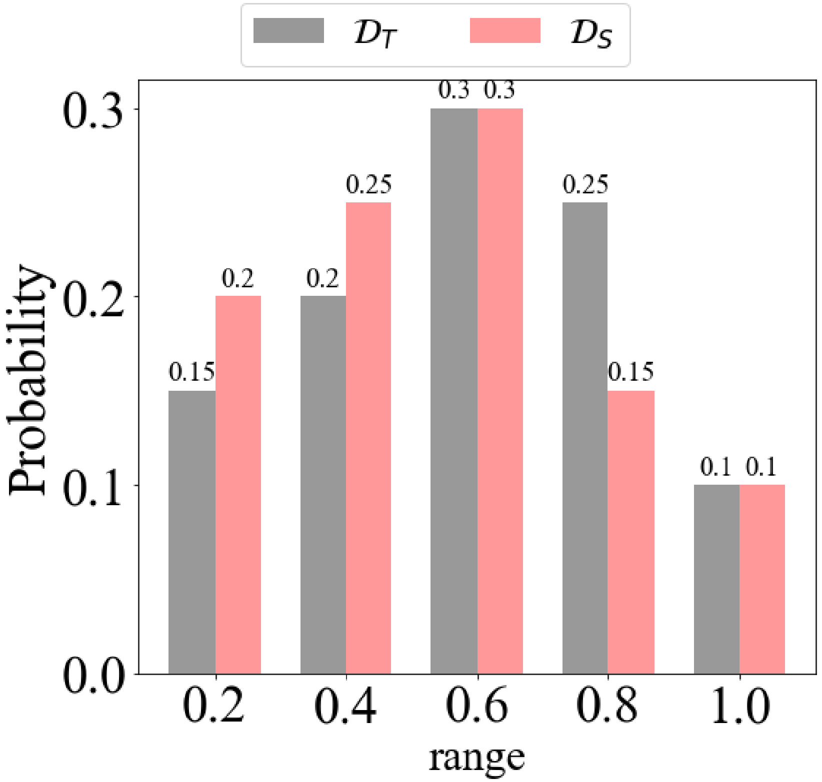
Methods

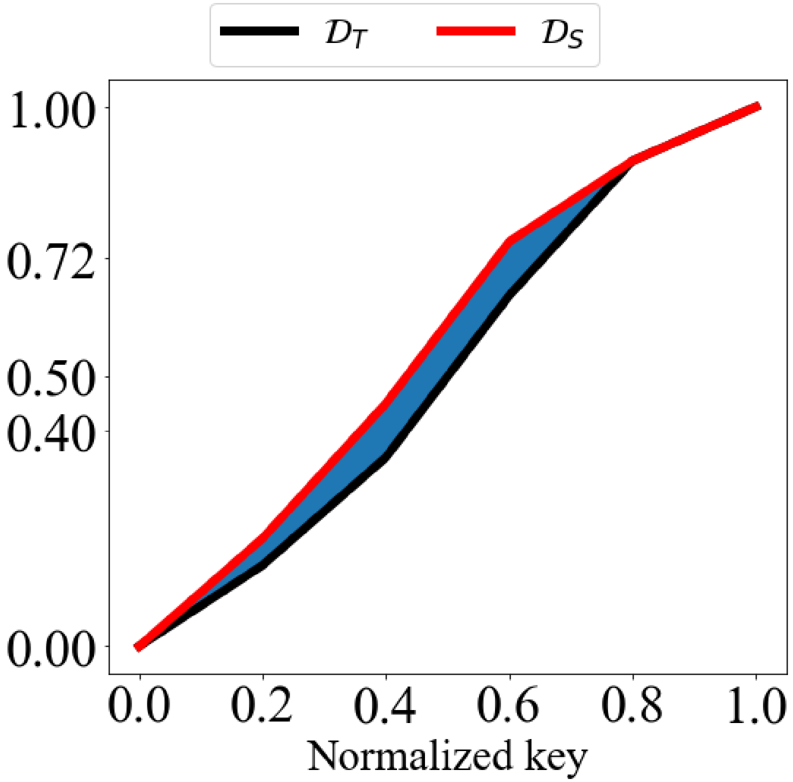
- **Overview**
  - Synthetic datasets
  - Pre-trained models
  - Similarity measurement of CDFs
  - Model adaptation + Fine-tuning



Methods

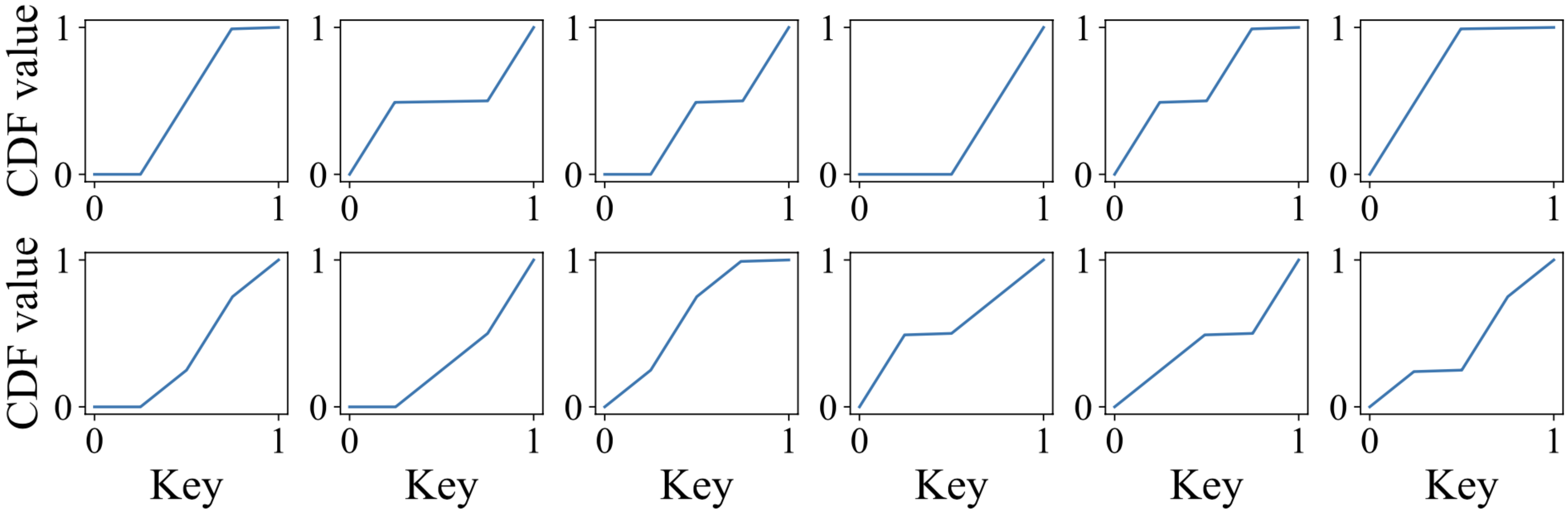
- **Similarity measurement**
  - **Definition**: given two datasets $\mathbf{D_S}$ and $\mathbf{D_T}$, the dissimilarity is the area between their empirical CDFs
  - **Method**: use relative frequency histograms





Methods

- **Synthetic datasets and Pre-trained models**
  - **Target**: a set of datasets to represent real datasets with a high similarity
  - **Method**: use ε to limit the bin size within {0, ε/2, ε}
  - **Examples**: 12 CDFs of generated datasets (ε = 0.5)



Experiments

- **Experimental Environment**
  - Hardware: 64-bit machine, 3.60 GHz Intel i9 CPU, RTX 2080Ti GPU, 64 GB RAM, and 1 TB HDD
  - Datasets:

    Real: amzn, face, osm, and wiki to follow SOSD benchmark[5]

    Synthetic: skewed datasets (200 million) to follow [6]
  - Implementation: Follow SOSD benchmark
  - We set ε=0.3 (987 pre-trained models)

Experiments: real datasets

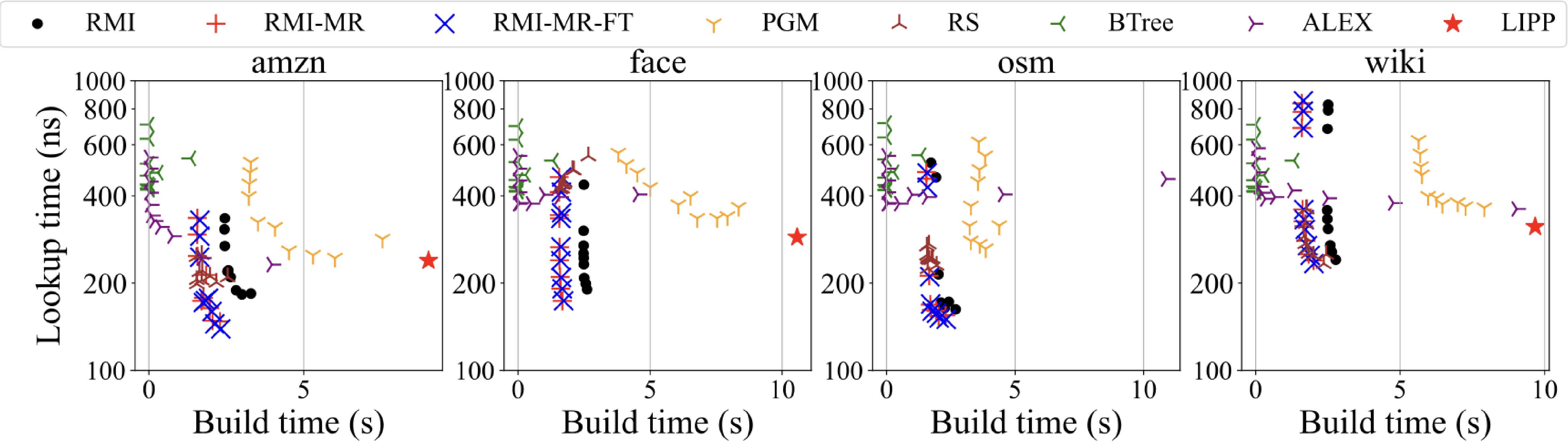


Figure: Build time vs. lookup time over real datasets

Experiments: synthetic datasets

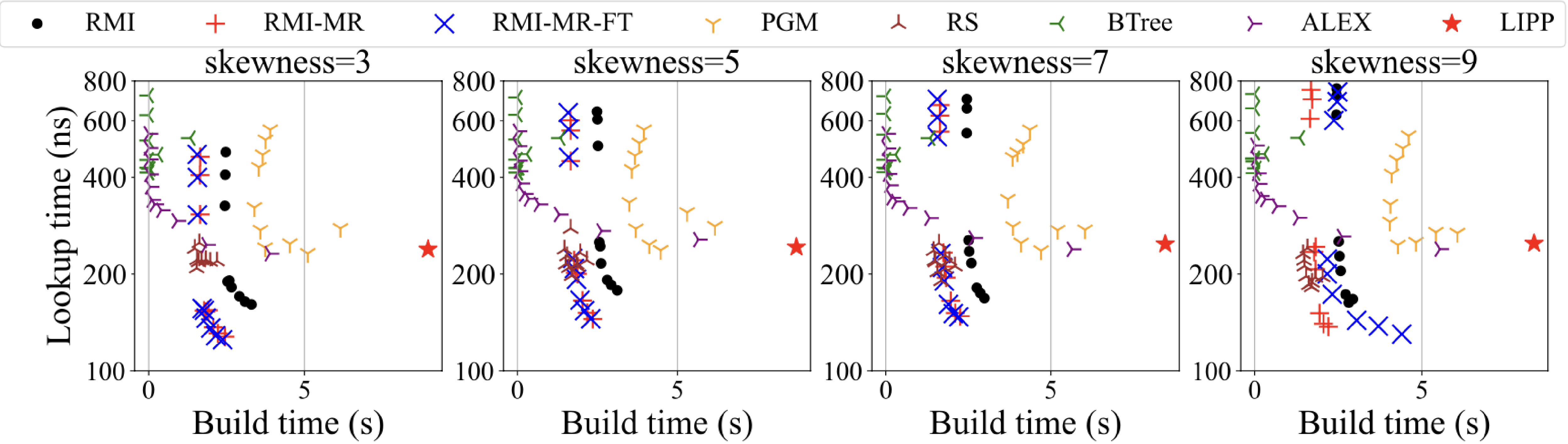


Figure: Build time vs. lookup time over skew datasets

Conclusions

– Enable model reuse + fine-tuning in 1-d learned indices

– Propose a synthetic dataset generation method

– Reduce the index build time

References

[1] T. Kraska, A. Beutel, E. H. Chi, J. Dean, and N. Polyzotis. The case for learned index structures. In SIGMOD, pages 489–504, 2018.

[2] J. Ding, U. F. Minhas, J. Yu, C. Wang, J. Do, Y. Li, H. Zhang, B. Chandramouli, J. Gehrke, D. Kossmann, D. Lomet, and T. Kraska. ALEX: An updatable adaptive learned index. In SIGMOD, pages 969–984, 2020.

[3] P. Ferragina and G. Vinciguerra. The PGM-Index: A fully-dynamic compressed learned index with provable worst-case bounds. PVLDB, 2020.

[4] A. Kipf, R. Marcus, A. van Renen, M. Stoian, A. Kemper, T. Kraska, and T. Neumann. RadixSpline: A single-pass learned index. In aiDM, pages 5:1–5, 2020.

[5] R. Marcus, A. Kipf, A. van Renen, M. Stoian, S. Misra, A. Kemper, T. Neumann, and T. Kraska. Benchmarking learned indexes. PVLDB, 2021.

[6] J. Qi, Y. Tao, Y. Chang, and R. Zhang. Theoretically optimal and empirically efficient R-trees with strong parallelizability. PVLDB, 2018.

Q & A

THE UNIVERSITY OF MELBOURNE

Thank you

Guanli Liu
guanli@student.unimelb.edu.au

Experiments

- **Experimental Results**
  - Datasets generation

Table: Summary of Synthetic Datasets

ϵ	0.2	0.3	0.4	0.5
Number of bins (m)	10	7	5	4
Number of datasets	8,953	987	95	19
Model training time (s)	839.5	63.5	8.8	2.1