E2E: Efficient Filtered AKNN Search via Adaptive Termination

Wenxuan Xia†, Mingyu Yang†*, Wentao Li, Wei Wang*

† Equal contribution · * Corresponding authors | HKUST (Guangzhou) · University of Leicester · HKUST

Contact: wxia248@connect.hkust-gz.edu.cn

香港科技大學(廣州) THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

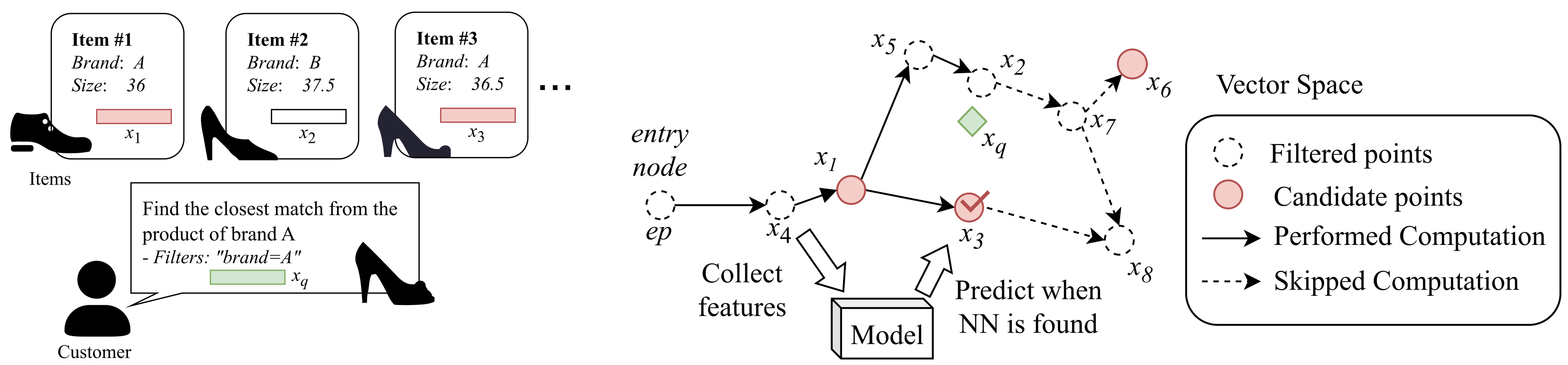
UNIVERSITY OF LEICESTER

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

Summary For **filtered approximate k-nearest neighbour (AKNN) search**, E2E augments the distance-based features of existing early-termination methods with **filter-aware features**. Each query then stops on its own predicted distance-computation budget.

1 Problem Definition

Every vector carries an attribute. A **filtered query** is a query vector plus a constraint on it: return the k nearest vectors that satisfy the constraint.



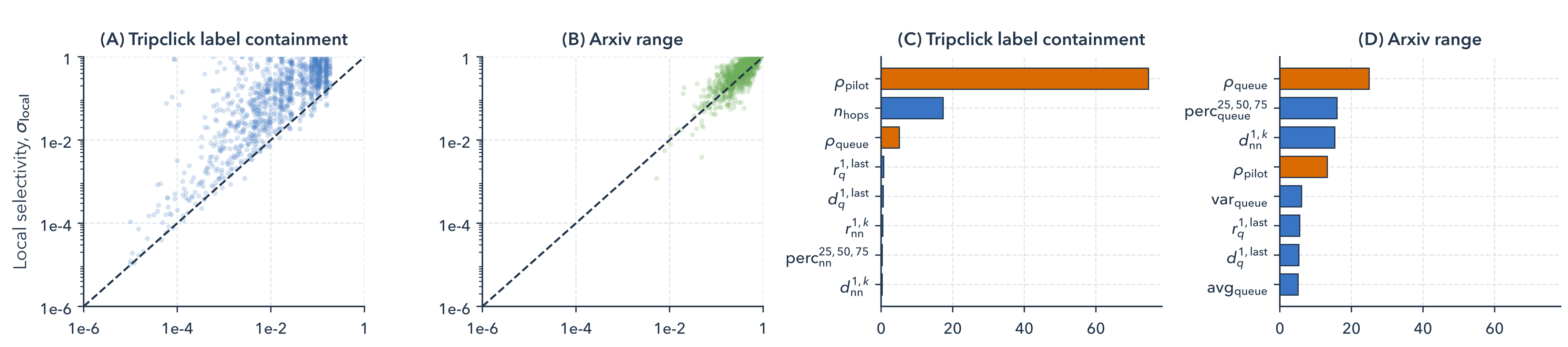
Filtered AKNN search on a graph index. The query is a vector plus the constraint `Brand = A`. Close neighbours such as x_2 and x_5 fail the filter, but their distances are computed first.

- → **Graph indexes and beam search.** Filtered search builds on graph indexes (e.g., HNSW). **Beam search** keeps a queue of the m best candidates and expands the nearest unexpanded one, stopping when none beats the current m-th best.
- → **Post-filtering.** Traversal ignores the filter; mismatched vectors are discarded at insertion into the result set, so their distances are computed and thrown away.
- → **Fixed termination.** The beam width m (efSearch) is a hyper-parameter fixed for the entire query workload: raise it to cover the hardest query and easy ones are over-computed; lower it and recall drops.

2 Motivation

Learned early termination already exists, but only for **plain** AKNN search.

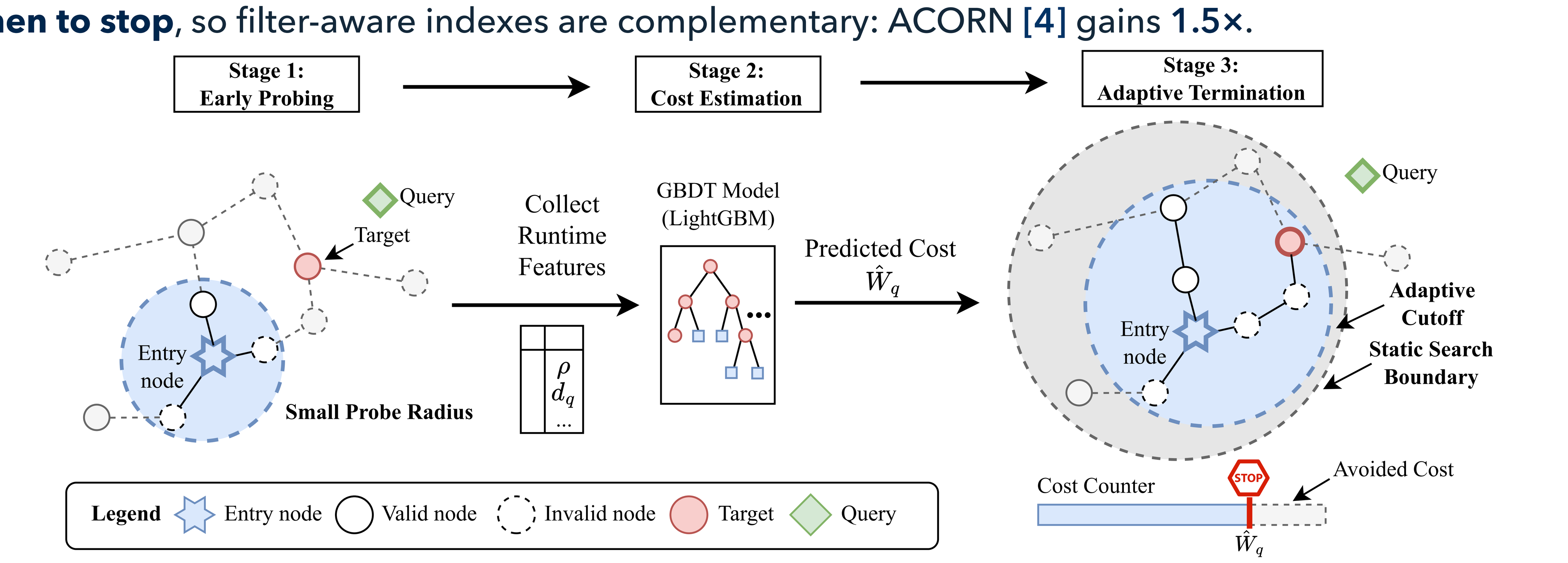
- → **How it works.** Existing methods [1], [2] use runtime distance-based features to determine query-specific stopping points, adapting search effort across the workload.
- → **Their assumption.** Distance is informative: if visited candidates quickly become close to the query, cost is likely small.
- → **Why a filter breaks it.** They ignore attributes: a query may look easy yet cost a lot when valid vectors are scarce, or look hard yet finish quickly in a high-selectivity region.



Where σ_{local} (share of the query's m nearest vectors passing the filter) and σ_{global} (share over the whole database) are **misaligned** (C), ρ_{pilot} alone carries **74.7%** of the model's gain. Where they are **aligned** (D), the distance features (blue) share the load and ours (gold) still hold **38.3%**.

3 Method: Early Probe to Adaptive Termination

E2E **augments the distance features** of [1], [2] with **filter-aware features** collected during greedy routing, then predicts $\hat{W}_q$: the distance computations that query needs for the exact top-k. E2E decides **when to stop**, so filter-aware indexes are complementary: ACORN [4] gains **1.5×**.



Observed Valid Ratio ρ_{pilot}

Fraction of the nodes visited during early probing that pass the filter.

Early Probing

Greedy routing runs anyway; E2E only records statistics over its first f distance computations.

Prospective Valid Ratio ρ_{queue}

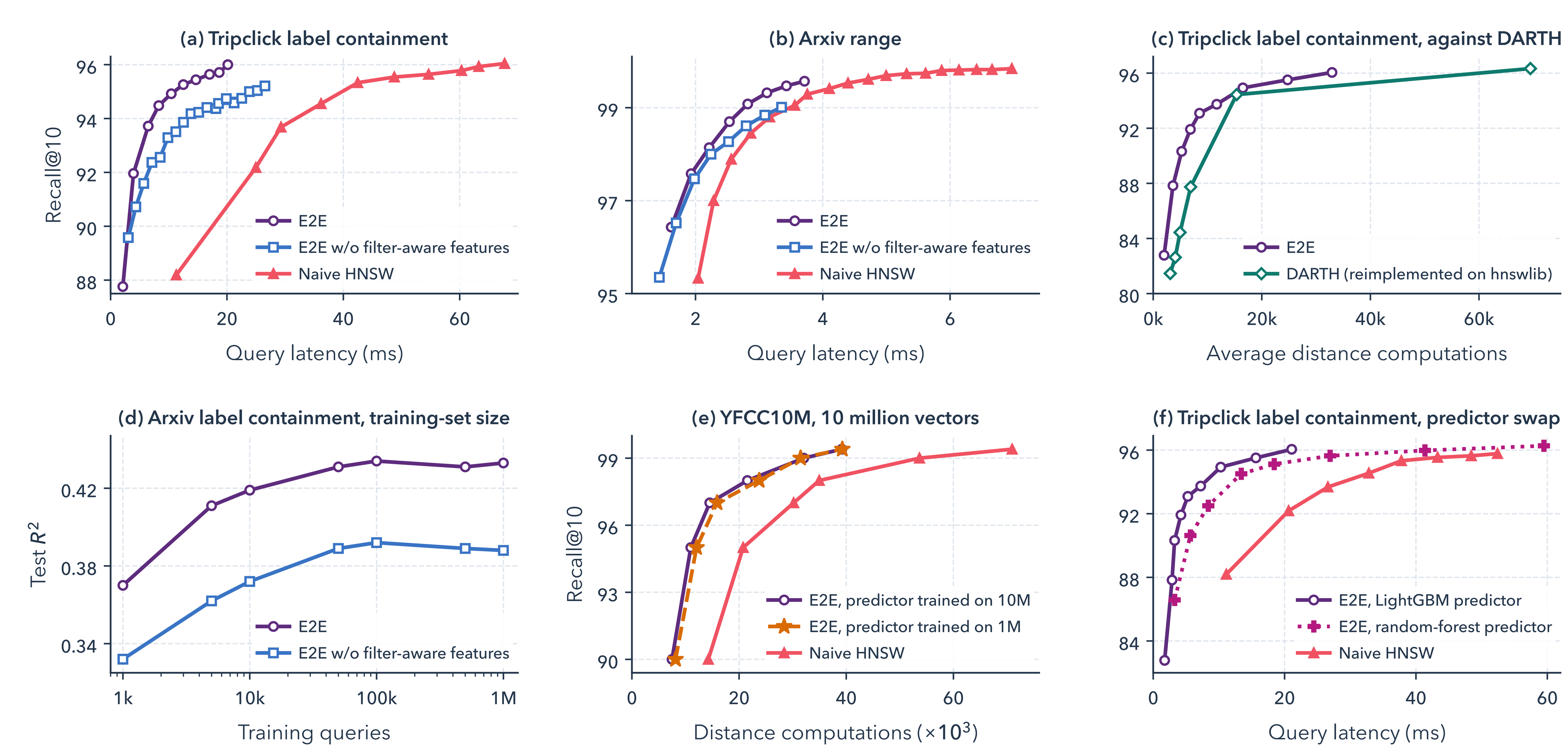
Fraction of the candidates in the queue that pass the filter.

Lightweight Deployment

Single-threaded LightGBM predicts $\hat{W}_q$ at **0.025 ms/query**, about **624 KB** at 1M vectors. Search stops at $\alpha\hat{W}_q$; one α per workload sets recall.

4 Experimental Results

Setup. $k = 10$, single-threaded HNSW (hnswlib) post-filtering. **E2E w/o filter-aware features** = the distance-only feature set of [1], [2]; **DARTH** reimplemented on the same hnswlib backbone.



[1] Li, Zhang, Andersen, He. Improving Approximate Nearest Neighbor Search through Learned Adaptive Early Termination. Proc. SIGMOD 2020, 2539–2554. [2] Chatzakis, Papakonstantinou, Palpanas. DARTH: Declarative Recall Through Early Termination for Approximate Nearest Neighbor Search. PACMMOD 3(4):242, 2025 (SIGMOD 2026). [3] Malkov, Yashunin. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. TPAMI 42(4):824–836, 2020. [4] Patel, Kraft, Guestrin, Zaharia. ACORN: Performant and Predicate-Agnostic Search over Vector Embeddings and Structured Data. PACMMOD 2(3):120, 2024 (SIGMOD 2024). [5] Ke, Meng, Finley, Wang, Chen, Ma, Ye, Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. NeurIPS 2017, 3146–3154.

Paper

Code: **github.com/gegeji/E2E**

DOI: 10.1145/3770855.3817860