

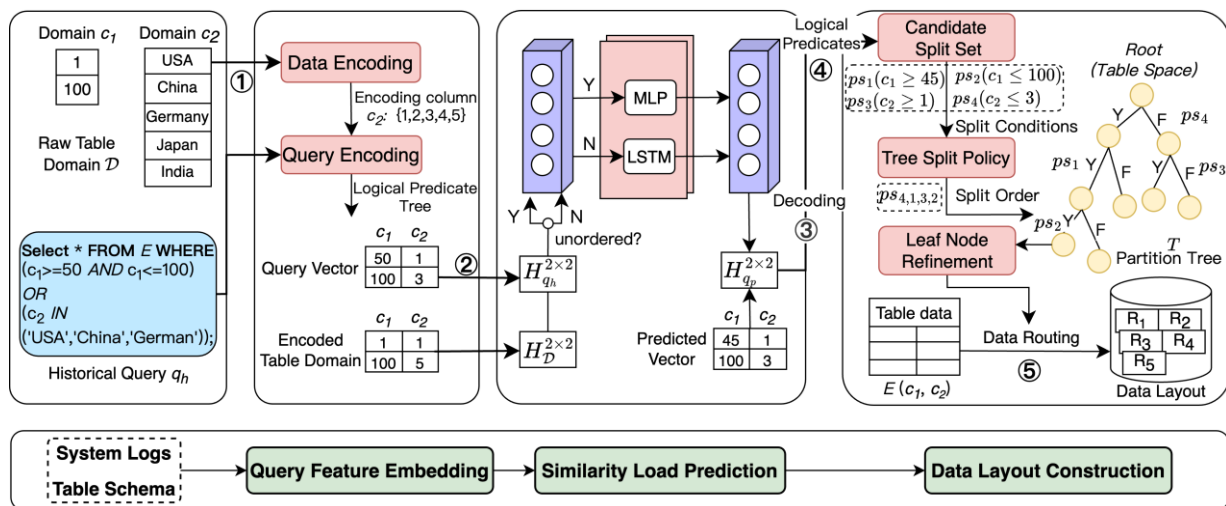
# LRP: Learned Robust Data Partitioning for Efficient Processing of Large Dynamic Queries

**Peng-Ju LIU, Pan CAI, Kai ZHONG, Cui-Ping Li,  
Hong CHEN**

Frontiers of Computer Science, DOI: [10.1007/s11704-024-40509-4](https://doi.org/10.1007/s11704-024-40509-4)

# Problems & Ideas

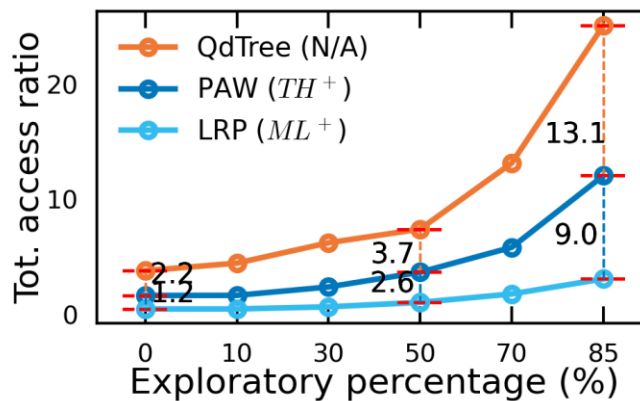
- Problems of dynamic horizontal partitioning approaches:
  - Non-numerical predicates in query plans are crucial features for data partitioning but are often overlooked.
  - Existing strategies primarily optimize partition layouts for static workloads, without accounting for query variations.
- Ideas: If text-type predicate information can be preserved through encoding, subsequent query vectorization and variation prediction become much easier.



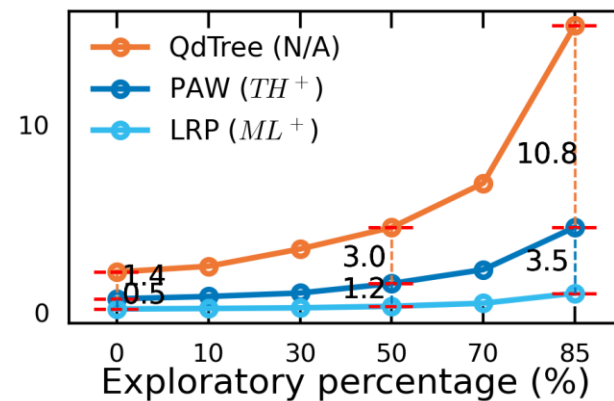
Our LRP system with a three-stage task. In the feature embedding stage, raw table and queries are encoded using predefined column and predicate coding rules. Next, LRP adopts the appropriate network, LSTM for ordered and MLP for unordered data, to predict changes in query vectors. In the layout construction stage, LRP decodes the predicted vectors into logical predicates, then incrementally constructs the partition tree, which is used to materialize the specified partition files.

# Main Contributions

- Contributions:
  - We propose multiple data and query encoding schemes to capture the often-overlooked access features of non-numeric columns.
  - We design two optional predictive networks, along with a loss function integrating partition semantics, to accurately predict changes in similarity queries, ensuring robust layout creation over these predictions.
  - Extensive experiments demonstrate that our method outperforms existing techniques in both performance and robustness.



(a) TPC-D



(b) JOB-D

We compare the data scan volumes of LRP with two other baseline methods when more exploratory (random) queries are mixed into the TPC-D and JOB-D workloads. It can be observed that our LRP demonstrates higher layout adaptability in dynamic load environments.