/TRP1¡¡/ca 1/CA 1¿¿

DBMS – PROJECTS

M.Tech. (CS), First Year, 2022–2023

Deadline: May 31, 2023

Total: 10 marks

SUBMISSION INSTRUCTIONS

STEP I:

1. Submit a solution sketch in a single file by the deadline. The solution must be self-explanatory.
2. The solution should include the sections (if applicable): Introduction, Related Work, Terminologies and Definitions, Theory, Methods, Results, Conclusion.
3. Include names and roll numbers of all of your group members (at most 3).
4. Naming convention for your submission file (assuming M is your project number): `projM` (.docx, .doc, .pdf, .tex, etc.).
5. To submit a solution file (say `projM.docx`), ensure that it is not password protected and mail to `malaybhattacharyya@isical.ac.in`.

STEP II:

1. Deliver a solution sketch through team presentation on (or immediately after) the deadline.
2. The presentation slides should contain the components (if applicable): Introduction, Related Work, Terminologies and Definitions, Theory, Methods, Results, Conclusion.

NOTE: The contribution must be novel and non-trivial.

Project 1: [**Row Pattern Recognition Using Joins**] The MATCH_RECOGNIZE was incorporated in SQL standard in 2016 for row pattern recognition. Since then, it was adopted by leading relational systems using Non-Deterministic Finite Automaton (NFA). While NFA is suitable for pattern recognition in streaming scenarios, the current uses of NFA by the relational systems for historical data analysis scenarios overlook important optimization opportunities.

A recent work has proposed a new approach to use join to speed up row pattern recognition in historical analysis scenarios for relational systems [1]. The approach primarily filters the input relation to MATCH_RECOGNIZE using joins constructed based on a subset of symbols taken from the PATTERN expression, then run the NFA-based MATCH_RECOGNIZE on the filtered rows, thereby reducing the net cost. The rule also includes a specialized cardinality model for the Joins and a cost model for the NFA-based MATCH_RECOGNIZE operator for choosing an appropriate symbol set.

It is suggested that this approach can gain further speedup through operator-level parallelism. Brainstorm on this and other possibilities and address the improvement accordingly.

[1] E. Zhu, S. Huang and s. Chaudhuri, High-Performance Row Pattern Recognition Using Joins. Proceedings of the VLDB Endowment, 16(5): 1181-1195, 2023. (Link: `https://www.vldb.org/pvldb/vol16/p1181-zhu.pdf`)

Project 2: [**Scalable Querying on Temporal Graphs**] Querying cohesive subgraphs on temporal graphs with various time constraints has caught recent research interest. Yang et al. have posed a novel temporal k-Core Query (TCQ) problem [1] and proposed a novel Temporal Core Decomposi- tion (TCD) algorithm that decrementally induces temporal k-cores from the previously induced ones and thus reduces "intra-core" redundant computation significantly. Then, we introduce an intuitive concept named Tightest Time Interval (TTI) for temporal k-core, and design an optimization technique with theoretical guarantee that leverages TTI as a key to predict which subintervals will induce duplicated k-cores and prunes the subintervals completely in advance, thereby eliminating "inter-core" redundant computation. They also proposed a compact in-memory data structure named Temporal Edge List (TEL) to implement OTCD algorithm efficiently in physical level with bounded memory requirement.

Identify limitations of this approach and suggest further improvements.

[1] J. Yang, M. Zhong, Y. Zhu, T. Qian, M. Liu and J. X. Yu. Scalable Time-Range k-Core Query on Temporal Graphs. Proceedings of the VLDB Endowment, 16(5): 1168 - 1180, 2023. (Link: `https://www.vldb.org/pvldb/vol16/p1168-zhong.pdf`)

Project 3: [**Feature-rich and Data-efficient Machine Learning**] The success of machine learning models depend upon the quality of data. There are some recent attempts that aim at achieving both feature-rich and data-efficient machine learning through feature augmentation and coreset selection, respectively. A recent study has shown that selection of the coreset without materializing the augmented table can make the overall process efficient [1].

Identify limitations of this approach and suggest further improvements.

[1] J. Wang, C. Chai, N. Tang, J. Liu and G. Li, Coresets over Multiple Tables for Feature-rich and Data-efficient Machine Learning. Proceedings of the VLDB Endowment, 16(1): 64 - 76, 2022. (Link: `https://www.vldb.org/pvldb/vol16/p64-wang.pdf`)

Project 4: [**Provenance-based Data Skipping**] Determining which data is needed for answering a query beforehand and accordingly plan for indexing and other physical design strategies to speed-up access is common in databases. For some important type of queries (e.g., having clause and top-k queries), it is impossible to determine up-front what data is relevant. To overcome this limitation, a recent study has developed a novel approach, namely provenance-based data skipping, for generating provenance sketches to concisely encode what data is relevant for a query. Once a provenance sketch has been captured it is used to speed up subsequent queries. This approach can exploit physical design artifacts such as indexes and zone maps.

It is interesting to investigate how to maintain provenance sketches under updates and extend the self-tuning techniques to support wider range of queries and more powerful strategies.

Brainstorm in these directions and propose a novel approach to improve the state-of-the-art.

[1] X. Niu, B. Glavic, Z. Liu, P. Li, D. Gawlick, V. Krishnaswamy, Z. H. Liu and D. Porobic. Provenance-based Data Skipping. Proceedings of the VLDB Endowment, 15(3): 451 - 464, 2022. (Link: `https://www.vldb.org/pvldb/vol15/p451-niu.pdf`)

Project 5: **[Cardinality Estimation with Machine Learning]** Machine Learning based cardinality estimation techniques have recently gained momentum for their superiority over the earlier approaches. These techniques are broadly categorized into two types – query-driven and data-driven methods. The success of query-driven methods heavily rely on the availability of high-quality training queries, whereas data-driven methods have no such limitation and have high adaptivity.

A recent study has proposed a data-driven cardinality estimation method, termed as FACE, which leverages the normalizing flow based model to learn a continuous joint distribution for relational data [1]. This approach significantly outperforms the state-of-the-art in terms of estimation accuracy while keeping similar latency and model size. Understand this approach and suggest how this could be further improved.

[1] J. Wang, C. Chai, J. Liu and G. Li. FACE: A Normalizing Flow based Cardinality Estimator. Proceedings of the VLDB Endowment, 15(1),: 72 - 84, 2022. (Link: `https://www.vldb.org/pvldb/vol15/p72-li.pdf`)

Project 6: **[Learning-Enhanced Range Filter]** A recent research has proposed a learned range filter method, termed as SNARF, which efficiently supports range queries for numerical data. It creates a model of the data distribution to map the keys into a bit array which is stored in a compressed form. SNARF provides a significant improvement over the state-of-the-art range filters, such as SuRF and Rosetta, without affecting the space requirement.

The SNARF method can be extended to filter other types of data. Finding better learning models or proving via a theoretical framework that linear spline model is a near-optimal model also remain as open questions. Finally, making SNARF workload dependent is an interesting direction for future work.

Make contributions in any one of these future directions of extending this work.

[1] K. Vaidya, S. Chatterjee, E. Knorr, M. Mitzenmacher, S. Idreos and T. Kraska. SNARF: A Learning-Enhanced Range Filter. Proceedings of the VLDB Endowment, 15(8): 1632 - 1644, 2022. (Link: `https://www.vldb.org/pvldb/vol15/p1632-vaidya.pdf`)

Project 7: **[Deep Learning from Relational Databases]** It is interesting to apply deep learning models directly on relational databases. Applying model deduplication can greatly reduce the storage space, memory footprint, cache misses, and inference latency. However, existing data deduplication techniques are not applicable to the deep learning model serving applications in relational databases. They do not consider the impacts on model inference accuracy as well as the inconsistency between tensor blocks and database pages. A recent work has proposed synergistic storage optimization techniques for duplication detection, page packing, and caching, to enhance database systems for model serving [1].

Identify limitations in this approach and propose a better method.

[1] L. Zhou, J. Chen, A. Das, H. Min, L. Yu, M. Zhao and J. Zou. Serving Deep Learning Models with Deduplication from Relational Databases. Proceedings of the VLDB Endowment, 15(10): 2230 - 2243, 2022. (Link: `https://www.vldb.org/pvldb/vol15/p2230-zou.pdf`)

Project 8: [**Data Integration**] Same kind of data from public and third-party sources suffer from formatting mismatch even if they describe the same set of entities. Such data must be joined or cross-referenced. Recent research highlights how tables can be made joinable under some transformations. There exist algorithms based on the common characteristics of the joined columns [1]. These approaches can be extended in a few directions like applying transfer learning where transformations obtained for one dataset can be adapted to another dataset, employing non-string-based transformations considering semantics of the words, etc.

You are required to pursue any one of these future directions by proposing novel solutions.

[1] A. D. Nobari and D. Rafiei, Efficiently Transforming Tables for Joinability. In Proceedings of the 38th IEEE International Conference on Data Engineering (ICDE), pp. 1649-1661, 2022. (Link: `http://webdocs.cs.ualberta.ca/~drafiei/papers/dargahi_joinability_ext.pdf`)

Project 9: [**Learned Indexes**] Latest progress in learned index structures propose replacing existing index structures, like B-Trees, with approximate learned models. A recent study presents a unified open source benchmark that compares well-tuned implementations of three learned index structures against several state-of-the-art traditional baselines [1]. They demonstrated that learned index structures can indeed outperform non-learned indexes in read-only in-memory workloads over a dense array. However, the learned structures generally had higher build times than insert-optimized traditional structures like B-Trees.

You are requested to combine a simple radix table with any of the recent learned index structures to overcome the said limitation.

[1] Ryan Marcus, Andreas Kipf, Alexander Van Renen, Mihail Stoian, Sanchit Misra, Alfons Kemper, Thomas Neumann and Tim Kraska. "Benchmarking Learned Indexes." Proceedings of the VLDB Endowment 14(1): 1-13, 2021. (`Link:http://vldb.org/pvldb/vol14/p1-marcus.pdf`)

Project 10: [**Understanding Tables through Deep Learning**] There has been promising developments in recent years on a variety of tasks in the area of table understanding. However, existing work generally relies on heavily-engineered task-specific features and model architectures. A recent work presents a novel framework that introduces the pre-training/fine-tuning paradigm to relational Web tables. During pre-training, our framework learns deep contextualized representations on relational tables in an unsupervised manner. Its universal model design with pre-trained representations can be applied to a wide range of tasks with minimal task-specific fine-tuning.

You are required to focus on other types of knowledge such as numerical attributes in relational Web tables, in addition to entity relations used in the said work. You can also incorporate the rich information contained in an external KB into pre-training for making the approach better.

[1] Xiang Deng, Huan Sun, Alyssa Lees, You Wu and Cong Yu. "TURL: Table Understanding through Representation Learning." Proceedings of the VLDB Endowment 14(3): 307-319, 2021. (Link:http://vldb.org/pvldb/vol14/p307-deng.pdf)

Project 11: **[Cardinality Estimation]** Query optimization requires accurate cardinality estimation for producing good execution plans. Unfortunately, state-of-the-art cardinality estimators are inaccurate for complex queries. A recent study has shown that it is possible to learn the correlations across all tables in a database without any independence assumptions [1]. This is possible by developing a join cardinality estimator that builds a single neural density estimator over an entire database. This approach, termed as NeuroCard, is both compact in space and efficient to construct or update.

You are required to identify the limitations of this approach and propose a better alternative. Note that, NeuroCard employs deep learning as a component that can be improved.

[1] Zongheng Yang, Amog Kamsetty, Sifei Luan, Eric Liang, Yan Duan, Xi Chen, and Ion Stoica. "NeuroCard: One Cardinality Estimator for All Tables." Proceedings of the VLDB Endowment, 14(1): 1033-1039, 2021. (Link:https://vldb.org/pvldb/vol14/p61-yang.pdf)

Project 12: **[Interpretable Data Cleaning]** ... [1]

[1] Jinfeng Peng, Derong Shen, Nan Tang, Tieying Liu, Yue Kou, Tiezheng Nie, Hang Cui, and Ge Yu. "Self-supervised and Interpretable Data Cleaning with Sequence Generative Adversarial Networks." Proceedings of the VLDB Endowment, 16(3): 433-446, 2022. (https://www.vldb.org/pvldb/vol16/p433-peng.pdf)

Project 13: **[Influence Maximization]** ... [1]

[1] Shixun Huang, Wenqing Lin, Zhifeng Bao and Jiachen Sun. "Influence Maximization in Real-World Closed Social Networks." Proceedings of the VLDB Endowment, 16(2): 180-192, 2022. (https://www.vldb.org/pvldb/vol16/p180-bao.pdf)

Project 14: **[Open Choice]** You can choose any single database-related research paper from these lists [1, 2, 3, 4] and suggest novel improvement over the method proposed over there.

[1] https://www.vldb.org/pvldb/volumes/16

[2] https://2023.sigmod.org/sigmod_research_list.shtml

[3] https://2023.sigmod.org/pods_list.shtml

[4] https://icde2023.ics.uci.edu/papers-research-track