Amalur: Next-generation Data Integration in Data Lakes

Rihan Hai Christos Koutras Andra Ionescu Asterios Katsifodimos
Delft University of Technology

Abstract

Data science workflows require extracting, preparing and integrating data from multiple data sources. Due to the lack of proper tooling this is a very cumbersome process that hinders the productivity of data scientists. Moreover, this is a very slow process: most of the times, data scientists prepare data in a data processing system or a data lake, and export it in the form of a table, in order for it to be consumed by a Machine Learning (ML) algorithm.

Recent advances in the area of factorized ML, allow us to push down certain linear algebra (LA) operators, and to execute them closer to the data sources [2, 1]. At the same time, we have a proliferation of novel data exploration and discovery tools as well as dataset relatedness and matching algorithms [6, 5]. With this work we argue that this is the right moment to revisit all the components of classic data integration (DI) systems, and to see how these fit into modern data lakes that are meant to support LA as a first-class citizen.

In this paper we first investigate how the advances in factorized ML and modern data integration techniques influence and can benefit from each other, forming new research opportunities. We then describe *Amalur*: a reference architecture of a next-generation data lake system which facilitates linear algebra processing over heterogeneous sources. We propose a formal representation based on matrices, which connects to the schema mapping formalism in first-order logic [3, 4], and enables LA factorization over joinable or unionable data in a data lake. Finally, we outline the future research challenges related to next-generation data lake systems.

References

[1] R. Alotaibi, B. Cautis, A. Deutsch, and I. Manolescu. Hadad: A lightweight approach for optimizing hybrid complex analytics queries. In *SIGMOD*, pages 23–35, 2021.

[2] L. Chen, A. Kumar, J. Naughton, and J. M. Patel. Towards linear algebra over normalized data. *PVLDB*, 10(11), 2017.

[3] R. Fagin. *Tuple-Generating Dependencies*, pages 3201–3202. Springer US, Boston, MA, 2009.

[4] R. Hai and C. Quix. Rewriting of plain so tgds into nested tgds. *VLDB*, 12(11):1526–1538, 2019.

[5] C. Koutras, K. Psarakis, G. Siachamis, A. Ionescu, M. Fragkoulis, A. Bonifati, and A. Katsifodimos. Valentine in action: Matching tabular data at scale. *Proc. VLDB Endow.*, 14(12):2871–2874, jul 2021.

[6] C. Koutras, G. Siachamis, A. Ionescu, K. Psarakis, J. Brons, M. Fragkoulis, C. Lofi, A. Bonifati, and A. Katsifodimos. Valentine: Evaluating matching techniques for dataset discovery. In *ICDE*, pages 468–479. IEEE, 2021.