Data Discovery in Data Lakes: Operations, Indexes, Systems

Ziawasch Abedjan
BIFOLD & TU Berlin
Berlin, Germany
abedjan@tu-berlin.de

Mahdi Esmailoghli
HU Berlin
Berlin, Germany
mahdi.esmailoghli@hu-berlin.de

Sainyam Galhotra
Cornell University
Ithaca, USA
sg@cs.cornell.edu

ABSTRACT

Data discovery has gained significant traction in the database community resulting in various discovery operations, index schemes, and discovery systems. This tutorial explores the architecture and components of data discovery systems, focusing on indexing structures and scalable algorithms for typical operations, such as join and union discovery. While giving insights into individual algorithms, we point out open challenges for holistic systems, data discovery evaluation, and discovery in federated setups.

PVLDB Reference Format:
Ziawasch Abedjan, Mahdi Esmailoghli, and Sainyam Galhotra. Data Discovery in Data Lakes: Operations, Indexes, Systems. PVLDB, 18(12): 5455 - 5459, 2025.
doi:10.14778/3750601.3750694

1 INTRODUCTION

Data lakes, whether intentionally created or emerging as a byproduct of organizational processes or open data initiatives, share a common challenge: they are often underspecified and underdocumented [1, 13]. Unlike curated data warehouses, where the schema and purpose are clearly defined, data lakes typically lack predefined objectives. Despite this, there is significant interest in leveraging these repositories for augmenting in-use datasets or pure exploratory purposes. To support such applications, data discovery as a process becomes essential, enabling users to interact with the data lake not by relying on detailed schema knowledge but by using search and matching techniques. By searching with user-provided keywords or table artifacts, users aim to uncover datasets that are relevant to their use cases.

Data discovery is currently an active field of research, as building scalable solutions to effectively capture user intent, such as augmenting a given dataset fragment, presents various challenges. A key issue is the need to index thousands to millions of tables for fast value look-ups and efficient alignment to evaluate joinability [9, 35] and unionability [19] with the user (query) table. After identifying relevant tables, assessing the usefulness of the retrieved results remains a complex problem. Often, users rely on proxy metrics, such as the downstream machine learning performance of the augmented dataset, to assess its effectiveness [7, 8, 22]. Finally, in distributed data lakes, additional constraints, such as privacy regulations and pricing structures, further complicate the process. In such cases, the usefulness and relevance of datasets must often be assessed based on available metadata and summaries, rather than direct access to the full datasets.

This tutorial is based on our previous study of data discovery use cases and our experience in building systems and algorithms that address some of its emergent challenges. Our goal is to give a structured overview of data discovery software components and to dive into technical aspects that differentiate individual prototypes and approaches. A meta-goal is to bridge the language and perspective differences on what is counted as data discovery and how discovery systems in database research can be compared. Figure 1 depicts a conceptual architecture of data discovery systems that we explore in this tutorial. The user formulates a discovery task that typically aims at enhancing an existing table or table fragment through additional features, rows, or individual values. Such tasks are then typically carried our on top of common operations, such as keyword search, join discovery, union discovery, and others that are built on top of dedicated indexes. In our tutorial, we cover the depicted architecture as follows. We first survey and discuss common data discovery use cases from industry and academia. We then present a classification of data discovery systems with layers that distinguish the locality, languages, and index structures of such systems. Locality captures the accessibility of repositories on monoliths and in distributed settings. Discovery languages can differ depending on how many different discovery tasks a system covers. Accordingly, we then dive deep into concrete index structures and how they serve individual discovery operations and languages.

Previous Tutorials: There have been similar tutorials in the past that relate to ours. In particular, the tutorial by Paton and Wu at EDBT 2024 [21] also covers various aspects of dataset search and navigation. In contrast to their tutorial, we have a strong focus on algorithmic details on how index structures and operations have evolved and can be categorized with a projection on how these approaches relate, can extend to distributed scenarios, and can be combined inside holistic systems. There has been a tutorial from the information retrieval perspective on web table extraction [32]. Our tutorial covers more recent advances and discusses the topic from the database management perspective. The tutorial on data exploration [15] is related but does not focus on data lakes.

2 TUTORIAL OUTLINE

In this tutorial, we first discuss the types of data lakes that set the motivation for building data discovery systems. We further categorize existing systems depending on their scope with regard to data locality, i.e., centralized and distributed repositories. We then detail common operations and fundamental indexing schemes in state-of-the-art that support such operations. Finally, we will discuss methods for evaluating and benchmarking data discovery systems and outline open problems and potential future research.

Proceedings of the VLDB Endowment, Vol. 18, No. 12 ISSN 2150-8097.
doi:10.14778/3750601.3750694

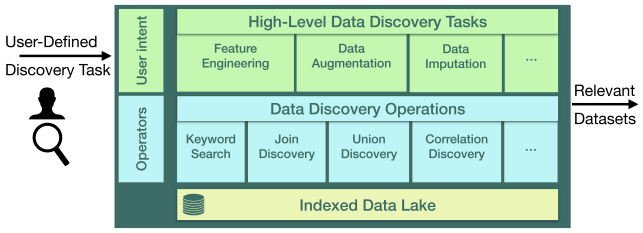

Figure 1: Vanilla data discovery system architecture

2.1 Data Lakes

With the awareness of how valuable data can be even without being currently under active usage, organizations started to accumulate datasets and make them accessible across silos. This is in contrast to the typical data warehousing approach, where handcrafted extract-transform-load pipelines are defined and documented to integrated datasets into a purposeful schema. Data lakes in contrast follow the integrate at application time paradigm. Often times the datasets not only are heterogeneous on schema level but can be subject to heterogenous data models or reside in heterogeneous storage systems. For the sake of scoping, we will focus on the algorithms and systems that have been designed for data lakes with tabular datasets.

2.2 Overview of Data discovery settings

Data discovery is a fundamental capability for organizations that manage large and often distributed collections of datasets. The need for robust data discovery systems has become even more pressing with the widespread adoption of machine learning and AI techniques, which depend critically on access to high-quality, relevant data. Numerous studies have explored different approaches to data discovery, with the optimal solution often varying based on the specific use case. This section provides an overview of the key architectural and interaction paradigms for data discovery.

Centralized vs. Federated Settings. Traditional data discovery systems typically assume *centralized access* to data, where all datasets are consolidated into a single repository, allowing the discovery engine to index and query them directly. However, this assumption is often impractical. In many cases, data is often dispersed across multiple repositories due to concerns related to data sovereignty, privacy regulations, and the high cost of data transfer.

To address these challenges, modern data ecosystems are shifting toward *federated* or *decentralized* discovery architectures. In such settings, data providers retain control over their datasets and share only metadata (or a summary of the dataset) with the discovery system. This model is common in commercial data marketplaces (e.g., Datarade, Dawex) and federated infrastructure initiatives (e.g., Gaia-X, Agora), where discovery must be performed over incomplete information, and access to data is negotiated only after identifying relevant resources.

Data Discovery Tasks and Use Cases. Users express their information needs in various ways, depending on the context and their familiarity with the data. Common interaction modalities include:

- **Keyword-based Discovery:** Users input textual queries along with optional filters. While straightforward and user-friendly, this method is often inadequate when users have nuanced requirements related to data semantics, coverage, or statistical characteristics.
- **Query-by-Example:** Users provide a small sample dataset and search for similar datasets based on structure or content. This approach is helpful when users struggle to verbalize their data needs but can illustrate them via examples.
- **Task-based Discovery:** Users specify a target task, such as training a classifier or evaluating a model, and seek datasets that are suitable for that purpose. This mode is especially relevant for machine learning workflows, where factors like class balance and feature alignment significantly impact performance.

In many real-world scenarios, users may only have a vague or evolving notion of what constitutes a “useful" dataset. Their needs may not be easily captured by keywords, examples, or tasks. To address this, recent work has focused on designing more expressive query languages that allow users to specify fine-grained predicates—such as schema constraints, statistical distributions, or semantic criteria—to better guide the discovery process.

2.3 Discovery Operations

Our tutorial will strongly focus on two major classes of operations that are common in data discovery research: Joins and Unions. note that each of the two can inhibit additional properties, such as fuzziness or conditional utility functions for the correspondingly joinable and unionable tables, as it is relevant for example for task-based discovery tasks. In our tutorial, we will define both operations and discuss the variations of each.

A join discovery operation aims at discovery tables that are joinable with a given user table. In the literature, variations, such as precalculated PK-FK relationships, single column joinability and multi-column joinability, as well as fuzzy joins have been considered. In our tutorial, we will cover the necessary algorithmic steps for each of the aforementioned variations, as well as their use cases and disadvantages. Furthermore, we will dive deep into the class of conditional join discovery approaches, where not only the joinability but also the utility of the obtained tables with regard to downstream tasks have been considered.

A union discovery operation aims at vertically expanding a table with more rows from tables inside a lake. Within this line of work, we will distinguish approaches that check for perfect unionability as in consistent with UNION in SQL as well as approaches that find partially unionable tables.

2.4 Indexing Data Lakes

An efficient index structure is necessary to support any of the aforementioned discovery operators. Index structures have long served as the foundation of traditional information systems, enabling efficient search and retrieval. In the era of large-scale data lakes and advanced data discovery techniques, these structures have evolved to meet two primary objectives: *scalability* and *customizability*. The large size and heterogeneity of data lakes require indexes that can efficiently scale to large volumes of data while supporting various

Table 1: Mapping between indexes and data discovery studies.

Index		Papers
Value (Cell)	Value → Location	[2, 4, 24, 29–31, 35]
	Location → Value	[2, 30, 35]
	Value → Class (Domain)	[24]
	Value → Embedding	[5]
Column	LSH Variations	[3, 5, 12, 14, 19, 28, 36]
	Column → Metadata	[2, 14, 24, 30, 35]
	Column Names → tables	[31]
	Column → Domain	[34]
	Column → Rank	[22, 23]
	Column → Embedding	[5, 11, 27]
Row	Row → Table	[2, 31]
Table	Table → Embedding	[31]
	Table → Metadata	[2]
Graph	V: Columns, E: Similarities	[4, 12, 16–18, 20, 26, 31, 33, 34]

custom search requirements. For instance, data discovery with the goal of benefiting a downstream machine learning (ML) task requires a task-specific similarity measure to find the most relevant data. In this example, the relevancy of a column is measured by how significantly the column correlates with a target column to be predicted in the ML task. These task-specific requirements demand customized index structures, leading to hybrid and more complex indexes.

Index structures for data discovery in data lakes can be categorized based on two key dimensions: query interface, determining the primary way users interact with the index, and search approximation, defining whether the index allows exact or approximate discovery.

Table 1 provides an overview of the state-of-the-art index structures, grouped into six categories based on their query interface: Value-based indexes, column-based indexes, row-based indexes, table-based indexes, and graph-based indexes.

Each of these index types serves a distinct purpose. Value-, column-, row-, and table-based indexes enable querying data lakes at different levels of granularity, from individual cell values to entire tables. Graph-based indexes facilitate exploratory search by capturing relationships between datasets, allowing users to navigate table connections.

Index structures can also be categorized based on whether they provide exact or approximate, i.e., fuzzy discovery. Exact indexes guarantee exact retrieval of the most relevant tables to the task at hand but may be computationally expensive. Examples of exact indexes are traditional inverted indexes. Approaches such as Josie [35] and MATE [9] benefit from exact indexes to find equi-joinable tables to a given key column. On the other hand, approximate indexes prioritize efficiency by sacrificing accuracy.

Hash- and sketch-based techniques have been widely adopted to improve scalability. For example, LSH-based indexes [6, 14, 36] are used for efficiently finding tables based on Jaccard, containment, and cosine similarities. However, significant research efforts focus on improving their accuracy and scalability, leading to variations such as LSH Ensemble [36], which reduces bias towards smaller tables, or Lazo [14], which uses OOPH sketches [25] to reduce the number of hashes required before constructing the LSH index. Sketch-based indexing techniques, such as Quadrant Count Ratio (QCR) [23], have also been employed to find correlating features to a target column.

High-dimensional vector indexes, such as HNSW, Inverted File Index (IVF), and Product Quantization (PQ), have recently been used to enable approximate data discovery based on semantic similarity between tables [11].

2.5 Evaluating Discovery Systems

One of the major challenges in data discovery research is the appropriate evaluation of proposed methods. Similar to other research problems in data integration, there is always a trade-off between resource consumption and quality of potential techniques. While measuring resource consumption and runtime is straightforward, it is often hard to set the proper criteria for the expected quality of data discovery results. In this tutorial, we want to delve into these challenges and discuss two different approaches that are currently carried out by researchers: 1) Close-world: A ground truth is created, often by fragmenting larger tables and trying to resynthesize the original tables. With this approach, metrics such as precision and recall can be calculated. 2) Open-world: Data discovery aims at finding serendipitous datasets that are useful for a downstream use case. Both approaches can bring insights into the capabilities of different techniques. In our tutorial, we want to highlight that the implication of each is different, which in turn might favor a certain class of techniques.

2.6 Future Directions

As our tutorial will show, research on data discovery has made significant progress in the past decade. Nevertheless, there are aspects that are still underexplored and impede the usage of data discovery solutions in practice. Among those, we will discuss the challenges in building holistic systems [10] that serve a variety of discovery tasks, data discovery in federated setups and under constraints as well as the expansion of existing techniques for additional modalities beyond relational tables.

3 TUTORIAL ORGANIZATION

We plan to carry-out the tutorial as an interactive lecture with live demonstration of discovery operations and systems. As several of the systems have been developed and reproduced at our labs, we can provide practical insights into several of the presented techniques. Overall, the structure as outlined in the previous section can be covered within 3 hours. We can trim the tutorial for a shorter session if needed, by focusing on fewer discovery operations.

Target audience. The topic of data discovery has a broad appeal within the data management community. We believe that anyone who is interested in data discovery, security, and privacy of data would be interested in this tutorial. More broadly, the tutorial is aimed at encouraging the DB community to work on novel problems in data discovery and make it more practical. All materials used will be released publicly, and the attendees will be given a hands-on experience to test the presented techniques.

Prerequisites. The background expected is that of an introductory course on data management. The tutorial has been carefully structured to accommodate both attendees unfamiliar with the topic and experienced participants by providing the required background knowledge and shared terminology.

4 TUTORIAL PRESENTERS

Mahdi Esmailoghli is a Postdoctoral researcher in the Database and Information Systems (DBIS) group at Humboldt-Universität zu Berlin. He holds a Ph.D. from TU Berlin. His Ph.D. research focused on data discovery in data lakes, in particular, he developed a holistic system to efficiently explore large data lakes to enhance the data at hand to train more effective machine learning models.

Sainyam Galhotra is an Assistant Professor in Computer Science at Cornell University and a field member for Computer Science, Statistics and Data Science. Previously, he was a Computing Innovation Fellow pursuing postdoctoral research at the University of Chicago. He received his Ph.D. from the University of Massachusetts Amherst. His research has been published in top-tier Data Management (SIGMOD, VLDB, PODS, & ICDE), AI (NeurIPS, AAAI & AIES) and Software Engineering (FSE) conferences. He is a recipient of the Best Paper Award in FSE 2017 and Most Reproducible Paper Award in both SIGMOD 2017 and 2018, and Best Artifact Paper Honorable Mention Award in SIGMOD 2023. He was recognized as a Data Science rising star, a DAAD AInet Fellow, and as the first recipient of the Krithi Ramamritham Award at UMass for contribution to database research.

Ziawasch Abedjan is Full Professor in Computer Science at the TU Berlin and Research Group Lead of the Berlin Institute for Foundations of Learning (BIFOLD) chairing the Research Group for Data Integration and Data Preparation. Previously, he chaired the Database and Information systems group at the Leibniz University Hannover. He held positions as Assistant Professor at TU Berlin, Postdoctoral Associate at MIT, Research Associate at QCRI, and Visiting Academic at Amazon Search. He received his PhD from the Hasso Plattner Institute in Potsdam and was awarded the University of Potsdam's best Dissertation Prize in 2014. He is a recipient of the Most Reproducible Paper Award in SIGMOD 2019, the Best Demo Award in SIGMOD 2015 and the Best Student Paper Award in CIKM 2014. Ziawasch Abedjan has published several papers on data discovery algorithms and systems that will be partially covered in the tutorial. His research has been published in top-tier Data Management (SIGMOD, VLDB, & ICDE) and Software Engineering (ICSE, ASE) conferences and is funded by the DFG and the German Ministry for Education.

REFERENCES

[1] Ziawasch Abedjan, Lukasz Golab, and Felix Naumann. 2015. Profiling relational data: a survey. *VLDB Journal* 24, 4 (2015), 557–581.

[2] Ziawasch Abedjan, John Morcos, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, and Michael Stonebraker. 2016. DataXFormer: A robust transformation discovery system. In *ICDE*. 1134–1145.

[3] Yael Amsterdamer and Moran Cohen. 2021. Automated Selection of Multiple Datasets for Extension by Integration. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM)*, Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong (Eds.). ACM, 27–36. https://doi.org/10.1145/3459637.3482322

[4] Sagar Bharadwaj, Praveen Gupta, Ranjita Bhagwan, and Saikat Guha. 2021. Discovering Related Data At Scale. *Proceedings of the VLDB Endowment (PVLDB)* 14, 8 (2021), 1392–1400. http://www.vldb.org/pvldb/vol14/p1392-bharadwaj.pdf

[5] Alex Bogatu, Alvaro A. A. Fernandes, Norman W. Paton, and Nikolaos Konstantinou. 2020. Dataset Discovery in Data Lakes. In *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE, 709–720. https://doi.org/10.1109/ICDE48307.2020.00067

[6] Andrei Z Broder. 1997. On the resemblance and containment of documents. In *Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171)*. IEEE, 21–29.

[7] Mahdi Esmailoghli and Ziawasch Abedjan. 2020. CAFE: Constraint-Aware Feature Extraction from Large Databases. In *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*. www.cidrdb.org. http://cidrdb.org/cidr2020/gongshow2020/gongshow/abstracts/cidr2020_abstract86.pdf

[8] Mahdi Esmailoghli, Jorge-Arnulfo Quiané-Ruiz, and Ziawasch Abedjan. 2021. COCOA: COrrelation COefficient-Aware Data Augmentation. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, Yannis Velegrakis, Demetris Zeinalipour-Yazti, Panos K. Chrysanthis, and Francesco Guerra (Eds.). OpenProceedings.org, 331–336. https://doi.org/10.5441/002/edbt.2021.30

[9] Mahdi Esmailoghli, Jorge-Arnulfo Quiané-Ruiz, and Ziawasch Abedjan. 2022. MATE: Multi-Attribute Table Extraction. *Proceedings of the VLDB Endowment (PVLDB)* 15, 8 (2022), 1684–1696. https://www.vldb.org/pvldb/vol15/p1684-esmailoghli.pdf

[10] Mahdi Esmailoghli, Christoph Schnell, Renée J. Miller, and Ziawasch Abedjan. 2025. BLEND: A Unified Data Discovery System . In *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE Computer Society, Los Alamitos, CA, USA, 737–750. https://doi.org/10.1109/ICDE65448.2025.00061

[11] Grace Fan, Jin Wang, Yuliang Li, Dan Zhang, and Renée J. Miller. 2023. Semantics-aware Dataset Discovery from Data Lakes with Contextualized Column-based Representation Learning. *Proceedings of the VLDB Endowment (PVLDB)* 16, 7 (2023), 1726–1739. https://www.vldb.org/pvldb/vol16/p1726-fan.pdf

[12] Raul Castro Fernandez, Ziawasch Abedjan, Famien Koko, Gina Yuan, Samuel Madden, and Michael Stonebraker. 2018. Aurum: A Data Discovery System. In *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE Computer Society, 1001–1012. https://doi.org/10.1109/ICDE.2018.00094

[13] Raul Castro Fernandez, Ziawasch Abedjan, Samuel Madden, and Michael Stonebraker. 2016. Towards large-scale data discovery: position paper. In *Proceedings of the Third International Workshop on Exploratory Search in Databases and the Web*, Senjuti Basu Roy, Kostas Stefanidis, Georgia Koutrika, Mirek Riedewald, and Laks V. S. Lakshmanan (Eds.). ACM, 3–5. https://doi.org/10.1145/2948674.2948675

[14] Raul Castro Fernandez, Jisoo Min, Demitri Nava, and Samuel Madden. 2019. Lazo: A Cardinality-Based Method for Coupled Estimation of Jaccard Similarity and Containment. In *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE, 1190–1201. https://doi.org/10.1109/ICDE.2019.00109

[15] Stratos Idreos, Olga Papaemmanouil, and Surajit Chaudhuri. 2015. Overview of Data Exploration Techniques. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, Timos K. Sellis, Susan B. Davidson, and Zachary G. Ives (Eds.). ACM, 277–281. https://doi.org/10.1145/2723372.2731084

[16] Fatemeh Nargesian, Ken Q. Pu, Bahar Ghadiri Bashardoost, Erkang Zhu, and Renée J. Miller. 2023. Data Lake Organization. *IEEE Trans. Knowl. Data Eng.* 35, 1 (2023), 237–250.

[17] Fatemeh Nargesian, Ken Q. Pu, Erkang Zhu, Bahar Ghadiri Bashardoost, and Renée J. Miller. 2018. Optimizing Organizations for Navigating Data Lakes. *CoRR* abs/1812.07024 (2018). arXiv:1812.07024

[18] Fatemeh Nargesian, Ken Q. Pu, Erkang Zhu, Bahar Ghadiri Bashardoost, and Renée J. Miller. 2020. Organizing Data Lakes for Navigation. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, David Maier, Rachel Pottinger, AnHai Doan, Wang-Chiew Tan, Abdussalam Alawini, and Hung Q. Ngo (Eds.). ACM, 1939–1950. https://doi.org/10.1145/3318464.3380605

[19] Fatemeh Nargesian, Erkang Zhu, Ken Q. Pu, and Renée J. Miller. 2018. Table Union Search on Open Data. *Proceedings of the VLDB Endowment (PVLDB)* 11, 7 (2018), 813–825. https://doi.org/10.14778/3192965.3192973

[20] Paul Ouellette, Aidan Sciortino, Fatemeh Nargesian, Bahar Ghadiri Bashardoost, Erkang Zhu, Ken Pu, and Renée J. Miller. 2021. RONIN: Data Lake Exploration. *Proceedings of the VLDB Endowment (PVLDB)* 14, 12 (2021), 2863–2866. http://www.vldb.org/pvldb/vol14/p2863-nargesian.pdf

[21] Norman W. Paton and Zhenyu Wu. 2024. Dataset Discovery and Exploration: State-of-the-art, Challenges and Opportunities. In *Proceedings 27th International Conference on Extending Database Technology, EDBT 2024, Paestum, Italy, March 25 - March 28*, Letizia Tanca, Qiong Luo, Giuseppe Polese, Loredana Caruccio, Xavier Oriol, and Donatella Firmani (Eds.). OpenProceedings.org, 854–857. https://doi.org/10.48786/EDBT.2024.87

[22] Aécio S. R. Santos, Aline Bessa, Fernando Chirigati, Christopher Musco, and Juliana Freire. 2021. Correlation Sketches for Approximate Join-Correlation Queries. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, Guoliang Li, Zhanhuai Li, Stratos Idreos, and Divesh Srivastava (Eds.). ACM, 1531–1544. https://doi.org/10.1145/3448016.3458456

[23] Aécio S. R. Santos, Aline Bessa, Christopher Musco, and Juliana Freire. 2022. A Sketch-based Index for Correlated Dataset Search. In *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE, 2928–2941. https://doi.org/10.1109/ICDE53745.2022.00264

[24] Anish Das Sarma, Lujun Fang, Nitin Gupta, Alon Y. Halevy, Hongrae Lee, Fei Wu, Reynold Xin, and Cong Yu. 2012. Finding related tables. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, K. Selçuk Candan, Yi Chen, Richard T. Snodgrass, Luis Gravano, and Ariel Fuxman (Eds.). ACM, 817–828. https://doi.org/10.1145/2213836.2213962

[25] Anshumali Shrivastava. 2017. Optimal Densification for Fast and Accurate Minwise Hashing. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017 (Proceedings of Machine Learning Research)*, Doina Precup and Yee Whye Teh (Eds.), Vol. 70. PMLR, 3154–3163. http://proceedings.mlr.press/v70/shrivastava17a.html

[26] Karamjit Singh, Kaushal Paneri, Aditeya Pandey, Garima Gupta, Geetika Sharma, Puneet Agarwal, and Gautam Shroff. 2016. Visual Bayesian fusion to navigate a data lake. In *FUSION*. IEEE, 987–994.

[27] Sahaana Suri, Ihab F. Ilyas, Christopher Ré, and Theodoros Rekatsinas. 2021. Ember: No-Code Context Enrichment via Similarity-Based Keyless Joins. *Proceedings of the VLDB Endowment (PVLDB)* 15, 3 (2021), 699–712. https://doi.org/10.14778/3494124.3494149

[28] Petros Venetis, Yannis Sismanis, and Berthold Reinwald. 2012. *CRSI*: a compact randomized similarity index for set-valued features. In *Proceedings of the International Conference on Extending Database Technology (EDBT)*, Elke A. Rundensteiner, Volker Markl, Ioana Manolescu, Sihem Amer-Yahia, Felix Naumann, and Ismail Ari (Eds.). ACM, 384–395. https://doi.org/10.1145/2247596.2247642

[29] Chuan Xiao, Wei Wang, Xuemin Lin, and Haichuan Shang. 2009. Top-k Set Similarity Joins. In *Proceedings of the 25th International Conference on Data Engineering, ICDE 2009, March 29 2009 - April 2 2009, Shanghai, China*, Yannis E. Ioannidis, Dik Lun Lee, and Raymond T. Ng (Eds.). IEEE Computer Society, 916–927. https://doi.org/10.1109/ICDE.2009.111

[30] Chuan Xiao, Wei Wang, Xuemin Lin, Jeffrey Xu Yu, and Guoren Wang. 2011. Efficient similarity joins for near-duplicate detection. *ACM Trans. Database Syst.* 36, 3 (2011), 15:1–15:41. https://doi.org/10.1145/2000824.2000825

[31] Mohamed Yakout, Kris Ganjam, Kaushik Chakrabarti, and Surajit Chaudhuri. 2012. InfoGather: entity augmentation and attribute discovery by holistic matching with web tables. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, K. Selçuk Candan, Yi Chen, Richard T. Snodgrass, Luis Gravano, and Ariel Fuxman (Eds.). ACM, 97–108. https://doi.org/10.1145/2213836.2213848

[32] Shuo Zhang and Krisztian Balog. 2019. Web Table Extraction, Retrieval and Augmentation. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval (SIGIR)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3331184.3331385

[33] Yi Zhang and Zachary G. Ives. 2019. Juneau: Data Lake Management for Jupyter. *Proceedings of the VLDB Endowment (PVLDB)* 12, 12 (2019), 1902–1905. https://doi.org/10.14778/3352063.3352095

[34] Yi Zhang and Zachary G. Ives. 2020. Finding Related Tables in Data Lakes for Interactive Data Science. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, David Maier, Rachel Pottinger, AnHai Doan, Wang-Chiew Tan, Abdussalam Alawini, and Hung Q. Ngo (Eds.). ACM, 1951–1966. https://doi.org/10.1145/3318464.3389726

[35] Erkang Zhu, Dong Deng, Fatemeh Nargesian, and Renée J. Miller. 2019. JOSIE: Overlap Set Similarity Search for Finding Joinable Tables in Data Lakes. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, Peter A. Boncz, Stefan Manegold, Anastasia Ailamaki, Amol Deshpande, and Tim Kraska (Eds.). ACM, 847–864. https://doi.org/10.1145/3299869.3300065

[36] Erkang Zhu, Fatemeh Nargesian, Ken Q. Pu, and Renée J. Miller. 2016. LSH Ensemble: Internet-Scale Domain Search. *Proceedings of the VLDB Endowment (PVLDB)* 9, 12 (2016), 1185–1196. https://doi.org/10.14778/2994509.2994534