EDITOR: Schahram Dustdar, dustdar@dsg.tuwien.ac.at

DEPARTMENT: INTERNET OF THINGS, PEOPLE, AND PROCESSES

Toward Building Edge Learning Pipelines

Anastasios Gounaris and Anna-Valentini Michailidou, *Aristotle University of Thessaloniki, 541 24, Thessaloniki, Greece*

Schahram Dustdar, *Technical University of Vienna, 1040, Vienna, Austria*

From a bird's eye point of view, large-scale data analytics workflows, e.g., those executed in popular tools, such as Apache Spark and Flink, are typically represented by directed acyclic graphs. Also, they are in a large scale in two dimensions: first, they are capable of processing big data (e.g., both in terms of volume and velocity) mainly through employing massive parallelism, and second, they can run over (powerful) distributed infrastructures. This article focuses on edge computing and its confluence with big data analytics workflows, which nowadays place special emphasis on deep learning and data quality.

Early examples of large-scale data analytics workflows were primarily MapReduce programs,[1–3] which, however, could only handle nonstreaming data over fixed data centers; streaming data analytics was initially evolving rather independently,[4] but nowadays, massive parallelism for processing data streams is the norm.[5] There were also several efforts that have tried to extend database query plan optimization technology to account for arbitrary user-defined functions, so that such plans can correspond to generic analytics workflows extending traditional ETL data pipelines.[6–9] Overall, it has been shown that database technology can offer a lot to large-scale data analytics workflows from its several decades of experience in terms of declarativeness and principled (cost based) optimization.[6,7,10,11]

More recently, research emphasis regarding executing big data analytics tasks is placed on the following topics.

- High-level system details, such as 1) derivation of the exact requirements from the software engineering point of view and the exact architecture to be adopted,[12] or 2) the data models underlying big data analytics.[13]
- Low-level execution engine details, e.g., with regards to aspects, such as state management.[14] This is also related to tuning and optimized resource usage in complex systems for big data analytics, such as Hadoop and Spark[15,16] along with service level agreement (SLA) management when such systems are deployed in the cloud.[17]
- In addition to the advances abovementioned, several efforts advocate taking a more holistic view in modern data analytics, i.e., address all components and steps involved in real applications, from storage to user interfaces and DevOps, and from data preparation to (iterative) model building and validation. Also, in practice, complete ecosystems need to be developed around processing engines for big data analytics.[18–20]

In all the aspects mentioned thus far, mature industry-level solutions exist and are adopted by both researchers and practitioners. Nevertheless, these aspects are not adequate for supporting large-scale data analytics workflows to their full extent, as defined at the beginning of this report. This is because the current solutions cannot support deployment over arbitrarily resource-constrained distributed computing infrastructures. Modern data analytics engines are decoupled from fixed data centers and are moved to cloud solutions, but their deployment remains largely centralized. In other words, there is a lack of mature support of deployment of data-intensive analytics workflow jobs over widely heterogeneous multiowner geo-distributed edge, fog and/or cloud resources. However, common IoT and edge computing settings are characterized by all these three factors, namely, 1) heterogeneity in several dimensions including resource characteristics, availability,

1089-7801 © 2023 IEEE
Digital Object Identifier 10.1109/MIC.2022.3171643
Date of current version 3 February 2023.

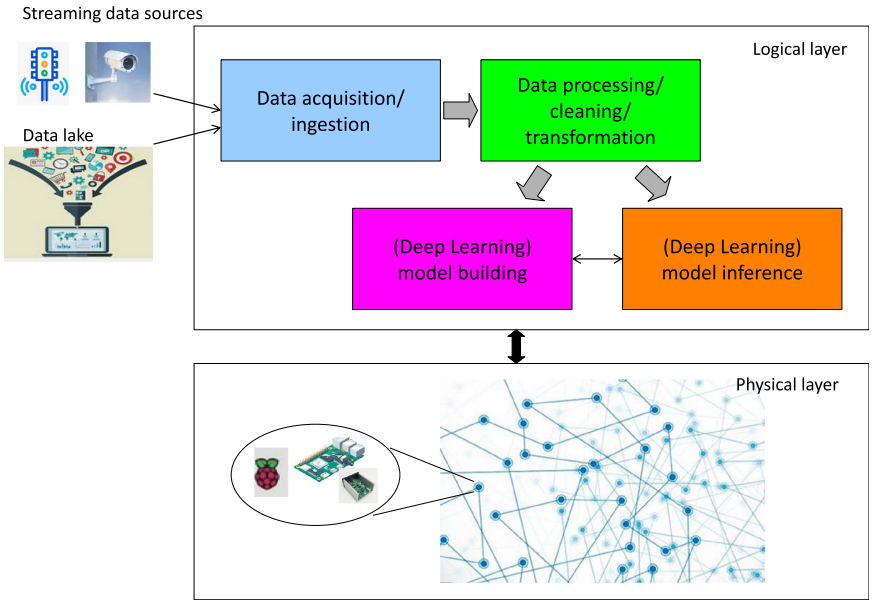

FIGURE 1. Example of an edge data acquisition, processing, and learning pipeline.

permissions and connection speeds, 2) geographical distribution, and 3) multi-ownership. Such settings are thus not adequately covered.

Our motivating remark: Future technical advances in data analytics pipelines should target to fill this gap, i.e., to account for heterogeneity, geographical distribution, and multiownership in large-scale data analytics workflows. To this end, there are several initiatives to adapt data-center-oriented solutions, such as Hadoop and Spark, to heterogeneous geo-distributed settings.[21,22] But all these fall short in dealing with primary concerns that edge and fog computing realms entail in a holistic manner. In Bansal *et al.*'s work,[23] which discusses the confluence between IoT and Big Data, several challenges are identified with regards to aspects, such as volume, velocity, variety, veracity, value, variability, visualization, validity, vulnerability, volatility, venue, vocabulary, and vagueness. Even when treating single aspects, such as venue, in isolation, the corresponding research is rather in its infancy, and many aspects are typically not considered. For example, for venue, commonly employed schedulers, resource managers, and orchestrators, such as Kubernetes, YARN, and MESOS, cannot place tasks at arbitrary geo-distributed places in a judicious manner. But this is just one part of the complete picture. It is important to acknowledge that, especially in an edge computing setting, there is a growing and demanding need for 1) treating data quality aspects as a first-class citizen and 2) move complex deep learning model training and inference to the edge,[24] which adds significant complexity to the analysis pipelines.

VISION FOR NEXT-GENERATION EDGE-ENABLED BIG-DATA ANALYTICS WORKFLOWS

Next generation edge-enabled big data analytics workflow solutions should not only address the current limitations but also go beyond them. We envisage a solution that would not only allow to run every analytics task everywhere but can also detect the appropriate data sources to feed the analysis tasks in an automated or at least semiautomated manner. By *everything*, we cover, for example, intelligent deep learning model training and inference. By *everywhere*, we cover cases, where a federation of low-end edge/fog devices forms the computation infrastructure to execute the workflows. For this vision to be realized, it is important to blend data lake technologies with edge learning so that local model training can benefit from all the relevant data required rather than locally produced ones solely.

Imagine a smart-city scenario, where advanced deep learning model construction, and inference are deployed on edge devices, e.g., to reduce latency.[25] In the rest of this article, edge and fog devices will be used interchangeably for simplicity. Such a scenario includes always-on surveillance coupled with the ingestion of data streams from third parties, e.g., to acquire meteorological conditions data. Similarly, in many application domains benefiting from edge learning, such as smart health and agriculture, it is common to join multiple data sources.[26] Retail is another field that can benefit from edge analytics. In this application domain, the real-time big data are

collected through various methods, including video cameras, basket analysis, POS terminals, and customer memberships.[27–29] Moreover, by combining these data with data from data lakes, for example, social media posts, demographic features, and customer information,[30] retailers can forecast product demand, predict customer purchases, provide personalized advertisements, discover trends, and grow their overall profit, all these based on edge learning techniques.

Edge learning implies several additional features of the corresponding workflow: 1) model building needs to be parallelized across several different computing nodes in an efficient heterogeneity-aware manner,[31–33] 2) data need to be shared only partially and after ensuring that any privacy concerns are addressed,[34] and 3) intensive data-quality actions, such as outlier removal need to take place to avoid data poisoning.[35]

The workflow, depicted as a DAG, comprises four main groups of tasks corresponding to data acquisition, data processing, model building, and model inference, respectively (see Figure 1). Each of these groups is something broader than, for instance, a single stage in Spark. Tasks may interact in complex manners and they may also involve human interaction.

Our vision includes the following three pillars.

1) Allow end users to define complex workflows in a mostly declarative manner, and these workflows to run on top of any (edge/fog) computational infrastructure judiciously in a massively parallel manner. This entails optimal resource usage and task allocation taking into account a wide range of data quality and optimization criteria, well beyond 2 or 3 typically employed in modern multiobjective scheduling/task allocation solutions.
2) Encapsulate integration with data lakes technology, possibly involving novel human-in-the-loop architectures to semiautomatically, detect the appropriate data sources that feed the remainder of the complex workflow analysis pipelines.
3) Account for edge learning scenarios, which impose strict constraints on which data can be shared and may require the existence of mutually trusted central cloud nodes that become responsible for specific parts of the model construction.

Building the aforementioned pillars should cover the big-data aspects in the Bansal *et al.*'s[23] work, also termed the 13 V's, as summarized in Table 1. In the rightmost column, we mention the challenges involved, which range from dealing with novel data and task placement problems to integrating data quality detection and improvement solutions, and appropriate source detection in data lakes.

Edge learning improvements: Our vision can be deemed as a call for extension to the state of the art in edge learning,[24] which currently focuses on building ML models over edge devices in a collaborative manner, while considering data, computational, communication, privacy, security, and incentive-related challenges. As reported in the Deng *et al.*'s[36] work, not only data analytics on the edge, but even the more restrictive scenario of AI on the edge employing a limited set of optimization criteria is a topic that requires much deeper investigation. In any case, the extensions are very important in that

1) they do not separate data acquisition and processing pipelines with the model training/inference ones;
2) they account for the full spectrum of big data aspects; and
3) they call for novel workflow management, i.e., expression, execution, and scheduling techniques.

These extensions are further analyzed in the following.

Toward Next-Generation Edge Learning: A Closer Look at the Three Axes

First, integrating data lake technology with database engines has already been identified as a key research direction[37]; what we advocate is such an integration to also cover the edge learning workflows that we aim to run over geo-distributed edge nodes. There are three main problems that are encountered:

(1) detection of the most appropriate sources
(2) optimized sharing of data across all nodes that run model construction tasks and may benefit from such sources
(3) including humans in the loop.

Why this is challenging? If a single computational node becomes responsible for source detection, this node may easily become a bottleneck. However, if multiple nodes undertake this task, it is unclear how to split the corresponding workload and synchronize their searching process. Finally, having the human-in-the-loop leads to the development of a whole new family of techniques.

Second, covering the full spectrum of Big Data aspects is strongly connected to meeting the 13 V's requirements abovementioned. In addition to the presentation in Table 1, which explains how all big data aspects are important in our vision, data quality issues

TABLE 1. Issues and challenges in multifaceted coverage of big data aspects in our vision.

Aspect	Description of impact on the solutions	Challenge
Volume	Relates to maximizing throughput, leveraging massive parallelism, moving filtering operations as close to the data sources as possible, reusing data, minimizing data transfers, and so on.	Data cannot be arbitrarily shared, which renders existing techniques inefficient or even inapplicable.
Variety	Relates to considering all kinds of resource heterogeneity involved.	The variety covers both computational and networking infrastructure and the local datasets available on each edge device, the combination of which is not currently considered.
Velocity	Emphasizes on minimizing latency, heavily relies on massive parallelism on top of heterogeneous resources, and poses restrictions on where model inference can run.	Incurs tradeoffs when deep learning models are large and need to be split across multiple nodes.
Veracity	Calls for detecting the most appropriate and trustworthy data sources on the fly.	Calls for the development of novel data lake-aware edge learning solutions that emphasize on both the training and the data acquisition process.
Value	Relates to including intelligent analytics and machine learning (ML) steps in the workflows apart from simpler data management tasks.	Such intelligent analytics may require synchronizations, which are difficult to be attained in a heterogeneous setting.
Variability	Relates to the capability of the solution to adapt to environmental changes, i.e., any task/data placement solutions may need to be adaptive.	Tasks are typically stateful.
Visualization	Relates to the fact that human-in-the-loop is a key distinctive feature (as also in the Industry 5.0 vision).	Impacts on metrics, such as latency, in a non-straightforward manner.
Validity	Envisaged as including data quality checks and enforcement steps as first-class citizens (in addition to data management and ML operations).	Data quality can be quantified in several manners and is not typically considered using execution plan optimization.
Vulnerability	Calls for addressing privacy and security requirements, an issue of paramount importance in edge learning.	Involves tricky tradeoffs with performance and placement flexibility.
Volatility	Calls for continuously refining analysis results as more data are produced, that is, the corresponding analytics workflows should run continuously to both refine and apply trained models.	Calls for novel techniques to reduce operations and data transmissions when no changes from previous values and/or results are detected/predicted.
Venue	Relates to the judicious placement of tasks to resources.	Need to account for resource heterogeneity and geographical distribution.
Vocabulary	Relates to the development of higher level (declarative) abstractions to describe tasks, resources, constraints, objectives, and so on.	No standardized approach exists to date for the relevant aspects.
Vagueness	Complements validity and veracity.	Same as validity and veracity abovementioned.

should be further emphasized. Data quality aspects are defined in multiple manners. For instance, in Deequ,[a] Apache Griffin,[b] and Great Expectations,[c] simple data checking operations are defined and implemented. However, data quality aspects can be described more broadly, e.g., through ISO-25012[d] with a view to covering the veracity, validity, and vagueness big data dimensions. In this standard, there are several relevant aspects of data quality. For example, completeness relates not only to the desire the input data to have non-NULL values but also all the corresponding data for model training to be available. Also, precision monitors IoT streams for unjustified data fluctuations, which are attributed to sensor malfunction; assessing precision in this sense entails the insertion of a lightweight statistics module in the complete analysis pipeline. As a third example, credibility relates to the accuracy of an ML model and is affected by the presence of a human in the loop. Data quality aspects are also directly relevant to optimization objectives, e.g., timeliness relates to the pipeline performance and its capability to perform model refinement and inference with low latency.

More specifically, examining the 15 data quality characteristics of the ISO-25012, we can extract eight of them, as presented in Table 2, which can be directly mapped to optimization objectives and encapsulation of data quality-oriented tasks in the pipeline. The other seven data quality characteristics are also relevant, but they cannot be easily quantified in our context, e.g., accessibility, understandability, and portability. The quantitative metrics in the table complement performance metrics, such as throughput, latency, power, resource, and network utilization, which are well understood.[4] Also, it is still important to consider quality of service (QoS), which may be deemed as quantifying accuracy after load shedding or can be application dependent.

Third, workflow management should be geared toward more declarativity, well beyond merely employing and calling complex ML libraries through user-friendly scripts, as is the main status to date.[10] The extended set of optimization criteria and constraints raise the need for a convenient manner to express them; similarly, the user feedback needs to be in a format amenable to immediate processing and enactment of corresponding actions, e.g., with regards to source selection. Thus, there is an interplay of expression and execution. Execution is also affected by the data and task placement decisions that also need to be controlled, at least partially, in a declarative manner. This implies changes in the underlying resource managers, negotiators, and schedulers.

TABLE 2. Data quality characteristics, as defined in ISO-25012 and the corresponding envisaged optimization metrics and tasks in data analytics pipelines.

Characteristic	Quantitative metric	Corresponding task
Accuracy	Degree to which values of ingested data deviate from their reference values.	Measure the accuracy; choose data sources based on their accuracy values.
Completeness	Number of data features extracted from external data sources employed in model building.	Seek for relevant and combinable data sources.
Consistency	Degree to which values for the same features from different sources are aligned.	Measure the consistency; consider consistency when choosing data sources.
Credibility	Degree to which data and models built is believable by users.	Receive human feedback on the credibility of external data-lake-based sources and ML models.
Currentness	The time difference between data generation and data processing.	Assess the currentness (also referred to as timeliness).
Compliance	Degree to which fields such as timestamps follow standards.	Task to assess compliance; choose data sources based on their compliance values.
Precision	Degree to which sensing mechanisms produce precise measurements.	Assess the measurements fluctuations due to sensing mechanism imprecision.
Availability	Degree to which external data sources are available.	Profiling of the availability of external sources.

[a][Online]. Available: https://github.com/awslabs/deequ
[b][Online]. Available: https://griffin.apache.org/
[c][Online]. Available: https://github.com/great-expectations/great_expectations
[d][Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards/iso-25012

TECHNICAL NOTES REGARDING THE RESEARCH ISSUES INVOLVED

The abovementioned discussion entails and touches upon several topics. In the following, we further elaborate on four of them.

Data source selection: This is the most challenging part when building data lake-aware edge learning pipelines. State-of-the-art solutions leverage LSH, while also focusing on feature engineering, consideration of both schema- and instance-level data, and advanced data transformations to reason about the relatedness of data sources.[38,39] The ultimate goal is to yield a list of combinable data sources. When the data sources are mapped to a relational schema, this goal is equivalent to detect joinable tables. Apart from the related data quality objectives, there are several performance-related optimizations that need to be taken into account given the high computation complexity of these tasks; such optimizations all target aggressive pruning of the search space. Combining these objectives, with data quality-related ones and human involvement, gives rise to optimization problems radically different than those encountered when considering task allocation. However, still, a promising approach is to aim to cast the whole problem as an integer linear programming one in a manner that no scalability problems are encountered, and enhance the initial solution with nature-inspired techniques,[40] something already shown to work very well in demanding geo-distributed, heterogeneous scenarios.[22,41]

Workflow expression: Declarative statement of service-level objectives (SLO) is not something new and is extensively employed in guiding elasticity in large-scale heterogeneous environments, e.g., Pusztai *et al.*'s[42] work. Such initiatives may serve as a basis to build more complete solutions that consider the full range of criteria and constraints involved, and also account for actions other than elasticity. More specifically, the elasticity actions need to be extended to allow not only the scaling and migration of tasks, but also the incorporation of additional data quality-oriented tasks and the reconfiguration of running tasks on the fly. This entails the development of novel schedulers and extensions to current state-of-the-art resource managers. It also implies more advanced optimization modules, which are discussed separately in the following.

Workflow optimization: Analyzing data closer to the edge devices, rather than in a central cloud, offers low latency, security, and scalability in many scenarios, such as smart cities. Several challenges arise when developing an edge computing-oriented analytics optimizer. Edge devices are highly heterogeneous in terms of resources, such as memory and computational capacity, and initial works that model such heterogeneity exist, e.g., Hiessl *et al.*'s[43] work. Furthermore, different cellular networks, as well as the emerging 5G technology, induce an additional challenge when combined.[44] Edge devices may also comprise smartphones and other mobile devices. This mobility needs to be taken into account when seeking an optimal service placement.[45] Overall, the corresponding service placement needs to be dynamic and adaptive to network and resource changes in real-time.[46] Finally, when dealing with sharing data across multiple edge devices, privacy constraints and restrictions may arise.[47] Thus, it becomes clear that multiple aspects should be considered when optimizing edge computing-oriented analytics. There is also a need to optimize multiple objectives at the same time or find a beneficial tradeoff between them. For example, response time, latency, energy consumption, and data transfer need to be minimized while resource utilization and QoS need to be maximized. Moreover, developing dynamic pricing models for service providers poses an additional challenge.[44] Independently optimizing workflows may be sufficient for scenarios with a small number of users; however, in edge computing applications, multiple users submit queries at the same time. Optimizing these queries simultaneously is complicated but would lead to more efficient resource utilization as techniques, such as service caching and resource sharing could be utilized.

Overall, the biggest challenges in the optimization of the workflows that we envisage stem from the combination of a much broader set of constraints and additional data quality-oriented objectives. Furthermore, the optimizations are not merely limited to judicious multiobjective task and data placement, configuration of parallelism degree, choice of the operator implementation, and so on. They should also cover modifications of the logical DAG execution plan, e.g., through inserting new data acquisition- and quality-specific operators. Also, modifying the type of tasks in the DAG based on the placement choices needs to be considered. For example, inference using complex deep networks could be allocated to either a set of edge nodes running different layers sequentially or to a single node, and thus, the workflow DAG is modified accordingly to reflect such decisions.

Workflow frameworks: In addition, the combination of large-scale data analytics frameworks, such as Apache Spark, Flink, and Storm, with edge learning frameworks, such as TensorFlow Federated[e] and Fate,[f] needs to be investigated in depth. Reinventing

[e][Online]. Available: https://www.tensorflow.org/federated
[f][Online]. Available: https://fate.fedai.org/

the wheel should be avoided, but it is unclear how this can be achieved in practice.

SUMMARY

Blending data acquisition, advanced ML, and analytics workflows to be executed over arbitrary heterogeneous, and geo-distributed computational resources both envisages and aspires to develop next-generation big data analytics and edge learning solutions. Current technologies need to be significantly extended in terms of the big data aspects directly considered, which in turn yields an updated list of optimization criteria, SLOs, and constraints. Data lake technologies, human intervention, and data quality guarantees become far more prevalent, while the underlying workflow execution engines need to be equipped with more advanced optimizers. Nevertheless, significant research efforts have already been conducted in several isolated aspects of the complete vision described hereby. Therefore, the technical roadmap is twofold: to both extend and judiciously combine existing solutions rather than starting from scratch, which is inefficient and unnecessary. To this end, we have identified the main research issues, and we sketched the current state of the art on top of which we advocate to build.

ACKNOWLEDGMENTS

The work of Anastasios Gounaris and Anna-Valentini Michailidou were supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.), Greece, through the First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment Grant under Project 1052.

REFERENCES

1. C. Doulkeridis and K. Nørvåg, "A survey of large-scale analytical query processing in MapReduce," *VLDB J.*, vol. 23, pp. 355–380, Dec. 2013, doi: 10.1007/s00778-013-0319-9.
2. S. Babu and H. Herodotou, "Massively parallel databases and MapReduce systems," *Found. Trends Databases*, vol. 5, pp. 1–104, 2013, doi: 10.1561/1900000036.
3. F. Li, B. C. Ooi, M. T. Özsu, and S. Wu, "Distributed data management using MapReduce," *ACM Comput. Surv.*, vol. 46, 2014, Art. no. 31, doi: 10.1145/2503009.
4. M. Hirzel, R. Soulé, S. Schneider, B. Gedik, and R. Grimm, "A catalog of stream processing optimizations," *ACM Comput. Surv.*, vol. 46, Mar. 2014, Art. no. 46, doi: 10.1145/2528412.
5. M. Fragkoulis, P. Carbone, V. Kalavri, and A. Katsifodimos, "A survey on the evolution of stream processing systems," 2020, *arXiv:2008.00842*, doi: 10.48550/arXiv.2008.00842.
6. A. Rheinländer, U. Leser, and G. Graefe, "Optimization of complex dataflows with user-defined functions," *ACM Comput. Surv.*, vol. 50, May 2018, Art. no. 38, doi: 10.1145/3078752.
7. G. Kougka, A. Gounaris, and A. Simitsis, "The many faces of data-centric workflow optimization: A survey," *Int. J. Data Sci. Analytics*, vol. 6, pp. 81–107, 2018, doi: 10.1007/s41060-018-0107-0.
8. S. M. F. Ali and R. Wrembel, "From conceptual design to performance optimization of ETL workflows: Current state of research and open problems," *VLDB J.*, vol. 26, pp. 777–801, Sep. 2017, doi: 10.1007/s00778-017-0477-2.
9. P. Jovanovic, O. Romero, and A. Abelló, "A unified view of data-intensive flows in business intelligence systems: A survey," in *Proc. Trans. Large-Scale Data-Knowl.-Centered Syst.*, 2016, pp. 66–107, doi: 10.1007/978-3-662-54037-4_3.
10. N. Makrynioti and V. Vassalos, "Declarative data analytics: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2392–2411, Jun. 2021, doi: 10.1109/TKDE.2019.2958084.
11. A. Modi *et al.*, "New query optimization techniques in the spark engine of azure synapse," *Proc. VLDB Endowment*, vol. 15, pp. 936–948, 2021, doi: 10.14778/3503585.3503601.
12. A. Davoudian and M. Liu, "Big data systems: A software engineering perspective," *ACM Comput. Surv.*, vol. 53, pp. 110:1–110:39, 2020, doi: 10.1145/3408314.
13. H. V. Olivera, G. RuiZhe, R. C. Huacarpuma, A. P. B. da Silva, A. M. Mariano, and M. Holanda, "Data modeling and NoSQL databases - A Systematic mapping review," *ACM Comput. Surv.*, vol. 54, pp. 116:1–116:26, 2021, doi: 10.1145/3457608.
14. Q.-C. To, J. Soto, and V. Markl, "A survey of state management in big data processing systems," *VLDB J.*, vol. 27, pp. 847–872, 2018, doi: 10.1007/s00778-018-0514-9.
15. H. Herodotou, Y. Chen, and J. Lu, "A survey on automatic parameter tuning for big data processing systems," *ACM Comput. Surv.*, vol. 53, 2020, Art. no. 43, doi: 10.1145/3381027.
16. I. A. T. Hashem *et al.*, "MapReduce scheduling algorithms: A review," *J. Supercomput.*, vol. 76, pp. 4915–4945, 2020, doi: 10.1007/s11227-018-2719-5.
17. X. Zeng *et al.*, "SLA management for big data analytical applications in clouds: A taxonomy study," *ACM Comput. Surv.*, vol. 53, pp. 46:1–46:40, 2020, doi: 10.1145/3383464.

18. S. Khalifa *et al.*, "The six pillars for building big data analytics ecosystems," *ACM Comput. Surv.*, vol. 49, 2016, Art. no. 33, doi: 10.1145/2963143.
19. C. C. Aggarwal *et al.*, "How can AI automate End-to-End data science?," 2019, *arXiv:1910.14436*, doi: 10.48550/arXiv.1910.14436.
20. S. Tang, B. He, C. Yu, Y. Li, and K. Li, "A survey on spark ecosystem: Big data processing infrastructure, machine learning, and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 71–91, Jan. 2022, doi: 10.1109/TKDE.2020.2975652.
21. S. Dolev, P. Florissi, E. Gudes, S. Sharma, and I. Singer, "A survey on geographically distributed big-data processing using MapReduce," *IEEE Trans. Big Data*, vol. 5, no. 1, pp. 60–80, Mar. 2019, doi: 10.1109/TBDATA.2017.2723473.
22. A.-V. Michailidou, A. Gounaris, M. Symeonides, and D. Trihinas, "EQUALITY: Quality-aware intensive analytics on the edge," *Inf. Syst.*, vol. 105, 2022, Art. no. 101953, doi: 10.1016/j.is.2021.101953.
23. M. Bansal, I. Chana, and S. Clarke, "A survey on IoT big data: Current status, 13 V's challenges, and future directions," *ACM Comput. Surv.*, vol. 53, 2021, Art. no. 131, doi: 10.1145/3419634.
24. J. Zhang *et al.*, "Edge learning: The enabling technology for distributed big data analytics in the edge," *ACM Comput. Surv.*, vol. 54, pp. 151:1–151:36, 2022, doi: 10.1145/3464419.
25. C.-J. Wu *et al.*, "Machine learning at facebook: Understanding inference at the edge," in *Proc. 25th IEEE Int. Symp. High Perform. Comput. Archit., HPCA*, 2019, pp. 331–344, doi: 10.1109/HPCA.2019.00048.
26. P. Raith and S. Dustdar, "Edge intelligence as a service," in *Proc. IEEE Int. Conf. Serv. Comput.*, 2021, pp. 252–262, doi: 10.1109/SCC53864.2021.00038.
27. H. B. Pasandi and T. Nadeem, "CONVINCE: Collaborative cross-camera video analytics at the edge," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, 2020, pp. 1–5, doi: 10.1109/PerComWorkshops48775.2020.9156251.
28. A. W. Senior *et al.*, "Video analytics for retail," in *Proc. 4th IEEE Int. Conf. Adv. Video Signal Based Surveill.*, 2007, pp. 423–428, doi: 10.1109/AVSS.2007.4425348.
29. A. Griva, C. Bardaki, K. Pramatari, and D. Papakyriakopoulos, "Retail business analytics: Customer visit segmentation using market basket data," *Expert Syst. Appl.*, vol. 100, pp. 1–16, Feb. 2018, doi: 10.1016/j.eswa.2018.01.029.
30. K. B. Subramanya and A. Somani, "Enhanced feature mining and classifier models to predict customer churn for an E-retailer," in *Proc. 7th Int. Conf. Cloud Comput., Data Sci. Eng. - Confluence*, 2017, pp. 531–536, doi: 10.1109/CONFLUENCE.2017.7943208.
31. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1–10. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html
32. H.-J. Jeong, H.-J. Lee, C. H. Shin, and S.-M. Moon, "IONN: Incremental offloading of neural network computations from mobile devices to edge servers," in *Proc. ACM Symp. Cloud Comput.*, 2018, pp. 401–411, doi: 10.1145/3267809.3267828.
33. J. Cipar *et al.*, "Solving the straggler problem with bounded staleness," in *Proc. 14th Workshop Hot Topics Oper. Syst.*, 2013, Art. no. 22, doi: 10.5555/2490483.2490505.
34. J. Zhao, "Distributed deep learning under differential privacy with the teacher-student paradigm," in *Proc. Workshops 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 404–407.
35. J. Steinhardt, P. W. Koh, and P. Liang, "Certified defenses for data poisoning attacks," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3520–3532, doi: 10.5555/3294996.3295110.
36. S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge intelligence: The confluence of edge computing and artificial intelligence," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7457–7469, Aug. 2020, doi: 10.1109/JIOT.2020.2984887.
37. D. Abadi *et al.*, "The Seattle report on database research," *SIGMOD Rec.*, vol. 48, pp. 44–53, 2019, doi: 10.1145/3385658.3385668.
38. A. Bogatu, A. A. A. Fernandes, N. W. Paton, and N. Konstantinou, "Dataset discovery in data lakes," in *Proc. 36th IEEE Int. Conf. Data Eng.*, 2020, pp. 709–720, doi: 10.1109/ICDE48307.2020.00067.
39. Y. Dong, K. Takeoka, C. Xiao, and M. Oyamada, "Efficient joinable table discovery in data lakes: A high-dimensional similarity-based approach," in *Proc. 37th IEEE Int. Conf. Data Eng.*, 2021, pp. 456–467, doi: 10.1109/ICDE51399.2021.00046.
40. J. Brownlee, *Clever Algorithms: Nature-Inspired Programming Recipes*, 1st ed. 2011. [Online]. Available: https://github.com/clever-algorithms/CleverAlgorithms
41. M. Nardelli, V. Cardellini, V. Grassi, and F. L. Presti, "Efficient operator placement for distributed data stream processing applications," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 8, pp. 1753–1767, Aug. 2019, doi: 10.1109/TPDS.2019.2896115.
42. T. W. Pusztai *et al.*, "SLO script: A novel language for implementing complex cloud-native elasticity-driven SLOs," in *Proc. IEEE Int. Conf. Web Serv.*, 2021, pp. 21–31, doi: 10.1109/ICWS53863.2021.00017.

43. T. Hiessl, V. Karagiannis, C. Hochreiner, S. Schulte, and M. Nardelli, "Optimal placement of stream processing operators in the fog," in *Proc. 3rd IEEE Int. Conf. Fog Edge Comput.*, 2019, pp. 1–10, doi: 10.1109/CFEC.2019.8733147.
44. W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, "Edge computing: A survey," *Future Gener. Comput. Syst.*, vol. 97, pp. 219–235, 2019, doi: 10.1016/j.future.2019.02.050.
45. N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018, doi: 10.1109/JIOT.2017.2750180.
46. Y. Teranishi, T. Kimata, H. Yamanaka, E. Kawai, and H. Harai, "Dynamic data flow processing in edge computing environments," in *Proc. 41st IEEE Annu. Comput. Softw. Appl. Conf.*, 2017, pp. 935–944, doi: 10.1109/COMPSAC.2017.113.
47. W. Yu *et al.*, "A survey on the edge computing for the Internet of Things," *IEEE Access*, vol. 6, pp. 6900–6919, 2018, doi: 10.1109/ACCESS.2017.2778504.

ANASTASIOS GOUNARIS is an associate professor at the Department of Informatics, Aristotle University of Thessaloniki, 541 24, Greece. His main research interests include large-scale data management, massive parallelism, workflow and business process optimization, big data analytics, and data mining. Gounaris received his Ph.D. degree from the University of Manchester, Manchester, U.K. Contact him at http://datalab.csd.auth.gr/~gounaris/.

ANNA-VALENTINI MICHAILIDOU is a Ph.D. student with the Department of Informatics, Aristotle University of Thessaloniki, 541 24, Greece. Her research interests include distributed data analytics, dataflow and workflow optimization, data-quality, and edge computing. Michailidou received her B.Sc. degree in informatics from the Aristotle University of Thessaloniki. Contact her at http://annavalen.webpages.auth.gr/.

SCHAHRAM DUSTDAR is full professor of computer science heading the Research Division of Distributed Systems, TU Wien, 1040, Austria. He is an IEEE fellow. Contact him at dustdar@dsg.tuwien.ac.at.