Big data analytics in telecommunications: literature review and architecture recommendations

Article *in* IEEE/CAA Journal of Automatica Sinica · December 2019

DOI: 10.1109/JAS.2019.1911795

4 authors, including:

Hira Zahid
Institute of Business Management
3 PUBLICATIONS **26** CITATIONS
SEE PROFILE

Ahsan Morshed
Central Queensland University
63 PUBLICATIONS **757** CITATIONS
SEE PROFILE

Timos Sellis
Swinburne University of Technology
437 PUBLICATIONS **13,516** CITATIONS
SEE PROFILE

Big Data Analytics in Telecommunications: Literature Review and Architecture Recommendations

Hira Zahid, Tariq Mahmood, Ahsan Morshed, and Timos Sellis, *Fellow, IEEE*

***Abstract*— This paper focuses on facilitating state-of-the-art applications of big data analytics (BDA) architectures and infrastructures to telecommunications (telecom) industrial sector. Telecom companies are dealing with terabytes to petabytes of data on a daily basis. IoT applications in telecom are further contributing to this data deluge. Recent advances in BDA have exposed new opportunities to get actionable insights from telecom big data. These benefits and the fast-changing BDA technology landscape make it important to investigate existing BDA applications to telecom sector. For this, we initially determine published research on BDA applications to telecom through a systematic literature review through which we filter 38 articles and categorize them in frameworks, use cases, literature reviews, white papers and experimental validations. We also discuss the benefits and challenges mentioned in these articles. We find that experiments are all proof of concepts (POC) on a severely limited BDA technology stack (as compared to the available technology stack), i.e., we did not find any work focusing on full-fledged BDA implementation in an operational telecom environment. To facilitate these applications at research-level, we propose a state-of-the-art lambda architecture for BDA pipeline implementation (called LambdaTel) based completely on open source BDA technologies and the standard Python language, along with relevant guidelines. We discovered only one research paper which presented a relatively-limited lambda architecture using the proprietary AWS cloud infrastructure. We believe LambdaTel presents a clear roadmap for telecom industry practitioners to implement and enhance BDA applications in their enterprises.**

***Index Terms*—Big data analytics, BDA pipeline, BDA technology stack, lambda architecture, python, systematic literature review, telecommunications.**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

This work was supported in part by the Big Data Analytics Laboratory (BDA-LAB) at the Institute of Business Administration under the research grant approved by the Higher Education Commission of Pakistan (www.hec.gov.pk) and in part by the Darbi company (www.darbi.io). We acknowledge useful inputs regarding LambdaTel with Mr Uzair Ahmed (Project Lead for Darbi) and with Muhammad Zahid Raza (Assistant Vice President IT) from Meezan Bank (www.meezanbank.com). Recommended by Associate Editor Qinglong Han. *(Corresponding author: Timos Sellis.)*

Citation: H. Zahid, T. Mahmood, A. Morshed, and T. Sellis, "Big data analytics in telecommunications: literature review and architecture recommendations," *IEEE/CAA J. Autom. Sinica*. DOI: 10.1109/JAS.2019.1911795

H. Zahid and T. Mahmood are with the Faculty of Computer Science, Institute of Business Administration, Karachi 75270, Pakistan (e-mail: hzahid@iba.edu.pk; tmahmood@iba.edu.pk).

A. Morshed is with the School of Engineering and Technology, CQUniversity, Melbourne Victoria 3000, Australia (e-mail: a.morshed@cqu.edu.au).

T. Sellis is with the Director of Swinburne's Data Science Research Institute, Melbourne Victoria 3122, Australia (e-mail: tsellis@swin.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2019.1911795

II. Introduction

THE telecommunications (telecom) industry is facing an avalanche of data on a daily basis due to smart phone usage and boom of social media and IoT along with availability of next generation communication networks. Data occurs in both batch and real-time modes. Notable data examples are call detail records, user clickstream, mobile network usage, geographical user data, network performance, network monitoring, customer/subcriber profiles, hardware and VOIP data. In telecom, big data can be characterized by the standard 3V's: volume, variety and velocity [1]–[4]. The value of this data (generally the 4th V) is Big Data Analytics (BDA) [5], [6] which is the process of extracting valuable insights from big data streams that can help align business strategies to meet critical KPIs. BDA can harness big data for telecom by employing knowledge from diverse domains notably machine learning, statistics, pattern recognition, and business intelligence. BDA is mostly implemented in the context of NoSQL databases which tear away from the tight relational storage to more loose, unstructured and semi-structured data models [7], [8]. Well-known examples include MongoDB (which stores data as JSON documents) and Redis (which stores data as key-value pairs), along with Apache Hadoop and its ecosystem [2], [9], [10]. These databases are capable of addressing the ACID (Atomicity, Consistency, Integrity and Durability) requirements of relational databases [8]. In telecom, BDA can enhance customer relationship management through more efficient resource management, identification of root causes of service failure, more intelligent marketing campaigns, boosted-up sales, detection of high-velocity fraud activities in real time, and timely inception of new business partnerships [5], [11].

BDA is an expensive, resource-intensive and a complicated process which is plagued by many problems leading to significant project failures in different industries [5], [6], [12]–[19]. According to Gartner, upto 85% of BDA projects were failing in 2017 [15]. A McKinsey survey has determined the impact of investment in BDA initiatives by telecom companies on the actual benefits; of the 273 telecom companies who invested in BDA, only 5% companies are getting more than 10% benefit. Also, 75% to 80% companies ran into a loss due to BDA application. [11]. The more important problems in BDA initiatives are lack of data quality, poor data management, mistakes in selecting the analytical model, lack of an existing BDA infrastructure, lack of expenditure, making non-scalable BDA infrastructures, difficulty in creating a roadmap for BDA skills, and a rising

complexity in integrating heterogeneous big data [20]. The BDA landscape is also increasing at an exponential pace; termed as "firing on all cylinders" in industry [21]. Hence, the speed of innovation largely outpaces the speed of adoption. Most of these tools are open-source initiatives and require expert skills to understand and employ directly in an operational environment. This time for learning slows down adoption and demotivates a majority of businesses to invest in BDA [5], [6], [17], [22].

BDA complexity is another challenge. In a BDA process, a considerable number of activities/tasks are executed as a pipeline. Each of these activities can be implemented through an increasing diversity of both open-source and proprietary tools. There is a lack of skilled BDA pipeline developers due to the diversity of tasks to be performed, e.g., data upload, data transformation/clearning, statistical analysis, communication of the back-end activities with front-end GUIs, along with different types of analytics and visualization activities. Each tool has a learning curve, and the problem becomes severe when BDA developers need to integrate several tools together in the same pipeline. Moreover, the BDA pipeline runs perpetually until the analytical requirement is fulfilled, which requires the automation of core tasks like ETL and Machine Learning. The progress and now the domination of Python as a pipelining language has largely facilitated development of BDA pipelines in the last decade [20]. Some tools have also matured and have seeded the rise of BDA applications in telecom, for instance, MongoDB, Redis, Hbase, Spark, Flink, and Hadoop (described in Section 2). Due to these technologies, the BDA applications in telecom are increasing and likely to increase further [5], [6], [23], [24]. For instance, BDA can identify traffic delay sensitivity and accurate identification of small packet traffic, and brings much-reduced delay and processing complexity from data [25]–[28].

In this paper, our intent is to determine the extent to which the huge potential of BDA has been realized by the telecom sector in academic research, and to identify and address the concrete challenges. We focus on academic research because the fast-changing BDA landscape leaves much space for formal research activities and projects to determine the impact of BDA tools on telecom. In other words, we want to gauge the actual benefits of BDA that the research community has brought to the telecom sector. For this, we formulate three research questions:

1) RQ1**:** How much research literature is focused on BDA applications to telecom sector and what is the BDA technology stack in these articles?

2) RQ2**:** What are the benefits and challenges mentioned in these articles and how much benefit has been actually realized?

3) RQ3**:** How can the challenges be strongly addressed to facilitate BDA applications to telecom sector?

To investigate these questions, we conduct a Systematic Literature Review (SLR) according to standard guidelines [29], [30]. To the best of our knowledge, this is the first SLR application for telecom sector. We have modeled the SLR and this paper from a big data perspective and avoid any operational detail of telecom domains and technologies. For this latter knowledge, we refer the readers to [31], [32]. Later on, we address the BDA challenges in telecom by proposing and describing a comprehensive, state-of-the-art BDA architecture called LambdaTel for telecom practitioners.

II. Background

Gartner defines big data as "high volume, high velocity and high variety information assets that demand cost effective, innovative forms of information processing for enhanced insight and decision making" [33]. Here, four properties pertinent to our SLR are: 1) Volume is the large size of big data reaching generally from terabytes and petabytes, 2) Velocity is the speed of data generation and required processing of both batch and real-time data, 3) Variety is different types of data from heterogenous data sources, grouped as structured, unstructured and semi-structured data, and 4) Value indicates the hidden, previously unknown information or knowledge in data that is potentially useful for business decision making. The process of extracting value from big data sets is called big data analytics (BDA) [5], [6], [12], [13].

A. Apache Hadoop and MapReduce

A big challenge facing telecommunication companies today is the difficulty of employing a software and hardware infrastructure to handle big data. Apache's Hadoop is an open source framework used for distributed processing of big data across a cluster of commodity hardware [34]. Each Hadoop cluster is highly available and fault tolerant. Hadoop version 2.x is a three-layered model classified as storage layer, processing layer and management layer (Fig. 1) described as follows. HDFS is Hadoop's file system which provides fault-tolerance and high throughput over low-cost commodity hardware. Large files are split into smaller blocks in a redundant fashion to achieve fault tolerance and stored across multiple machines to provide easy access. HDFS also provides file permission and authentication rights. MapReduce is the batch processing framework which works over Hadoop based on divide and conquer rule. It comprises of a 'Map' and 'Reduce' function. Input key-value pairs process during map step which generates intermediate key-value pairs. Then, all the intermediate values related to the same key will combine so that reduce function is able to access them and compress the value set into a smaller set.

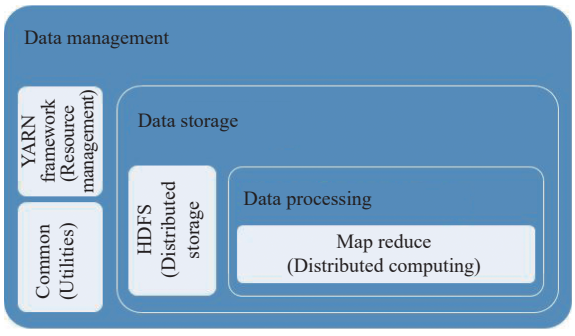

Fig. 1. Hadoop V2.x Architecture.

Overhead of steps like data scheduling, fault-tolerance, and inter-node communications are eliminated in MapReduce [18]. YARN is Hadoop's resource management framework which abstracts MapReduce from managing resources (as was the case in Hadoop version 1.x). Finally, we have the common utilities which are components needed to operate Hadoop sub-modules and projects. Shared libraries support other operations like error detection, compression codes implementation, and I/O utilities etc.

Hadoop's data management occurs through a master-slave architecture (Fig. 2). Master is called Name Node and slaves are Data Nodes. Name Node manages the file system name space, regulates clients access to files and executes file system operations such as renaming, closing, and opening files and directories. Data nodes perform read-write operations on HDFS, as per client request. They also perform operations such as block creation, deletion, and replication. Name Node runs the Job Tracker process to process MapReduce tasks that distributes and assigns work to Task Tracker daemon processes running on Data Nodes. The Hadoop ecosystem is a set of software APIs available as open source Apache projects which use Hadoop to provide different functionalities, e.g., database (Hbase), data warehouse (Hive), SQL Querying (Hive and Drill), stream processing (Spark, Storm, Flink), machine learning (Mahout,H2O), MapReduce programming (Pig) and cluster coordination (Zookeper) [34]–[36]. The ecosystem relevant to our SLR is:

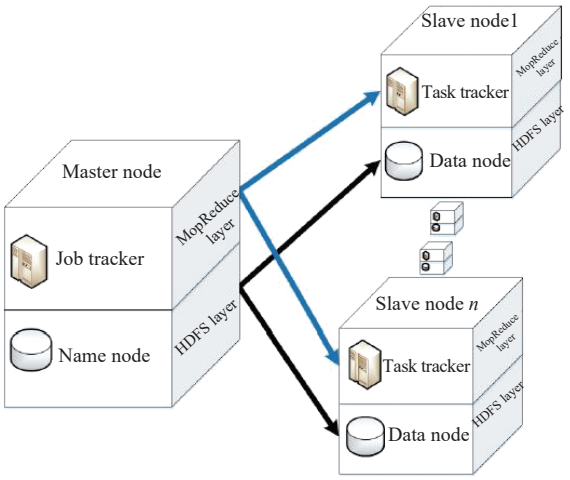

Fig. 2. Master slave architecture of hadoop.

1) Apache Hbase: This is Hadoop's database built on HDFS [37]. It is capable of providing real-time read and write operations on big data sets stored as a wide columnar store (discussed below). In Hbase, data is stored column wise, with each row having a sorted key indexed with timestamp. Columns can be grouped together to form column families which can be grouped in super column families. These column families are the basic units for access control. The time stamps are 64-bit integers to maintain different editions for a cell's content in Hbase. Clients flexibly determine the number of cell editions stored. These editions are sequenced in the descending order of time stamps, so the latest edition will always be read. Fig. 3 shows column families over *voice* and *sms* entities being grouped into a single super column family.

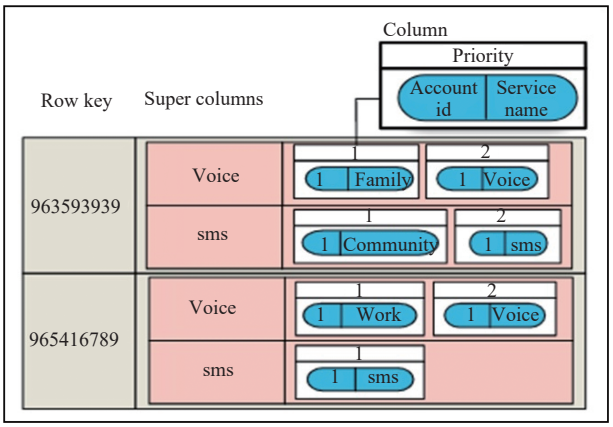

Fig. 3. A snapshot of a super column family for the choice of services.(Adapted from [35]).

2) Apache Hive: Hive provides the SQL interface and a relational model for big data processing over Hadoop [34]. Hive is also considered a data warehousing application infrastructure on top of Hadoop that provides summarization, query and analysis.

3) Apache Pig: Pig Latin is an ETL-level language which facilitates textual programming, parallel execution and optimization of complex tasks comprised of multiple interrelated data transformations, by encoding them as data flow sequences [34]. It also provides users the facility to encode their own user defined functions.

4) Apache Spark: Spark is an execution engine in which data streams are interpreted as a series of deterministic batch-processing jobs, making traditional MapReduce 100 times faster [38]. Spark is based on master/slave architecture. Master instance runs on user-defined driver program and can launch a set of workers in the cluster and read data from HDFS. Spark uses resilient distributed datasets (RDDs) that are partitioned across multiples machines to achieve fault-tolerance and slaves create partitions on RAM for RDDS as defined by the driver program. Spark Streaming is a Spark API for stream data processing.

5) Apache Kafka: Kafka is an ingestion API which processes real-time data streams and stores them into the queue [39]. Each queue has a topic component and it is a user defined category. The topic decides which event put in which queue. As events arrive randomly, they are sorted and arranged in a queue so that they consumed by the message broker component easily, which are servers consuming the queue. Servers can be based on Apache Spark, Apache Flink or Apache Storm.

6) Apache Flink: Flink is a data flow streaming engine and implements "true streaming" in that the whole job is deployed concurrently in the cluster [40]. Operators in the long run continuously consume input and produce output. These output tuples are immediately forwarded to further processing by next level operators which enables pipeline parallelism.

7) Apache Storm: Apache Storm is a distributed realtime

computation system which can reliably process data streams [41]. In Storm, spouts represent information sources and bolts represent data manipulations. Storm architecture is a processing pipeline modeled as directed acyclic graph with spouts and bolts as vertices and data streams as edges. Streams can be repartitioned as per need to enhance efficiency (over a million tuples processed per second per node). It is efficient, fault-tolerant and can integrate with database sources.

Spark Streaming, Flink, Storm (along with Kafka ingestion) have successful use cases in realtime analytics, online machine learning, continuous computation, distributed RPC, and data preprocessing (ETL). From SLR, we found that social network analysis (SNA), machine learning, stochastic modeling, data mining, cluster computing and cloud computing have been proposed/applied. As these domains are vast and generally well-known we do not present any background here.

B. Big Data Storage Technologies

NoSQL (Not Only SQL) is a new breed of databases that address the high scalability, complexity, and elastic schema requirements of big data [42]. They allow storage over four data models: wide columns, documents, key-value pairs, and graphs. Initially NoSQL compromised somewhat on ACID, formalized through CAP (Consistency, Availability and Partition Tolerance), i.e., given a tolerance to definite partitioning of nodes through system failures, we can provide availability at cost of consistency, or vice versa. In the case of latter, the system was in BASE i.e., basically available in a soft (temporarily inconsistent) state which will eventually become consistent with time. CAP and BASE are still used in NoSQL, e.g., in Amazon's DynamoDB which forms the storage backbone of Amazon Web Services. However, NoSQL now largely caters for ACID in powerful databases such as MongoDB and Redis [42].

We now define key-value, columnar, document and graph stores with examples of telecom data as shown in Fig 4.

1) Key-Value Data Stores: In key-value stores, data is input and accessed using key-value pairs. Keys are randomly generated and value can be any data type associated with in-built database objects. Notable examples include Redis and DynamoDb. Keys are stored in hash tables and logically grouped into a 'bucket'. Both bucket and key can be used to access the value as they are hashed for unique indexing. Key-value stores provide much faster query response times than relational or other NoSQL stores due to indexing and simplicity of storage. They also support adhoc querying for complicated unstructured analytical applications like web usage, social network feeds, and real-time response processes. Fig. 4(a) shows telecom call detail records (CDRs) contained in a key-value schema. In this scenario, a CDR instance (or a collection of instances) can be inserted as value while key is a set of CDR flags.

2) Document Data Stores: In document stores, data is stored in documents comprising a collection of key-value pair data. Every document has its unique identifier (key) and serializes data in semi-structured formats, particularly JSON which provides a widespread, flexible and information-rich structure for data modeling. A new field can be added at anytime without considering its schema. The document data model maintains data locality and is hence easy to distribute. It is useful to store complicated data formats related to web applications, blogs, mobile/smart phone usage, chat applications, and social media clients. MongoDB is a world-renowned NewSQL document store. A sample document model for a telecom's data is shown in Fig. 4(b). Here, JSON document with key 1001 is storing a set of key-value pairs (attributes) related to a CDR instance.

3) Wide Columnar Data Stores: In wide columnar stores, data is stored and processed in the form of tables which are schema-free; it is not necessary to provide a value for each cell and each row can have its own schema. For instance, data in HBase is stored in tables which are further stored in logical spaces called regions. Due to large size of Hbase table, it is partitioned into multiple regions and assigned to region servers across the cluster. Each region server contains multiple regions and each region contain multiple storage units. Fig. 4(c) shows an Hbase table of telecom CDR data segmented in three regions being managed by two region servers. The data model is a multi-dimensional sorted map as discussed above (refer to Fig. 3). Each row is indexed by key and columns can be combined together to form column families. Two or more column families form a super column family.

4) Graph Data Stores: Graph data stores store data consisting of objects (nodes) and edges linking nodes through relationships. Indexes are used to traverse the graph (either directed or undirected) which can be scaled out and distributed across nodes. Frequent analytical queries include identification of clusters, shortest path between two nodes and community detection. New edges can be added and existing edges removed so that social graph entities like friends, followers, endorsements, messages, and responses can be accommodated along with their relationships. Time-evolving graphs can be analyzed by monitoring changes to architecture over time. Fig 4(d) shows a sample graph store for telecom. Nodes are entities (company, call date, voice call, call from, call to) or actions (connect) with relevant attributes (connection time, company name). Edge labels are characterized by their roles, e.g., caller, callee, type of call etc.

III. SLR Research Methodology

We focus our SLR on the following domains of research: telecom analytics, big data applications to telecom, big data analytics (BDA) applications to telecom, amd NoSQL applications to telecom. We selected these domains to be generic enough to constitute all the possible (hundreds of) different BDA solutions available in the market today and so, this domain list is complete to the best of our knowledge. We use the more common and popular terms from these domains to develop our search queries (described below). We targeted digital sources commonly known for computer science related publications, i.e., Institute of Electrical & Electronics Engineers (IEEE), Association for Computing Machinery (ACM), Elsevier, Springer and Google Scholar. We relied on

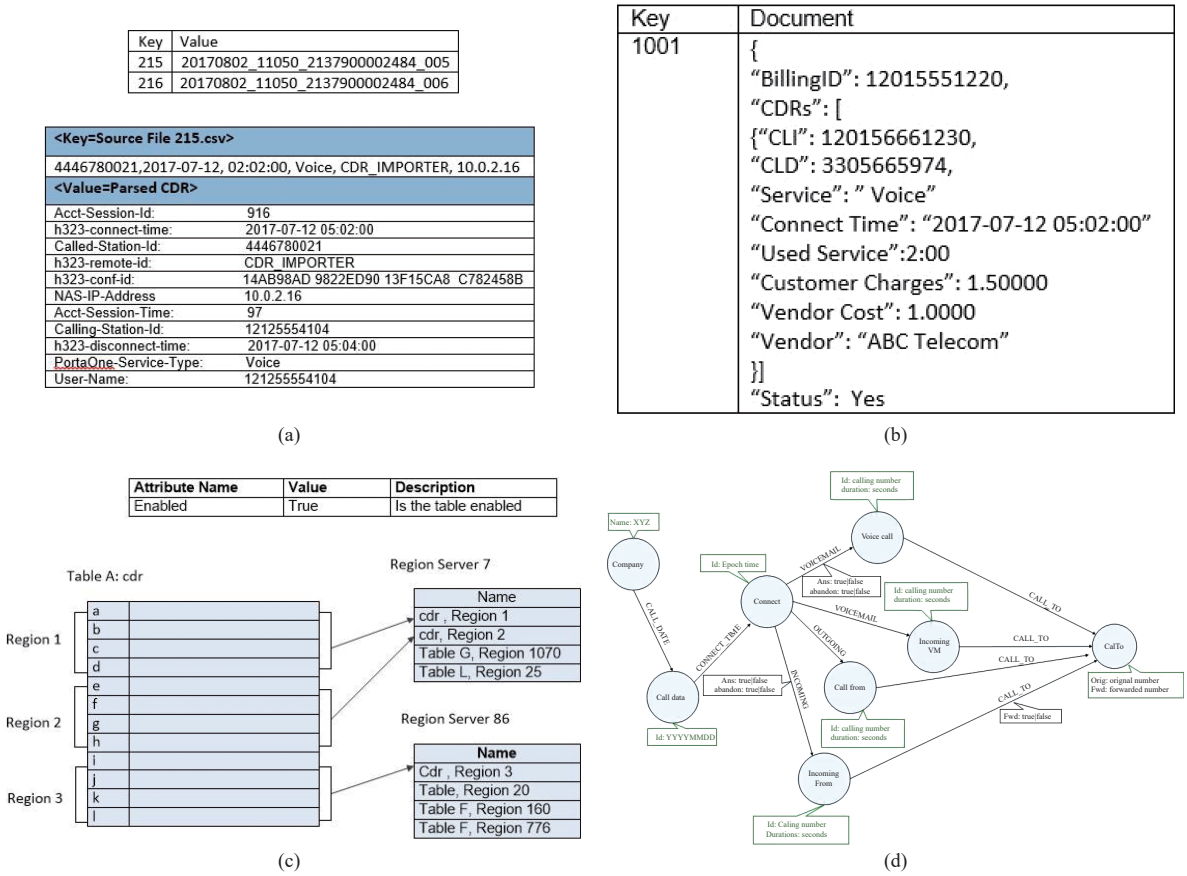

Fig. 4. NoSQL stores' examples for telecom: (a) key value store; (b) document data store; (c) wide columnar data store; (d) graph data store.

Google Scholar for coverage of the remaining digital sources as it is the most frequently referenced digital source in 2018 and continues to approach and make contracts with sources for indexing their research databases [3], [43][1]. To ensure state of the art results, we focused on research content from 2010 onwards but decided to include past content also if we deemed it critical. To manage the retrieved articles, we used the Mendeley tool as we found it to be more acclaimed and comprehensive for our needs. It is also broadly connected in scholarly groups and has large online community [44].

To formulate search queries, we selected eleven (11) keywords related to big data, i.e., big data, NoSQL, NewSQL, Hadoop, columnar store, key-value store, document store, graph database, big data tools, big data techniques, and big data analytics. We also selected three keywords related to telecom, i.e., telecommunications, VOIP, and mobile communication. Then, we combined each big data keyword with a telecom keyword to generate 33 queries, for instance big data AND telecommunications. We executed these queries on our selected digital sources.

We adopted a three-step methodology to extract the relevant research articles pertaining to our research questions. In the first step, we scanned the title of each paper (on each digital source) to determine its significance to our scope. We excluded completely irrelevant papers but we did not exclude papers which had fuzzy or unclear titles. This gave us 233 articles, which we added to our Mendeley database.We considered the most appropriate resulting literature papers from each source that are closely related to the topic. We used Mendeley to remove articles duplicated through Google Scholar, leaving us with 222 articles. In the second step, we read the abstracts of the papers filtered from first step. This provided more insights about the scope and helped in further filtering out irrelevant papers. This gave us 61 papers. In the third step, we read the first two sections of the articles filtered from second step, to perform a final filtration of papers which did not map our scope. This finally gave us 38 articles matching our scope, which we review in this paper. The breakdown of our 222 papers filtered in initial step with respect to digital sources is shown in Fig. 5. Google Scholar retrieved the maximum articles (60), followed closely by IEEE (57) and ACM (56), while Elsevier provided the least

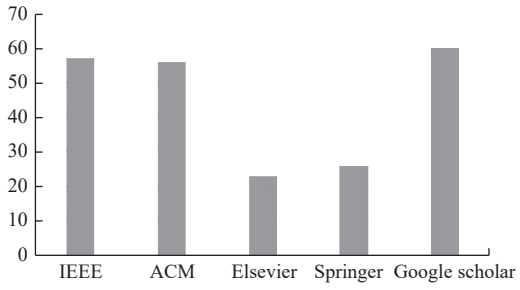

Fig. 5. Distribution of 222 articles in first step of filtration with respect to digital sources.

[1]The complete list of indexed resources is not made public by Google.

hits (23). Moreover, Fig. 6 shows the distribution of our selected 38 articles with respect to the complete set of 33 search strings, summed over all digital sources. The horizontal axis shows the big data keywords, and different colors represent different telecom keywords. There were no articles for search strings combining any telecom keyword with NewSQL, columnar store, key-value store, document store and graph database (hence, we do not display them). This shows that our selected literature does not use the more technical terms related to BDA. Also, a majority of the articles are retrieved through combination with telecom keyword followed by mobile communication and sparsely by VOIP.

In Table I, we show the distribution of selected articles with

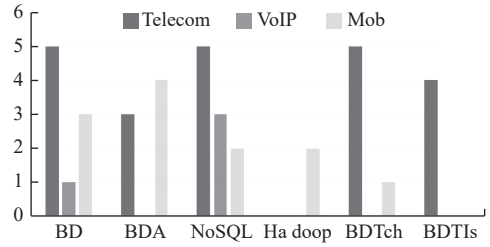

Fig. 6. Distribution of selected 38 articles with respect to 33 search strings over all digital sources. BD=big data, BDA=big data analytics, BDTch = big data techniques, BDTls = big data tools, Mob = mobile communication.

respect to type of research content. Majority of the content came from the top conferences (17), and journals (8) followed by a couple of magazines. master thesis, workshop or symposium papers and commentaries had limited contributions. We did not find any book matching our scope.

Finally, we show the distribution with respect to citation year in Fig 8. From 2011–2015, ≤ 5 articles were published each year. A spike was observed in 2016 with 18 publications, signaling the time when major interest and research was generated. This again fell in 2017.

A. Quality Assessment

We assessed the quality of the selected 38 studies against eight (8) selected quality criteria (QC), to determine whether our selected research articles suitably address our research objectives. We created 8 questions to assess the quality of our articles, with "Yes"/"No" responses having weights "1" and "0" respectively. After these responses, we evaluated the results through Cohen's score for inter-rater agreement. We debated on the discrepancies using the Delphi method [77] until consensus was reached on filtered articles. This activity was carried out by five researchers who are faculty members of three different universities in Pakistan (not anyone in the authors' universities), including three females and two males. Based on a selected threshold of 5, our QC process did not exclude any of the 38 selected articles (all obtained a score ≥5 (see Table II which records the mode response for each QC). Following are our quality checklist questions:

1) QC 1: Are research objectives clearly stated?

2) QC 2: Is methodology well-defined?

3) QC 3: Is big data tool utilization present?

4) QC 4: Any characteristics of big data out of 4V's catered in the study clearly described and its solution with respect to telecommunication?

5) QC 5: Is the process of BDA or any one step of BDA process clearly stated to resolve telecom's big data major issues?

6) QC 6: Does the study perform the comparison of proposed approach with existing baseline approaches?

7) QC 7: Were the performance metrics fully defined?

8) QC 8: Are results properly interpreted and discussed and does the conclusion reflect the research findings?

IV. RESULTS

We now discuss the 38 articles extracted from SLR. Article frequency distribution over telecom domain is shown in Fig. 9; mobile telecom is most frequent (23), followed by general telecom (12) and VoIP (3). In mobile, 20 articles (87%) discuss BDA applications to 2G, 3G, 4G and 5G networks, 2 on cellular networks and 1 on wireless networks. In Table IX, we show a classification of our 38 articles with respect to telecom domain, big data technology stack and the type of article. We identified 4 such types: literature reviews, frameworks, use cases, white papers, and validation (with experiments). BDA frameworks have been proposed in 23 (61%) articles, out of which 15 (40%) include validation with experiments, 5 (13%) present a use case without experiments, while 3 (8%) discuss only the framework without any use case or experiments. Also, in 7 (18%) articles, the authors do an experimental validation without proposing any framework. There are 4 (11%) literature reviews, a single white paper (commentary), and 3 (8%) use cases. Of the 22 (58%) articles on experimental validation, 16 (42%) employ Hadoop or its ecosystem APIs, and remaining use different NoSQL databases. Of the 16 (42%) papers that do not involve any experiments, only 5 (13%) papers employ Hadoop while the remaining focus on NoSQL. Of these 16, 2 papers also focus on stream analytics. The analytical applications targeted in our articles focus on Hadoop ecosystem, machine learning/data mining, deep learning, distributed databases, self organizing networks, computational visualization, graph analytics, monte carlo simulations, social network analysis and cloud computing-based data processing particularly for mobile telecom. Finally, Spark has also been used in 2 works for experimental validation and also once in a case study. We also estimated the number of articles focusing on BDA benefits and challenges of its implementation, along with future research directions and characteristics of telecom big data (shown in Fig 10). The authors discuss benefits most frequently, probably due to hype of big data and BDA, followed by challenges which are significant as outlined in Section I. Apparently, benefits/opportunities of BDA are clearly known and also its implementation challenges. However, experimental validations to realize these benefits and address the challenges remain very limited in research literature. We now discuss our 38 papers in detail. We have further classified them according topics (labels) and sub-topics of telecom domain which we identified during our SLR (shown in Fig 11).

TABLE I
CLASSIFICATION OF PAPERS WITH RESPECT TO ARTICLE TYPE AND THEIR RANKING

Papers	Publication type	Venue	Ranking
[45],[46],[25]	Magazine	IEEE Communications Magazine	H-Index : 126 Impact Factor : 10.356 CiteScore : 11.72
[47]	Conference	2012 International Conference on Information Security and Intelligent Control	H-Index : 6
[48]	Journal	Journal of Innovation in Digital Ecosystems	CiteScore: 3.25
[49]	Conference	2015 International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering, ICCNEEE	H-Index : 4
[27],[50]	Journal	IEEE Transactions on Cloud Computing	H-Index : 32 Impact Factor : 2.63 CiteScore : 5.92
[51],[25]	Journal	IEEE access	H-Index : 56, Impact Factor : 4.098 CiteScore: 5.38
[52],[53]	Symposium	16th International Symposium on Communications and Information Technologies (ISCIT)	H-Index : 4
[54]–[57]	Journal	IEEE Network	H-Index : 111 Impact:7.5 CiteScore: 8.98
[58]	Conference	12th International Conference on Network and Service Management	H-Index : 5
[59]	Conference	2015 IEEE International Conference on smart city	H-Index : 6
[60]	Conference	11th International Conference on Ubiquitous Information Management and Communication	H-Index : 3
[61]	Conference	11th International Conference on the Design of Reliable Communication Networks	H-Index : 8
[62]	Conference	41st Conference on Local Computer Networks	H-Index : 45
[63]	Conference	2013 High Capacity Optical Networks and Emerging/Enabling Technologies	H-Index : 6
[64]	Conference	15th Asia-Pacific Network Operations and Management Symposium (APNOMS)	H-Index : 9
[65]	Conference	2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery	H-Index : 7
[66]	Conference	7th International Conference Application of Information and Communication Technologies (AICT)	H-Index : 5
[67]	Workshop	IEEE 21st International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD)	H-Index : 7
[68]	Conference	2016 IEEE Global Communications Conference	H-Index : 15
[69],[70]	Conference	2014 IEEE International Conference on Mobile Services	H-Index : 6
[71]	Conference	IEEE Conference on Information and Communication Technologies, ICT 2013	H-Index : 15
[72]	Conference	IEEE 33rd International Conference on Data Engineering (ICDE)	H-Index : 16
[73]	Journal	APSIPA Transactions on Signal and Information Processing	H-Index : 10
[74]	Master Thesis	School of Electrical Engineering, Aalto University	
[18]	Journal	Mobile Networks and Applications, Springer	H-Index :79
[36]	Conference	2nd International Conference on Cloud Computing Technologies and Applications (CloudTech), IEEE	H-Index : 5
[75]	Journal	ACM Transactions on Multimedia Computing, Communications, and Applications	H-Index : 38 Impact Factor : 0.57 CiteScore: 5.68
[26]	Conference	39th International Convention on Information and Communication Technology, Electronics and Microelectronics, (MIPRO), IEEE	H-Index : 7
[76]	White paper	Huawei	

A. Frameworks

We now discuss big data frameworks over following five topics: mobile network operators, 5G, network optimization, CDR analytics and streaming data.

1) Mobile Network Operators: Authors target MNO's [2] in [36], [45], [51] to address challenges in BDA implementation. One framework proposed automation of a manual reporting system of a Moroccan MNO to deal with unstructured data and CDRs, server logs, billing, and social network data. Kafka is used for data ingestion and Flink to process HDFS data in both streaming and batch mode, with final visualizations being shown on dashboards. Another framework uses Hadoop, Spark and machine learning to achieve network KPIs and enhance revenue and the third proposes a lambda architecture that caters for both batch and stream data processing, and the self-organizing network (SON) approach [78] work by inferencing data from a relevant knowledge base. Case studies are presented which extend previous works to create data intervals for data reduction, identify sleeping telecom cells, and find correlations in telecom data, all employing MapReduce to estimate parameters before a self-organizing network (SON) application. Generally, authors propose key-value data stores for storing mobile data.

2) 5G: In [54], the authors discuss a novel SON-based approach for 5G networks. They first identify challenges hindering SONs to meet 5G requirements and then propose a BDA framework for SON geared towards 5G based on machine learning and data analytics which could be exploited

[2]Mobile Network Operators.

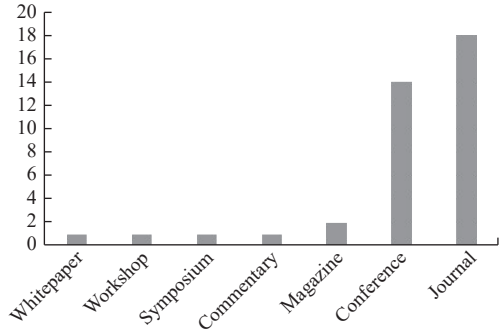

Fig. 7. Distribution of selected 38 papers with respect to article type.

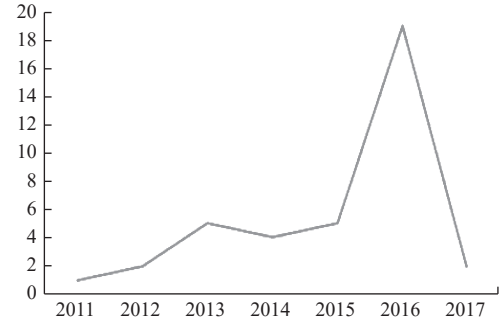

Fig. 8. Distribution of selected 38 papers with respect to year of citation.

to extract insights for creating end-to-end network intelligence. Through a case study, they show how this approach can diagnose a sleeping telecom network cell. Authors also conduct simulations to demonstrate superior performance over 3G/4G SON. Also, in [25], the authors propose a BDA architecture for mobile wireless 5G communication to achieve optimized resource allocation, mobile content distribution and network optimization. The authors categorize four types of big data, i.e., application, user, network and link channel data, and describe the protocol stack, procedure for signaling as well as operations at the physical layer. It is claimed that this architecture can provide a drilled-down view of operations and customers. There is no mention of any specific big data storage or processing technology in this paper although the benefits of machine learning are mentioned.

3) Network Optimization: Authors expound on the benefits of using machine learning and deep learning approaches [28], [54], [56] for network optimization. To meet the network requirements (for e.g. 5G services), generalized BDA framework is presented. NoSQL databases are proposed in ETL step. BDA through machine learning extracts insights for creating end-to-end network intelligence and generates network optimization solutions for different types of big data (private, redundant, distributed etc.) Also authors investigate the application of BDA to 5G in [28] and show that it can produce architectures which are flat, both green and soft, i.e., more agile and efficient.

4) CDR Analytics: In [49], the authors consider CDRs and prove it is a big data source and a candidate for BDA with respect to storage, processing and CDR analysis. Considerable research has been done to address the CDR analytics challenges along specification of BDA architecture, utilization of big data tools and techniques, and use case scenarios that presents better performance measures and cost efficient solutions in batch processing and real time. Authors stress the importance of using Apache Hive for querying with HDFS as a file system, while using Apache Cassandra for storing streaming data.

5) Streaming Data: In [26], the author presents an architecture for real-time predictive BDA for telecom domain. This model has four BDA capabilities: 1) real-time analytics, 2) detecting most probable cause of network failure, 3) modeling the users' telecom usage experience, and 4) real-time actuation of business goals. There are three relevant business areas: 1) optimizing management of telecom customer experience, 2) enhancing business efficiency through real-time (stream) processing, and 3) social network analysis (SNA) to analyze social and business relations among telco customers. The salient features of proposed architecture are: 1) gathering more reatime data from all nodes, 2) ETL connectors for extracting data from any source (NoSQL, relational etc.), 3) real-time analytics, opinion mining of users along with real-time SNA, 4) developing business processes and rules within the organization, and 5) measurement of relevant KPIs.

B. Literature Reviews, White Papers, Use Cases

We now explain literature reviews, white papers and use cases over two topics: BDA applications to telecom sector and BDA usecases.

1) BDA Applications to Telecom Sector: In order to reap BDA benefits for wireless telecom, authors in [46], [76] present actionable steps such as to identify big data sources in their domain, selection of new technology (particularly open-source), vendors, resource management including employees, BDA architecture, its execution with optimization. Further, identified BDA techniques according to network type particularly for wireless are stochastic modeling (e.g., Markov models, time series), machine learning (e.g., classification, regression, dimensionality reduction, reinforcement learning, deep learning) and data mining (pattern matching, clustering) algorithms as a solution to a specific problem such as predicting mobility of a user.

The authors in [57] direct attention towards format of input data generated by IoT, crowdsourcing and social media in structured, semi-structured or unstructured format, and either pseudo-random sampling, compressive sampling or distributed source coding being used to shred the streams for management. Such a streaming situation occurs when Internet-based mobile networking being performed in real-time on the cloud and there is a need to manage the off-loaded and uploaded streams in a real-time fashion. Therefore, a case study on BDA for streaming big mobile telecom data is presented via streaming architecture which utilizes distributed database technology to manage the streams; none of the well-known BDA tools like Spark, Storm and Kafka have been used.

2) BDA Usecases: Hadoop, NoSQL and machine learning

TABLE II
QUALITY ASSESSMENT OF 38 SELECTED ARTICLES WITH DELPHI METHOD OVER 8 QUESTIONS QC1-QC8; MODE RESPONSE IS SHOWN FOR EACH QC FOR FIVE DIFFERENT RATERS

Papers	QC1	QC2	QC3	QC4	QC5	QC6	QC7	QC8	Overall score
[46]	√	×	√	√	√	√	×	×	5
[47]	√	√	√	√	√	×	√	√	7
[36]	√	√	√	√	√	√	×	×	6
[48]	√	√	√	×	√	√	√	√	7
[49]	√	√	√	√	√	×	×	×	5
[50]	√	√	√	√	√	√	√	√	8
[51]	√	√	√	√	√	√	×	×	6
[52]	√	√	√	√	√	×	√	√	7
[54]	√	√	×	√	√	√	×	×	5
[58]	√	√	×	√	√	√	√	√	7
[35]	√	√	√	√	√	√	√	√	8
[59]	√	×	√	√	√	√	×	×	5
[45]	√	√	√	√	√	√	×	×	6
[55]	√	√	√	√	√	×	×	×	5
[60]	√	√	×	√	√	√	√	√	7
[61]	√	√	√	√	√	√	√	√	8
[27]	√	√	√	√	√	√	√	√	8
[62]	√	√	√	×	√	√	√	√	7
[63]	√	√	√	√	×	√	×	√	6
[28]	√	√	×	√	√	√	×	√	6
[64]	√	√	√	√	√	×	√	√	7
[65]	√	√	×	√	√	√	×	×	5
[75]	√	√	×	×	√	√	√	√	6
[66]	√	√	×	√	√	√	×	×	5
[56]	√	√	×	√	√	√	×	×	5
[53]	√	√	√	√	√	√	√	√	8
[68]	√	√	×	√	√	×	√	√	6
[67]	√	√	×	√	√	√	×	×	5
[26]	√	√	×	√	√	√	×	×	5
[76]	√	√	×	√	√	√	×	×	5
[57]	√	√	×	√	√	√	×	×	5
[24]	√	√	×	√	√	√	×	×	5
[69]	√	√	√	√	√	√	√	√	8
[70]	√	√	√	√	√	√	×	√	7
[71]	√	√	√	√	√	√	×	×	6
[72]	√	√	×	√	√	√	×	×	5
[73]	√	√	×	√	√	√	×	×	5
[25]	√	√	×	√	√	√	×	×	5

are the primary big data technologies being employed by the companies. In [24], [59], [65], [73], authors review BDA usecases identified through interviews, online research and through critical analysis from a representative sample of global telecom companies. The domains identified are marketing, sales, customer analysis, security, business development, innovation of new business models, products and services development, billing, intelligent transportation service, quality control, partner analysis, cost and contribution analysis, public sector, healthcare, media and entertainment, banking and insurance, quality of experience (QoE) or satisfaction, mobile retail shopping, mobile pricing analysis of products and SIM box detection, i.e., recognizing fraudsters who do not use their mobile sims as per policy.

The findings of review show that the remarkable benefits can be achieved through the earlier adoption of BDA with

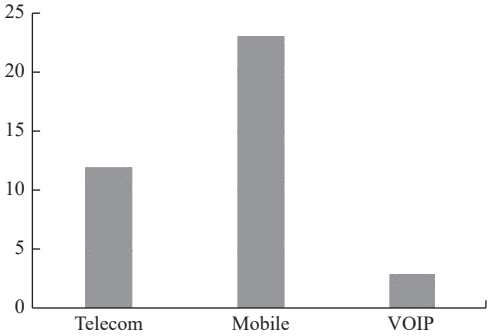

Fig. 9. Distribution of 38 papers with respect to telecom domain.

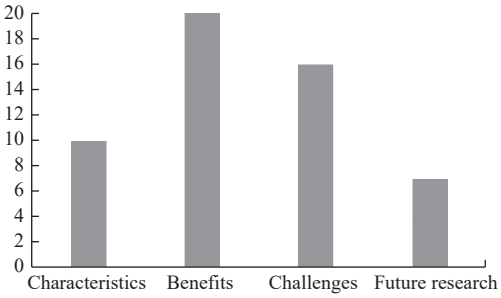

Fig. 10. Distribution of 38 articles with respect to telecom big data characteristics, benefits of applying BDA, challenges faced and specification of future research directions.

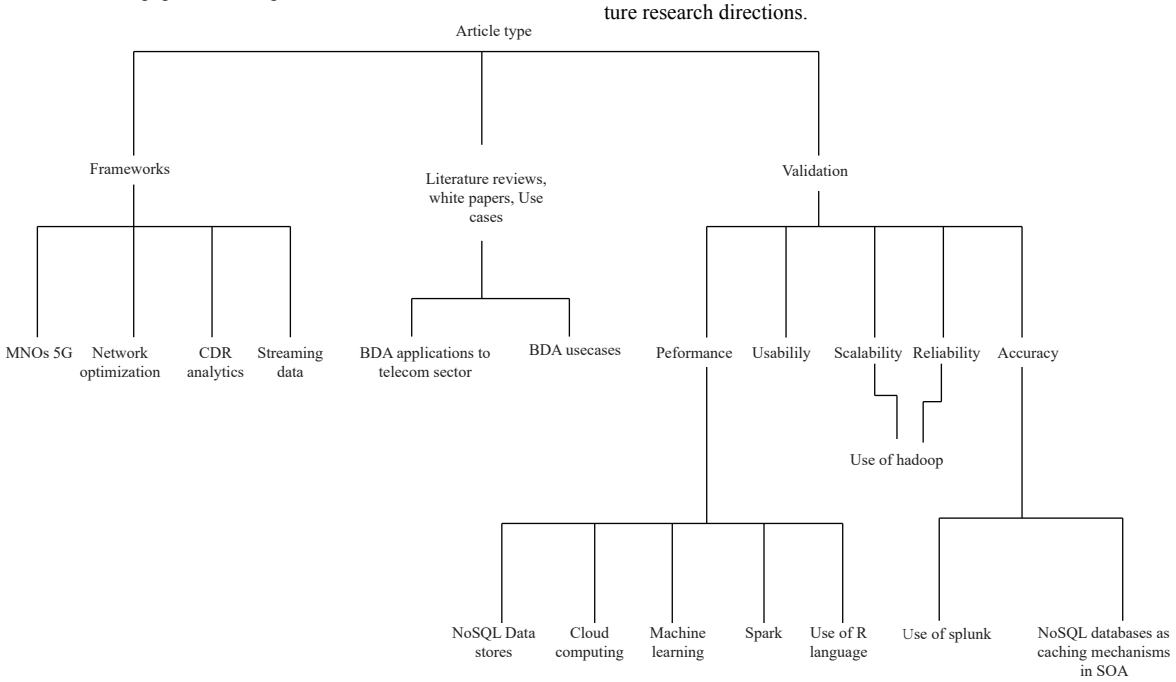

Fig. 11. Classification and sub-classifications of article types of our 38 SLR papers.

significant challenges. Also, real-time analytics and data management are core BDA requirements nowadays in telecom, which is expected to get serious due to IoT boom [79]. The primary motivation for a BDA initiative is to enhance customer satisfaction by providing a unique customer experience each time. The author recommends telecom companies to focus BDA efforts on satisfying customers, develop BDA architectures that are applicable to the complete organizational business process, not to wait for more data but to initiate BDA with currently available data to achieve streaming results, build BDA skillset based on business requirements through defining measurable outcomes.

C. Validation

We now discuss experimental validation articles based on the following architectural topics: performance, accuracy, scalability, reliability, security and usability. In some cases, we identified classifications of these aspects and discuss papers according to these classifications[3].

1) Performance: We found following six sub-classifications of articles related to performance of BDA applications to telecom: NoSQL Data Stores, Cloud Computing, Machine Learning, Use of Hadoop with Spark, and Use of R Language. A. NoSQL Data Stores: The objective is to highlight the role of NoSQL in improving performance of a system and provide some guidelines for selecting the data store that is most efficient in terms of computational complexity.

- In [35], the authors present the method of migrating from a relational store to a NoSQL store and prove superior performance of Apache Cassandra over PostgreSQL relational store in a telecom scenario. The data model consists of customer data, customer account data, bucket (balance of customer), bucket type, subscribed service, service type, and tariff plan. The specific query is determining the list of services subscribed by a customer (ordered by priority) and account information on receiving an in-coming call. The authors create a super column family for each user (row key is the caller number) where super columns contain the list of services (and details) being subscribed by each user. A regular day workload and a Christmas workload is created. In the former, Cassandra is able to handle 0.24 million calls over 600 seconds and in the latter 0.34 million calls, as compared to

[3]Where necessary, we have itemized the paper discussion of validation-related papers to enhance readability.

0.16 million and 0.18 million respectively for PostgreSQL. Similarly, In [69], the authors utilize the NoSQL data storage technology

- In [66], the authors employ a NoSQL document store in an enterprise BDA telecommunication application. This system collects, merges and analyzes data from several sub-systems, with major entities being staff, shift, permit, work activity, service, turnstile transaction, and department. Traditional relational storage is converted to document model; for instance, shift, permit, work activity, and turnstile transaction tables are combined into a document structure called Activity Package through a synchronization service. The use of NoSQL facilitates much better automatic load balancing in case of heavy querying load. The author's do not mention the specific document store they have implemented and detailed experimentation is not shown.
- In [52], the authors implement a BDA cellular network planning system which is based on a common telecom use case of combining OLAP and OLTP technologies for a real-world concurrency scenario. The authors focus on HP's Vertica, which is a MPP columnar warehouse providing support for both OLTP and OLAP type queries. In a cluster with standard configuration, inserting 10 K records in a table consumes an average of 200 ms, updating 10 K records consumes 3500 ms on average and deleting 10 K records consumes 2000 ms on average. A theoretical comparison with SAP HANA is also presented, which caters for the limitation in older versions of Hadoop to provide transaction processing, i.e., low-latency, SQL-oriented Hadoop solution was inefficient. The authors then propose a BDA architecture which can perform unified data processing to process analytical and highly concurrent transactional tasks efficiently within one system which run above various applications with loaded data and answering different end users requests.

B. Cloud Computing: Latency problem effects the quality of service and thus lowers the performance as well. This section highlight the importance of cloud computing as this solution is found abundantly.

- In [67], [68], the authors tackle the problem of delayed transfer of mobile data from smartphones and related devices to mobile cloud computing data centers over wireless channels. To overcome this latency, an efficient BDA-based data transfer approach is proposed that employs overlapping features of heterogeneous wireless networks (HetNets) by splitting data into chunks transferred simultaneously. In this way, data is first transferred to small clouds called cloudlets (associated with small telecom cells) which then transmit to public cloud. Through monte carlo simulations, the efficiency of the proposed approach is demonstrated. The authors do not mention any use of standard NoSQL databases for big data storage management.
- In a similar vein, the authors in [58] justify the complexity of maintaining QoS guarantees in a cloud computing (software defined network) scenario, due to un optimized network design, load balancing, access control and prioritization of traffic. Here, we report the findings from a couple of papers to exhibit the role of ML algorithm in improving the performance of the telecom big data system.

C. Machine Learning: A number of ML algorithms are available that can be leveraged. These ML algorithms range from supervised learning (e.g., Logistic Regression, Support Vector Machine, Naive Bayes, Random Forest, and Decision Trees) to unsupervised learning algorithms 14 (e.g., K-means and Neural Networks). When selecting a ML algorithm, several factors need to be considered. These factors include the time complexity, incremental update capability, offline/online mode, and generalization capacity of the algorithm, and most importantly the impact of the algorithm on the detection rate (accuracy) of a system. Due to the diverse role of the algorithm, it is quite challenging to pick the most appropriate and efficient algorithm.

- The authors in [58] propose and implement a BDA approach for QoS optimization, based on quantifying the correlation between telecom KPIs (e.g., packet size, number of packets transmitted, length of queues) and QoS metrics (e.g., end-to-end delay, throughput, packet loss etc). Initially, correlation coefficients are computed for each KPI and then machine learning using a combination of decision trees and linear regression is used to predict the QoS metrics. One core finding shows that CPU utilization is correlated with number of packets transmitted and packet loss with average communication delay. There are some unexpected results, e.g., bi-directional delay increases end-to-end delay by a factor of 4. Similarly to above, the authors do not mention any use of standard NoSQL databases for big data storage management.
- In [53], the authors combined the output of random forest, logistic regression and support vector machines algorithms to build a predictive model for their customers In [70], machine learning algorithms are also involved specifically neural networks, decision trees and logistic regression to identify influential telecom subscribers

D. Use of Hadoop With Spark: In [62], the authors propose an efficient statistical BDA solution to identify encrypted, non-encrypted, or tunneled VoIP media (voice) flows. A system is also proposed to efficiently process high-speed real-time network traffic. The BDA solution uses association rule mining to extract rules regarding VoIP parameters/KPIs, for instance, packet size, current flow, and packet transmission time. The authors treat this as a classification problem, with classes depicting different outputs being generated by the rules' combination. A single-node Hadoop setup is used for batch processing along with Spark for processing of streaming VoIP data. The authors demonstrate that the system efficiency in terms of number of packets transmitted outperforms performance of existing VoIP analytics systems.

E. Use of R Language: In [60], the authors tackle the problem of understanding and analyzing VoIP applications through standard parameters such as bandwidth, packet loss rate, delay, jitter, codec type and CPU power of the end devices. Performing ETL on VoIP big data is a challenge due to the diversity of Internet data being generated by a single device. As part of a performance measurement research, a robust ETL approach is proposed which is based on execution of certain scripts during extraction, conversion to XLS and txt formats of extracted data in transformation, and loading of

TABLE III
DISTRIBUTION OF 38 ARTICLES WITH RESPECT TO TELECOM DOMAIN, BIG DATA TECHNOLOGY STACK AND ARTICLE TYPE

Paper reference	Telecom domain	Big data technology stack	Article type
[46]	Mobile telecommunication	Statistical Modeling, Data Mining, Machine Learning	Literature Review
[47]	VOIP	Stream Processing, Hadoop, Cloud Computing, Machine Learning	Framework, Validation
[36]	Mobile telecommunication	Kafka, Hadoop, Flink	Framework, Use Case
[48]	Mobile telecommunication	Graph Analytics, Visualization, Social Network Analysis	Framework, Validation
[49]	telecommunication	Hive, HBase	Literature Review, Use Case
[50]	Mobile telecommunication	Hadoop, MapReduce	Framework, Validation
[51]	Mobile telecommunication	Hadoop, Spark, Machine Learning	Framework, Use Case
[52]	Mobile telecommunication	Hadoop, Columnar Database& Validation	
[54]	Mobile telecommunication(5G)	Machine Learning	Framework, Use Case
[58]	Mobile telecommunication	Cloud Computing, Machine Learning	Validation
[35]	Mobile telecommunication	Cassandra, PostgreSQL, NoSQL Databases	Validation
[59]	telecommunication	Cloud Computing, Machine Learning, NoSQL Databases	Literature Review
[45]	Mobile telecommunication	MapReduce, NoSQL DB	Framework, Use Case
[55]	Mobile telecommunication(2G,3G)	Hadoop, MapReduce, HBase	Validation
[60]	VOIP(QoS)	ETL, Statistical Analysis	Validation
[61]	telecommunication	Splunk, Statistical Analysis	Validation
[27]	Mobile telecommunication	Hadoop, Pig	Framework, Validation
[62]	VOIP	Hadoop, Spark, Association Rule Mining	Framework, Validation
[63]	telecommunication	Hadoop, HBase, Pig, Hive	Framework, Validation
[28]	Mobile telecommunication	Machine Learning, Optimization, NoSQL Databases	Framework
[64]	telecommunication	Hive, Hadoop, MapReduce	Framework, Validation
[65]	telecommunication	NoSQL Databases, Stream Processing, IOT	Literature Review
[75]	Mobile telecommunication	Deep Learning, ETL, Stream Processing	Framework, Validation
[66]	telecommunication	Document Store, NoSQL databases	Validation
[56]	Mobile telecommunication(5G)	Machine Learning, NoSQL Databases, Deep Learning	Framework, Use Case
[53]	Mobile telecommunication(4G)	Spark, Machine Learning	Framework, Validation
[68]	Mobile telecommunication	Big Data Storage, Cloud Computing	Framework, Validation
[67]	Mobile telecommunication(5G)	Cloud Computing, Monte Carlo Simulation	Framework, Validation
[26]	telecommunication	Machine Learning, Social Network Analysis, NoSQL Databases	Framework
[76]	telecommunication	Machine Learning, NoSQL Databases	White Paper
[57]	Mobile telecommunication	Cloud Computing, Stream Processing, Distributed Databases	Use Case
[24]	Mobile telecommunication	Machine Learning, Hadoop	Use Case
[69]	Mobile telecommunication	NoSQL Databases	Framework, Validation
[70]	Mobiletelecommunication	Hadoop, MapReduce, Social Network Analysis	Framework, Validation
[71]	Mobile telecommunication	Hadoop, MapReduce	Framework, Validation
[72]	Mobile telecommunication	Hadoop, Spark, Exploration, IoT	Framework, Validation
[73]	Mobile telecommunication	Machine Learning, Hadoop, NoSQL Databases	Use Case
[25]	Mobile telecommunication, wireless(5G)	Machine Learning	Framework

data into R for analytics in loading phase. More important results show that GoogleTalk, Skype and Express Talk are sensitive to packet loss rate and jitter rather than to delay. Also, bandwidth and de-jitter buffer and gateway CPU and memory are important in order to produce a good quality VoIP service.

2) Accuracy: The following articles dealing with accuracy of BDA application results have used Spark:

● In [53], the authors attempt to increase the adoption of 4G technology by predicting the relevant customers currently using 2G/3G from big data streams of a Chinese telecommunication firm. The exact work is to enhance 4G transfer rates through prediction of peer influence in CDR graphs of customers with data like service subscription, service usage, demographic information, and calling and messaging history. The graph contains 0.15 million nodes and

TABLE IV
CLASSIFICATION AND SUB-CLASSIFICATIONS (IN ITALICS) OF ARTICLE TYPES OF OUR 38 SLR PAPERS

Article Type	Topics and Sub-Topics Classification
Frameworks	Mobile network operators (MNOs), 5G, Network optimization, CDR analytics, Streaming data
Literature reviews, White papers, Use cases	BDA applications to telecom sector BDA usecases
Validation	Performance (NoSQL Data Stores, Cloud Computing, Machine Learning, Use of Hadoop with Spark, Use of R Language) Scalability (Use of Hadoop, Social Network Analysis) Accuracy, Security, Usability Reliability (Use of Hadoop, Use of Splunk, NoSQL databases as Caching Mechanisms in SOA)

62000 edges. In a field study, the authors first perform feature selection and then build a predictive model that combines outputs of ML algorithms using an Apache Spark cluster setup. The authors demonstrate excellent predictive accuracy by comparing real-vs-predicted values.

● In [75], the authors propose a deep learning framework for video analytics. Specifically, convolution neural networks are used to classify each frame of the video in real-time to determine its importance, in which case it will be retained. This activity controls the size of the input video hence leading to load reduction by discarding frames unnecessary for analytics. The action decision to discard is taken by determining correlation between consecutive frames in a streaming fashion. The authors show that their system has better accuracy as compared to other deep learning models for stream processing (temporal stream convnet and two stream model).

● The authors in [27] also use neural networks to predict the anomalies in their mobile network when implementing a BDA framework for analyzing customer-centric mobile wireless big data using Hadoop for processing and Apache Pig for ETL. Initially, customer CDR data are used to detect anomalous behavior using k-means and hierarchical clustering, for instance, unusual traffic at a given location and time. These outputs are compared with ground truths for verification, and held identify regions in the network for specific actions such as resource allocation.

3) Scalability: We now discuss papers that have implemented scalable BDA applications, over two classifications: Use of Hadoop and Social Network Analysis.

1. Use of Hadoop:

● In[55], [63], MapReduce is used for processing and analysis of data sets at different layers of a telecom platform.For this purpose, employ Apache pig or hive for programming MapReduce. Hbase is used as persistent NoSQL data store The authors in [64] implement a NoSQL infrastructure for criminal investigation through telecom CDRs. The infrastructure uses Apache Hive with Apache Hadoop/HDFS at backend. The authors vary several parameters including block and partition size in HDFS. The ideal parameters demonstrate the superiority of using Hive, with considerable improvements to the efficiency of query execution and scalability along with reducing cost of cloud usage.

● In [55], the authors implement a scalable network traffic monitoring for a large scale telecommunications company based in China. MapReduce is used to execute traffic analysis, application analysis, and user behavior analysis on telecom big data streams. Some specific KPIs which are estimated are traffic statistics in terms of bytes and packets, traffic classification at the application level, web service provider analytics from the perspective of mobile Internet, and clustering the user behavior data to extract useful homogeneous groups. The authors persist data in Hbase and employ Apache Pig for programming MapReduce. It is shown that Hadoop can efficiently processes 4.2 terabytes of traffic daily from 123 Gb/s links with high performance and reduced cost.

● In [63], the authors implement a big data platform with a domain specific language (DSL) for telecom sector, which uses MapReduce for processing and HBase as persistent NoSQL store, along with a visualization layer for displaying results. A specific type of file descriptor is also created for organizing communication in the above process. DSL is a high level Language which abstracts the user from directly writing MapReduce code or performing ETL through Apache Pig or Hive. It transforms the datasets in such a way that various information about a particular grid such as multiple calls and SMS activities on a particular date and time could be done with few lines of code. The authors demonstrate the scalability (rate of change throughput is more than the increase in number of nodes), reduced average execution times and linearly increasing write performance of HBase with increase in data.

● In [69], the authors propose a three-tiered BDA architecture for the agriculture domain to solve the problems of farmers not being able to access crop yield information online due to unstable nature of wireless communication. A middleware allows farmers to access the data with WiFi or 3G/4G connection. Another tier stores farmers' requests in a NoSQL database which acts as a cache to avoid DoS messages. Finally, in an offline mode, farmers can communicate through Bluetooth to synchronize information. Strangely enough, the authors do not specify the exact NoSQL technology which could be useful in their application.

2. Social Network Analysis: In [70], the authors propose a scalable BDA solution for identifying influential telecom subscribers through several social network analysis metrics combined through machine learning, specifically neural networks, decision trees and logistic regression. To support scalability, a prototype system is implemented on Hadoop and algorithms are executed through MapReduce. Results show that the system can scale to millions of users through actual data from a telecom company with 2.4 million subscribers and experimental data for networks with 100 million subscribers.

4) Reliability: We now discuss articles focusing on

reliability aspect of validating BDA applications to telecom sector, on following classifications: Use of Hadoop, Use of Splunk and NoSQL databases as Caching Mechanism in Service Oriented Architectures (SOAs).

1. Use of Hadoop:

● In [27], the authors implement a BDA framework for analyzing customer-centric mobile wireless big data using Hadoop for processing and Apache Pig for ETL. Initially, customer CDR data are used to detect anomalous behavior using k-means and hierarchical clustering, for instance, unusual traffic at a given location and time. These outputs are compared with ground truths for verification, and held identify regions in the network for specific actions such as resource allocation. The authors also use neural networks to predict these anomalies in advance.

● In [50], the authors develop a MapReduce framework over a previously implemented distributed system for mobile cloud computing (MDFS). In this way, due to the parallel processing nature of Hadoop, no single mobile device can become a bottleneck for the mobile cloud. Also, resource allocation and task management are handled efficiently by Hadoop in a fault-tolerant manner, and Hadoop has proven its worth in many applications in varied domains. The authors do not employ HDFS as it does not cater for the energy limitations of mobile devices and it requires heavy I/O processing to maintain fault-tolerance, as compared to the lightweight nature of mobile data processes. In experiments, authors achieve optimal parametric settings for HDFS block size, Hadoop cluster size, and node failure rate, along with the effect of changing input sizes on the throughput. The superior performance of MDFS over traditional HDFS is also validated.

● In [47], a real-time video/voice over IP (VVoIP) application is implemented using Hadoop cloud computing system to resolve head-of-line blocking, handover interruption, and non-real-time transmission problems in VoIP communication. The authors employ TCP-based Real Time Messaging Protocol instead of the traditional Stream Control Transmission Protocol, and also employ a neural network to tune parameters to optimize handover and analyzing network traffic at any time. VoIP data is moved from user devices to Hadoop cloud with access control to implement rapid facial/fingerprint identifications and reduce the amount of processing data. The authors demonstrate that their system has a faster response time and lesser misclassification rate in access control.

● In [71], the authors employ a Hadoop application to solve the problem of detecting signal discontinuity regions for 3G connectivity through a combination of standard KPIs. Experiments are performed through simulations over a cluster of 3 nodes with commodity configuration, where the cluster received around 1 million messages from different access regions daily and MapReduce is used to execute the queries using KPI values. The results prove that the Hadoop system has superior accuracy and efficiency as compared to traditional approaches.

2. Use of Splunk: In [61], the authors evaluate the health of network of a university wireless network, in order to analyze patterns of outages and failures for reliability improvement. For this, the proprietary Splunk tool is used for analyzing the huge volume big data generated by node outages, link failures and topology information. Simple Network Management Protocol and syslog data are used to investigate reliability along with standard documents of causes and recommended actions, and input of network operators' on special events and actions. The overall result is that wireless networks are less reliable as compared to wired network.

3. NoSQL Databases as Caching Mechanisms in SOA: In [69], the authors propose a three-tiered BDA architecture for the agriculture domain to solve the problems of farmers not being able to access crop yield information online due to unstable nature of wireless communication. A middleware allows farmers to access the data with WiFi or 3G/4G connection. Another tier stores farmers' requests in a NoSQL database which acts as a cache to avoid DoS messages. Finally, in an offline mode, farmers can communicate through Bluetooth to synchronize information. Strangely enough, the authors do not specify the exact NoSQL technology which could be useful in their application.

5) Security: In this section, we discuss the single paper related to security aspects of BDA applications to telecom. Particularly, in [48], a graph analytics platform is implemented which provides the network operator with an extended toolkit to obtain an overview of the whole network and allowing the operator to gradually focus on the desired information and acquiring useful insights. It facilitates data mining by providing modules for extraction of behavioral patterns, detection of attacks against network, behavioral similarity and detection of anomalies and attacks against networks. For validation, the authors perform root cause analysis of denial of service (DoS) attacks on a mobile network operators, along with early detection of an emerging (hot) event in Twitter streams. Most of the previous solutions have not managed graph based data mining at this level of adequate depth and GAP is a much better visualization platform for big data. The authors do not mention about any NoSQL graph database in their work.

6) Usability: Finally, this section discusses the single paper related to usability aspect of BDA applications to telecom. In [72], the authors implement a BDA application called SPATE, which is an innovative big data exploration framework for telecom data using Hadoop and Spark to achieve comparable response times with orders of magnitude lesser storage space for spatio-temporal queries. The authors use lossless compression to ingest streaming telecom data and use a concept of *decaying* to distinguish between 'old' and 'new' data. Experiments have been conducted using network data traces and a variety of telecom analytics tasks. SPATE's future includes advanced smart city application scenarios namely an automated car traffic mapping system and an emergency recovery system which is critical after natural disasters.

V. Challenges and Benefits

We now mention the challenges and benefits of telecom BDA applications described in our 38 papers. We also enlist

concrete gaps identified from challenges to realize the benefits properly.

A. Benefits of BDA in Telecom

We have categorized the benefits of BDA applications to telecom as follows:

1) BDA As a Smart Solution: BDA brings special infrastructures and tools that provide considerable advantages for telecom industry in terms of infrastructure, programming models, high performance schema free databases and process analysis, all of which offer new and innovative opportunities to telcos, for instance, lesser power consumption and optimized resource management and network performance [35], [36], [46], [59], [63].

2) Cost Reduction and Revenue Generation: BDA can assist in reducing cost of different operations of communication networks. BDA stream processing technologies help to process complex events with real time requirements which reduce risks, cost, and improve decision making and revenues [76], [80].

3) Improving Customer Care Services: Business case for big data is substantially focused on addressing customer-centric objectives. Companies can use BDA to enhance the customer care services as a result of being able to truly understand customer needs and anticipate future behaviors. Operators can make automated procedure to meet customer requirements such as faster calling [65], [80].

4) Improving Diverse Usecases: BDA can be applied for Sim-box detection and quality of experience (QoE), the two most compelling use cases in the telecom domain. Important telecom applications that can benefit from big data include QoE analysis, churn prediction, target marketing, and fraud detection [73].

5) BDA for Next Generation Mobile Communication: BDA can be used to analyze 4G LTE and 5G network from multiple dimensions and provide optimized solutions. Some examples include end-to-end visibility of the wireless network, self-coordination among network functions and entities, building faster and proactive network, smart and proactive caching and energy efficient network operation [27], [74]. The future 5G network design will be greener and softer and will better meet the user requirements of mobile communication [28].

6) Future BDA: Future BDA will encapsulate many different data models and algorithms as well as data integration components, for instance, advanced probes and adapters for retrieving data from all network nodes in real-time, advanced adapters for pulling relevant customer data from traditional big data and data warehouse systems, real-time analytics for customer activities, quality indexes for sentiment analysis, and opinion mining in real-time [26].

B. Challenges of BDA Application in Telecom

In the research papers which address challenges [36], [46], [48], [51], [54], [60], [65], [72], [73], [76], we have identified three categories:

1) Lack of a formal Architecture for BDA Pipeline Implementation: BDA initiatives are posing serious challenges in integration of different data sources, complicated and time-consuming ETL activities and ensuring quality of the BDA outputs by uncovering correlations and actionable insights using distributed machine learning. In fact, BDA offers two types of architectures (pipelines) to streamline the process and solve these problems, i.e., lambda and kappa [17]. Briefly, lambda allows processing of streaming and batch data in parallel, while kappa considers everything as streaming data; first data has to be processed as stream and then in batch-mode (if required). We have not found any lambda or kappa implementations or proposals in any of our reviewed papers. Perhaps the best match is the work done in [81] who propose an AWS-based (Amazon's cloud service) lambda architecture for IoT data processing. However, this paper does not propose any much innovation (due to the already available AWS infrastructure) and lacks several important components (shown in next section) which are critical to deal with the heterogeneous nature of telecom's BDA requirements. Surprisingly, state-of-the-art NoSQL solutions (e.g., MongoDB and CouchDB document stores and Redis and Riak key-value stores) which have had a global impact are not demonstrated in published research [82], [83].

2) Lack of BDA Expertise and Knowledge: The BDA technology stack is increasing exponentially and it is difficult to find the relevant human resource who are technology experts for operational BDA architecture implementations. Definitely, guidance is needed in selecting the best architecture and the best combination of BDA technologies in this architecture. We also need to select the standard Python development language which has a massive online community in BDA pipeline development.

3) Lack of Security Policies: The problems of ensuring data security, privacy, confidentiality and protection are quite significant in a BDA application for telecom companies. Many companies will avoid a BDA initiative if the security policy are not specified or enforced. Cloud computing (through Amazon AWS and Microsoft Azure) have resolved doubts over privacy invasion of companies of diverse types [84], [85]. However, it is still not an acceptable solution for many companies in general because the 'data is not available within the company's data center' and its increased cost as compared to an in-house BDA architecture based on open-source technologies where the data security policies as specified by the IT department can be applied on the BDA pipeline.

In our opinion, the only solution to these challenges is to propose a formal BDA architecture for the telecom sector, to be implemented in-house with standard open-source technologies and programming languages in order to reduce cost and increase privacy. We firmly believe this is the need of the moment and it will provide a clear roadmap for telecom's BDA practitioners.

VI. LAMBDATEL: PROPOSED LAMBDA ARCHITECTURE (BDA PIPELINE) FOR TELECOM SECTOR

Our proposed BDA architecture for the telecom sector called LambdaTel is shown in Fig 12. The engineering behind LambdaTel is lambda in nature, allowing both batch and streaming data processing to execute in parallel with each other[4]. It consists of seven layers which we describe below.

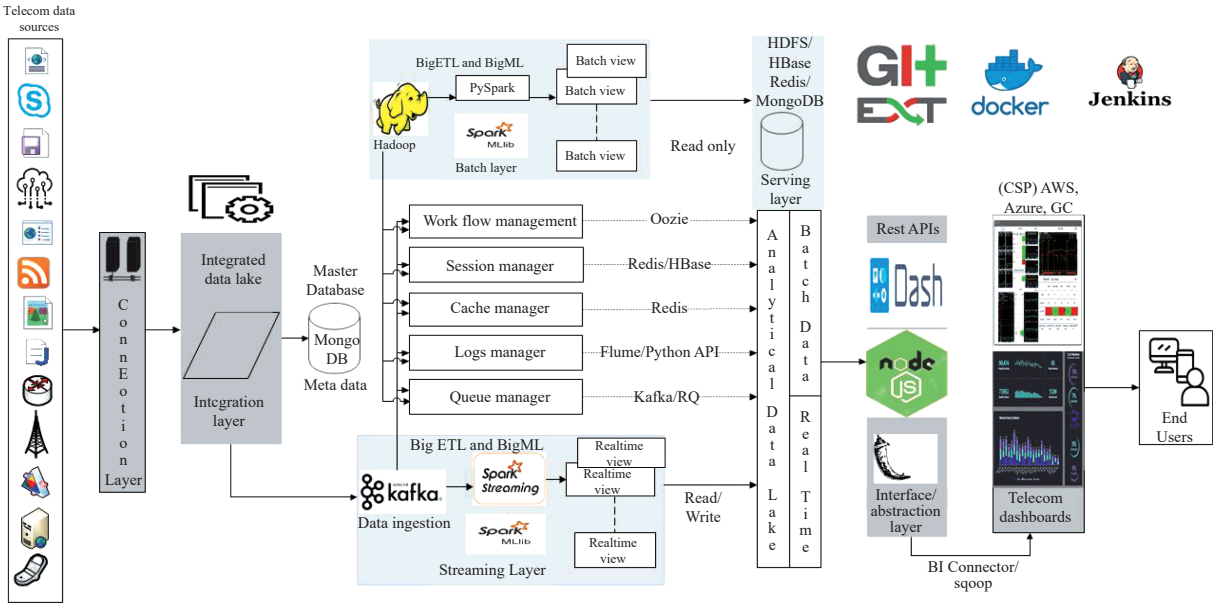

Fig. 12. LambdaTel: Proposed Lambda Architecture (BDA Pipeline) for Telecommunication Companies (adapted from our previous work [86]).

1) Connection Layer: The Connection layer allows the different types of telecom data sources to feed data to our BDA pipeline. In other words, this layer implements an application programming interface (API) of data connectors for a potentially large set of standard No-SQL databases, SQL databases, IoT feeds and other streaming or batch telecom data feeds. Python's support for connecting to NoSQL and other databases facilitates the implementation of this layer, e.g., pymongo API for connecting to a MongoDB instance and redis-py API for connecting to Redis database instance.

2) Integration Layer: The Integration layer is responsible for integrating telecom data from the Connection layer and inserting that in an integrated data lake. We propose to deploy a master database (either on a single or multiple servers) to store the data lake. We also propose to use MongoDB for this purpose for its flexibility of storage schema and support for both batch and streaming data. The actual integration of data can be done by storing each individual telecom source data in its relevant database (preferably NoSQL) and then implementing a controller API over these different stores for coordination. For example, newsfeeds from social networks can be continuously stored in Neo4J and call detail records in MongoDB where the controller keeps meta-data information of associations, data, storage capacity activities to provide access. Redis is our recommendation for metadata store to encourage quicker recovery and storage with minimize management overhead. Talend and Pentaho tools are not suitable here for data incorporation.In order to maintain the efficiency of BDA process, our proposal is towards Python programming for every stage.

3) Batch Layer: The Batch layer is responsible for batch (static) processing of big telecom data from the master database. We recommend a using a Hadoop cluster to tackle the major ETL tasks for telecom big data in this layer. If some tasks are required to be processed faster, these can be done through Apache Spark and the more lengthy and time-taking tasks can be processed through MapReduce, for instance, computing the average call time for five years over 250 TB of data through MapReduce [87]. The Batch layer provides a thorough drilled-down analysis to supplement the processing done in the streaming layer.

4) Streaming Layer: The Streaming layer is responsible for processing of real-time/dynamic) telecom streams. This layer presents real-time views and basic analytics at an abstract level. We recommend the use of Apache Kafka for ingesting the streams Apache Spark's SparkStreaming feature for processing. Another competitor to Kafka is Flume (geared particularly for log analysis) although Kafka's usecases outnumber outnumber Flume's by a large margin. Similarly, Apache's Storm API is a competitor to SparkStreaming but the latter is considered to be more applicable with respect to usecases. The selection of the tool should be preceded by a concrete analysis as BDA technologies continue to evolve. High velocity dynamic data streams could be stored in MongoDB or Redis if any use case occur otherwise it could be inefficient [88].

5) Serving Layer: The Serving layer consolidates the results of both Batch and Streaming layers. It acts as a staging area where the batch and stream processing results are integrated together as per the requirements of the end-users, e.g., C-level executives of the telecom company. We recommend implementing the layer on a separate server machine, which we call the analytical data lake. This layer prepares processed data for displaying the end-user dashboards.

6) Interface Layer: The Interface layer combines all the back-end layers (Connection, Integration, Batch, Streaming and Serving) with the front-end layer (Dashboard layer). Here, the well-known Python API's for implementation of REST interfaces, e.g., Flask and Dash can be employed. Along with this, we recommend using Node.JS web server technology due to its enhanced scalability.

7) Dashboard Layer: The Dashboard layer involves

[4]We have adapted this architecture from one of our previous works [86].

displays a series of dashboards to be seen by various telecom end-users. Every dashboard connects through the Serving layer through standard connectors, e.g., BI connectors given by various BI tools (such as Tableau, Oracle's NI and QlikView), or Apache's Sqoop. Plotly, an on open-source Python APIs is useful to create dashboards. As per client needs, the whole pipeline (or simply the front-end) could be deployed on any of the well-know cloud service providers, e.g., Amazon's AWS, Microsoft's Azue or Google's Cloud; we recommend using SSL technology to connect dashboards to Serving layer through Interface layer.

Parallel and exclusive data streams can be process by both layers. Static and dynamic both layers ought to include ETL (data cleaning) exercises and statistical modeling, e.g., big data predictive analytics can be marked as BigML. During this process, program the data management modules in python which further develop the background routines for processing static and dynamic data:

8) Workflow Management: This module manages the large assortment of potential workflows that are possible in a BDA pipline for telecom data. Apache's Oozie is recommended here as a task scheduler. Oozie is Python-compatible and allows creation of formation of Hive, MapReduce, and Sqoop tasks as Directed Acyclic Graphs (DAGs).

9) Session Management: This modules stores the session of every activity in the BDA pipeline in a stateless manner. As this is likely to generate much big data itself with high velocity, we recommend creating archives with respect to a windowing period. Here, our recommendation is to use Redis as the session database. In case of storage requirement over a larger timeframe, we can employ Apache Cassandra or HBase.

10) Cache Management: This module speeds up telecom BDA processing through implementation of a separate caching mechanism. Redis data structure is highly robust in nature and is the best to use for cache management with more usecases, variety of data structures better strategies for removing data from main memory. [89], [90].

11) Log Management: Client logging, server preparing and troubleshooting information can be served through log management, as indicated by standard practice. There should no negligence on administrator side for coverage of client clickstream in sessions and logs. Logged information can be acquired through Flume and processed as MapReduce tasks.

12) Queue Management: Due to the diverse requirement of analytical task at different events, queuing up the tasks (e.g., in an Oozie instantiation) can be required. Kafka is an ideal data queueing system for streaming real-time data. For static data, we recommend RQ (Redis Queue) programming, which executes queues in Redis scripted in Python. RQ is also utilized for real-time data incase Kafka is not applicable.

13) Resource Management: This module coordinates the different BDA pipeline resources and activities. The best software for this task is Apache's Zookeeper which can detect master node and slave node failures and help recover from such faults. It also provides interfaces to manage cluster resources in an effective and efficient manner.

In our opinion, the aforementioned BDA pipeline is standard as per the current BDA technology stack. Last but not the least, we recommend implementing this pipeline in a Dockerized manner, with each activity running in its own docker container for more efficient processing and ease of coordination with other dockers. Also, BDA pipelining requires a development operations (DevOps) type of structure, with continuous integration and continuous deployment being managed by the standard Jenkins tool. All data and results being generated should be stored in private GitHub repositories for enhanced security and identity and access management (IAM) solution.

It is important to discuss the strengths and weaknesses of both lambda and kappa architectures to justify our selection. Our motivation for selecting lambda is that telecommunication analytics use cases all require both batch-level analyses as well as real-time analyses (primarily due to the requirement of data cleaning and machine learning at the batch level). In a kappa architecture, there is no operational component for batch-level analyses and analyses are computed on-the-fly. As proof, let us discuss two important use cases of telecom industry for LambdaTel: 1) Customer Relationship Management (CRM): call detail records (CDRs) of customers (streaming in nature) are fed into both batch layer and streaming layer in parallel. In batch layer, the CDRs are pre-processed and cleaned and then machine learning is applied to extract important customer segments, all in batch mode over a period of one hour. These segments are then fed to the serving layer. In streaming layer, real-time analytics starts to immediately show basic results like customer call throughput and average calling time per unit time, which are also fed to the serving layer. Here, real-time results are then shown with respect to segments available from batch layer, to present the required CRM picture to business decision makers. This use case can also be applied for other machine learning applications like prediction (classification, regression, time series forecasting) of telecom KPIs (calling time, SMS per second, mobile data usage frequency, revenue, sales, etc.) 2) Customer Attrition: CDRs and historical attrition data are fed to the batch and streaming layer in real-time, while marketing data and competitors' data are fed to the batch layer only at a pre-determined time. The batch layer performs ETL to clean all data to predict customer churn, with result sent to serving layer. The streaming layer presents basic attrition analytics with respect to customer segments (computed previously), and in serving layer, the results are combined to present attrition prediction for each segment. We can similarly prove the need for batch-level analytics in other use cases related to marketing, cross-selling/up-selling, human resource management and operational analyses.

Also, lambda architecture guarantees an error-free data execution process due to the presence of batch layer, hence maintaining a good balance between speed and reliability along with a fault-tolerant and scalable architecture due to Hadoop (plus Spark) implementation at batch level. It is interesting to note a blog questioning the lambda architecture by Jay Kreps in 2014 [91], who actually proposed kappa. According to him, lambda brings much coding overhead for ETL at batch layer, primarily required for machine learning.

However, the conquest of Python as a data science and big data language has made coding practices much simpler in the last 5 years. In lambda, we may need to re-process or repeat executions per batch but this can be catered by using in-memory and/or columnar storage solutions, which have also matured since 2014. Finally, a lambda architecture is still difficult to migrate or re-organize but considering the lack of any published lambda architecture for telecom, we think the time for this migration is still far; the need of the moment is to first implement and use it. the kappa use case allows execution of real-time queries, either on real-time data or data previously stored in some in-memory or streaming database without focusing on ETL. These situations can also be tackled by lambda, in which we can temporarily disable the batch layer for such requirements (for more information, kindly refer to [17], [91]–[95]).

A. An On-Going Application of LambdaTel

We are currently implementing LambdaTel for a local company (jazz.com.pk) to conduct a proof of concept (POC) application for their use case related to cross-selling/up-selling of customer services for marketing division. This is an on-going work in which we can mention the current results without providing sensitive information. The company currently maintains an enterprise resource planning (ERP) implementation of SAP, with an Oracle back-end consisting of 5 different databases and around 1450 tables in all. There was no analytical infrastructure in-place previously. All queries were executed through structured query language (SQL) which was generating delays for several complex queries. Results were shown through a standard business intelligence (BI) tool. We also discovered a major problem of data quality in these tables, particularly missing values, incomplete values, inconsistent data, and data entry errors. For the cross-selling/up-selling use case, the requirement was to get the customers to purchase more than one service or upgrade in a single attempt; specifically, to predict which customer will purchase in this manner. For this, we identified 410 relevant tables, measuring around 825 GB. We extracted this data through connection layer, and inserted it in a MongoDB cluster (Master Database) with the metadata. This activity consumed two months. Then, we implemented the batch layer through a Hadoop cluster with Spark front-end. We cleaned data thoroughly through ETL functions encoded in Pig Latin and running on Hadoop. We then used processed data for prediction through MLLib over Hadoop, and fed the results to serving layer. This activity also consumed two months. Once the customers to target were identified, we started inserting their real-time CDRs to streaming layer to compute day-to-day behavioral analyses of mobile phone usage, which were all fed to serving layer. This layer now shows the predictions of customers to target for cross-sell/up-sell, along with their real-time behavior to put things in perspective. This activity consumed one month. Currently (as of September 2019), the marketing division is testing the predictions at serving layer, and has demonstrated a small increase in customer loyalty due to some successful predictions. We have used exactly the same technology stack as proposed for LambdaTel. For the company, LambdaTel has brought the following advantages: 1) implementation of a complete data quality execution pipeline which can be replicated for different use cases, 2) implementation of a master database (data lake) for analytics which was previously unavailable, 3) a combination of both real-time analyses and batch-level machine learning to understand the consumers more deeply, and 4) implementation of a personalized dashboard using Apache Flask which was more convenient for top-level management as compared to the current BI tool implementation. The batch layer keeps on updating its machine learning model on a daily basis, while the streaming layer always gets the CDR's for the predicted customers, and all this in an automated manner through Python scripts. All of this has been very effective, efficient and reliable for the marketing department.

We must mention that LambdaTel is not a solution for companies which don't require machine learning or real-time analytics on-the-fly, or don't tend to focus much on maintaining data lakes for multiple analytical use cases.

VII. Answering the Research Questions

We now answer our research questions as follows:

A. RQ1: How much research literature is focused on BDA applications to telecom sector and what is the BDA technology stack in these articles? Answer: In all, 38 articles are focused on BDA applications to telecom sector (primarily from 2010 - March 2018). Technology stack includes Hadoop and some of its ecosystem APIs, MapReduce, Spark and some of its component APIs, Kafka, Flink, R, NoSQL databases, statistical analysis, machine learning, deep learning, cloud computing and social network analysis.

B. RQ2: What are the benefits and challenges mentioned in these articles and how much benefit has been actually realized? Answer: Optimized costs and better customer experience, revenues and security are the major benefits. Challenges include lack of a standard BDA architecture for implementation in industries, along with a lack of security and BDA expertise. Benefits are realized in a limited manner in academic research with respect to frequency of articles and experimental validations in industry and use of standard tools from ever-expanding BDA landscape. Although cloud service providers like AWS and Azure can address data security concerns of telecom practitioners, many telcos' policies prevent data from leaving the premises [96], [97].

C. RQ3: How can the challenges be strongly addressed to facilitate BDA applications to telecom sector? Answer: We have answered this by proposing a state-of-the-art lambda architecture for telecom practitioners called LambdaTel; we specify the exact components of this architecture and propose the use of Python to implement it due to its massive online community and availability of Python's APIs in all NoSQL solutions.

VIII. Conclusions and Future Work

Big data analytics (BDA) has much to offer for telecommunications industry and its importance can hardly be underestimated. In this paper, we determined through a

systematic literature review that the practical applications of BDA to telecom are limited in academic research with respect to a lack of architecture and usage of latest solutions in an expanding technology stack. To solve this problem and address other challenges, we have proposed and described LambdaTel, a state-of-the-art lambda architecture for BDA implementations in telecom sector. It is important to note that we have successfully implemented LambdaTel in a telecom solution called Darbi (https://www.darbi.io/). Darbi currently has one implementation in the military but the experiments cannot be discussed due to the confidential nature of the application. A limitation of LambdaTel is that the BDA implementation solutions it proposes are not eternal in nature, e.g., MongoDB could be replaced by another better NoSQL document store five years later. This defines our future work also: there is a need to "keep up" with the pace of BDA innovation and "keep on" modifying LambdaTel's implementation solutions accordingly. We believe LambdaTel gives a strong opportunity to telecom practitioners to implement BDA pipelines now in their own enterprises. In fact, fulfilling the requirements of [15], LambdaTel presents a type of roadmap for BDA application to telecom sector through technology and process improvements. We strongly believe that it can be directly implemented in the industry with minor modifications if needed. As future work, we intend to target BDA applications to Telecom (and related sectors) with respect to IoT domain, which has an apparently rapidly-expanding user base with many companies/startups planning IoT applications in their business operations along with increasing publication of research papers [98], [99]. We would be interested in determining the exact number of Telecom applications and then to propose and implement an IoT-BDA framework for Telecom sector.

REFERENCES

[1] D. Laney, "3D data management: Controlling data volume, velocity, and variety," META Group, Tech. Rep., February 2001. [Online]. Available: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

[2] P. Zikopoulos, C. Eaton *et al.*, *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*. McGraw-Hill Osborne Media, 2011.

[3] M. E, "The world according to linq," *Communications of the ACM*, vol. 10, no. 54, pp. 45–51, 2011.

[4] F. X. Diebold, "Big data dynamic factor models for macroeconomic measurement and forecasting," in *Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress of the Econometric Society, "(edited by M. Dewatripont, LP Hansen and S. Turnovsky)*, 2003, pp. 115–122.

[5] J. Liebowitz, *Big Data and Business Analytics*, 1st ed., Amazon, Ed. CRC Press, 2013.

[6] F. J. Ohlhorst, *Big Data Analytics: Turning Big Data into Big Money*, 1st ed., Amazon, Ed. John Wiley & Sons, 2012.

[7] G. C. Deka, *NoSQL: Database for Storage and Retrieval of Data in Cloud*, Amazon, Ed. Chapman and Hall/CRC, 2017.

[8] C. M. Ricardo and S. D. Urban, *Databases Illuminated*, 3rd ed., Amazon, Ed. Jones & Bartlett Learning, 2015.

[9] M. D. D. Silva and H. L. Tavares, *Redis Essentials*, Amazon, Ed. Packt Publishing - ebooks Account, 2015.

[10] M. D. D. Silva and H. L. Tavares, *MongoDB: The Definitive Guide: Powerful and Scalable Data Storage*, 2nd ed., Amazon, Ed. O'Reilly Media, 2013.

[11] J. Bughin, "Telcos: The untapped promise of big data," https://www.mckinsey.com/industries/telecommunications/our-insights/telcos-the-untapped-promise-of-big-data, 2016.

[12] P. Singh, "10 reasons why big data and analytics projects fail," https://analyticsindiamag.com/10-reasons-big-data-analytics-projects-fail/, 2017.

[13] B. Violino, "How to avoid big data analytics failures," https://www.infoworld.com/article/3212945/big-data/how-to-avoid-big-data-analytics-failures.html, 2017.

[14] H. Demirkan and B. Dal, "The data economy: Why do so many analytics projects fail?" http://analytics-magazine.org/the-data-economy-why-do-so-many-analytics-projects-fail/, 2014.

[15] M. Asay, "85% of big data projects fail, but your developers can help yours succeed," https://www.techrepublic.com/article/85-of-big-data-projects-fail-but-your-developers-can-help-yours-succeed/, 2017.

[16] Datafloq, "Top reasons of hadoop - big data project failures," https://datafloq.com/read/top-reasons-of-hadoop-big-data-project-failures/2185, 2017.

[17] N. Marz and J. Warren, *Big Data: Principles and Best Practices of Scalable Realtime Data Systems*. Manning Publications Co., 2015.

[18] M. Chen, S. Mao, and Y. Liu, "Big data: a survey," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 171–209, 2014.

[19] A. Kumari, S. Tanwar, S. Tyagi, N. Kumar, M. Maasberg, and K.- K. R. Choo, "Multimedia big data computing and internet of things applications: A taxonomy and process model," *J. Network and Computer Applications*, vol. 124, pp. 169–195, Dec. 2018.

[20] A. Bahga and V. Madisetti, *Big Data Science & Analytics: A Hands-On Approach*, Amazon, Ed. VPT, 2016.

[21] M. Turck, "Firing on all cylinders: The 2017 big data landscape," http://mattturck.com/bigdata2017/, 2017.

[22] J. Manyika, M. Chui, M. G. Institute, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. Byers, *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. McKinsey, 2011.[Online]. Available: https://books.google.com.pk/books?id=APsUMQAACAAJ

[23] S. Parise, "Big data: a revolution that will transform how we live, work, and think, by viktor mayer-schonberger and kenneth cukier," *J. Information Technology Case and Application Research*, vol. 18, no. 3, pp. 186–190, sep. 2016. [Online]. Available: https://doi.org/10.1080/15228053.2016.1220197

[24] J. Bughin, "Reaping the benefits of big data in telecom," *J. Big Data*, vol. 3, no. 1, 2016.

[25] S. Han, C. -L. I, G. Li, S. Wang, and Q. Sun, "Big data enabled mobile network design for 5g and beyond," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 150–157, 2017. [Online]. Available: https://doi.org/10.1109/mcom.2017.1600911

[26] D. Sipus, "Big data analytics for communication service providers," in *39th IEEE Int. Conv. Information and Communication Technology, Electronics and Microelectronics,* May 2016.

[27] M. S. Parwez, D. Rawat, and M. Garuba, "Big data analytics for user activity analysis and user anomaly detection in mobile wireless network," *IEEE Trans. Industrial Informatics*, 2017.

[28] I. Chih-Lin, Y. Liu, S. Han, S. Wang, and G. Liu, "On big data analytics for greener and softer ran," *IEEE Access*, vol. 3, pp. 3068–3075, 2015.

[29] R. F. Baumeister and M. R. Leary, "Writing narrative literature reviews," *Review of General Psychology*, vol. 1, no. 3, pp. 311–320, 1997.

[30] C. M. Murphy, "Writing an effective review article," *Journal of Medical Toxicology*, vol. 8, no. 2, pp. 89-90, Jun 2012.[Online]. Available: https://doi.org/10.1007/s13181-012-0234-2

[31] A. Leon-Garcia and I. Widjaja, *Communication Networks*, 2nd ed., Amazon, Ed. USA: McGraw-Hill Education, 2003.

[32] L. Goleniewski and K. W. Jarrett, *Telecommunications Essentials: The Complete Global Source*, 2nd ed., K. W. Jarrett, Ed. USA: Addison Wesley Professional;, 2006.

[33] Gartner, "It glossary," https://www.gartner.com/it-glossary/big-data, 2018.

[34] T. White, *Hadoop: The Definitive Guide*, 3rd ed., Amazon, Ed. USA: Yahoo Press, 2012.

[35] F. Cruz, P. Gomes, R. Oliveira, and J. Pereira, "Assessing NoSQL databases for telecom applications," in *Proc. 13th IEEE Conf. Commerce and Enterprise Computing*, Sept. 2011.

[36] H. Daki, A. El Hannani, A. Aqqal, A. Haidine, A. Dahbi, and H. Ouahmane, "Towards adopting big data technologies by mobile networks operators: A moroccan case study," in *Proc. 2nd IEEE Int. Conf. Cloud Computing Technologies and Applications*, 2016, pp. 154–161.

[37] L. George, *HBase: The Definitive Guide: Random Access to Your PlanetSize Data*, 1st ed., Amazon, Ed. USA: O'Reilly Media, 2011.

[38] M. A. Abbasi, *Learning Apache Spark 2.0*, 1st ed., Amazon, Ed. USA: Packt Publishing - ebooks Account, 2017.

[39] N. Garg, *Learning Apache Kafka, Second Edition*, 2nd ed., Amazon, Ed. USA: Packt Publishing - ebooks Account, 2015.

[40] F. Hueske and V. Kalavri, *Stream Processing With Apache Flink: Fundamentals, Implementation, and Operation of Streaming Applications*, 1st ed., Amazon, Ed. USA: O'Reilly Media, 2018.

[41] S. T. Allen, M. Jankowski, and P. Pathirana, *Storm Applied: Strategies for Real-Time Event Processing*, 1st ed., Amazon, Ed. USA: Manning Publications, 2015.

[42] P. J. Sadalage and M. Fowler, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*, 1st ed., Amazon, Ed. USA: Addison-Wesley Professional, 2012.

[43] G. Scholar, "Inclusion guidelines for webmasters," https://scholar.google.com/intl/en/scholar/inclusion.html, 2018.

[44] Mendeley, "Mendeley feed," www.mendeley.com, 2018.

[45] E. J. Khatib, R. Barco, P. Muñoz, I. De La Bandera, and I. Serrano, "Self-healing in mobile networks with big data," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 114–120, 2016.

[46] S. Bi, R. Zhang, Z. Ding, and S. Cui, "Wireless communications in the era of big data," *IEEE Communications Magazine*, vol. 53, no. 10, pp. 190–199, 2015.

[47] B. R. Chang, H. F. Tsai, Z. -Y. Lin, and C. -M. Chen, "Access-controlled video/voice over ip in hadoop system with bpnn intelligent adaptation," in *Proc. IEEE Int. Conf. Information Security and Intelligence Control*, 2012, pp. 325–328.

[48] A. Drosou, I. Kalamaras, S. Papadopoulos, and D. Tzovaras, "An enhanced graph analytics platform (gap) providing insight in big network data," *J. Innovation in Digital Ecosystems*, vol. 3, no. 2, pp. 83–97, 2016.

[49] S. B. Elagib, A.-H. A. Hashim, and R. Olanrewaju, "CDR analysis using big data technology," in *Proc. IEEE Int. Conf. Computing, Control, Networking, Electronics and Embedded Systems Engineering*, 2015, pp. 467–471.

[50] J. George, C. -A. Chen, R. Stoleru, and G. Xie, "Hadoop MapReduce for mobile clouds," *IEEE Trans. Cloud Computing*, pp. 1–1, 2016.[Online]. Available: https://doi.org/10.1109/tcc.2016.2603474

[51] Y. He, F. R. Yu, N. Zhao, H. Yin, H. Yao, and R. C. Qiu, "Big data analytics in mobile cellular networks," *IEEE Access*, vol. 4, pp. 1985–1996, 1985.

[52] X. Lu, F. Su, H. Liu, W. Chen, and X. Cheng, "A unified OLAP/OLTP big data processing framework in telecom industry," in *Proc. 16th IEEE Int. Symp. Communications and Information Technologies*, Sept. 2016, pp. 290–295.

[53] Y. Ouyang, L. Shi, A. Huet, M. M. Hu, and X. Dai, "Predicting 4g adoption with apache spark: A field experiment," in *Proc.16th Int. Symp. Communications and Information Technologies*, 2016, pp. 235-240.

[54] A. Imran, A. Zoha, and A. Abu-Dayya, "Challenges in 5G: how to empower son with big data for enabling 5G," *IEEE Network*, vol. 28, no. 6, pp. 27–33, 2014.

[55] J. Liu, F. Liu, and N. Ansari, "Monitoring and analyzing big traffic data of a large-scale cellular network with hadoop," *IEEE Network*, vol. 28, no. 4, pp. 32–39, Jul. 2014. [Online]. Available: https://doi.org/10.1109/mnet.2014.6863129

[56] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5g," *IEEE Network*, vol. 30, no. 1, pp. 44–51, 2016.

[57] E. Baccarelli, N. Cordeschi, A. Mei, M. Panella, M. Shojafar, and J. Stefa, "Energy-efficient dynamic traffic offloading and reconfiguration of networked data centers for big data stream mobile computing: review, challenges, and a case study," *IEEE Network*, vol. 30, no. 2, pp. 54–61, 2016.

[58] S. Jain, M. Khandelwal, A. Katkar, and J. Nygate, "Applying big data technologies to manage qos in an sdn," in *Proc. 12th IEEE Int. Conf. Network and Service Management*, 2016, pp. 302–306.

[59] R. I. Jony, A. Habib, N. Mohammed, and R. I. Rony, "Big data use case domains for telecom operators," in *Proc. IEEE Int. Conf. Smart City/SocialCom/SustainCom*, Dec. 2015, pp. 850–855.

[60] A. Saad, A. R. Amran, I. W. Phillips, and A. M. Salagean, "Big data analysis on secure voip services," in *Proc. 11th Int. Conf. Ubiquitous Information Management and Communication*. ACM, 2017, pp. 5.

[61] H. Park, H. Gebre-Amlak, B. Choi, S. Song, and D. Wolfinbarger, "Understanding university campus network reliability characteristics using a big data analytics tool," in *Proc. 2015 11th Int. Conf. Design of Reliable Communication Networks*, March 2015, pp. 107–110.

[62] M. Rathore, A. Paul, A. Ahmad, M. Imran, and M. Guizani, "Highspeed network traffic analysis: Detecting voip calls in secure big data streaming," in *Proc. IEEE 41st Conf. Local Computer Networks*, Nov. 2016, pp. 595–598.

[63] C. Şenbalcı, S. Altuntaş, Z. Bozkus, and T. Arsan, "Big data platform development with a domain specific language for telecom industries," in *Proc. High Capacity Optical Networks and Emerging/Enabling Technologies*, Dec. 2013, pp. 116–120.

[64] J. -C. Tseng, H. -C. Tseng, C. -W. Liu, C. -C. Shih, K. -Y. Tseng, C. -Y. Chou, C. -H. Yu, and F. -S. Lu, "A successful application of big data storage techniques implemented to criminal investigation for telecom," in *Proc. 15th IEEE Conf. Asia-Pacific Network Operations and Management Symposium*, 2013, pp. 1–3.

[65] R. Van Den Dam, "Big data a sure thing for telecommunications: telecom's future in big data," in *Proc. IEEE Int. Conf. CyberEnabled Distributed Computing and Knowledge Discovery*, 2013, pp. 148–154.

[66] T. Yigit, M. A. Cakar, and A. S. Yuksel, "The experience of nosql database in telecommunication enterprise," in *Proc. 7th IEEE Int. Conf. Application of Information and Communication Technologies*, 2013, pp. 1–4.

[67] R. Siddavaatam, I. Woungang, G. Carvalho, and A. Anpalagan, "An efficient method for mobile big data transfer over hetnet in emerging 5G systems," in *Proc. 21st IEEE Int. Workshop on Computer Aided Modelling and Design of Communication Links and Networks*, 2016, pp. 59–64.

[68] R. Siddavaatam, I. Woungang, G. Carvalho, and A. Anpalagan, "Efficient ubiquitous big data storage strategy for mobile cloud computing over hetnet," in *Proc. IEEE Global Communications Conf.*, Dec. 2016, pp. 1–6.

[69] R. K. Lomotey and R. Deters, "Management of mobile data in a crop field," in *Proc. IEEE Int. Conf. Mobile Services*, 2014, pp. 100–107.

[70] J. Magnusson and T. Kvernvik, "Subscriber classification within telecom networks utilizing big data technologies and machine learning," in *Proc. 1st Int. Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications*, ser. BigMine '12. ACM, 2012, pp. 77–84.

[71] Ö. F. Çelebi, E. Zeydan, O. F. Kurt, O. Dedeoglu, Ö. Ileri, B. AykutSungur, A. Akan, and S. Ergüt, "On use of big data for enhancing network coverage analysis," *ICT*, pp. 1–5, 2013.

[72] C. Costa, G. Chatzimilioudis, D. Zeinalipour-Yazti, and M. F. Mokbel, "Efficient exploration of telco big data with compression and decaying," in *Proc. IEEE 33rd Int. Conf. Data Engineering*, 2017, pp. 1332–1343.

[73] C.-M. Chen, "Use cases and challenges in telecom big data analytics," *APSIPA Trans. Signal and Information Processing*, vol. 5, pp. 12, 2016.

[74] R. Jony, "Preprocessing solutions for telecommunication specific big data use cases," Master's thesis, 02010.

[75] K. Wang, J. Mi, C. Xu, Q. Zhu, L. Shu, and D. -J. Deng, "Realtime load reduction in multimedia big data for mobile internet," *ACM Trans. Multimedia Computing, Communications, and Applications*, vol. 12, no. 5s, pp. 1–20, oct 2016. [Online]. Available: https://doi.org/10.1145/2990473

[76] J. van der Lande, "The future of big data analytics in the telecoms industry," *White Paper*, 2014.

[77] N. Dalkey and O. Helmer, "An experimental application of the DELPHI

method to the use of experts," *Management Science*, vol. 9, no. 3, pp. 458–467, Apr. 1963. [Online]. Available: https://doi.org/10.1287/mnsc.9.3.458

[78] R. R. Panko, *Business Data Networks and Telecommunications*, 4th ed., Amazon, Ed. USA: Prentice Hall, 2002.

[79] D. Slama, F. Puhlmann, J. Morrish, and R. M. Bhatnagar, *Enterprise IoT: Strategies and Best Practices for Connected Products and Services*, 1st ed., Amazon, Ed. O'Reilly Media, 2015.

[80] A. Banerjee, "Big data & advanced analytics in telecom: a multibillion-dollar revenue opportunity (technical report)," New York: Heavy Reading, 2013.

[81] M. Kiran, P. Murphy, I. Monga, J. Dugan, and S. S. Baveja, "Lambda architecture for cost-effective batch and speed big data processing," in *Proc. IEEE Int. Conf. Big Data*, 2015, pp. 2785–2792.

[82] C. E. Perkins and P. R. Calhoun, "Authentication, authorization, and accounting (AAA) registration keys for mobile ipv4," *RFC*, vol. 3957, pp. 1–27, 2005.

[83] SolidIT. (2017) Db-engines, ranking of key-value stores@ONLINE. [Online]. Available: https://db-engines.com/en/ranking/key-value+store, https://db-engines.com/en/ranking/document+store, https://db-engines.com/en/ranking/wide+column+store

[84] A. Anthony, *Mastering AWS Security: Create and Maintain A Secure Cloud Ecosystem*, 1st ed., Amazon, Ed. USA: Packt Publishing - ebooks Account, 2017.

[85] D. S. Yuri Diogenes, Tom Shinder, *Microsoft Azure Security Infrastructure (IT Best Practices - Microsoft Press)*, 1st ed., Amazon, Ed. USA: Microsoft Press, 2016.

[86] H. Zahid, T. Mahmood, and N. Ikram, "Enhancing dependability in big data analytics enterprise pipelines," in *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, G. Wang, J. Chen, and L. T. Yang, Eds. Cham: Springer International Publishing, 2018, pp. 272–281.

[87] G. Weiss, *Data Mining in the Telecommunications Industry*. GI Global, 2009.

[88] W. Queiroz, M. A. Capretz, and M. Dantas, "An approach for SDN traffic monitoring based on big data techniques," *J. Network and Computer Applications*, vol. 131, pp. 28–39, Apr. 2019.

[89] I. Haber, "Why redis beats memcached for caching," https://www.infoworld.com/article/3063161/nosql/why-redis-beats-memcached-for-caching.html, 2018.

[90] Redis, "Using redis as an lru cache," https://redis.io/topics/lru-cache, 2018.

[91] J. Kreps, "Questioning the lambda architecture," https://www.oreilly.com/ideas/questioning-the-lambda-architecture, Jul. 2, 2014.

[92] I. Samizadeh, "A brief introduction to two data processing architectures — lambda and kapp. for big data," https://towardsdatascience.com, 2018.

[93] J. Forgeat, "Data processing architectures - lambda and kappa," https://www.ericsson.com/en/blog/2015/11/data-processing-architectures--lambda-and-kappa, 2015.

[94] M. Verrilli, "From lambda to kappa: a guide on real-time big data architectures," https://www.talend.com/blog/2017/08/28/lambda-kappa-real-time-big-data-architectures/, Augist 28, 2017.

[95] J. ZAGELBAUM, "Kapp. architecture: a different way to process data," https://www.blue-granite.com/blog/a-different-way-to-process-data-kappa-architecture, Jan. 25, 2019.

[96] L. Zhou, A. Fu, S. Yu, M. Su, and B. Kuang, "Data integrity verification of the outsourced big data in the cloud environment: a survey," *J. Network and Computer Applications*, vol. 122, pp. 1–15, Nov. 2018.

[97] R. Nachiappan, B. Javadi, R. N. Calheiros, and K. M. Matawie, "Cloud storage reliability for big data applications: a state of the art survey," *J. Network and Computer Applications*, vol. 97, pp. 35–47, 2017.

[98] W. Xu, H. Zhou, N. Cheng, F. Lyu, W. Shi, J. Chen, and X. Shen, "Internet of vehicles in big data era," *IEEE/CAA J. Autom. Sinica*, vol. 5, no. 1, pp. 19–35, Jan. 2018.

[99] Z. Sheng, S. Pfersich, A. Eldridge, J. Zhou, D. Tian, and V. C. M. Leung, "Wireless acoustic sensor networks and edge computing for rapid acoustic monitoring," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 1, pp. 64–74, 2019.

Hira Zahid is a Doctoral Candidate at the Faculty of Computer Science, Institute of Business Administration (IBA), Karachi, Pakistan. Her research revolves around designing novel big data architectures to support real-time machine learning. Her research interests include big data analytics and deep learning. She has published a conference paper on enhancing dependability in big data analytics pipelines, published in the 4th Int. Symp. Dependability in Sensor, Cloud, and Big Data Systems and Applications (DependSys 2018), Australia. She has an M.S. and B.S. degrees with distinction from NED University of Engineering and Technology, Karachi, Pakistan, in computer and information systems. She is also serving as a Faculty Member with Iqra University since Spring 2016 in Department of Computer Science and has been employed at various software houses previously.

Tariq Mahmood is an Associate Professor at the Faculty of Computer Science, Institute of Business Administration (IBA), Karachi, Pakistan. He has a Ph.D. degree in machine learning from University of Trento, Italy, and an M.S. degree in statistical machine learning from Universite Pierre et Marie Curie (Paris 6), France. He has published around 20 international journal and 35 conference publications with total 691 citations and h-index of 12 (Google Scholar). His research interests include BDA, deep learning and machine learning/data science. He heads the Big Data Analytics Laboratory at IBA, with a focus on imparting data science and big data certifications to students and industry professionals, implementing BDA-related industrial projects and researching in BDA technology stack, particularly to develop BDA architectures for different types of streaming and non-streaming data. He also consults in various local industries regarding business intelligence, data governance, BDA and machine learning.

Ahsan Morshed is a Lecturer in ICT at CQUniversity, Melbourne. Previously, he was a Research Fellow in Data Analytics at Swinburne University of Technology and a senior project officer at RMIT University. He was also a Postdoctoral fellow at CSIRO (Australia) on sensor data integration and machine learning, and an Information Management Specialist in the OEKC division at Food and Agriculture Organization (FAO) of UN in Rome, Italy. During his time in FAO, he acquired extensive skills in metadata standards, knowledge organization systems, ontologies, Linked Open Data management and information management tools. His research interests are the big data, data science, semantic Web, linked open data and semantic machine learning. He holds a Ph.D. from the University of Trento, Italy. Dr. Morshed has 50 peer-reviewed publications (book, book chapter, journals, conference and workshop papers), with 229 citations and an h-index of 6 (Google Scholar).

Timos Sellis (F'09) is a Professor at Swinburne University of Technology, Australia. He holds a Diploma from National Technical University of Athens (NTUA), an M.Sc. degree from Harvard University, and a Ph.D. from the University of California at Berkeley. Timos has a significant international research reputation in big data, data analytics, data integration and spatio-temporal database systems. He is a Fellow of the Association for Computing Machinery (ACM) for his contributions to database query optimisation, spatial data management and data warehousing and also an Institute of Electrical and Electronics Engineers (IEEE) Fellow for his contributions to database query optimisation and spatial data management. In 2018 he was awarded the IEEE TCDE Impact Award, in recognition of his impact in the field and for contributions to database systems research and broadening the reach of data engineering research. Before joining Swinburne, Timos was the Director of the Institute for Management of Information Systems and Professor at the National Technical University of Athens. He has also held the role of Director, Big Data Lab at RMIT University.