Journal of Big Data

RESEARCH

Open Access

Evaluation of big data frameworks for analysis of smart grids

Mohammad Hasan Ansari[1], Vahid Tabatab Vakili[1*] and Behnam Bahrak[2]

*Correspondence: vakily@iust.ac.ir
[1] Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran
Full list of author information is available at the end of the article

Abstract

With the rapid development of smart grids and increasing data collected in these networks, analyzing this massive data for applications such as marketing, cyber-security, and performance analysis, has gained popularity. This paper focuses on analysis and performance evaluation of big data frameworks that are proposed for handling smart grid data. Since obtaining large amounts of smart grid data is difficult due to privacy concerns, we propose and implement a large scale smart grid data generator to produce massive data under conditions similar to those in real smart grids. We use four open source big data frameworks namely Hadoop-Hbase, Cassandra, Elasticsearch, and MongoDB, in our implementation. Finally, we evaluate the performance of different frameworks on smart grid big data and present a performance benchmark that includes common data analysis techniques on smart grid data.

Keywords: Smart grid, Big data, Data generator, Performance

Introduction

The smart grid combines the traditional power system with information technology leading to a one of most important modern cyber-physical systems. Smart grid is envisioned to fully integrate high-speed and two-way communication technologies into millions of power equipment to establish a dynamic and interactive infrastructure with new energy management capabilities, such as advanced metering infrastructure (AMI) and demand response. Smart grid heavily relies on information and communication technology to achieve efficient and reliable operation [1]. At the same time, smart grid big data has provided new opportunities for electric load forecasting, anomaly detection (e.g power theft), and demand side-management. However, the high-dimensional and massive smart grid big data creates new challenges in data transmission, data storage, and data analysis. This paper addresses the problem of creating a benchmark for big data frameworks used in smart grid big data analysis. We also develop a realistic smart grid data generator for performance analysis in real conditions.

Motivation

The rapid growth of smart grid, deployment of modern information and communication technologies and millions of newly deployed smart meters, will generate large amount of smart grid. Smart grid big data analysis is considered to be the key to solving significant problems of this industry. With the exponential growth of data, how to efficiently utilize this data becomes a critical issue. Most of recent work have focused on smart grid analysis and big data technology separately, without considering a benchmark for different big data frameworks. But, it is important to understand different big data frameworks' performance in analyzing millions of smart meters' data.

Contributions

The main contributions of this paper are as follows:

1. A smart grid data generator is designed based on big data platforms, taking into account the practical concerns of realistic smart grid.
2. The data generator is developed and implemented using Spark and HDFS file-systems.
3. The four most efficient open source big data frameworks are selected and used to analyze smart grid big data.
4. We propose a benchmark for evaluating smart grid big data frameworks in term of different criteria such as run time, throughput, latency and etc.
5. Based on the proposed benchmark, we compare four big data frameworks and select the best one for smart grid big data analysis

Paper structure

The rest of the paper is organized as follows. "Related work" section discusses related work. In "Methods" section, we explain the design of our scalable smart meter data generator. We also provide the required background on big data concepts and the four big data frameworks for smart grid big data analysis. In "Implementation and benchmarking of big data technology in smart grid" section we propose and deploy our benchmark for comparing smart grid big data frameworks. In "Results and discussion" section, we evaluate different frameworks for smart grid data analysis and present the benchmark results. Finally, "Conclusion" section provides some concluding remarks and outlines directions of future research.

Related work

There is a wide range of studies that are related to the big data challenges in smart grids. In this section, we try to briefly cover researches that their results are mostly related to our work.

Big data concepts in smart grid

Ye et al. in [2] illustrate big data analytics and cloud computing in a smart grid with focus on relevant issues in demand response and wide-area monitoring. To fulfill the potential of energy big data and obtain insights to achieve smart energy management, a comprehensive study of big data-driven smart energy management presented in [3]. Vasilakos et al. in [4] present a survey and discuss about new findings and developments in the smart grid big data analytics and security. A big data framework that can be a start for innovative research in smart grids and implementation of the framework on a secure cloud-based platform is presented in [5]. In [6], an overview on the state-of-the art of big data technology in smart grid is presented. A comprehensive survey on the life cycle of data and data trading, such as studying a variety of data pricing models, categorizing them into different groups, and comparison of these models is presented in [7]. Kang et al. in [8] introduce some recent progress in the application of big data analytics in China's electric power industry. A study on the compression techniques for smart meter big data is presented in [9]. A smart grid big data ecosystem based on the state-of-the-art Lambda architecture is proposed in [10] that is capable of performing parallel batch and real-time operations on distributed data. Jiang et al. in [11] proposed a high performance and privacy-preserving query scheme over encrypted multidimensional big metering data to address how to efficiently query encrypted multidimensional metering data stored in an untrusted heterogeneous distributed system environment. In [12], a lightweight communication approach for big data traffic is presented that can prevent or quickly respond to the occurrence of network congestion in situations that computing power and bandwidth are limited. A comprehensive tutorial and survey to highlight research challenges on the vulnerabilities in smart grid big data is presented in [13]. Hou et al. in [14] propose a novel temporal, functional, and spatial big data computing framework for large-scale smart grid. The potential applications of big data analytics in electric grids is discussed in [15]. A study on 0.1 billion data point, collected by various smart Internet of Things (IoT) devices in power system of China is done to analyze consumption characteristics of power users in [16].

Big data analysis in smart grid

Here, we briefly survey the papers that focused on using big data analysis technology in different domains such as anomaly detection, marketing and electricity price forecasting, fault tolerance etc. in smart grid. Liu et al. in [17] proposed a lambda system for detecting anomalous consumption patterns, aiming at assisting decision makings for smart energy management. A real-time anomaly detection framework, which can be built based upon smart meter data collected in smart grid big data is proposed in [18]. Vimalkumar et al. in [19] designed a big data framework using various machine learning techniques to detect intrusions. An unsupervised progressive incremental data mining mechanism applied to smart meters big energy consumption data through frequent pattern mining is presented in [20]. Wu et al in [21] proposed a security situational awareness mechanism based on the analysis of big data in the smart grid. A novel electricity price forecasting model that handle with huge price data in the smart grid big data is introduced in [22]. Ye et al. in [23] proposed a secure data learning scheme for big data applications in the information and communication technology infrastructure of the smart grid. In [24], a new model for the microgrids optimal scheduling and load curtailment based on big data concept is proposed. A data-driven method of association rule mining for transformer state parameters based on big data is proposed by combining the a-priori algorithm and probabilistic graphical model in [25]. From performance point of view only in [26], smart meter analytic from a software performance perspective is examined. Also a smart meter data generator using Spark that can generate semi-realistic data sets is presented in [27].

Methods

Figure 1 presents the overall architecture of our smart grid big data framework and data analysis system model based on this architecture. The system can be divided in to three blocks: data generator, database, and data analysis. Data initialization module is a smart meter dataset that has been provided by the Irish Social Science Data Archive [28]. This real data is collected during 2009 and 2010 with over 5000 Irish homes and businesses participating. The data set includes over 150 million records of 5000 customers for 18 months. A smart meter measures consumption in a given interval every 30 min. Figure 2 illustrates the typical power consumption of customer from ISSDA dataset. We

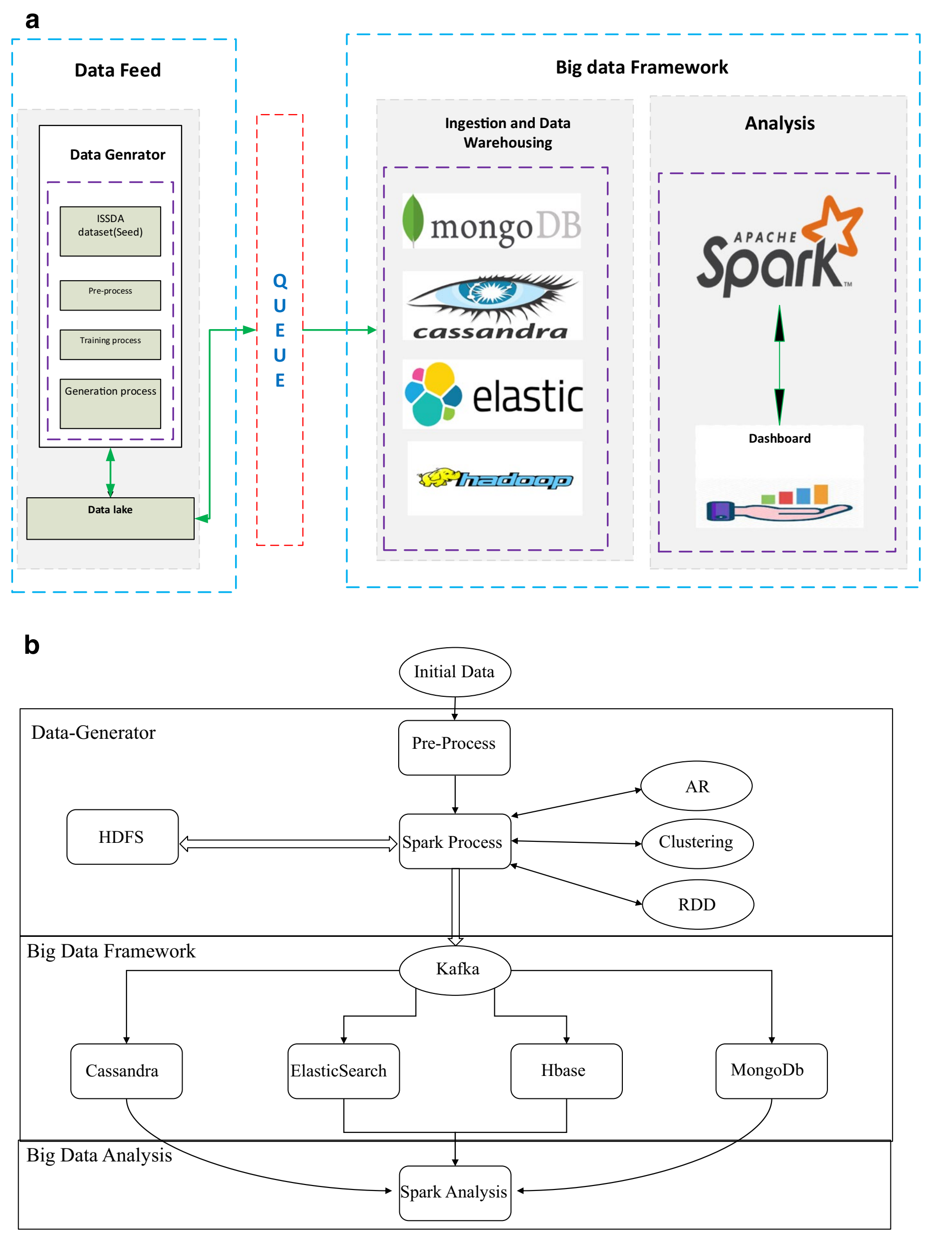

Fig. 1 **a** Overall architecture of our smart grid big data framework, **b** smart grid big data analysis system model

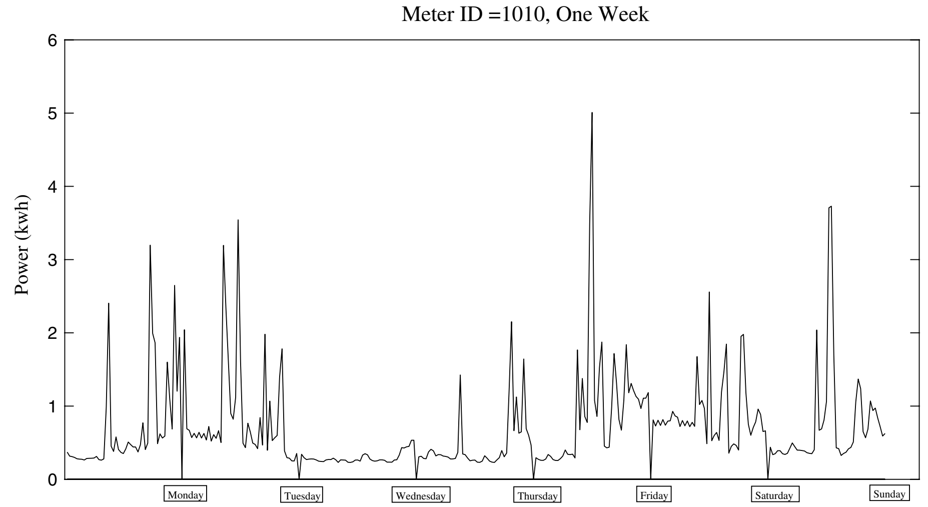

Fig. 2 A week smart meter reading of customer

can see a fixed pattern in workdays and small differences in weekends. In the following we describe each part of the smart grid big data system model in details.

Smart grid data generator model

In order to investigate the performance and benchmarking of big data technologies in smart grids, we need to process petabytes of data for smart grid data management and analytics. Indeed using realistic data for performance evaluation is much better, but obtaining datasets with records of millions customers is challenging because of privacy issues and the lack of truthful data sources. Therefore, in this section, we first describe a comprehensive smart meter data generator based on big data frameworks that can generate semi-realistic datasets with large number of customers. Our proposed data generator use a supervised machine learning algorithm that can generate data based on small real datasets. As shown in Fig 1, the data generator has different blocks: pre-process module, Spark module and HDFS file system module. The pre-process module, first checks the format and values in each field of the seed data, then clustering the customers in order to offer personalized energy-efficiency services [29]. It also prepares time series of data to send to the Spark module. The engine of the data generator includes two modules: training module and data generation module which are implemented using memory-based distributed computing framework Spark. In the first stage, Resilient Distributed Dataset (RDD) is built in RAM to store temporary data. The seed data is deseasonalized using a moving average time series model in order to flatten the periodic variations and create an adjusted time series. The moving average by smoothing the peak periods reduces the periodic variations from the actual time series and it uses autoregressive (AR) model trained with the flattened time series to predict meter reading. In the end, the periodic variations are added back. Actually, the training process will take a clustered seed data as the input to create the models and writes the output in HDFS. Since, generated data has a stream-oriented nature, we propose using Apache Kafka [30] as a message queue in big data framework in our model. The output is inserted in NoSQL database through Kafka queue. The generated data has four fields including meter ID,time, power consumption and environment temperature. The meter ID is consider as 1000 to 5001000. The time is stored as a five-digit numeric field, the first three numbers representing the day n and the next two digits representing the clock that it's value set between 1 to 48 which time of day start 0 correspond to 12 PM . The power consumption field is considered as numerical data. We see that, all data is structured column-based and range of changes is fully specified.

NoSQL database

NoSQL provides the new data management technologies designed to meet the features of big data i.e. volume, velocity, and variety of data. It can store and retrieve data that is modeled in means other than the tabular relations used in relational databases [31]. In comparison to relational databases, NoSQL databases are more scalable and provide superior performance when we are dealing with big data. Based on our experiences and domain experts' recommendations, we select four open source NoSQL database: Hbase, Cassandra, MongoDB and Elastic search. In the following we describe each technology.

Hbase

Hbase is a column store based on Apache Hadoop and concepts of BigTable. It is a NoSQL key/value store which runs on top of HDFS [32]. It supports four primary operations: *put* to add or update rows, *scan* to retrieve a range of cells, *get* to return cells for a specified row, and *delete* to remove rows, columns or column versions from the table. Although Hbase includes tables, a schema is only required for tables and column families, but not for columns, and it includes increment/counter functionality. Hbase has centralized architecture where the Master server is responsible for monitoring all RegionServer (responsible for serving and managing regions) instances in the cluster, and is the interface for all metadata changes. Hbase is optimized for reads, supported by single-write master, and resulting strict consistency model, as well as use of Ordered Partitioning which supports row-scans. It provides CP (Consistency, Partition tolerance) form the CAP theorem. The most important feature of Hbase is strong consistency and fast read and write with high scalability.

Cassandra

Cassandra is also a column store based distributed database management system based on ideas of BigTable and DynamoDB. Unlike Hbase, Cassandra has decentralized architecture and each node can perform any operation that it provides AP (Availability, Partition-Tolerance) from the CAP theorem [33]. Cassandra has excellent single-row read performance as long as eventual consistency semantics are sufficient for the use-case. Cassandra quorum reads, which are required for strict consistency will naturally be slower than Hbase reads. Cassandra is well suited for supporting single-row queries or selecting multiple rows based on a Column-Value index [34].

MongoDB

MongoDB is a document oriented database. All data in MongoDB is treated in JSON/BSON format. It is a schema less database which goes over tera-bytes of data in database.

It also supports master slave replication methods for making multiple copies of data over servers making the integration of data in certain types of applications easier and faster [31]. MongoDB combines the best of relational databases with the innovations of NoSQL technologies, enabling engineers to build modern applications. MongoDB provides the data model flexibility, elastic scalability, and high performance of NoSQL databases. As a result, engineers can continuously enhance applications, and deliver them at almost unlimited scale on commodity hardware [35].

Elasticsearch

Elasticsearch is an open-source search engine built on top of Apache Lucene^{TM}, a full-text search engine library. Elasticsearch is a real-time distributed search and analytics engine. It is used for full text search, structured search, analytics, and all three in combination [36]. It has a very flexible query API, that supports filtering, sort, pagination, and aggregations in the same query. Elasticsearch handles unstructured data automatically, meaning you can index JSON documents without predefining the schema. It will try to figure out the field mappings, and adding/removing the new/existing fields automatically. Also task such as clustering, replication of data, and automatic fail-over are supported out of the box and are completely transparent to the user.

Implementation and benchmarking of big data technology in smart grid

In this section the architecture of the platform is briefly introduced and principles for building real-time processing systems are presented. Furthermore, the advantages of using the architecture in smart grids are summarized.

Cluster specification

To implement our ecosystem, we use a cluster of 6 nodes. Each node is a server with 20 physical CPU core and 40 in hyper-thread mode, 128GB RAM, and 24TB HDD. We assign a node to Spark, a node to HDFS file system, and a node to Kafka queue. Each of the database technologies (i.e. Hbase, Cassandra, MongoDB, and Elasticsearch) is implemented on a cluster with three nodes.

Smart grid data generator implementation

The smart meter data generator is implemented on two nodes. The implementation is consisted of two functions: *data-preparator* and *data generator*. The initial value for training algorithm and specifications of data generated are summarized in Table 1.

The generated data is stored in HDFS and the format of each data file is as follows: 3 columns corresponding to Meter ID, a five digit code composed of day code: digits 1–3 (day 001 = 1st of January 2009), and time code: digits 4–5 (1–48 for each 30 min with 01= 00:00:00–00:29:59), and electricity power consumed during each 30 min interval (in kWh). By means of this implementation we generate a big data-set with 5,000,000 home and commercial customers for 1.5 year that might be used to evaluate big data platforms performance for smart grid data.

Table 1 Initial and generated data description

	Data description	Value
Initial data	Data country	Irish
	Data supplier	ISSDA
	Start time	1st January 2009
	End time	1st June 2010
	Number of meters	5000
	Number of records	131.4×10^6
	Recording frequency	30 min
	Length	48 observations
Generated data	Number of meters	5,000,000
	Start time	1st January 2017
	End time	1st June 2019
	Recording frequency	30 min
	Size of data	3T byte
	Number of records	131.4×10^9
	Length	48 observations

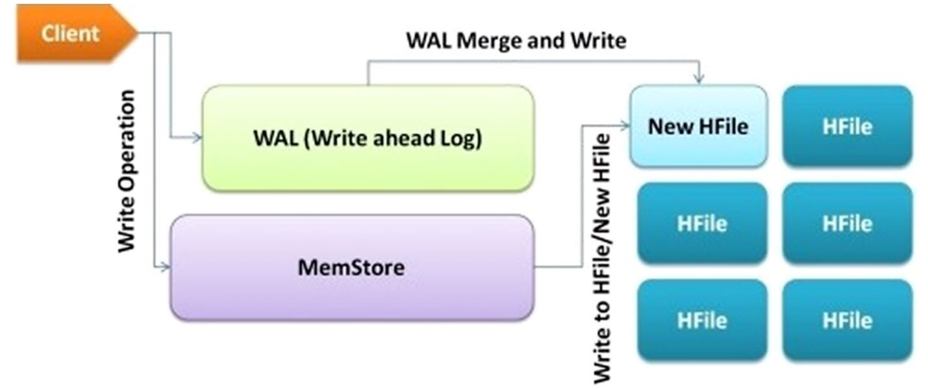

Fig. 3 Hbase data model

Big data backend implementation

Hbase

In Hbase, read and write processes have two stages: data transfer to RAM memory data-node that is called memstore and then copying on hfile. Figure 3 illustrate the process of Hbase read and write.

Cassandra

We implement our cluster with three nodes as a ring. Figure 4 shows the mechanism of data processing in Cassanadra. In data insertion, each write command changes store in memtable memory on RAM and then on SSTable. Also in order to protect data against fault and Cassanadra nodes down, replica of data is stored temporarily in an optimized disc called commitlog for each change caused by write command.

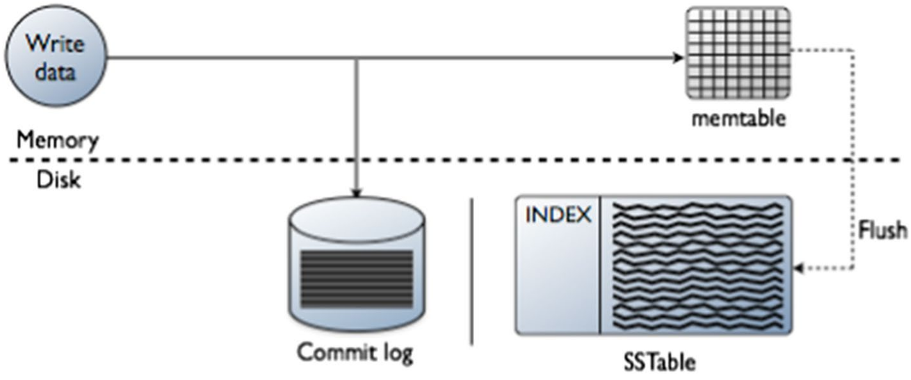

Fig. 4 Cassandra data model

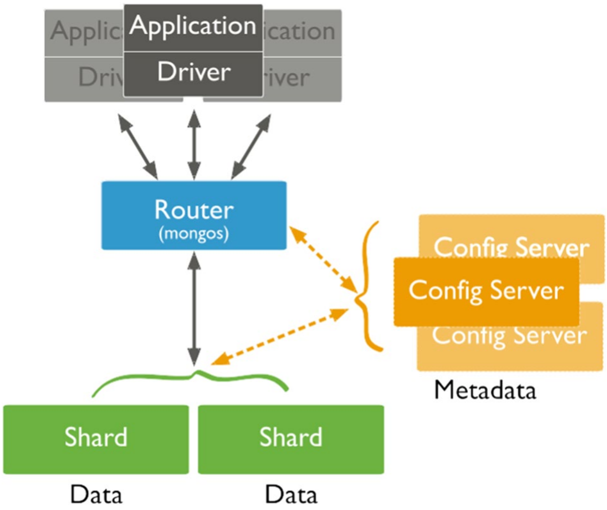

Fig. 5 MongoDB data model

MongoDB

In our MongoDB implementation we consider a process depicted in Fig. 5. The implementation has a mongos function that acts as a router and determines the path of data store. The data directly is written on tables, after the data store path is determined by mongos.

Elasticsearch

The data model of Elasticsearch that we have implemented is shown in Fig. 6.

Results and discussion

In this section, we evaluate the performance of each big data technology with default configurations in terms of effectiveness and scalability. The effectiveness will be evaluated by comparing WRITE and READ parameters. The scalability will be assessed by measuring the execution performance of the full mechanism. Table 2 summarizes the

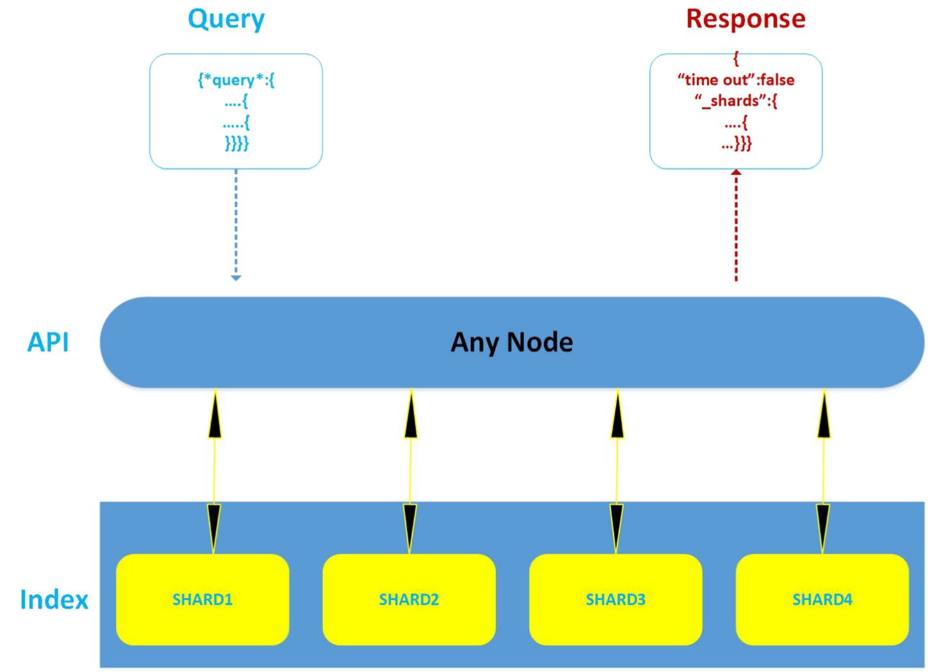

Fig. 6 Elasticsearch data model

Table 2 The summarized performance criteria results

	Criteria	Elastic	MongoDB	Hbase	Cassandra
Write process	Run time (ms)	351	430	450	300
	Throughput (ops/s)	185,000	100,308	150,462	230,030
	Operations (number of ops)	131.4×10^9	131.4×10^9	131.4×10^9	131.4×10^9
	Average latency (ms)	2.8	2.36	1.37	0.8
	Min latency (ms)	0.967	0.1	0.57	0.01
	Max latency (ms)	139	160	124	197
	95th percentile latency (ms)	2.489	2.125	2.943	2.343
	99th percentile latency (ms)	5.859	5.32	4.719	4.519
Read process	Run time (ms)	400	480	550	400
	Throughput (ops/s)	195,000	110,508	180,432	250,400
	Operations (number of ops)	131.4×10^9	131.4×10^9	131.4×10^9	131.4×10^9
	Average latency (ms)	2.1	3.39	3.898	1.898
	Min latency (ms)	0.825	0.21	0.62	0.021
	Max latency (ms)	219	349.52	397.24	218.497
	95th percentile latency (ms)	2.04	2.86	4.37	2.19
	99th percentile latency (ms)	6.18	5.83	6.34	5.79

comparison of the four technologies performance. We compare big data technologies based on the most famous criteria including: Run time, Throughput, number of operations, and latency.

Figure 7 shows the run time of each big data database technology. We see that Cassanadra in write and read stages has less run time and consequently better performance.

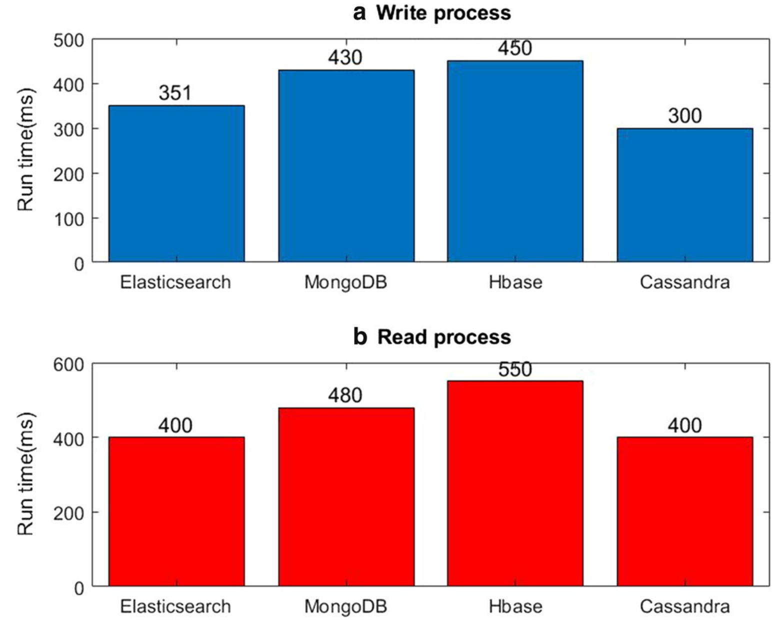

Fig. 7 Run time of different technologies

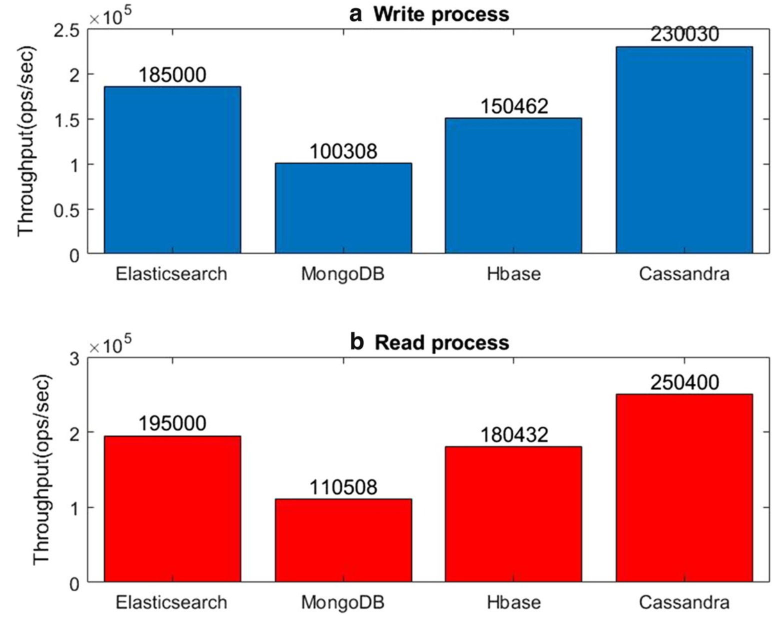

Fig. 8 Throughput of different technologies

Comparison of throughput's in ops/sec is presented in Fig. 8. It is shown that Cassandra has the best throughput and MongoDB performs the worst.

Figure 9 shows the percentile latency of different technologies. If *A*% latency of a technology is *B*, it means that *A*% of read or write latencies of this technology is less than or equal to *B*. For example in Fig. 9, we see that the 95% latency of MongoDB is 2 ms, thus 95% of read/write latencies of MongoDB are less than or equal to 2 ms. We

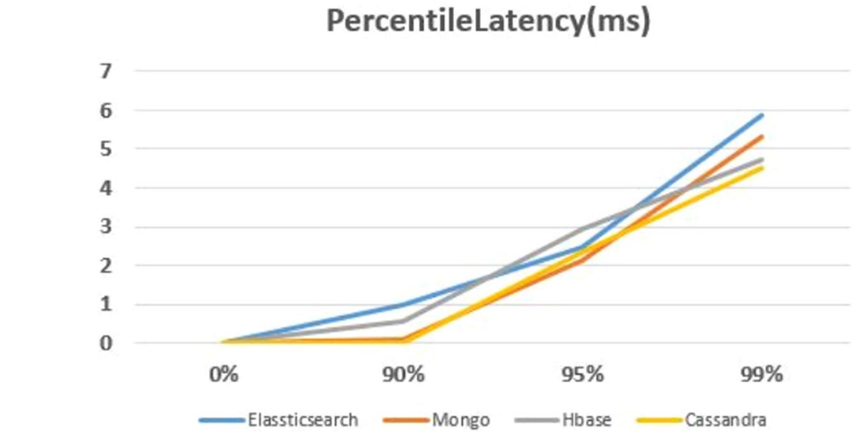

Fig. 9 Percentile-latency of Elasticsearch, MongoDB, Hbase and Cassandra comparison

see that Cassanadra has the smallest average latency in both read and write processes. Based on these results we observe that Cassandra has the best overall performance among the four open source big data database technologies that are used for analysis of smart grid big data in real scale. Generally as outlined in "Methods" section, because of smart grid meter data is structural column based data, that in our implementation the smart meter ID and time fields are considered as key partition for storing these data. Since the Cassandra is one of the best column base databases, writing on this database is better than other technologies. Also, Cassandra read operation is faster than other technologies due to key structure and sorting data according to time.

Conclusion

In this paper we proposed a framework to analyze smart grid big data. At first a comprehensive architecture for evaluation of smart grid big data is proposed. Then we introduce a scalable data generator to overcome to lack of access to real smart grid big data. Since there are different big data technologies at this time, we selected four of the best open source big data NoSQL databases and integrated them in the architecture of our large scale smart grid data generator. Based on our implementation, we evaluated the performance of Elasticsearch, MongDB, Hbase and Cassandra in terms of different criteria. Our results show that among these database technologies, Cassandra has the best overall performance and satisfies all requirements for smart grid bid data analysis. In the future, we plan to perform cyber security analysis using smart grid big data based on Cassandra. Also we are going to extend our data generator to generate data related to electricity theft and crypto-currency mining.

Acknowledgements
Not applicable.

Authors' contributions
MHA performed the study, developed the models and implemented it, performed the analysis and drafted the manuscript. VT is a leader of research, conducted the research and approved the results. BB worked with MHA and VT to develop the articles framework and interpreted results. All authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests
The authors declare that they have no competing interests.

Author details
[1] Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran. [2] Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran.

Received: 25 September 2019 Accepted: 14 November 2019
Published online: 03 December 2019

References

1. Ansari MH, Vakili VT, Bahrak B, Tavassoli P. Graph theoretical defense mechanisms against false data injection attacks in smart grids. J Mod Power Syst Clean Energy. 2018;6(5):860–71.
2. Ye F, Qian Y, Hu RQ. Big data analytics and cloud computing in the smart grid. Smart Grid Commun Infrastruct. 2018;171–185. https://doi.org/10.1002/9781119240136.ch8. http://doi.wiley.com/10.1002/9781119240136.ch8.
3. Zhou K, Fu C, Yang S. Big data driven smart energy management: from big data to big insights. Renew Sustain Energy Rev. 2016;56:215–25. https://doi.org/10.1016/j.rser.2015.11.050.
4. Vasilakos A, Hu J. Energy big data analytics and security: challenges and opportunities. IEEE Trans Smart Grid. 2016;PP(99):1. https://doi.org/10.1109/TSG.2016.2563461.
5. Munshi AA, Mohamed YA. Big data framework for analytics in smart grids. Electric Power Syst Res. 2017;151:369–80. https://doi.org/10.1016/j.epsr.2017.06.006.
6. Tu C, He X, Shuai Z, Jiang F. Big data issues in smart grid—a review. Renew Sustain Energy Rev. 2017;79(May):1099–107. https://doi.org/10.1016/j.rser.2017.05.134.
7. Liang F, Yu W, An D, Yang Q, Fu X, Zhao W. A survey on big data market: pricing trading and protection. IEEE Access. 2018;6:15132–54. https://doi.org/10.1109/ACCESS.2018.2806881.
8. Kang C, Wang Y, Xue Y, Mu G, Liao R. Big data analytics in china's electric power industry: modern information, communication technologies, and millions of smart meters. IEEE Power Energy Mag. 2018;16(3):54–65.
9. Wen L, Zhou K, Yang S, Li L. Compression of smart meter big data: a survey. Renew Sustain Energy Rev. 2018;91(April):59–69. https://doi.org/10.1016/j.rser.2018.03.088.
10. Munshi AA, Mohamed YARI. Data lake lambda architecture for smart grids big data analytics. IEEE Access. 2018;6:40463–71. https://doi.org/10.1109/ACCESS.2018.2858256.
11. Jiang R, Lu R, Choo KKR. Achieving high performance and privacy-preserving query over encrypted multidimensional big metering data. Future Gener Comput Syst. 2018;78:392–401. https://doi.org/10.1016/j.future.2016.05.005..
12. Chang YC, Huang TC. An interactive smart grid communication approach for big data traffic. Comput Electr Eng. 2018;67:170–81. https://doi.org/10.1016/j.compeleceng.2018.03.024.
13. Chin WI, Li W, Chen Hh. Energy Big Data Security Threats in IoT-Based Smart Grid Communications (October), 2017;70–75. https://ieeexplore-ieee-org.udel.idm.oclc.org/document/8067688/.
14. Hou W, Ning Z, Guo L, Zhang X. Temporal, functional and spatial big data computing framework for large-scale smart grid. IEEE Trans Emerg Topics Comput. 2017;6750(1):1–1. https://doi.org/10.1109/TETC.2017.2681113. http://ieeexplore.ieee.org/document/7876814/.
15. Junaidi N. Big Data Applications in Electric Energy Systems. In: 2018 international conference on computational approach in smart systems design and applications (ICASSDA). 2017; p. 1–5.
16. Shuai C, Yang H, Ouyang X, He M, Gong Z, Shu W. Analysis and identification of power blackout-sensitive users by using big data in the energy system. IEEE Access. 2018; PP(c):1–1 . https://doi.org/10.1109/ACCESS.2018.2886551. https://ieeexplore.ieee.org/document/8576506/.
17. Liu X, Nielsen PS. Scalable prediction-based online anomaly detection for smart meter data. Inf Syst. 2018;77:34–47. https://doi.org/10.1016/j.is.2018.05.007.
18. Moghaddass R, Wang J. A hierarchical framework for smart grid anomaly detection using large-scale smart meter data. IEEE Trans Smart Grid. 2018;9(6):5820–30. https://doi.org/10.1109/TSG.2017.2697440.
19. Vimalkumar K, Radhika N. A big data framework for intrusion detection in smart grids using Apache Spark. 2017 International conference on advances in computing, communications and informatics (ICACCI). IEEE, p. 198–204, 2017.
20. Singh S, Yassine A. Mining energy consumption behavior patterns for households in smart grid. IEEE Trans Emerg Topics Comput. 2017;6750(c):1–1 . https://doi.org/10.1109/TETC.2017.2692098. http://ieeexplore.ieee.org/document/7894203/.
21. Wu J, Ota K, Dong M, Li J, Wang H. Big data analysis based security situational awareness for smart grid. IEEE Trans Big Data 2016;7790(c):1–1 . https://doi.org/10.1109/TBDATA.2016.2616146. http://ieeexplore.ieee.org/document/7587350/.
22. Wang K, Xu C, Zhang Y, Guo S, Zomaya A. Robust big data analytics for electricity price forecasting in the smart grid. IEEE Trans Big Data 2017;7790(c):1–1 . https://doi.org/10.1109/TBDATA.2017.2723563. http://ieeexplore.ieee.org/document/7968454/.

23. Ye F, Qian Y, Hu RQ. A secure data learning scheme for big data applications in the smart grid. 2017.
24. Nguyen HK, Khodaei A, Han Z. A big data scale algorithm for optimal scheduling of integrated microgrids. IEEE Trans Smart Grid. 2018;9(1):274–82. https://doi.org/10.1109/TSG.2016.2550422.
25. Sheng G, Hou H, Jiang X, Chen Y. A novel association rule mining method of big data for power transformers state parameters based on probabilistic graph model. IEEE Trans Smart Grid. 2018;9(2):695–702. https://doi.org/10.1109/TSG.2016.2562123.
26. Liu X, Golab L, Golab W, Ilyas IF, Jin S. Smart meter data analytics. ACM Trans Database Syst. 2016;42(1):1–39. https://doi.org/10.1145/3004295. http://dl.acm.org/citation.cfm?doid=3015779.3004295.
27. Iftikhar N, Liu X, Danalachi S, Nordbjerg FE, Vollesen JH. A scalable smart meter data generator using spark. In: OTM Confederated International Conferences "On the move to meaningful internet systems". Cham: Springer; 2017. p. 21–36.
28. Electricity customer behaviour trial. http://www.ucd.ie/issda/data/commissionforenergyregulationcer/.
29. Iftikhar N, Liu X, Nordbjerg FE, Danalachi S. A Prediction-Based Smart Meter Data Generator. In: NBiS 2016—19th international conference on network-based information systems. 2016; p. 173–80. https://doi.org/10.1109/NBiS.2016.15.
30. Kafka, Walkenbach J. The definitive guide. 2010;. https://doi.org/10.1017/CBO9781107415324.004.
31. Harrison G. Next generation databases. 2015;. https://doi.org/10.1007/978-1-4842-1329-2. http://link.springer.com/10.1007/978-1-4842-1329-2.
32. Vaish G. Getting Started with NoSQL. Packt Publishing Ltd, 2013.
33. Hewitt E. Cassandra: the definitive guide, first. edit ed. Sebastopol: O'Reilly; 2011.
34. Kan C. Cassandra data modeling and analysis. Birmingham: Packt Publishing Ltd; 2014.
35. Kristina Chodorow and Michael Dirolf: MongoDB: The Definitive Guide (2010).
36. Gormley C, Tong Z. Elasticsearch: the definitive guide: a distributed real-time search and analytics engine. Sebastopol: O'Reilly Media, Inc.; 2015.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.