Analysis Big DATA Framework for Smart Meter

Korranat Siripakthanakan*

Received : September 22, 2020
Revised : November 10, 2020
Accepted : November 16, 2020

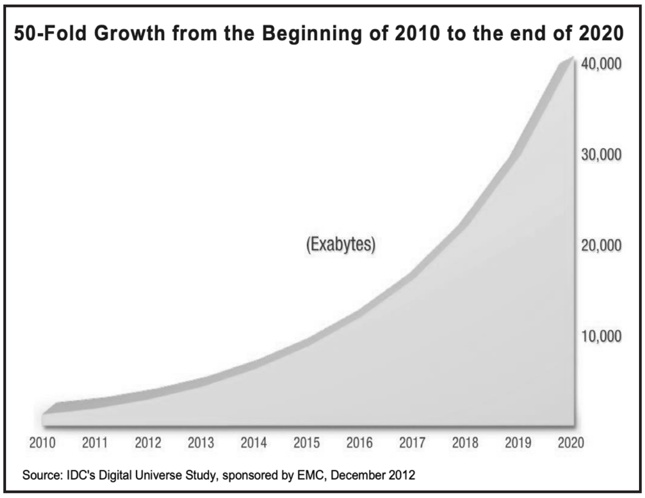
Abstract

A general meter has been replaced by a smart meter in usage to be able to automatically compile data on power usage. However, big data developed from a meter need to be systematically managed to enhance reliability, which is a challenge for big data showing both data characteristics, which require advanced information techniques and infrastructure to store and analyze big data. For this reason, this unprecedented amount of information needs a powerful platform. This paper presents a functional framework that can be a start for further research in focusing on comparing elements of the data processing framework that require processing speed and reliability as an open source platform.

Keywords: Big Data, Smart Meter, Framework, Data Processing, Opensource.

1. Introduction

From the estimate, global data volumes from 2010 to 2020 [1] are expected to grow 300 times from 130 exabytes (EB: Exabytes (1018)) to 40,000 exabytes as shown in Figure 1.



Source: IDC's Digital Universe Study, sponsored by EMC, December 2012

Figure 1. *50-Fold Growth from the Beginning of 2010 to the end of 2020 [1].*

Due to an increasing amount of data in every industry, especially those involved in alternative or renewable energy, solar cell energy that data is derived from a smart meter received from the promotion from the government in a concrete manner, which can be seen that Energy Policy and Planning Office, Ministry of Energy has developed a Alternative Energy Development Plan (AEDP 2015) [2] to strengthen sustainable energy in Thailand as a guideline to drive Thailand's alternative energy model in 2015-2036, resulting in a continuous increase in data rates. The rate increases exponentially. The information obtained is diverse, inconsistently fast, and complex. Then, data become specific information, called Big Data. Nowadays, big data cannot be immediately processed or displayed.

Big data are currently becoming the focus of technology in science and engineering. Big data structures consist of processes, data collection, data storage, data sort, data processing, and data analysis. However, the design and implementation of big data structures are beyond the capabilities of today's commonly used hardware and software, which are currently used to compute data.

The alternative energy industry is interested in bringing technology "a smart meter" to become a part of the industry in Thailand. In April 2020 renewable energy power plants in Thailand, shown in Figure 2. Data on the number of alternative power plants which have supplied power to the system (COD: Commercial Operation Date) in 2016 included 968 power plants and, in 2020, there are 1,140 power plants [3].

In the past, the tools used in the analysis and in the process showed reports regarding energy consumption to a certain extent. With the increasing data rates, a meter system that can read the data every few minutes is shifting to millions of readings per hour. The end result is a vast increase of data. If the resulting data were managed properly and efficiently, they

* *Department of Information Technology Management, Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok.*

can help to systematically understand the behavior of the industry through a smart meter.

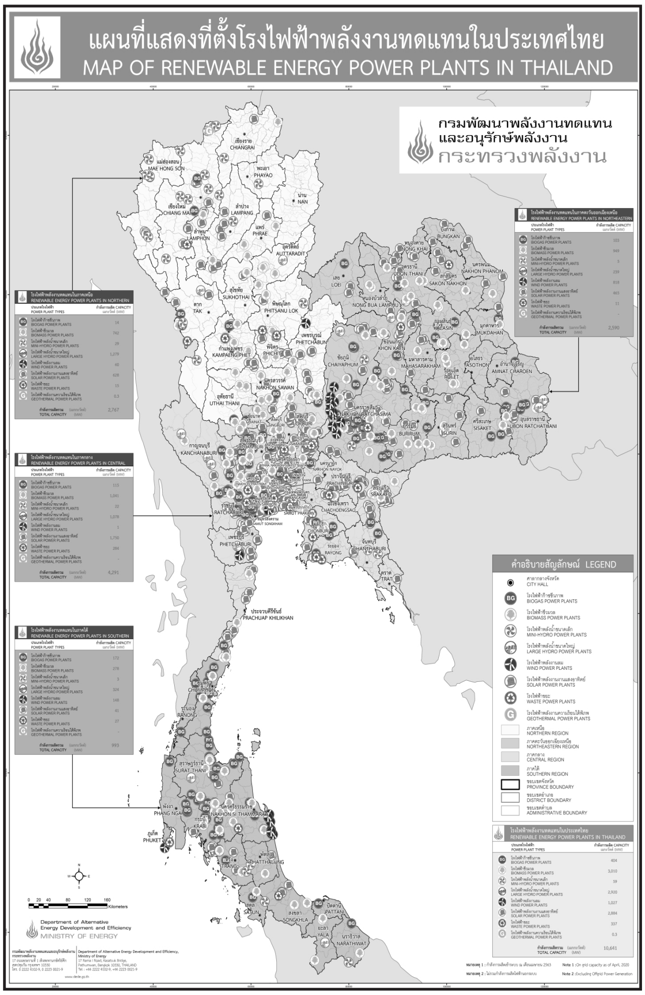

แผนที่แสดงที่ตั้งโรงไฟฟ้าพลังงานทดแทนในประเทศไทย
MAP OF RENEWABLE ENERGY POWER PLANTS IN THAILAND
กรมพัฒนาพลังงานทดแทน
และอนุรักษ์พลังงาน
กระทรวงพลังงาน
คำอธิบายสัญลักษณ์ LEGEND
Department of Alternative Energy Development and Efficiency
MINISTRY OF ENERGY

Figure 2. Map of Renewable Energy Power Plants in Thailand 2020 [4].

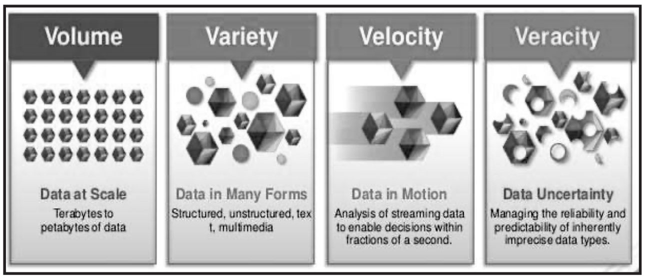

Volume
Data at Scale
Terabytes to petabytes of data
Variety
Data in Many Forms
Structured, unstructured, text, multimedia
Velocity
Data in Motion
Analysis of streaming data to enable decisions within fractions of a second.
Veracity
Data Uncertainty
Managing the reliability and predictability of inherently imprecise data types.

Figure 3. The Era of Big Data Demands [6].

2. Related Literature

2.1 Big Data

The definition of large data, which general software or hardware cannot be used to manage or analyze efficiently as shown in Figure 3.

2.1.1 The Characteristics of Big Data in 4Vs Era [5].

1) Volume: a large volume of data is obtained through business operations, such as data generated from customer activities, the activity of exchanging corporate data in online and offline formats, often greater than terabytes.

2) Variety: various data in texts, videos, images, and a variety of sources, such as the Omni Channel, can be both structured and unstructured.

3) Velocity: data are rapidly changing over time and continuously transmitted in streaming, making manual data analysis very limited to capture patterns or directions of data.

4) Veracity: big data require speed of use and are highly diverse. Therefore, the data itself contains veracity, which may be caused by various errors during data creation or data outside the box. Before entering into the model, data must be cleaned up again to ensure that the data, which are the model, is under a practical framework.

2.1.2 New era of Big Data

In the new era of Big Data as shown in Figure 4, more features are being defined, which increase the characteristics as 6Vs.

1) Visibility: big data can be visible and safe.

2) Value: data are useful and relevant in business. It must be understood first that not all data are useful for collection and analysis. Useful data must be relevant to the business purpose. For example, if you want to increase the competitiveness in the market of the products you sell, the

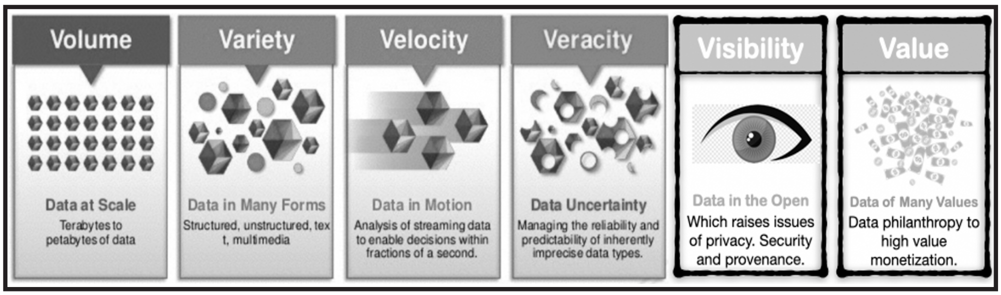

Volume
Data at Scale
Terabytes to petabytes of data
Variety
Data in Many Forms
Structured, unstructured, text, multimedia
Velocity
Data in Motion
Analysis of streaming data to enable decisions within fractions of a second.
Veracity
Data Uncertainty
Managing the reliability and predictability of inherently imprecise data types.
Visibility
Data in the Open
Which raises issues of privacy. Security and provenance.
Value
Data of Many Values
Data philanthropy to high value monetization.

Figure 4. Transformation Big Data & Analytics.

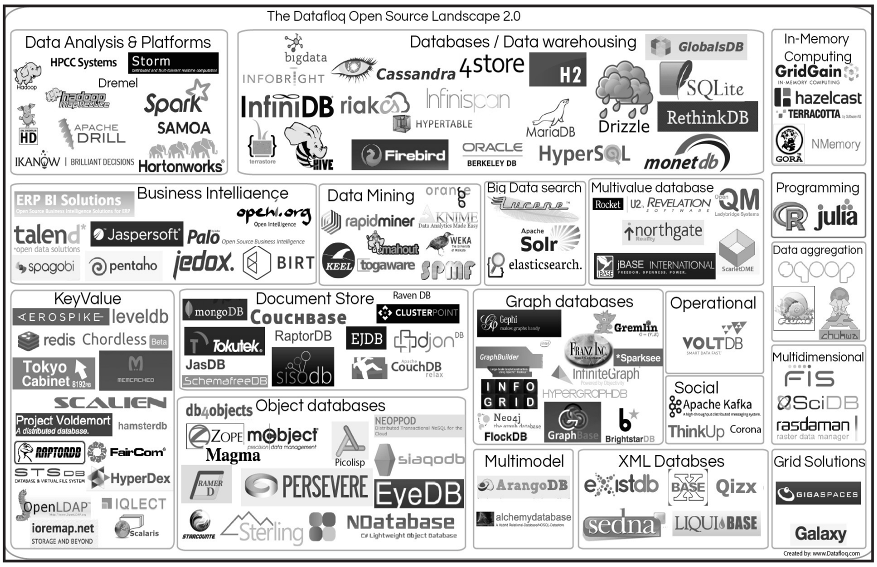
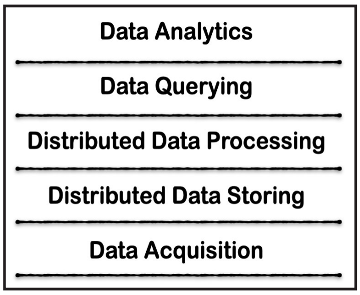


Figure 5. The Big Data Open Source Tools [7].



Data Analytics

Data Querying

Distributed Data Processing

Distributed Data Storing

Data Acquisition

Figure 6. Hierarchical architecture of the core components big data.

most useful data are probably competitor product data.

2.2 Big Data Components (Opensource)

There is currently a wide variety of tools which support big data processing, and those can be used free of charge, this articles education open source core components tools used in framework and easy and cost-effective and most popular on framework's stages including data management using Apache Hadoop platform as shown in Figure 5.

2.3 Big Data Framework

Big Data Framework is the main components of Big Data used for a smart meter [8] by tools which can be used to process and analyze data including data acquisition, data storing, data processing, data querying and data analysis using opensource tools as shown in Figure 6.

2.3.1 Data Acquisition Component

The data acquisition components [9], [10] of big data are the process of data acquisition, which collect digital data for further storage and analysis. The process consists of 3 steps: Data Collection, Transmission and Data Processing as shown in Figure 7, without definite sequence during transmission and data processing. Therefore, data processing steps may take place before data are transmitted and/or after transmission.

Data Collection

Data collection refers to the process collecting raw data from data source. This process must be designed well. Otherwise, incorrect data collection will affect the subsequent analysis process and eventually lead to inaccurate results.

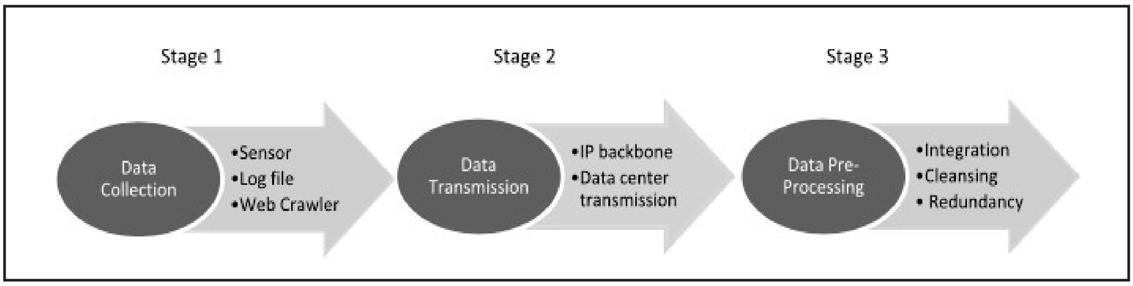


Figure 7. The data acquisition stage consists of three-tasks: collection, transmission and pre-processing [9].

Data Transmission

After collecting raw data from the data collection process, the data need to be transferred to the storage infrastructure at the data center waiting for further processing.

Data Pre-processing

From a variety of sources, the level of data may be different in quality, such as redundancy, consistency of data, etc. Therefore, this data process is designed using high techniques to improve the quality of data obtained from previous processes, such as integration and cleansing techniques.

1) Apache Flume

Flume [11] was originally developed by Cloudera before it was transferred to Apache Community Version 13 and is now renamed to Flume NG (Next Generation) where the Flume feature is designed to run in diffusion for accessing real-time data collection or streaming from multiple web servers, temporary storage and a delivery to the destination of the processing needs. Also, Flume is recognized as a reliable tool and can be configured according to requirements. The data are sent to HDFS, and, technically, Flume agent creates a channel to connect the source to the target via Flume.

• The Source: Flume functions as collecting data from various sources.

• The Flume Channel: a Buffer collects information before being used in general memory.

• The Flume Target: can be used from Flume Channel to HDFS.

2) Kafka

Kafka [12] is an open source stream processing software platform developed by the Apache Software Foundation written in Scala and Java. The project aims to provide a platform that can handle with high volumes of data with low computing times. For managing data in real time, it can be connected to external systems (for data import / export) via Kafka Connect and Kafka Streams [13] which is a Java binary stream processor. Kafka can also use the binary TCP protocol which has been optimized for better performance. If you are interested in finding out more about Kafka, please look at Ref [12-16].

2.3.2 Distributed Data Storing Component

The data were collected from many nodes [17], [18] and are scattered in databases on various computer networks from the center that is not designed according to the principles to a database principle for quick access to information, such as Google's Big Table, Amazon Dynamo, Windows Azure Storage. Increased storage is required. In term of high-speed read and write access, the design must always take into account availability as well.

1) Hadoop’s Distributed File System (HDFS) [19], [20] is an open source software platform which supports storing and processing big data. It is proprietary to store and process data, enabling distributed big data processing on server clusters. Hadoop's core consists of two main components: storage components and processing components, which are important parts of architecture in big data [19-23].

2.3.3 Distributed Data Processing Component

The definition of data processing is to have good performance, to respond quickly. Also, it must be reliable, ready to use, low cost, highly-flexible and easy to customize.

1) MapReduce

MapReduce [24], [25] is the Hadoop processing component of the Job Tracker and Task Tracker per cluster node. The main unit is responsible for scheduling tasks for secondary tasks and is responsible for reviewing and processing the work again in the event that the work fails.

2) YARN

YARN Hadoop has developed YARN [26], [27] (Yet Another Resource Negotiator) to improve performance efficiency to be better than MapReduce or called MapReduce 2. YARN is a resource manager for Hadoop which opens applications from other computing frameworks to run on Hadoop clusters in a distributed manner. It also initiates the process for the data streaming application.

Basic idea of YARN is to separate MapReduce's two major Job Tracker / Task Tracker responsibilities into separate entities. Fundamentally, YARN consists of the Resource Manager and Node Manager for application management in a distributed manner. Resource Manager is the highest authority that decides the use of resources among all applications in a cluster of servers. The Resource Manager has a scheduler that allocate resources to applications running in the cluster. Unlike Map Reduce, this architecture allows an unlimited number of nodes to be added to the cluster all the time without affecting the functionality of the existing cluster.

2.3.4 Data Querying Components

Hive and Impala are high-level languages like SQL that represents big data analysis tasks, facilitating the query and manipulation of large data in distributed storage.

1) Hive

Hive [28] facilitates to read, write and manage data stored in HDFS using a SQL-like interface in the framework to read smart grid data files from HDFS, and to create table of content. To use this task, a specific table with a smart meter and smart meter usage [29] must be created.

2) Impala

Impala [30] is a real-time interactive SQL query engine on big data. Impala query calls are performed on Parallel in memory of each cluster node. The intermediate results of the node are sent and combined and then returned. As a result, Impala queries are able to produce results in near real time.

Impala [31] is specially targeted for integration with smart and standardized environments, which support the most relevant standards, connecting via ODBC or JDBC; authenticated with Kerberos or LDAP.

2.3.5 Data Analysis Components

1) Mahout [32], [33], [34] is a dataset installed on Hadoop and processed in a group and contains various core algorithms for robust applications. Apache Mahout, which is an open source, was developed by the Apache Software Foundation to create a scalable machine learning algorithm that uses three factor techniques: guidance, classification and clustering. In addition, using Apache Hadoop allows Mahout to be scalable efficiently.

2) SAMOA (Scalable Advanced Massive Online Analysis) [35],[36],[37] is a distributed streaming framework that is programmed for distributed streaming algorithms for the most common work, data and machine learning.

3) Tableau is an interactive data visualization tool that allows users to analyze, visualize and share information in a form of a dashboard, including effectively studying and understanding large volumes of data with limited human and financial resources, worthing an ability to interact smoothly with data.

3. Hierarchical Architecture

Hierarchical architecture of opensource core components for smart grid Big Data with popular on framework's stages using Apache Hadoop platform. Represent this focus stage components distributed data processing comparison between Hadoop and spark of hierarchical architecture as shown in Figure 8.

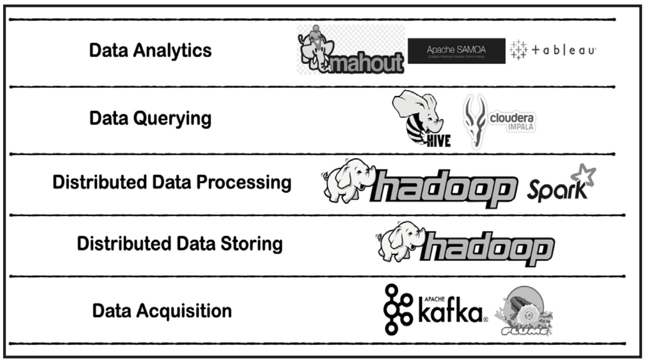


Figure 8. A hierarchical architecture of the core components for smart grid big data.

3.1 Core Components for Smart Grid

Big Data Framework with energy industry to smart meter as shown in Figure 9. Smart meter's big data management extends from the beginning to data analysis. Please look ref [8].

3.1.1 Data Generation

Data from thousands of smart meters come from the source such as alternative energy industry.

3.1.2 Data Acquisition

The smart meter's data acquisition can be divided into

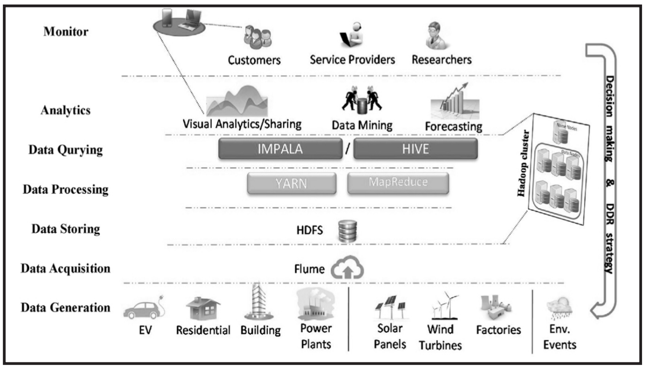


Figure 9. *The framework to smart grid big data [8].*

3 parts: data collection, transmission and pre-data processing. Once the data is collected, the data will be sent to the storage infrastructure for subsequent processing. Due to a wide variety of sources of data, the data may be different. Data integration techniques aim to combine data from different sources and provide a unified view of the data. In this framework, the data are transferred to a comma-separated value file (CSV format). The data attributes include time, meter code, consumption rate and location. Additionally, in pre-data processing, inaccurate and incomplete information will be corrected or removed to improve the quality of the data.

3.1.3 Data Storing and Processing

After receiving data from a smart meter, at this stage, Hadoop's HDFS manages the storage for further processing. The HDFS cluster consists of a Name Node, and Data Node. The data received are divided into one or more blocks, and these blocks are stored in a set of Hadoop Data Nodes in computation for the analysis of big data by HDFS.

3.1.4 Data Querying

Hive and Impala use this framework to read data from HDFS and choose to analyze or generate data of interest. For example, electricity consumption can be obtained in certain areas. The retrieval process runs at the top of the Hadoop cluster, enabling efficient results.

3.1.5 Data Analysis

The data received and optimized will be able to use big data tool through Mahout SAMOA and continue to use Tableau to build a complete Dashboard.

4. Education

Performance comparison between Hadoop and Spark framework using HiBench benchmark suite. The focus of smart grids frameworks lies in the distributed data processing using Hadoop and Spark in processing with standardized comparison to test performance. In this education execution result including execution time, throughput and speedup for all data load. Using nine benchmarks of HiBench suite including.

- Aggregate input data size is 12 x104 Pages
- Bayesian input data size is 105 pages
- Join input data size is 12 x105 pages
- Pagerank input data size is 5,000 pages
- Scan input data size is 183 MB
- Sleep input data size is 0
- Sort input data size is 328 MB
- Tera sort input data size is 320 MB
- Wordcount size of Data set 1GB, 5GB and 10GB

For benchmarks of workload [38].

4.1 Execution Time

Execution time meaning period of time in which an event is active. The comparison of the execution time Spark shows less execution time as compared to Hadoop, Pagerank gives better execute time of more than 90%, Join and Aggregate gives second to third best between 80-70% of the execution time but Spark's Sleep workload execute time very slow to Hadoop as show in Figure 10. Finally, for Wordcount workload input data [1,5,10 GB] Spark show less execution time as compared to Hadoop [38] as shown in Figure 11.

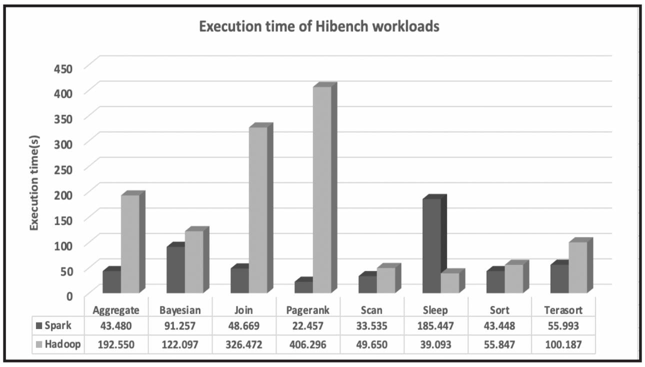


Figure 10. *Execution time of HiBench workloads [38].*

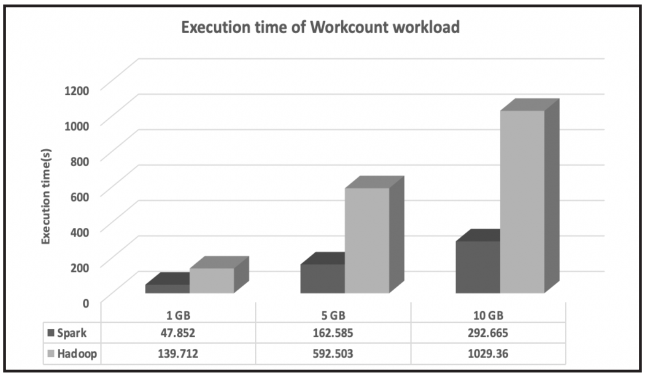


Figure 11. *Execution time of Wordcount workload [38].*

4.2 Throughput

Throughput is the ratio of input data size over execution time. Spark show higher performance of throughput as Hadoop with Jone workload as gives better throughput from eight benchmarks as shown in Figure 12 and Figure 13 Wordcount workload in different data sizes increase in throughput when data size from 1 GB, 5GB, 10 GB throughput remains stable for Spark and Hadoop [38].

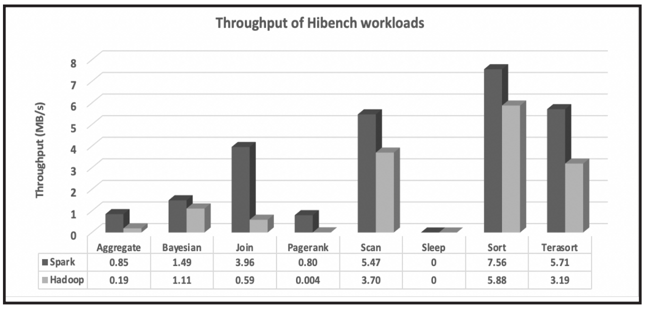


Figure 12. *Throughput of Hibench workloads [38].*

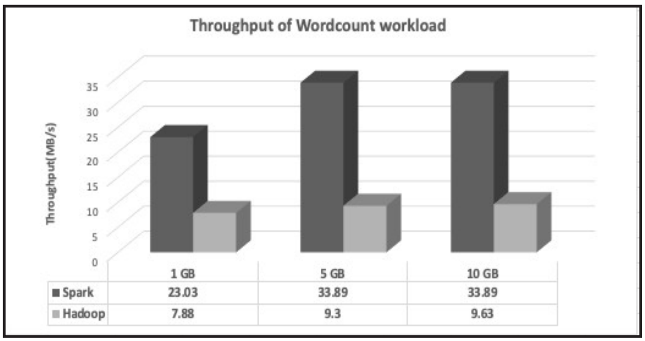
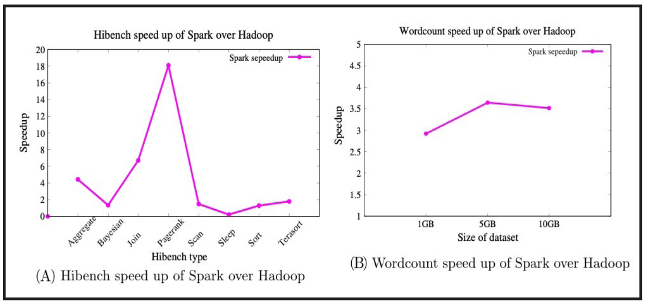


Figure 13. *Throughput of Wordcount workload [38].*

4.3 Speedup

Speedup is the ratio of the performance improvement on different input sizes between two systems that process the same problem. Spark's speedup over Hadoop on eight workloads with same data sizes. Pagerank gives better speedup and best Join and Aggregation workloads. Spark gives performance than Hadoop with maximum speedup to 3.64 times with 5GB in shown Figure 14A and 14B. Please look at ref [38].



(A) Hibench speed up of Spark over Hadoop

(B) Wordcount speed up of Spark over Hadoop

Figure 14. *Speedup of Spark over Hadoop. (A) Hibench speedup of Spark over Hadoop; (B) Wordcount speedup of Spark over Hadoop [38].*

5. Conclusions

This article presents a framework for the alternative energy industry that introduced a smart meter technology to measure energy efficiency. The concept of big data and its core framework elements focuses on an open source format of utilizing opensource tools and providing and cost-effective to be used in tool performance experiments, framework procedures, including data acquisition, storage, data processing, data search and data analysis via tools.

Regarding the implication of tools with big data in the data processing stage, the framework also needs the most instantaneous processing speed with effective results. Because Hadoop is batch processing the results are very efficient but a lack of processing speed. Regarding a Spark as a steam processing, the results are less efficient but faster processing.

Hence, there should be an idea and investigate of combining the computation in this component into a hybrid processing or eco processing with good performance in terms of reliability, quality, execution, throughput and speedup. They may be tested against a standardized agency such as HI Beach Benchmark to help the test.

6. References

[1] EMC Corporation. *Executive summary: A universe of opportunities and challenges*. Available Online at http://www.emc.com/leadership/digital-universe/2012iview/executive-summary-a-universe-of.htm, accessed on 1 August 2020.

[2] Ministry of Energy in Thailand: *Alternative Energy Development plan: AEDP2015*. Available Online at https://www.dede.go.th, accesses on 1 August 2020.

[3] Energy Regulatory Commission. Available Online at http://www.erc.or.th/ERCSPP. accessed on 1 August 2020.

[4] Department of Alternative Energy Development and Efficiency. *Renewable energy map on April 2020*. accessed on 1 August 2020.

[5] D. Corrigan. *IBM Building Confidence in Big Data*. Smart Business 2013.

[6] IBM. *The Era of Big Data Demands Confidence 2013*. Available online at https://www.slideshare.net/ibmsverige/building-confidence-in-big-data, accessed on 1 August 2020.

[7] The Big Data Open Source Tools Landscape. Available online at https://datafloq.com/big-data-open-source-tools/os-home/, accessed on 1 August 2020.

[8] A. Munshi, A. Yasser and R. Mohamed. “Big data framework for analysis in smart grids.” *Electric power systems research*, Vol. 151, pp. 369-380, 2017.

[9] H. Hu et al. “Toward Scalable Systems for Big Data Analysis: A Technology Tutorial.” *IEEE Access*, DOI:10.1109/Access.2014.2332453, 2014.

[10] M. Di Paolo Emillo. “Data Acquisition Systems.” *Springer New York Heidelberg Dordrecht London*, DOI: 10.1007/978-1-4614-4212-1, 2013.

[11] M. Birjali, A. Beni-Hssane and M. Erritali. “Analyzing Social Media through Big Data using Info Sphere Big Insights and Apache Flume.” *The 8th International Conference on Emerging Ubiquitous Systems and Pervasive Network*, Lund, Sweden, pp. 280-285, 2017.

[12] Kafka Steams. Available online at https://kafka.apache.org/documentation/streams. accessed on 1 August 2020.

[13] Kafka Ecosystem. Available online at https://cwiki.apache.org/confluence/display/KAFKA/Ecosystem, accessed on 1 August 2020.

[14] N. Garg. “Apache Kafka.” *Publishing by Packt Publishing Ltd.*, Birmingham, UK, October 2013.

[15] K. Me Me Thein. “Apache Kafka: Next Generation Distributed Messaging System.” *International Journal of Scientific Engineering and Technology Research*, Vol. 3, Issue 47, pp. 9478-9483, 2014.

[16] P. Le Noac, A. Costan and L. Bouge. “A Performance Evaluation of Apache Kafka in Support of Big Data Streaming Application.” *IEEE International Conference on Big Data 2017*, Boston, MA, USA, 2017.

[17] R. Mayer et al. “FogStore: Toward a Distributed Data Store for Fog Computing.” *IEEE Fog World Congress (FWC) 2017*, Santa Clara, CA, USA, 2017.

[18] Whitehead et al. *Distributed Data Store*.US2014/0181041, 2014.

[19] Apache Hadoop. Available online at http://hadoop.apache.org/hdfs/, accessed on 1 August 2020.

[20] The Hadoop Distributed File System: Architecture and Design. Available online at http://svn.apache.org/repos/asf/hadoop.,accessed on 1 August 2020.

[21] K. Shvachko et al. “The Hadoop Distributed File System.” *IEEE Access Sunnyvale*, California. USA, 2010.

[22] K. Karun and A. Chitharanjan. “A Review on Hadoop – HDFS Infrastructure Extensions.” *IEEE Conference on Information and Communication Technologies (ICT 2013)*, Thuckalay, Tamil Nadu, India, 2013.

[23] B. Ristevski, M. Stevanovska and B. Kostovski. “Hadoop as a Platform for Big Data Analysis in Healthcare and Medicine.” *Faculty of Information and Communication Technologies*, Republic of Macedonia, 2017.

[24] J. Dean and S. Ghemawat. “MapReduce: Simplified Data Processing on Large Clusters.” *Communication of ACM.*, Vol. 51, No. 1, 2012.

[25] T. Condie et al. “MapReduce Online.” *UC Berkeley*, 2010.

[26] A. Pramod Kulkarni and M. Khandewal. “Survey on Hadoop and Introduction to YARN.” *International Journal of Emerging Technology and Advanced Engineering*. Vol. 4, Issue 5, 2014.

[27] V. Kumar Vavilapalli et al. “Apache Hadoop YARN: Yet Another Resource Negotiator.” *SoCC’13*, California, USA, pp. 1-3, 2013.

[28] A. Thusoo et al. “Hive – A Petabyte Scale Data Warehouse Using Hadoop.” *IEEE access ICDE Conference*, 2010.

[29] Y. Huai et al. “Major Technical Advancements in Apache Hive.” *SIGMOD’14*, Showbird, UT, USA, DOI 10.1145/2588555.2595630, 2014.

[30] Impala: A Modern Open-source SQL Engine for Hadoop, Available online at https://2013.berlinbuzzwords.de/sites/2013.berlinbuzzwords.de/files/slides/Impala%20tech%20talk.pdf, accessed on 1 August 2020.

[31] M. Kornacker et al. “Impala: A Modern Open-source SQL Engine for Hadoop.” *7th Biennial Conference on Innovative Data Systems Research (CIDR’15)*, Asilomar, CA, USA, 2015.

[32] Introducing Apache Mahout. Available online at https://www.ibm.com/developerworks/library/j-mahout/, accessed on 1 August 2020.

[33] J. Prakash Verma, B. Patel, and A. Patel. “Big Data Analysis: Recommendation Systems with Hadoop Framework.” *IEEE International Conference on Computational Intelligence & Communication Technology*, 2015.

[34] U. Demirbaga and D. Nandan Jha. “Social Media Data Analysis using MapReduce Programming Model and Training a Tweet Classifier using Apache Mahout.” *IEEE Access.*, DOI 10.1109/SC2.2018.00024., 2018.

[35] G. De Francisci Morales., “SAMOA: A Platform for Mining Big Data Streams.” *WWW 2013 Companion*, 13-17 May, 2013.

[36] A. Bifet and G. De Francisci Morales. “Big Data Stream Learning with SAMOA.” *IEEE International Conference on Data Mining Workshop*, DOI 10.1109/ICDMW.2014.24, 2014.

[37] B. Rohit Prasad and S. Agarwal. “Critical parameter analysis of Vertical Hoeffding Tree for optimized performance using SAMOA.” *Indian Institute of Information Technology Allahabad*, India, DOI 10.1007/s13402-016-0513-3, 2016.

[38] Y. Samadi, M. Zbakh and C. Tadonki. “Performance comparison between Hadoop and Spark frameworks using HiBench benchmarks.” *Nation School of Computer Science and System Analysis*, Paris, France. DOI 10.1002/cpe.4367, 2017.