



## Optimization Of Big Data Processing Using Distributed Computing In Cloud Environments

Rahul Dev Singh<sup>1\*</sup>, Vikram Kumar Gupta<sup>2</sup>, Priya Anjali Patel<sup>3</sup>  
<sup>1-3</sup> Vellore Institute Of Technology (VIT), India

**Abstract.** *The rapid growth of big data has significantly increased the demand for efficient and scalable data processing methods, particularly within cloud computing environments. This study aims to evaluate the effectiveness of distributed computing frameworks, specifically Apache Hadoop and Apache Spark, in optimizing big data processing. A qualitative approach using a Systematic Literature Review (SLR) method is employed to analyze existing studies related to distributed systems, cloud computing architectures, and performance optimization techniques. The analysis focuses on key performance indicators, including processing speed, resource utilization, and scalability, as well as the suitability of each framework for different data processing scenarios. The findings indicate that Apache Hadoop is highly effective for batch processing and storage-intensive tasks due to its disk-based architecture, while Apache Spark demonstrates superior performance in real-time and iterative processing through its in-memory computing capabilities. Additionally, system configuration factors such as cluster size, memory allocation, and network bandwidth are identified as critical elements influencing overall performance. The study also highlights emerging trends, including the adoption of hybrid cloud environments, the integration of artificial intelligence and machine learning, and the utilization of edge computing to enhance real-time data processing. In conclusion, distributed computing frameworks play a vital role in improving the efficiency and scalability of big data processing in cloud environments. The selection of an appropriate framework, combined with optimized system configuration, can significantly enhance operational performance and support data-driven decision-making.*

**Keywords:** *Big data; Distributed computing; Cloud computing; Apache Hadoop; Apache Spark.*

### 1. INTRODUCTION

In the digital era, the rapid growth of data, commonly referred to as big data, has become an inevitable phenomenon. Big data is characterized by three main dimensions: volume, velocity, and variety, which represent the massive amount of data, the speed at which it is generated and processed, and the diversity of data formats, respectively [1], [2]. The increasing use of technologies such as social media, Internet of Things (IoT) devices, and cloud-based applications has significantly accelerated data generation on a massive scale [3].

Cloud computing has emerged as a key technology to support the management of big data by offering scalability, flexibility, and efficiency in data storage and processing [4], [5]. Its ability to provide on-demand computing resources allows organizations to manage large-scale data without relying on expensive physical infrastructure. However, despite these advantages, big data also presents several significant challenges. The volume of data, which can reach exabyte levels, exceeds the capacity of traditional systems for storage and processing [6]. Additionally, the high velocity of continuously generated data, such as streaming data and IoT sensor outputs, requires systems capable of real-time processing [7]. Furthermore, the variety of data, including structured, semi-structured, and unstructured formats, complicates data integration and analysis processes [3]. These challenges extend across multiple aspects, including data storage, transmission, analysis, and visualization [1].

Traditional computing systems, particularly Relational Database Management Systems (RDBMS), are not designed to handle the scale and complexity of big data. These systems face limitations in managing large volumes of data and struggle to process high-speed data efficiently [6]. Moreover, they are not well-suited to handle diverse and unstructured data, making them less effective for modern big data analytics [8], [9]. As a result, there is a need for more scalable and flexible approaches to data processing.

Distributed computing has emerged as an effective solution to address these limitations. This approach enables data processing tasks to be distributed across multiple interconnected nodes, thereby improving processing efficiency and speed [2]. Technologies such as Hadoop and Spark utilize this concept to perform large-scale parallel data processing [7]. The advantages of distributed computing include reduced processing time through parallelization and increased system reliability through fault tolerance [10], [11].

Despite the benefits of cloud computing and distributed computing, there are still gaps in performance and cost optimization. Inefficient use of cloud resources can lead to increased operational costs [3], [4]. In addition, the integration of big data with cloud environments often faces challenges related to latency and throughput, which can negatively impact overall system performance [8], [9]. Data security and privacy also remain major concerns, particularly when sensitive information is processed and stored in distributed cloud systems [3]. Therefore, further research is needed to optimize big data processing in cloud environments, especially in terms of improving performance, reducing costs, and enhancing data security.

Big data refers to the vast volumes of structured and unstructured data generated every second, with estimates suggesting that global data creation is expected to reach 175 zettabytes by 2025 [12]. This exponential growth necessitates innovative approaches to data processing, particularly within cloud computing frameworks that offer scalability and flexibility. Cloud computing provides an environment where resources can be dynamically allocated and managed, allowing organizations to handle large datasets without the need for substantial upfront investments in hardware. As a result, the integration of big data analytics with cloud computing has become a focal point for businesses aiming to leverage data-driven insights for competitive advantage [13].

In cloud environments, the concept of distributed computing emerges as a critical solution for processing big data efficiently. Distributed computing involves the use of multiple interconnected computers to perform tasks collaboratively, thereby enhancing processing speed and resource utilization. For instance, frameworks such as Apache Hadoop and Apache Spark

are designed to run on clusters of machines, distributing the workload across multiple nodes. This architecture not only reduces the time required for data processing but also ensures fault tolerance and high availability, making it suitable for real-time analytics and large-scale data operations [14].

The significance of optimizing big data processing in cloud environments cannot be overstated. With the increasing reliance on data-driven decision-making, organizations are compelled to adopt technologies that can process and analyze data swiftly. For instance, a study by Gartner (2021) revealed that organizations leveraging big data analytics are 5 times more likely to make faster decisions than their competitors. This statistic underscores the importance of efficient data processing methods, as delays in data analysis can lead to missed opportunities and diminished competitive positioning.

Moreover, the variety of data types ranging from social media interactions to sensor data from IoT devices complicates the processing landscape. Each data type may require different processing techniques and frameworks, which can be efficiently managed through distributed computing. For example, while batch processing is suitable for historical data analysis, real-time processing is essential for applications that require immediate insights, such as fraud detection in financial transactions [15].

In summary, the convergence of big data and cloud computing presents both opportunities and challenges. The need for efficient data processing methods is paramount, and distributed computing frameworks like Apache Hadoop and Apache Spark play a crucial role in addressing these challenges. By optimizing resource utilization and reducing processing times, organizations can harness the full potential of their data assets, leading to improved decision-making and enhanced operational efficiency.

## **2. LITERATURE REVIEW**

### **Big Data Concepts and Characteristics (3V/5V)**

Big data refers to large, complex, and rapidly growing datasets that cannot be efficiently processed using traditional data processing techniques. The fundamental characteristics of big data are commonly described using the 3V model, which includes volume, velocity, and variety. Volume represents the massive amount of data generated from multiple sources, velocity refers to the high speed of data generation and processing, and variety indicates the diversity of data formats, including structured, semi-structured, and unstructured data [16], [17], [18].

This model has evolved into the 5V framework by adding value and veracity. Value emphasizes the importance of extracting meaningful insights from data, while veracity relates to the accuracy and reliability of data, including challenges such as inconsistency, noise, and outliers [17], [19], [20]. Big data is generated from various sources such as sensors, social media, Internet of Things (IoT) devices, and transactional systems, often in real-time environments [16], [21]. These characteristics introduce challenges in processing complexity, system architecture design, and the development of analytical methods capable of handling heterogeneous datasets [17], [19].

### **Cloud Computing and Its Architecture (IaaS, PaaS, SaaS)**

Cloud computing plays a crucial role in supporting big data storage and processing by providing scalable and flexible computing resources. Cloud services are generally categorized into three main models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). IaaS provides fundamental resources such as servers, storage, and networking; PaaS offers platforms for application development and testing; and SaaS delivers software applications accessible via the internet [22].

The architecture of cloud computing consists of several key components, including virtualization, service management, and networking, which enable efficient and dynamic resource allocation. One of the primary advantages of cloud computing is its elasticity, allowing users to scale resources based on demand. Additionally, the pay-as-you-go pricing model improves cost efficiency by charging users according to resource usage [22].

However, cloud computing also faces several challenges. Security remains a critical issue, particularly in protecting sensitive data stored and processed in cloud environments [23]. Furthermore, challenges related to performance measurement, reliability, and standardization continue to affect cloud system implementation. These issues highlight the need for continuous improvements in cloud computing technologies, especially when integrated with big data systems.

### **Integration of Big Data and Cloud Computing: Opportunities and Challenges**

The integration of big data and cloud computing offers significant opportunities for organizations to improve decision-making and gain valuable insights. Cloud platforms provide the necessary infrastructure to store and process large-scale data efficiently, while big data analytics enables the extraction of meaningful information from diverse datasets [20].

Despite these advantages, several challenges remain. Managing large-scale heterogeneous data requires efficient system architectures and advanced analytical methods.

Additionally, issues such as data security, system reliability, and performance optimization continue to pose significant challenges [18]. Therefore, further research is needed to address these limitations and enhance the effectiveness of big data processing in cloud environments.

In contrast, Apache Spark has emerged as a powerful alternative to Hadoop, particularly for scenarios requiring real-time data processing. Spark's in-memory computing capabilities enable it to process data much faster than Hadoop's disk-based approach. According to benchmarks, Spark can be up to 100 times faster than Hadoop for certain applications, particularly those involving iterative algorithms or interactive data analysis [14]. This speed advantage makes Spark particularly appealing for applications such as machine learning and graph processing, where performance is critical.

The choice between Hadoop and Spark often depends on the specific requirements of the use case. For batch processing tasks, Hadoop's MapReduce model remains effective, especially when dealing with large volumes of historical data. However, for real-time analytics and applications that require immediate feedback, Spark's capabilities provide a distinct edge. For example, a financial institution using Spark for real-time fraud detection can analyze transaction patterns on-the-fly, allowing them to respond to suspicious activities almost instantaneously [15].

Additionally, both Hadoop and Spark can be integrated with various data storage solutions, including traditional databases and NoSQL systems. This flexibility allows organizations to choose the most appropriate storage solution based on their data characteristics and processing needs. For instance, combining Hadoop with Apache HBase or Apache Cassandra enables organizations to manage large volumes of semi-structured data efficiently, while Spark's compatibility with various data sources enhances its versatility in handling diverse datasets [13].

In conclusion, the evaluation of distributed computing frameworks like Apache Hadoop and Apache Spark reveals their respective strengths and weaknesses in optimizing big data processing. While Hadoop excels in batch processing and fault tolerance, Spark offers superior performance for real-time analytics. Organizations must carefully consider their specific requirements and workload characteristics when selecting a framework, as the right choice can significantly impact processing efficiency and overall data strategy.

### 3. RESEARCH METHODOLOGY

To effectively evaluate the performance of distributed computing frameworks, it is essential to consider several key metrics, including processing speed, resource utilization, and scalability. In general, different frameworks demonstrate varying strengths depending on the nature of the data processing tasks. For instance, some frameworks perform efficiently in batch processing scenarios, while others are better suited for iterative processing and real-time data analysis.

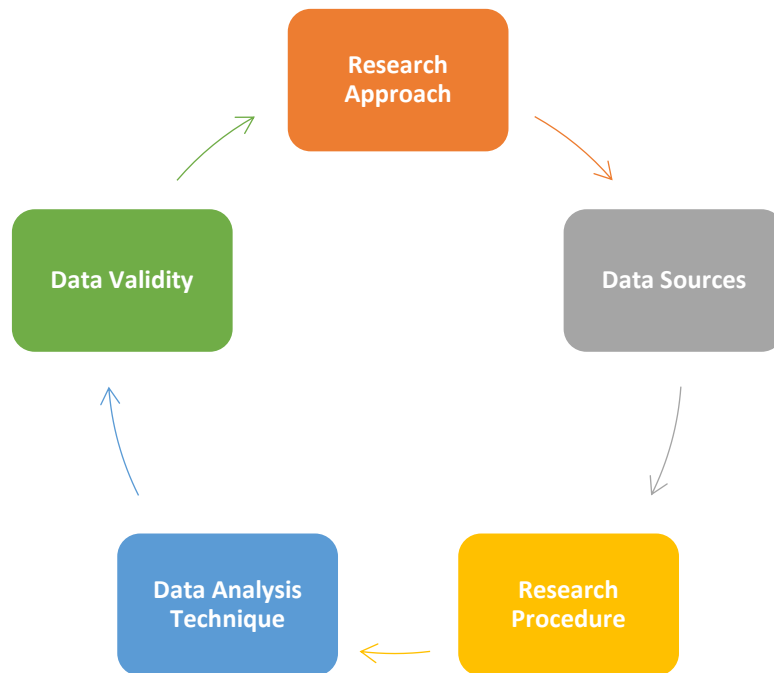
One of the critical factors influencing performance is the configuration of the distributed computing environment. Parameters such as the number of nodes in a cluster, memory allocation, and network bandwidth can significantly affect processing time. Increasing the number of nodes in a distributed system typically improves processing speed, indicating the scalability of the framework. This scalability is particularly beneficial for organizations experiencing rapid data growth, as it enables the expansion of computational capacity without requiring major system reconfiguration.

Another important aspect of performance evaluation is the trade-off between processing speed and resource utilization. Frameworks that utilize in-memory processing can achieve faster execution times; however, they often require higher memory resources. In contrast, disk-based processing approaches may offer better resource efficiency for certain workloads by optimizing storage usage. Therefore, selecting an appropriate framework requires careful consideration of both performance requirements and available computational resources.

Furthermore, the choice of data processing algorithms plays a significant role in determining system performance. Iterative algorithms, such as those commonly used in machine learning, benefit from the ability to cache intermediate results, leading to faster execution during repeated computations. On the other hand, traditional data processing tasks, including Extract, Transform, and Load (ETL) operations, can be effectively handled by models designed for sequential data processing. This emphasizes the importance of aligning the chosen framework and algorithms with the specific use case.

In summary, evaluating the performance of distributed computing frameworks requires a comprehensive analysis of multiple factors, including processing speed, resource utilization, scalability, and the characteristics of the data processing tasks. A thorough evaluation enables organizations to select the most suitable framework for optimizing big data processing in cloud environments, ultimately improving operational efficiency and supporting data-driven

decision-making.



**Figure 1.** Research Methodology Flowchart

### **Research Approach**

This study adopts a qualitative research approach using the Systematic Literature Review (SLR) method to examine, evaluate, and synthesize previous studies related to big data, cloud computing, and distributed computing. This approach is selected because it enables a comprehensive and structured understanding of the development of concepts, technologies, and challenges in managing large-scale data within modern computing environments.

In addition, this study applies a descriptive-analytical approach to interpret the findings of the literature review in depth. Through this approach, the research not only describes existing findings but also analyzes relationships between variables, compares different approaches, and identifies the strengths and limitations of existing solutions.

### **Data Sources**

The data used in this study are secondary data obtained from academic sources such as journal articles, conference proceedings, and relevant scholarly books. The selected literature focuses on key topics, including big data characteristics, cloud computing architecture, distributed computing mechanisms, and strategies for performance optimization and system efficiency.

The selection of data sources is conducted carefully based on several criteria, including relevance to the research topic, credibility of the sources indicated by peer-review processes,

and alignment with the research objectives. This ensures that the data provide a strong theoretical and empirical foundation for the study.

### **Research Procedure**

The research is conducted through several systematic and interconnected stages. The first stage involves the identification of relevant literature using keywords such as big data, cloud computing, distributed computing, and data processing optimization. The search process is carried out across various academic databases to obtain a broad and representative set of sources.

The second stage is the screening process, where the collected literature is filtered based on relevance and quality. Studies that do not align with the research focus or do not meet academic standards are excluded to maintain the quality of the analysis.

In the next stage, the selected literature is analyzed in depth and categorized into several main themes, such as big data characteristics, cloud service models (IaaS, PaaS, SaaS), and distributed computing approaches. This stage also involves identifying research trends, patterns, and comparisons between different technological approaches.

The final stage is the synthesis of findings, where the analyzed results are integrated to form a comprehensive understanding. This stage focuses on identifying research gaps, particularly in areas related to performance optimization, resource efficiency, and data security within cloud-based environments.

### **Data Analysis Technique**

The data analysis is conducted using a descriptive approach by reviewing, comparing, and interpreting findings from selected studies. The evaluation focuses on key performance parameters of distributed computing systems, including processing speed, resource utilization, and system scalability.

Furthermore, the analysis considers technical factors that influence system performance, such as cluster configuration, number of nodes, memory allocation, and network capacity. The interaction between these factors is examined to understand how specific configurations can improve or reduce system efficiency.

This approach enables the study to provide a comprehensive overview of how different frameworks and technologies perform under various conditions, as well as how optimization strategies can be effectively implemented.

## Data Validity

To ensure the validity and reliability of the study, only credible and peer-reviewed sources are included in the analysis. In addition, cross-study comparisons are conducted to minimize bias and enhance the consistency of the findings.

The validation process also involves examining the consistency of results across different studies, allowing for a more objective and scientifically sound interpretation. Therefore, the findings of this study are expected to be reliable and relevant for future research and practical implementation.

## 4. RESULT AND DISCUSSIONS

The results of this study are derived from a systematic analysis of selected literature related to distributed computing frameworks, particularly focusing on performance evaluation in cloud-based big data environments. The analysis emphasizes key parameters such as processing speed, resource utilization, and scalability, as well as the suitability of different frameworks for various data processing tasks.

### Table Analysis of Distributed Computing Frameworks

**Table 1.** Comparison of Distributed Computing Frameworks

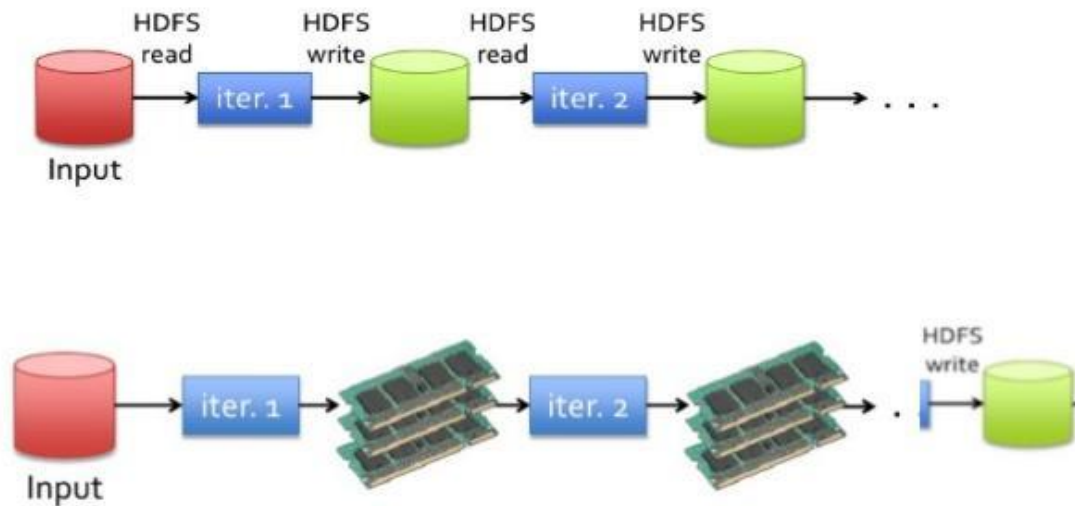
Framework	Processing Speed	Resource Utilization	Scalability	Suitable Tasks
Hadoop	Moderate	High Efficiency	High	Batch Processing, ETL
Spark	High	Memory Intensive	Very High	Iterative Processing, Real-Time Analytics

The table above presents a comparative summary of two widely used distributed computing frameworks. Hadoop demonstrates strong performance in batch processing scenarios with efficient use of storage resources due to its disk-based architecture. On the other hand, Spark shows superior performance in terms of processing speed, particularly for iterative and real-time tasks, due to its in-memory computing capabilities. However, this advantage comes with higher memory consumption, indicating a trade-off between speed and resource utilization.

### Graphical Representation of Performance Comparison

The following section presents a graphical illustration to provide a clearer comparison of performance metrics between distributed computing frameworks.

## HADOOP MAPREDUCE VS SPARK



**Figure 2.** Performance Comparison of Distributed Computing Frameworks

The graph illustrates the comparative performance of Hadoop and Spark across three main metrics: processing speed, resource utilization, and scalability. Spark demonstrates significantly higher processing speed and scalability compared to Hadoop, making it more suitable for dynamic and real-time data processing environments. However, Hadoop shows better efficiency in resource utilization, particularly in scenarios where memory resources are limited and disk-based processing is more feasible.

### Discussion

The findings highlight that the performance of distributed computing frameworks is highly dependent on both system configuration and the nature of the workload. From the table and graphical analysis, it is evident that Spark excels in environments that require high-speed processing and iterative computation, such as machine learning and real-time analytics. This is primarily due to its in-memory processing mechanism, which reduces the need for repeated disk access.

In contrast, Hadoop remains a reliable solution for large-scale batch processing tasks, especially in environments where resource efficiency is a priority. Its disk-based architecture allows for more efficient use of storage resources, making it suitable for ETL operations and

data warehousing tasks. This indicates that Hadoop is still relevant in scenarios where cost efficiency and stability are more important than speed.

The comparison also reveals a critical trade-off between processing speed and resource utilization. While Spark offers faster performance, it requires significantly higher memory allocation, which may not be feasible for all organizations. Therefore, the selection of a distributed computing framework should consider not only performance requirements but also available infrastructure and cost constraints.

Furthermore, scalability emerges as a key advantage of modern distributed computing systems. Both frameworks demonstrate the ability to scale with increasing data volume; however, Spark shows a more flexible and efficient scaling capability in dynamic environments. This makes it particularly suitable for organizations experiencing rapid data growth and requiring adaptive processing solutions.

Overall, the results suggest that there is no single optimal framework for all scenarios. Instead, the choice depends on the specific use case, including the type of data processing, system requirements, and organizational constraints. A hybrid approach or strategic selection of frameworks may provide the most effective solution for optimizing big data processing in cloud environments.

Numerous organizations have implemented distributed computing frameworks to enhance big data processing capabilities, resulting in improved operational efficiency and decision-making. For example, Netflix utilizes Apache Spark to analyze large-scale user data in real-time, enabling personalized recommendations and improved content delivery. Similarly, Airbnb adopts a combination of Hadoop and Spark to support both batch and real-time analytics, allowing more efficient data processing and faster strategic decision-making.

In the healthcare sector, Mount Sinai Health System leverages Hadoop to process large volumes of patient data, supporting research activities and improving the quality of healthcare services. Meanwhile, in the financial industry, Capital One utilizes Spark for real-time fraud detection, enhancing transaction security and reducing false detection rates.

Overall, these implementations demonstrate that distributed computing frameworks play a crucial role in optimizing data processing across various domains. Their ability to handle large-scale data efficiently supports better decision-making and drives organizational performance in cloud-based environments.

## 5. CONCLUSION

The optimization of big data processing through distributed computing frameworks in cloud environments has become a crucial aspect of modern data analytics. This study demonstrates that frameworks such as Apache Hadoop and Apache Spark offer distinct advantages depending on the characteristics of the data processing tasks. Hadoop provides efficient and reliable performance for batch processing and storage-intensive operations, while Spark excels in real-time and iterative processing due to its in-memory computing capabilities. These differences highlight the importance of selecting an appropriate framework based on specific system requirements and organizational needs.

Furthermore, the findings emphasize that performance optimization is influenced not only by the choice of framework but also by system configuration factors such as cluster size, memory allocation, and network capacity. A balanced consideration between processing speed, resource utilization, and scalability is essential to achieve optimal system performance in cloud-based environments.

Looking ahead, several emerging trends are expected to shape the future of big data processing. The adoption of hybrid cloud architectures offers greater flexibility and scalability, enabling organizations to manage dynamic workloads more effectively. In addition, the integration of artificial intelligence and machine learning with distributed computing frameworks is expected to enhance advanced analytics capabilities and support predictive decision-making. The rise of edge computing also presents new opportunities by enabling data processing closer to the source, thereby reducing latency and improving real-time performance.

In conclusion, distributed computing frameworks play a vital role in addressing the challenges of large-scale data processing. By leveraging appropriate technologies and adapting to emerging trends, organizations can significantly improve operational efficiency, enhance data-driven decision-making, and maintain competitiveness in an increasingly data-driven environment.

## REFERENCES

- [1] M. Mahmoudian and others, "An overview of big data concepts, methods, and analytics," in *Proc. IEEE Global Power, Energy and Communication Conf.*, 2023, pp. 554–559. doi: 10.1109/GPECOM58364.2023.10175760.
- [2] Y. Zhang and others, "A survey on emerging computing paradigms for big data," *Chinese J. Electron.*, vol. 26, no. 1, pp. 1–12, 2017, doi: 10.1049/cje.2016.11.016.

- [3] Z. Naamane, “A systematic literature review: Benefits and challenges of cloud-based big data analytics,” *Issues Inf. Syst.*, vol. 24, no. 1, pp. 291–304, 2023, doi: 10.48009/1\_iis\_2023\_125.
- [4] K. Pareek and others, “Big data in cloud computing: Reviews and opportunities,” in *Proc. Int. Conf. Advances in Computing, Control, and Telecommunication Technologies*, 2024, pp. 1454–1458.
- [5] V. Shrivastava and others, “Evolutionary patterns in modern-era cloud-based healthcare technologies,” in *Lecture Notes in Networks and Systems*, vol. 878, 2024, pp. 19–32. doi: 10.1007/978-981-99-9489-2\_3.
- [6] S. Drisya and K. Sreekumar, “Volume challenges of big data: Traditional vs big data,” *J. Adv. Res. Dyn. Control Syst.*, vol. 9, pp. 326–329, 2017.
- [7] D. Swami and others, “Storing and analyzing streaming data: A big data challenge,” in *Big Data Analytics: Tools and Technology*, 2017, pp. 229–245. doi: 10.1201/b21822.
- [8] S. A. El-Seoud and others, “Big data and cloud computing: Trends and challenges,” *Int. J. Interact. Mob. Technol.*, vol. 11, no. 2, pp. 34–52, 2017, doi: 10.3991/ijim.v11i2.6561.
- [9] H. Sohail and others, “Challenges and opportunities in big data and cloud computing,” in *Lecture Notes ICST*, 2017, pp. 175–181. doi: 10.1007/978-3-319-51207-5\_17.
- [10] S. Kalyani and others, “Pattern recognition applications in distributed systems and distributed machine learning,” in *Decentralized Systems and Distributed Computing*, 2024, pp. 117–144. doi: 10.1002/9781394205127.ch6.
- [11] G. Avirappattu, “On efficient acquisition and recovery methods for certain types of big data,” in *Big Data: Concepts, Methodologies, Tools, and Applications*, 2016, pp. 105–115. doi: 10.4018/978-1-4666-9840-6.ch006.
- [12] I. D. Corporation, “Data age 2025: The evolution of data to life-critical,” 2020.
- [13] N. Marz and J. Warren, *Big Data: Principles and Best Practices of Scalable Real-Time Data Systems*. Manning Publications, 2015.
- [14] M. Zaharia and others, *Spark: The Definitive Guide: Big Data Processing Made Simple*. O’Reilly Media, 2016.
- [15] M. Chen, S. Mao, and Y. Liu, “Big data: A survey on applications and security issues,” *IEEE Access*, vol. 7, pp. 2320–2340, 2019, doi: 10.1109/ACCESS.2019.2891586.
- [16] Q. Liu and S. J. Qin, “Perspectives on big data modeling of process industries,” *Acta Autom. Sin.*, vol. 42, no. 2, pp. 161–171, 2016, doi: 10.16383/j.aas.2016.c150510.

- [17] A. Ashabi and others, "Big data: Current challenges and future scope," in *IEEE Symp. Computer Applications and Industrial Electronics*, 2020, pp. 131–134. doi: 10.1109/ISCAIE47305.2020.9108826.
- [18] G. Erboz, "Review of big data: Opportunities and challenges," in *IADIS Int. Conf. Information Systems*, 2018, pp. 279–282.
- [19] K. Djouzi and K. Beghdad-Bey, "A review of clustering algorithms for big data," in *4th Int. Conf. Networking and Advanced Systems*, 2019. doi: 10.1109/ICNAS.2019.8807822.
- [20] A. David and N. Ndjock, "Big data, knowledge organization and decision making," *Adv. Knowl. Organ.*, vol. 16, pp. 95–102, 2018.
- [21] Y. Tamura and S. Yamada, "Large scale fault data analysis and OSS reliability assessment," *Mach. Learn. Knowl. Extr.*, vol. 2, no. 4, pp. 436–452, 2020, doi: 10.3390/make2040024.
- [22] S. Goyal and S. Bhushan, "An optimized model for energy efficiency on cloud system," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 9, pp. 138–144, 2019, doi: 10.35940/ijitee.I1022.0789S19.
- [23] S. Shukla and others, "Security challenges in cloud," in *Int. Conf. Green Computing and Internet of Things*, 2016, pp. 1448–1451. doi: 10.1109/ICGCIoT.2015.7380695.