ISSN: 2754-6659

Journal of Artificial Intelligence & Cloud Computing

Conference Proceedings

Open Access

Optimizing Database Management Systems for Big Data Applications: Techniques and Challenges (Approved by ICITET 2024)

Srinivasa Rao Maka[1]*, Suneel Babu Boppana[2], Gangadhar Sadaram[3], Laxmana Murthy Karaka[4], Vasu Velaga[5] and Janardhana Rao Sunkara[6]

[1]North Star Group Inc, Software Engineer, USA

[2]iSite Technologies, Project Manager, USA

[3]Bank of America, VP DevOps/ OpenShift Admin Engineer, USA

[4]Code Ace Solutions Inc, Software Engineer, USA

[5]Cintas Corporation, SAP Functional Analyst, USA

[6]AXS Group LLC, , Senior Database Engineer, USA

ABSTRACT

The extraordinary growth in data-driven critical changes how Database Management Systems (DBMS) need to function. Modern Database Management Systems have become inadequate because it has no longer meet the needs of big data requirements, which include massive scales and rapid data processing. The research analyzes the development of DBMS technologies designed for big data systems by examining three essential challenges, which include scalability needs as well as performance improvements, and handling different kinds of data. In order to increase system speed and response time, the study looks at optimizing system performance using query optimization, data compression, parallel processing, and vectorization techniques. The paper examines major data security matters and system timing issues, which both present significant performance challenges during increasing data growth. This paper evaluates upcoming distributed computing frameworks and machine learning strategies, and automated database tuning through AI, which can empower the next level of DBMS solutions. The research survey explores optimal techniques for DBMS systems and defines upcoming research routes that focus on developing big data DBMS performance.

***Corresponding author**
Srinivasa Rao Maka, North Star Group Inc, Software Engineer, USA.

Received: May 09, 2024; **Accepted:** May 13, 2024; **Published:** May 27, 2024

Keywords: Big Data, Database Management Systems (DBMS), Performance Optimization, Scalability, Query Optimization, Data Security, Vectorization, Data Heterogeneity, Scalability Techniques, Data Privacy

Introduction

The digital era sees organizations from every sector producing massive data sets through business operations and user engagement, and sensor-based information acquisition. Big Data is the name for this extensive data collection that brings specific technological problems from data quantity and speed, and information diversity [1]. The capacity of relational database management systems (RDBMSs) to process and manage big data volume and complexity becomes limited when facing such magnitude and intricacy [2].

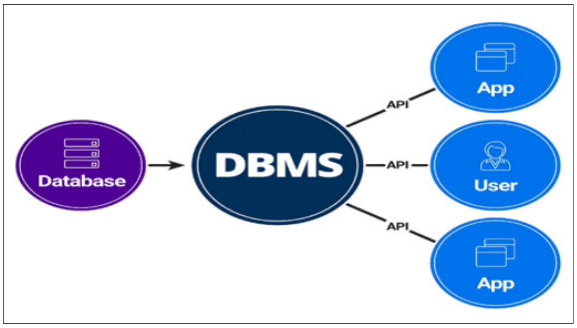

Figure 1: Database Management System

The DBMS operates as a program solution in Figure 1 that optimizes data storage and retrieval functions. The system gives access control capabilities combined with data abstraction which enables transaction management, thus acting as the essential basis for data-intensive digital applications. Modern DBMSs operate as complex systems that depend on multiple adjustable parameters which professionals call configuration knobs to determine system operational characteristics [3]. System efficiency evaluation within big data environments depends heavily on these performance benchmarks which are strongly affected by adjustable parameters.

The recent trend has been shifted towards more flexible as well as more scalable alternate, such as NoSQL databases for big data applications. These systems can run schema-less data models, will distributed architecture, and are high availability, and are suitable for processing unstructured and semi-structured data at scale. NoSQL databases such Since key-value stores, document databases, column family stores, and graph-based systems are flexible, it can handle data that is constantly being created from a variety of sources.

One of the most crucial aspects of DBMS performance when the database accommodates enormous data is query optimization. Efficiency in query processing is crucial in order to reduce resource utilization and have a quick response time. However, with the continuous growth of data, the need to optimize queries increases to let real-time analytics, decision making and user experience [4].

Structure of the Paper
The following paper organized as: Section II describes overview of database management system in big data and Section III provide the optimization techniques of DBMS in bigdata, then Section IV challenges in optimizing DBMS for big data, Section V characteristics of NoSQL database models and relational databases and Section VI give the review of relevant literature, together with a conclusion outlining future research.

Overview of Database Management Systems in Big Data
Database management systems, also known as real-time database systems (DBMS), act as data repositories and offer effective data loading, storage, and manipulation [5]. It must have the fundamental features of a broad database management system. A database system is the collective term for the database, the DBMS, and the software program. It is usually hard to separate the database and the DBMS without exporting the database in a different format, particularly when it comes to relational databases [6]. In these cases, the database is frequently not accessible by the DBMS until it is imported back into a format that the DBMS can understand. The underlying database and the DBMS are frequently referred to simply as databases informally, maybe because of this inseparability.

An example of a simplified system that highlights the elements that are essential to a database system and the scope of this research. Throughout this investigation, make reference to the elements shown in Figure 2. The Figure has sacrificed technical accuracy and comprehensiveness for presentational convenience by showing only one end-user, one software program, one database management system, one hardware component, and one database.

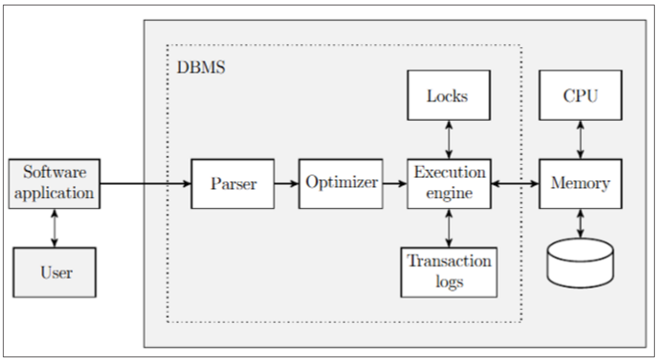

Figure 2: Overview of DBMS

Types of Databases
Types of the databases management system in big data are as follows:

- **Bibliographic Databases:** Information that is format-free (unformatted data). This is made up of textual data, which by definition doesn't show much or any format. Libraries and information systems frequently employ these kinds of databases. Keywords and phrases from book abstracts and other materials might be used to create this data. One can judge whether or not a document is of interest by looking at its abstract. Document titles, authors, journal names, volume and number, dates, keywords, abstracts, and other descriptive information are all included in bibliographic database.
- **Knowledge Databases:** AI applications make advantage of knowledge databases. There is formatted and discrete data in them. There are usually many different types of data in these, but there are extremely few instances of each type. The amount of these databases' data matches the size of their description [7,8].
- **Graphic-Oriented Databases:** Computer-Aided Design (CAD) may make use of graphic-oriented databases. This type of database's data is described as active. Accordingly, data is a process that may be carried out. Since the aforementioned 1 and 2 cannot be carried out on a computer, any changes can be made to the data.
- **Decision-making Databases:** Corporate management and related administrative duties make use of decision-making datasets. One may deal with issues like resource planning and sales forecasting by using the data in these databases. The fact that these databases' data contents are formatted, somewhat longer than descriptions, and passive.

A common term for these decision-making databases is "just databases." Database Management Systems (DBMS) are categorized according to the type of databases it handle. For instance, Knowledge database management systems and bibliographic database management systems.

Numerous enabling technologies are connected to the wide-ranging subject of big data studies. The Big Data Technology Map connects a number of enabling technologies to various stages of the Big Data value chain, both proprietary and open-source [9]. As shown in Figure 3, the large data system may also be broken down into a layered structure. From bottom to top, the layered structure may be separated into three levels: the infrastructural layer, the computational layer, and the application layer. Only a conceptual hierarchy is provided by this layered approach to highlight the intricacy of a huge data system.

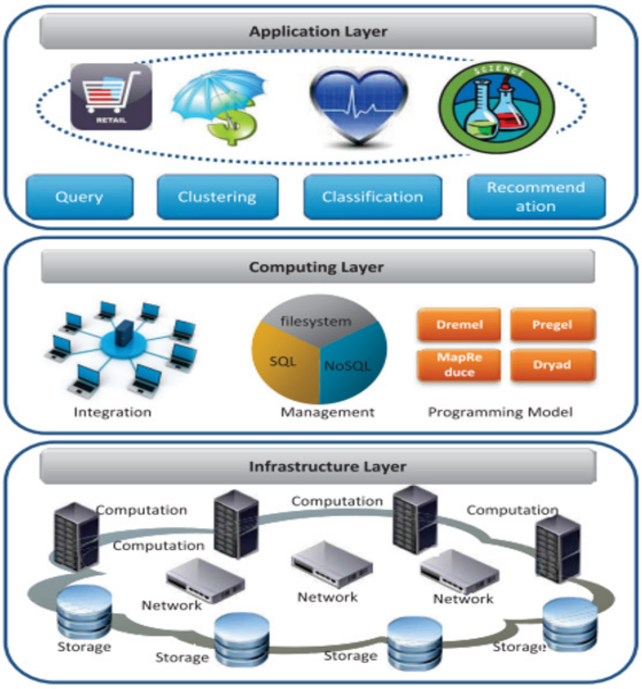

Figure 3: Layered Architecture of Big Data System

The function of each layer is as follows:

- The collection of ICT resources that make up the infrastructure layer can be arranged using cloud computing architecture and made possible by virtualization technologies.
- The computer layer integrates many data tools into a middleware layer that runs on raw ICT resources. The programming model, data management, and data integration are common technologies in the big data setting.
- The application layer integrates fundamental analytical techniques to create a variety of field-related applications by using the interface that the programming models give to accomplish different data analysis operations, such as statistical analyses, clustering, classification, and querying

Characteristics of Big Data

Some features of big data include the following, and Figure 4 illustrates the three V's of big data.

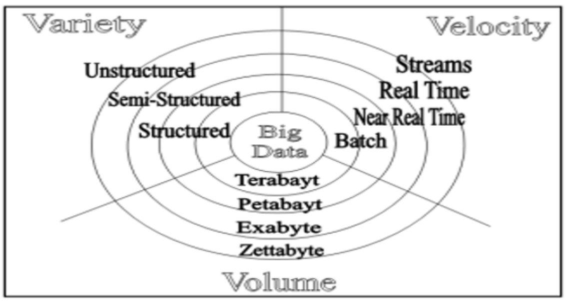

Figure 4: The three Vs of big data

- **Big Velocity:** Big velocity is correlated with data delay and throughput. Big velocity for throughput refers to the pace at which data enters and exits networked systems in real time [10]. Stated differently, it refers to the rapid flow of data and information into and out of interconnected systems in real-time.
- **Big Variety:** The phrase "big variety" describes the vast array of data sources available, their many forms, and the kinds of data that are accessible to everybody [11]. At a higher level, big data may be divided into three categories: semi-structured, structured, and unstructured. Relational database systems such as Oracle store structured data.
- **Big Volume:** Data volume or size has surpassed terabytes and petabytes in recent years [12]. The vast volume and growth of data surpasses conventional storage and processing methods.

Optimization Techniques for DBMS in Big Data

Almost any quantity of data can be scaled with Hadoop/Hive. However, optimization is still required to process data rapidly and run queries effectively [13]. To enhance Hive efficiency, Hive query optimization techniques may be used while running Hive queries.

- **Tez Execution Engine:** Built on top of Hadoop Yarn, an application framework called Apache Tez is used for both interactive and batch data processing, speeding up query execution. For interactive queries, Tez is more adaptable and effective than MapReduce. It reduces query execution time by one to three times [14]. Because of less job splitting and access to HDFS, Tez provides faster response times; MapReduce divides tasks into more jobs and requires more HDFS access.
- **Vectorization:** Through the retrieval of 1024 rows in a single operation, vectorization enhances efficiency for processes like scanning, filtering, aggregation, and joining. Rather than processing each row separately, Hive can handle a batch of rows thanks to vectorization. To enhance cache utilization, every batch comprises a column vector, and operations are performed on the entire vector.
- **Parallel execution:** The parallelism that MapReduce processes on Hadoop provide is automatically utilized by a number of executable searches on Hive. The query's total execution time can be decreased in this fashion. It is possible to transform the queries into several phases during execution, which are typically carried out sequentially.
- **Compression:** The quantity of data that is exchanged between mappers and reducers is greatly decreased via compression techniques. The size of the compressed file should not exceed a few hundred megabytes. Snappy, lzo, bzip, and gzip are a few of the Hive compression techniques. The scenario determines whether compression techniques are appropriate.
- **File format:** Certain file formats, like JSON, are not suitable for huge systems with a lot of data since it require a lot of storage space. Depending on the data, a suitable file format can significantly improve query performance. Because ORC format speeds up data processing and reduces up to 75% of the original data's size, it makes it possible to store Hive data efficiently.

Query Optimization

Enhancing the method for handling a database query is known as query optimization. As a result, it is a crucial stage in query processing. The variety of tasks required to get data from a database is referred to as query processing. In theory, every option must be considered in order to choose the one with the most promising performance forecast [15]. Figure 5 depicts an abstraction of the procedure for creating and evaluating these options; it is basically a query optimizer's modular design.

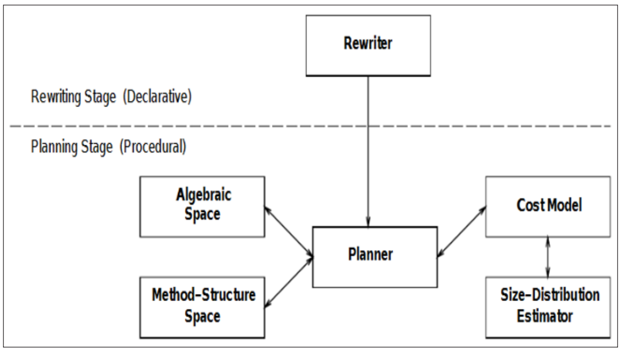

Figure 5: Query Optimizer Architecture

Although an optimizer might be constructed using this design, the modules' borders are not usually as distinct in actual systems.

Performance Optimization

As a modelling and computational tool, the PBO methodology makes use of the method of finite element analysis (FEA). The stresses or strain energy densities of components can be calculated using the results of finite element analysis [16]. Elements with low levels of stress or strain are seen as ineffective and should be removed. By gradually removing inefficient components from a structure, its performance can be enhanced. Until the structure's performance is optimized, the FEA, performance evaluation, and element removal processes are repeated. Additionally, the optimization process is summed up as follows:

- A fine-grained representation of the original continuum structure.
- The use of finite element analysis to the structure is necessary.
- After building the structure, use performance indices to evaluate it.
- This enables us to determine each load case's stresses and strain energy densities of the elements.
- Reduce the number of ineffective elements in the design domains.
- It is necessary to verify the continuity of the resultant structure.
- Symmetry of the resulting structure has to be checked
- Select the best design and plot the performance index history.

Scalability Techniques

Database management systems (DMS) have the need to be scaled since an adaptable infrastructure is needed to accommodate data growth and expanding workloads. For start-ups, flexibility and growth, knowing specific scalability measures with their impact on DMS performance are very important. Metrics that are part of achieving high throughput, low latencies and data availability include partitioning, replication, concurrency control, and consistency. Designing for these challenges is a goal of existing scalable DB vendors, and such a comparison can show which changes might be made for future applications. It is also important to make the collaboration between database software and hardware, and storage devices more efficient and effective. Work in this area has been to minimize disk accesses, use memory, and to ensure that hardware resources are sustainably and predictable used.

Challenges in Optimizing DBMS for Big Data

There are several challenges facing the optimization of Database Management Systems (DBMS) for big data applications that may seem unique to big data. There are data security, heterogeneous datasets, scale, timeliness, and enterprise-specific issues. Here is an overview:

Data Privacy and Security

Security may be a big worry because Big Data is a new technology that not all businesses fully understand. Protecting the data from security breaches is of utmost importance since the majority of the data in the data sets is crucial [17]. Think about information obtained from a location-based service, for instance. For this service to work, the user must let the service provider know where that are right now. Because the user's location is disclosed in the event of a security breach, there may be major privacy problems.

Heterogeneity and Incompleteness

In contrast to humans, machines can only comprehend homogeneous data. In contrast to humans, machines are incapable of understanding subtleties. Therefore, meticulously organized, Accurate, and fruitful data analysis requires data. Furthermore, inaccuracies in data analysis might arise from inadequate data. Think of a health record database where each patient's blood type, employment, and date of birth are recorded. Patients frequently fail to submit all the data that was asked for. Therefore, the data's value is set to null.

Scale

A novel approach to processing big data is needed as computer resources cannot keep up with the growing volume of data, and CPU speeds are stagnating due to power constraints. First, CPUs with more cores are being produced, and the clock speed of processors has essentially stagnated over the past five years. Previously, huge data processing systems had to deal with parallelism within a single node, whereas now it has to deal with parallelism among cluster nodes.

The second is the move to cloud computing, which now creates massive clusters by combining several workloads with various performance goals. This level of resource sharing on expensive and large clusters calls for new ways to Figure out how to run and execute data processing jobs in order to meet the goals of each workload in a cost-effective manner and to handle system failures, which become more frequent as it operate on larger and larger clusters.

Time Liness

Timeliness is the ability to get data considerably more quickly in real time. massive volumes of data must be handled, and analysis takes longer, since the digitization of almost "everything" has caused the development of new types of huge, real-time data in several fields. It requires a system architecture that will address each of these factors. need the analysis's results in a matter of seconds in their fast-paced environment. For example, if a suspected fraudulent credit card transaction is discovered, it is best to report it before the transaction is completed, possibly preventing it completely.

Business/Enterprise Challenges

The amount, value, and significance of the internal and external data required to manage the company and compete in various international marketplaces are constantly growing. The actual difficulty is to extract as much value as possible from this data in an efficient and economical manner. To create new business intelligence (BI) using rich analytics, one must be able to cross-reference seemingly unrelated data sources in creative ways. Each business segment handles its own data, even if the master data is accessible and used to run the firm.

Features of Relational Databases and NoSQL Database Models
Relational databases and NoSQL databases have distinct features that make them suitable for different applications. Here's an overview of their key features based on the provided content [18]:

Closed and Open Source
There are both proprietary and open source systems for relational databases [19]. Relational databases that are proprietary, like Oracle, may frequently scale more effectively than their open-source counterparts, like MySQL. Nonetheless, a lot of NoSQL database models, like MongoDB, CouchDB, and Cassandra, are open source [20]. NoSQL's open-source nature gives academics more chances to study a database's characteristics and gives customers who can't afford proprietary database models access to cheaper storage.

Scalability
Relational databases often expand up, requiring one server's hardware to be changed in order to improve efficiency. Because of this, administrators find it more challenging to update relational databases [21]. The hardware constraints that are established by design and cannot be changed provide another difficulty for this upgrade approach. For instance, the manufacturers of the hardware set a predetermined Figure for the most auxiliary storage or RAM that the system can manage.

Volume and Variety of Data
The amount of data that databases are required to manage has expanded due to internet applications [22]. The rise of web 2.0 and 3.0 on the internet has increased the amount and diversity of data that has to be saved. The emergence of big data has increased the amount and variety of data. These sources have produced massive amounts of data that relational databases have not been able to handle. NoSQL is ideal for data-intensive web applications because of its exceptional ability to manage massive amounts of data. Businesses that have switched to NoSQL, including Google, Facebook, and Yahoo, demonstrate this.

Literature of Review
This section reviews various studies emphasizing how big data technology may improve decision-making and data processing, and service delivery across domains such as libraries, enterprise management, and cloud computing, emphasizing performance enhancement, data modeling, and system integration strategies.

Zalipynis examines how data compression affects array processing operations. Compress the data using a range of methods, then look into how compression impacts processing speed. It conduct an extensive comparison of Chronos DB and SciDB's source and compressed data performance on real-world data on computer clusters in the Microsoft Azure Cloud [23].

The problem of library big data applications is examined in this research, which then builds a model system for library big data applications using the method of large-scale network analysis. The challenge of big data services in academic libraries can be met by this library's big data application model, which is founded on knowledge management theory and an open source cloud computing platform that uses a large-scale network analysis method. It can also successfully encourage the growth of big data services in academic libraries [24].

An increasing number of businesses are realizing that data collection, analysis, filtering, and application are critical components of company decision-making management in the face of intense market competition. In enterprise management circles, bid data has emerged as a hotspot for study and has greatly affected relevant ideas of corporate decision-making management. In order to investigate the practical difficulties of corporate decision-making management against a backdrop of big data and propose solutions, this study developed a type of business decision-making management system based on big data [25].

Chang et al. project aims to optimize a high-performance, high-availability infrastructure for processing various large data sets. The platform can run SQL queries in a big data environment, automatically select the top-performing big data warehouse platform for computation, and use the high-performance cache system to get the same result much faster owing to optimizations done to integrate Apache Hive, Cloudera Impala, and BDAS Spark SQL. Especially when running several repeated SQL queries in multi-user mode, the recommended approach significantly decreases query/response time. This greatly enhances overall performance [26].

Rahman et al. offers a data clustering method grounded on the frequency of data access. Only the hot and cold statistics have been taken into account. In this case, they separated the entire database into two distinct files. Only hot data is included in the first file, and only cold data is included in the second. various application domains will have various hot and cold data time periods. The first database file will be directly accessible to the database engine searching for the second database file next should the data be missing [27].

Benjelloun et al. government and a variety of companies may use big data to gain insightful information. This kind of information can assist decision makers in improving their plans and strategies. It offers additional value to several commercial and social areas and aids the organization in gaining a competitive edge. Several governments have really started initiatives to improve research and development in the Big Data space, with significant funding. The private sector has likewise invested heavily in order to optimize resources and maximize earnings [28].

Table 1 summarizes key literature on optimizing database management systems for big data applications, highlighting diverse approaches, focus areas, findings, and associated challenges across academic, enterprise, and public sector contexts

Citation: Srinivasa Rao Maka, Suneel Babu Boppana, Gangadhar Sadaram, Laxmana Murthy Karaka, Vasu Velaga, et al. (2024) Optimizing Database Management Systems for Big Data Applications: Techniques and Challenges . Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-479.
DOI: doi.org/10.47363/JAICC/2025(4)453

Table 1: Literature of review on Optimizing Database Management Systems for Big Data Applications

Reference	Key Topic	Focus Area	Findings/Insights	Challenges
Zalipynis	Data Compression in DBMS	Impact of compression on array processing in Chronos DB and SciDB	Compression techniques significantly affect processing speed; evaluated on Microsoft Azure cloud clusters	Managing trade-off between compression ratio and query performance
Liu	Library Big Data Applications	Academic library big data service using network analysis	suggested a framework to enable library services that is built on open-source cloud platforms and knowledge management	Effective integration of big data technologies in traditional academic settings
Yang and Zhang	Enterprise Decision-making with Big Data	Management decision system for enterprises	Developed a data-driven decision-making model to improve enterprise strategy under big data contexts	Adapting organizational structures to exploit big data fully
Chang et al.	SQL Optimization in Big Data	Auto-selection of data warehouse for SQL processing	Enabled faster query responses using high-performance caching and optimal platform selection	Ensuring consistent results across diverse platforms
Rahman et al.	Data Clustering in DBMS	Access frequency-based clustering (Hot/Cold data separation)	Separated hot and cold data for performance gains; dynamic access patterns considered for various applications	Defining hot/cold data thresholds and managing dynamic data transitions
Benjelloun et al.	Big Data Utilization in Public/Private Sectors	Government and industry investments in big data exploitation	Big data improves decision-making, adds economic value, and supports innovation; highlighted global governmental and private investments	Data privacy, infrastructure scalability, and efficient insight extraction from diverse sources

Conclusion and Future Work

The limits of conventional database management systems in managing the volume, velocity, and diversity of big data have been made clear by the exponential development of data in recent years. To overcome these obstacles, the optimization of DBMSs has become a critical research and development area. This paper reviewed key optimization techniques such as parallel query execution, vectorization, advanced compression, and efficient storage architectures that significantly improve DBMS performance in big data environments. The adoption of scalable frameworks, specialized file formats, and adaptive resource management strategies plays allowing real-time analytics and informed decision-making depends on a key part.

Future research must tackle unresolved issues related to data heterogeneity, security, and interoperability. The design of next-generation DBMSs should emphasize extensibility, high throughput, low latency, and seamless integration with big data analytics ecosystems. Additionally, the exploration of hybrid architectures that bridge relational and non-relational paradigms, as well as innovations in query optimization and data indexing, will be essential for sustaining performance at scale. Through continuous innovation in these areas, organizations can unlock deeper insights from their data assets and stay ahead of the competition in the age of data [29-34].

References

1. Hwang JS, Lee S, Lee Y, Park S (2015) A Selection Method of Database System in Bigdata Environment: A Case Study from Smart Education Service in Korea. Int J Adv Soft Comput its Appl 7.
2. Muniswamaiah M, Agerwala T, Tappert C (2019) Big Data in Cloud Computing Review and Opportunities. Int J Comput Sci Inf Technol https://arxiv.org/pdf/1912.10821.
3. Van Aken D, Pavlo A, Gordon GJ, Zhang B (2017) Automatic Database Management System Tuning Through Large-scale Machine Learning. Proceedings of the 2017 ACM International Conference on Management of Data, New York, NY, USA: ACM 1009-1024.
4. John R, Palaskar N (2017) A Survey of Various Query Optimization Techniques. Int J Comput Appl https://ijcaonline.org/archives/volume173/number5/john-2017-ijca-915286.pdf.
5. Jian Wu, Yong Cheng, Schulz NN (2006) Overview of Real-Time Database Management System Design for Power System SCADA System. Proceedings of the IEEE SoutheastCon 62-66.
6. Gogineni A (2017) Novel Scheduling Algorithms For Efficient Deployment Of Mapreduce Applications In Heterogeneous Computing. Int Res J Eng Technol 4: 6
7. Gunjal B, Koganurmath MM (2014) Database System: Concepts and Design. Res Sch Natl Inst Technol Rourkela 1-19.
8. Neeli SSS (2019) Serverless Databases : A Cost-Effective and Scalable Solution. IJIRMPS 7.
9. Hu H, Wen Y, Chua TS, Li X (2014) Toward Scalable Systems for Big Data Analytics: A Technology Tutorial. IEEE Access 2: 652-687.
10. Pathak P, Shrivastava A, Gupta S (2015) A survey on various security issues in delay tolerant networks. J Adv Shell Program 2: 12-18.
11. Sun Z, Strang K, Li R (2018) 10 Bigs: Big Data and Its Ten Big Characteristics. Manag Perspect Intell Big Data Anal 1-14.
12. Sagiroglu S, Sinanc D (2013) Big data: A review. International Conference on Collaboration Technologies and Systems (CTS) 42-47.
13. Wankhade P, Deshmukh V (2008) An Overview of Query Optimization Techniques in Database Systems Int Res J Eng Technol https://www.irjet.net/archives/V6/i2/IRJET-V6I2217.pdf.
14. Hazarika AV, Ram GJSR, Jain E (2017) Performance comparision of Hadoop and spark engine. International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), IEEE 671-674.
15. Zager Al Saedi, Ghazali RB, Bin Mat Deris M (2014) An Efficient Multi Join Query Optimization for DBMS using Swarm Intelligent Approach. 4th World Congress on

Information and Communication Technologies (WICT 2014), IEEE 113-117.
16. Liang QQ (2007) Performance-Based Optimization: A Review. Adv Struct Eng 10: 739-753.
17. Kanchi S, Sandilya S, Ramkrishna S, Manjrekar S, Vhadgar A (2015) Challenges and Solutions in Big Data Management -- An Overview. 3rd International Conference on Future Internet of Things and Cloud, IEEE 418-426.
18. Neeli SSS (2019) The Significance of NoSQL Databases: Strategic Business Approaches and Management Techniques. J Adv Dev Res 10.
19. Mohamed MA, Altrafi OG, Ismail MO (2014) Relational vs . NoSQL Databases : A Survey. Int J Comput Inf Technol 3.
20. Kim W (2014) Web Data Stores (aka NoSQL databases): A Data Model and Data Management Perspective. Int J Web Grid Serv 10: 100.
21. Abourezq M, Idrissi A (2013) NoSQL Database: New Era of Databases for Big data Analytics - Classification, Characteristics and Comparison. Int J Database Theory Appl 6.
22. Moniruzzaman ABM, Hossain SA (2013) NoSQL Database: New Era of Databases for Big data Analytics - Classification, Characteristics and Comparison. Int J Database Theory Appl 6.
23. Zalipynis RAR (2019) Evaluating Array DBMS Compression Techniques for Big Environmental Datasets. 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) 859-863.
24. Liu Y (2018) Research on the application of big data in academic libraries. Proceedings - 3rd International Conference on Intelligent Transportation, Big Data and Smart City, ICITBS.
25. Yang L, Zhang JJ (2017) Realistic Plight of Enterprise Decision-Making Management Under Big Data Background and Coping Strategies. IEEE 2nd International Conference on Big Data Analysis (ICBDA) 402-405.
26. Chang BR, Tsai HF, Wang YA (2016) Optimized Multiple Platforms for Big Data Analysis. IEEE Second International Conference on Multimedia Big Data (BigMM), IEEE 155-158.
27. Rahman MH, Bin Al Abid F, Zaman MN, Akhtar MN (2015) Optimizing and Enhancing Performance of Database Engine Using data Clustering Technique. International Conference on Advances in Electrical Engineering (ICAEE), IEEE 198-201.
28. Benjelloun FZ, Lahcen AA, Belfkih S (2015) An Overview of Big Data Opportunities, Applications and Tools,” in 2015 Intelligent Systems and Computer Vision (ISCV), IEEE 1-6.
29. Kuraku S, Kalla D, Samaah F, Smith N (2023). Cultivating proactive cybersecurity culture among IT professional to combat evolving threats. International Journal of Electrical, Electronics and Computers 8.
30. Kalla D, Smith N, Samaah F, Polimetla K (2022) Enhancing Early Diagnosis: Machine Learning Applications in Diabetes Prediction. Journal of Artificial Intelligence & Cloud Computing 2-7.
31. Kuraku DS, Kalla D (2023). Impact of phishing on users with different online browsing hours and spending habits. International Journal of Advanced Research in Computer and Communication Engineering 12.
32. Kalla D, Kuraku S (2023) Phishing website url’s detection using nlp and machine learning techniques. Journal of Artificial Intelligence 5: 145.
33. Kuraku DS, Kalla D, Samaah F (2022) Navigating the link between internet user attitudes and cybersecurity awareness in the era of phishing challenges. International Advanced Research Journal in Science, Engineering and Technology 9.
34. Kuraku DS, Kalla D, Smith N, Samaah F (2023) Exploring How User Behavior Shapes Cybersecurity Awareness in the Face of Phishing Attacks. International Journal of Computer Trends and Technology https://www.ijcttjournal.org/archives/ijctt-v71i11p111.