International Conference on
Global Engineering & Management Trends

International Journal of Science,
Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Analyzing the security during Big Data Transmission Over the Network

Ms. Sugandha, Dr. Kavita Mittal
Engineering & Technology, Jagannath University, Bahadurgarh

Abstract- This research aims to enhance the security and performance of big data transmission across networks by proposing a secure clustering approach that integrates encryption mechanisms. The study addresses four primary objectives: analyzing security risks to structured and unstructured data through a literature review, evaluating clustering and encryption techniques to improve big data security, developing a robust model for secure data transmission, and conducting a comparative analysis of the proposed model against existing methods. The proposed framework leverages a combination of clustering, and encryption to safeguard data while ensuring efficient transmission. The data flow diagram illustrates a multi-stage process, beginning with parallel operations to extract relevant datasets and keywords. The frequency of these terms is determined using MapReduce and grouping methods. Enhancements in K-means clustering involve iteratively linking data entries to the nearest centers, recalculating centroids, and optimizing cluster configurations until stability is achieved. The secure framework is built upon a layered architecture comprising client-side validation, server-side storage, database management, map reduction, and encryption for content protection. The implementation involves the use of a Hadoop-based environment, integrating tools such as Spark and Hive, along with Python scripts for data processing. The comparative analysis highlights the efficiency of Spark's in-memory execution, which significantly outperforms MapReduce in processing speeds. Hive, with its concise query capabilities, is shown to be more efficient than Pig in handling large datasets. Python scripts and Cloudera integration further streamline data processing, ensuring scalability and reliability. Experimental results underscore the performance advantages of the proposed model. Accuracy metrics, are derived from confusion matrices for both optimization and non-optimization datasets. The proposed clustering and encryption techniques achieved an overall accuracy of 93.4% with optimized datasets, compared to 92.2% for non-optimized datasets. The study also demonstrates the effectiveness of the K-means algorithm in identifying clusters, with an accuracy of 86.05% for raw data.

Keywords- Blockchain Technology, Internet of Things (IoT), Blockchain-based Internet of Things (BIoT), Cloud Computing.

I.INTRODUCTION

The need for big data in many applications has been covered in this section. One definition of "big data" is the study and practice of managing and analyzing massive data volumes. The medical field, the industrial industry, the media and entertainment industries, and many more have made use of it (Al-Sai, 2020). Big data sets are essential in marketing, but they are difficult to manage and analyze. Government procedures also make use of Big Data, which impacts pricing, production, technology, etc. With the current rate of data expansion, the transport industry is almost unable to cope (Bag, S., 2020). Rich sources are part of the data collected by the transportation industry. While big data is finding applications in many fields, there are still obstacles to overcome when working with massive data sets. Skilled employers are in short supply. Such data must be evaluated by these competent employers (Bante, P. M., 2017). Along with this, processing massive data volumes takes time. The ability to do real-time evaluations of data is something that many firms have been considering from the beginning. Another fundamental issue that is becoming more pressing is the massive amount of unstructured data. Another obstacle is the absence of competent management to analyse Big Data effectively (Barragán, D., 2020). Not only can it manage a vast amount of independent data in a short amount of time, but it is also expected to provide businesses with improved understanding. Processing massive data has been a common use case for the Hadoop architecture. It is a popular tool for handling and analysing large amounts of data. Hadoop provides Map Reduce HDFS, and there are a number of distributed programming frameworks that can handle massive data with it. In this part, we will go over Hadoop and its related technologies (Batko, K., 2022). The characteristics of map reduction and spark are described. Pig, HDFS, and Hive are all seen. Also taken into account is the idea of a web server that might run a Python script. To host web applications that communicate with databases, a web server must be deployed on the environment (Belcastro, L., 2022). Databases such as HIVE, PIG, and HDFS do exist. Data has been sent from the operator end to the database using Python on the server side. On the server side, a Python script may make decisions.

BIG DATA

The organization running in existing world always wants that there should be mechanism to examine large quantity of data in a relevant time (Bhandarkar, M., 2010). Due to this requirement of customer and it's paying capacity require to be considered. On basis of this, extra offers to customer are provided. Moreover recognition of accurate quantity of data and its nature is also required because it could affect output of a business (Bharati, T. S., 2020). Big data contains all data whether it is organized or unorganized, data from e-mail, Facebook, WhatsApp, Instagram, Twitter etc. In data handling system it is must to organize information in controlled manner. Organizations can harness Big Data to gain insights, improve decision-making, and create value by analyzing patterns and trends that were previously undetectable. However, the sheer scale of Big Data necessitates advanced technologies and methodologies in case of storage, processing, and analysis, surpassing the capabilities of traditional data management tools (Bobade, 2016). The primary advantage of Big Data lies in its potential to reveal hidden correlations and insights. In healthcare, Big Data can lead to breakthroughs in personalized medicine and predictive diagnostics. Governments can use it to enhance public services and improve urban planning. Despite these benefits, the complexity of Big Data also raises concerns about data privacy, security, and ethical use, necessitating robust governance frameworks and policies (Chebbi, I., 2018). Big Data technologies include distributed storage systems like Hadoop, processing frameworks such as Spark, and various analytics tools that facilitate real-time data processing and visualization. These technologies enable the efficient handling of data from diverse sources, supporting complex queries and advanced machine learning models. The integration of (AI) with Big Data further amplifies its impact, enabling predictive analytics and automation across industries. As Big Data continues to evolve, it is essential for organizations to develop necessary skills and infrastructure to stay competitive and leverage data-driven insights effectively (Cheng, D., 2017).

HADOOP

Hadoop is an Apache freely available structure. It is scripted in java. It provides an opportunity in case of circulation of big datasets working with in bundle of computers. To fulfill this purpose uncomplicated software are adopted by it. The Hadoop structure program operates in an atmosphere that gives circulated storage space and calculation within group of computers (Rossi, R., 2022). The formation of Hadoop is done in order to improve from an individual server to thousands of machines, each contribute to narrow calculation along with storage. Apache has introduced Hadoop as well-known distributed data processing platform. Massive informations could be executed by Hadoop. It assigns categorized information to different servers that are known as clusters. Various parts of problem are defeated by them. After that outputs are withdrawn. Two most important sections are available in middle of Hadoop to perform its job. The formation of large servers with intense arrangements that manage massive dispensation is pretty valuable. In its place, operator could wrap countless items of computers with single-CPU, as a separate useful distributed system. Simultaneously records are studied by combined machines and supplied a much better output (S. Ikhlaq, 2016). Additionally, in comparison to one high-end server it is cost effective. Thus, it is primary influential reason of using Hadoop.

MapReduce

It has been known as a programming model to write applications. These applications could process BD in parallel on several nodes. MapReduce is able to offer the analytical ability. It has been used for analyzing huge volume of complex data (Stergiou, C. L., 2020). The Data has been considered a collection of huge datasets. It has been processed with existing techniques used in computing. The example may be volume of data Facebook or User tube needed for assembling and managing of daily category related Big Data. MapReduce has indeed been recognized as a powerful programming model in case of developing applications capable of processing Big Data in parallel across multiple nodes (Sun, J., 2022). This framework has been extensively used for analyzing massive datasets, including those generated by platforms like Facebook and YouTube, which produce vast amounts of data daily across various categories. For instance, in the case of Facebook, MapReduce could be utilized to analyze user and shares, across millions of posts and profiles. This analysis could provide valuable insights into user behavior, preferences, along with trends, which can inform content recommendations, targeted advertising, and overall platform optimization. Similarly, for YouTube, MapReduce could be employed to process and analyze the enormous volume of video content uploaded and viewed by users worldwide. By analyzing viewer engagement, video metadata, and user-generated content, YouTube can improve recommendations, enhance content discovery, and optimize ad targeting to better serve its user base.

CLUSTERING

The goal of cluster analysis, also known as clustering, is to organise data sets into meaningful groupings wherein items within each cluster are more comparable to one another than to those outside of that cluster. Pattern recognition, data compression, ML, bioinformatics, information retrieval, pattern recognition, along with image analysis are just a few of the several domains that make use of this basic statistical data analysis approach. It is also a primary role of exploratory data analysis. It is instead than being a particular technique, cluster analysis is the overarching problem that needs solving. This may be accomplished by a variety of algorithms, each with its own unique take on clustering and the best ways to locate them (Tang, L., 2022). Clusters are often understood as groupings of closely related data points, dense regions of the data space, intervals, or specific statistical distributions. One way to look about clustering is as an optimisation issue with several objectives. In and of itself, cluster analysis is not a robotic process but rather an iterative, trial-and-error method of discovering new information or optimising interactive multi-objectives. Until the outcome has the expected qualities, it is often required to tweak data preparation and model parameters. To arrange a collection of items into clusters, where objects within each cluster are more similar to each other than to objects outside of that cluster, cluster analysis—sometimes called clustering—is a basic approach in data analysis. The end objective is to help in areas like pattern recognition, machine learning, and exploratory data analysis by revealing latent structures and patterns in the data (Taylor, 2010).

Security Of Digital Data In Cloud

It is important to maintain authoritative or customer information. If the information of this association is not secured in the cloud, conventional cloud engineering cannot be transferred (Vasa, J., 2022). Many non-profit organizations support cloud security concerns and pay attention. One of them is Cloud Security Alliance, which distributes a report on most prominent cloud security problems every year. In 2013, the CSA reports identified eight notable security hazards in the cloud that may interfere with specific customer information without understanding them (Vorapongkitipun, C., 2014). These are the security problems most reported by CSA. Despite these security concerns, researchers are exploring a lot more issues in the next parts. Cloud computing is a popular platform for various purposes, including illegal, criminal, and aggressive use. Some malicious individuals may exploit the cloud for malicious purposes, such as stealing data or providing unauthorized access to services. The programmable interface of cloud information and services can be weak and insecure, leading to potential insider theft. Security issues in shared cloud include virtualization, where customers can create multiple virtual computers on the same server, and combined services, where customer data is shared among all services that comprise the service (Wahyudi, M., 2022). This can lead to the misuse of unique information by backend services without the customer's knowledge. Data damage and loss are two main threats to cloud services. Common reasons for data damage include server issues like earthquakes, fires, physical problems, and equipment malfunctions. Malicious clients can erase or overwrite all information, even if all information is moved. Data breach is considered the highest security risk, as demonstrated by the Computer Security Alliance's review article (Wang, J., 2020).

II. LITERATURE REVIEW

This review explores the security and performance of big data transmission across networks, focusing on challenges in securing structured and unstructured data. It evaluates various clustering and encryption techniques, including MapReduce, K-means clustering, and encryption mechanisms. The study aims to address gaps in current methodologies and explores the applications of Hadoop, MapReduce, and HBase in big data environments. The review also discusses the security paradigm for cloud computing data and upgrades to HDFS infrastructure. The paper also discusses the integration of big data into SVM models for enhanced anomaly detection and the role of machine learning mechanisms. The review also discusses new research on social data processing, big data analysis, cross-platform resource scheduling, data migration processes, word count jobs, and data mining operations. The review concludes by analyzing the literature to identify security risks and factors influencing big data transmission performance.

Table 1 Literature review

Author / Year	Objective	Methodology	Limitation
K. Rajesh kumar / 2024	Secure medical large data transfer via the cloud with shaxecc	SHAXECC, MFCM, LSCOA	As clustered network expands, suggested technique may struggle to effectively transmit enormous amounts of data.

Md. A / 2024	Network intrusion detection utilising ML for large and unbalanced datasets via feature extraction,	PCA, DT, RF	Clustered approaches may increase network overhead owing to node communication, reducing transmission efficiency.
S. Guan / 2024	A safe way to store large amounts of data in the cloud using Hadoop	HDFS, ECC	Clustered may restrict resource use, resulting in computational resource inefficiency.
Wong and Arjunan (2024)	Real-time detection of network traffic anomalies in big data environments using deep learning models.	Utilized deep learning models for anomaly detection in large-scale network traffic data.	High computational requirements for real-time analysis.
Haddad et al. (2024)	Sentiment prediction in social networks using batch and streaming big data analytics with deep learning.	Combined deep learning techniques with batch and streaming analytics for sentiment prediction.	Complexity in managing batch and real-time data integration.
Arjunan (2024)	Detection of network traffic anomalies in big data in real-time using deep learning.	Employed deep learning algorithms to detect traffic anomalies in real-time.	Scalability issues with increasing data volume.
Stephen et al. (2024)	Development of a deep learning-based cotton plant monitoring system using big data.	Applied deep learning models to big data for monitoring cotton plant growth.	Limited field validation; primarily simulation-based.
Sun et al. (2024)	Intelligent big data analysis without using MapReduce.	Introduced a non-MapReduce for analyzing big data using AI.	Lack of standardized frameworks for implementation.
Mitra (2024)	MapReduce using cellular automata for transitioning big data processing from Industry 4.0 to 5.0.	Designed a cellular automata-based MapReduce framework for Industry 5.0.	Compatibility issues with existing Industry 4.0 infrastructure.
Wu et al. (2024)	Big data quality scoring for structured data using MapReduce.	Implemented MapReduce to evaluate data quality scoring in structured big data.	Limited to structured data; lacks focus on unstructured data.
Lawrance et al. (2024)	To enhance big data anonymization using parallel Fuzzy C-Means clustering approach with Hadoop MapReduce.	Implemented a parallel FCM algorithm using Hadoop MapReduce to improve scalability and efficiency in big data anonymization.	High computational complexity of the FCM algorithm and performance issues in handling large data volumes.
Kavitha et al. (2024)	To analyze the application of K-Means clustering in smart city development and big data analysis.	Used K-Means clustering to analyze smart city data for optimizing urban services	Sensitivity to initial centroid selection and difficulties in determining the optimal number of clusters in complex datasets.
Jiang et al. (2024)	To enhance big data security and privacy through a prototype-assisted clustered federated learning framework.	Developed a federated learning framework with prototype-assisted clustering to improve privacy and accuracy in distributed model training.	Complexity in coordinating multiple clusters and challenges in ensuring scalability and fault tolerance in larger, more distributed environments.
L. Xing / 2023	A Secure Data Transmission Method for the Social Internet of Vehicles Based on Feature Clusters	(FC-SDTM)	Integrating suggested technique with big data frameworks infrastructures may cause compatibility challenges.
M. Safa / 2023	Improved quality of service in the prediction of heart illness using IoT devices using a real-time health care big data analytics approach	TFW, TCW, IoT	While seeking safe transmission, cryptographic methods or security protocols may add computational overhead, reducing data transmission efficiency.
R. Rawat / 2023	A Novel Machine Learning-Based Approach to Big Data Cybersecurity	KNN, SVM and MLP	A secure clustered environment may take more time and resources, raising operating overhead and decreasing efficiency.
J. Vasa / 2022	A Deep Dive into Deep Learning: Differential Privacy Protection in the Big Data Age	Deep Learning	To ensure data integrity and availability during node failures interruptions, clustered environments may struggle with fault tolerance.
U. Narayanan / 2022	Secure authentication and data sharing in a cloud-enabled Big Data Environment	SADS-Cloud	The clustered technique may create delay in data transfer owing to cluster node cooperation, reducing network efficiency.
Y. Himeur / 2022	AI and big data analytics for building management and automation systems	BAMSs	Setting up and administering the cluster infrastructure may make the clustered solution difficult to install and use.
Mahdi, M. A., (2021)	Review of scalable clustering algorithms for big data	Review of scalable clustering	Limited focus on real-time clustering and edge computing environments
C. L. Stergiou / 2020	Protected IoT-Based Machine Learning Environment for Big Data on the Cloud	IoT, Big Data	Maintaining data integrity and coherence across data clusters, particularly in dispersed situations,

			may be difficult.
S. Riaz / 2020	The present state of big data privacy and security as well as future directions for research in the cloud	big data	Cluster formation and maintenance may require computer resources
T. S. Bharati / 2020	Problems, difficulties, privacy, and security with large data	Challenges, issues, security	Inefficient in diverse operating settings due to its inability to react to changing network conditions and workloads.
Jain, P., (2019)	Enhance security and privacy of Big Data in MapReduce.	Improved MapReduce layer with security measures.	Potential overhead affecting processing speed.
S. Ikhlaq / 2016	Processing Large Data Sets in a Hadoop and Cloud Setting	Big Data	Research is not providing any practical solution
S. Y. Inamdar / 2016	Hadoop Distributed File System Data Security	Big Data	Need to enhance the security of system
V. B. Bobade / 2016	Research Report on Hadoop and Big Data	Big Data	Need to do more work on performance
A. Fuad / 2014	Apache Hive, MySQL cluster, and Apache Pig processing efficiency	Clustering Method	Concept of reduction is missing in this research.
A. Pal / 2014	A large data experiment to examine memory use on a Hadoop cluster	MapReduce	The influencing factor must be considered.
C. Vorapong-kitipun / 2014	Hadoop speed optimisation for small-file access	Hadoop	Need to do more work on performance
P. S. Patil / 2014	An Overview of Hadoop and Big Data Processing	Big Data Processing	There is lack of technical work
A. Kala Karun / 2013	review on hadoop - HDFS infrastructure extensions	MapReduce	There is need to do work in field of security
C. Zhang / 2013	An improved K-means clustering algorithm	Clustering Method	Lack of reduction space and time consumption
Eman Meslhy / 2013	Data Security Model for Cloud Computing	Cloud Computing	The scope of the research work is limited.
F. Roohi / 2013	ANN approach to clustering	Clustering Method	Lack of real implementation which is very important.
Shunmei Meng / 2013	Big Data Service on Map Reduce: KASR	MapReduce	To discuss architecture is not sufficient, all factors.
S. Narayan / 2012	Hadoop acceleration in an openflow-based cluster	Clustering methods	Research has provided limited security.
Youguo Li / 2012	A Clustering Method Based on K-Means Algorithm	Clustering Method	It is suffering due to lack of concept of security
G. Kumar / 2011	Review of cloud-based e-learning security concerns	Cloud computing	There is lack of performance during implementation.
A. R. Taylor / 2010	Hadoop, MapReduce, and HBase in bioinformatics	MapReduce	The scope of work is limited
M. Bhandarkar / 2010	Using Apache Hadoop for MapReduce programming	MapReduce	Need to enhance the security of system
M. G. H. Omran / 2007	An overview of clustering methods	Clustering methods	There is need to do work on performance.

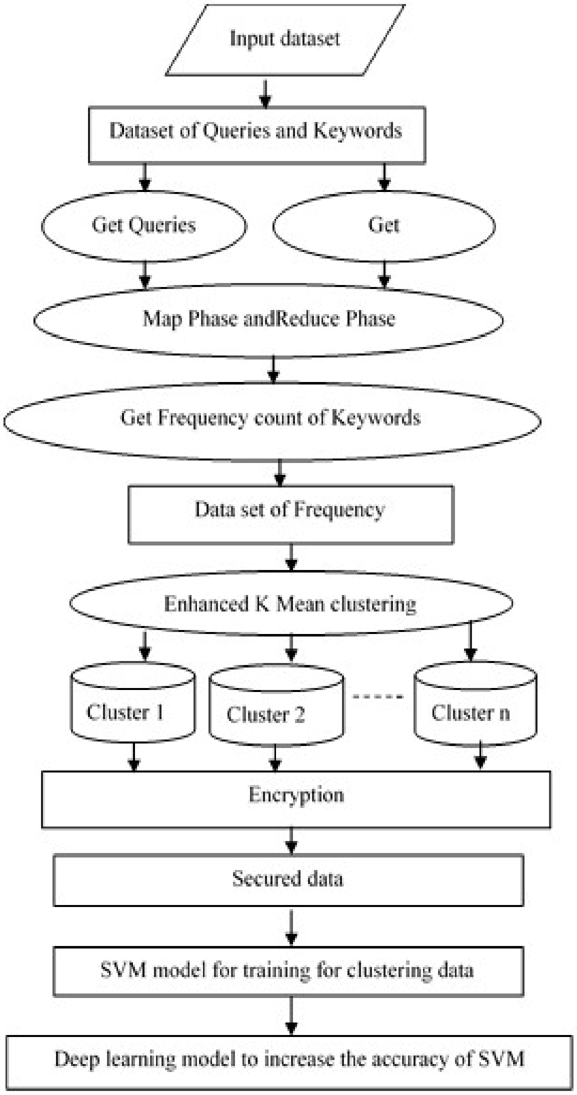
III. PROBLEM STATEMENT

Multiple big data studies have argued in favour of clustering. In order to create intelligent IoT solutions, some researchers advocated using big data platforms for malware detection, while others honed in on data clustering. Providing a more secure clustering approach is essential for better protecting Big Data during data transfer across networks. Managing massive volumes of data is doable or not. The results of this study indicate that research on enormous datasets that are not well-managed has been limited. Big data researchers often used the map reduction environment, but these methods weren't great for handling massive structured data sets. In addition, the clustering algorithms that were used in the previous research are not as successful. Research indicates that large data management systems should be more secure to prevent cyberattacks.

IV. PROPOSED WORK

The proposed Mapreduce and deep learning integrated big data security model system involves collecting a dataset of searches and keywords as input. The queries and keywords are derived from this dataset. With such queries and phrases, researchers are running the map and reduce phases of their study. At this point, the process gathers the dataset of frequency and obtains the count of keywords. Through research, K-mean clustering is improved and numerous clusters are obtained. Finally, the data is encrypted and safeguarded by the research. Dataset and frequency dataset, both of which are output by the map reduction technique, are represented by the genuine datasets by the rectangular rectangles. Original data flow activities are described using the elliptical boxes. The aforementioned data flow diagram depicts a series of operations. To begin, two processes are running in parallel to get the relevant dataset keywords and queries. Next, the gathered keywords and questions are subjected to the map reduction procedure to ascertain their frequency. Following the map reduction method, the acquire frequency approach is used to draw frequency of keywords and inquiries. The acquired frequency is saved in the frequency data set. The next step in creating the clusters is to run the clustering algorithm.

Data flow diagram for secure clustered approach for efficient protection of big data


Input dataset
Dataset of Queries and Keywords
Get Queries
Get
Map Phase andReduce Phase
Get Frequency count of Keywords
Data set of Frequency
Enhanced K Mean clustering
Cluster 1
Cluster 2

Cluster n
Encryption
Secured data
SVM model for training for clustering data
Deep learning model to increase the accuracy of SVM

Fig 1 Process flow of secure clustering logic.

To provide the clustered data even greater security, every cluster is secured using Advanced Encryption Standard (AES). Regarding safeguarding important enormous data, the symmetric encryption method AES is second only in speed and power. This protects the grouped data from any possible invaders during storage and processing. Further study on the encrypted clusters is conducted using Support Vector Machine thereafter. Regarding challenging classification tasks, supervised learning models such as SVM are very suited. We here pass the encrypted and grouped data into SVM for secondary clustering. SVM divides data into hyperplanes to help to further hone the clusters. Maximizing the performance of a clusterering and classification model depends on tuning hyperparameters like learning rate, epoch count, and batch size. Perfecting these values improves metrics like F1-score, recall, accuracy, and precision. We also use a deep learning approach to improve the SVM model's performance as it is not very precise. Grid search is one of the optimization techniques used to fine-tune the model so as to increase its accuracy and classification results. Finally, these measures evaluate the performance of the model to ensure that the goals related to security and clustering are satisfied. By means of encryption and enhanced machine learning techniques to ensure high data security, the proposed method not only overcomes the issue of clustering vast datasets but also makes big data management practical and scalable.

Process Flow for Clustering and Security of Big Data

- **Big Data Collection:** Whether they are structured, semi-structured, or unorganized, gather big data from many sources.
- **Data Preprocessing:** Using MapReduce, process and find the word frequency in the enormous data.
- **Initial Clustering (K-Means Clustering):** Perform K-Means clustering based on the word frequency to create initial clusters.
- **Data Encryption (AES Encryption):** Encrypt the data in each cluster using AES for data security. Store the encrypted data in the corresponding clusters.
- **Secondary Clustering (SVM-Based Clustering):** Apply SVM for further clustering by considering the already clustered big data. Optimize hyperparameters such as batch size, epoch, and learning rate to improve model performance.
- Deep learing model has been used to improve the accuracy.
- **Model Evaluation:** Evaluate model performance using accuracy, precision, recall, and F1-score metrics.

Proposed Mapreduce And Deep Learning Integrated Big Data Security Model

The architecture illustrated here has five layers of organisation. The operator may try to send data to the server-side script from the first layer, which is a client-side layer. Before being transmitted to server-side script for processing, his data would be checked by the client-side script. The second layer consists of a server-side script that stores data in a database after validation. Data storage and maintenance occur at the database layer, the third tier of data management. Levels two and three are connected in both directions. As part of the insert query process, server transmits data to the database layer. The execution of a selection query involves the transmission of data from the database layer to server side. The fourth layer employs map reduction and grouping to generate frequency. Lastly, the content is protected by using encryption. Following figure is presenting the interconnectivity between front end and backend where frontend makes input to application layer and application layer is connected to map reduce, clustering and security module.

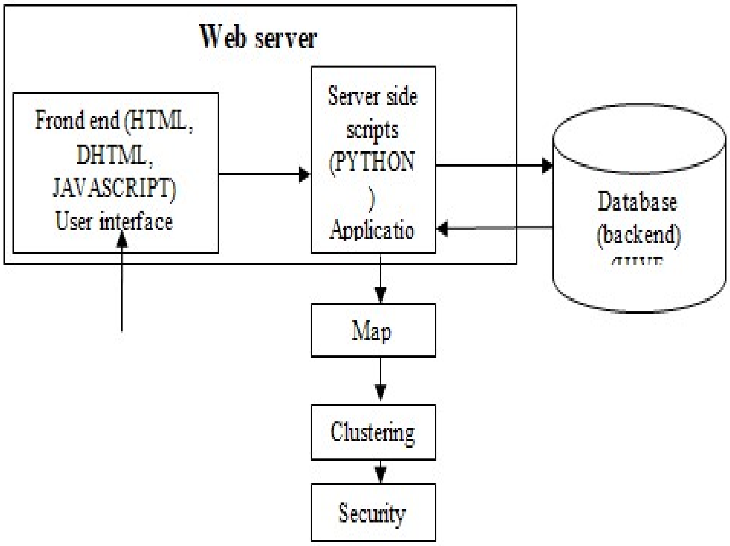

Web server
Frond end (HTML, DHTML, JAVASCRIPT) User interface
Server side scripts (PYTHON) Applicatio
Database (backend)
Map
Clustering
Security

Fig 2 Interconnectivity between frontend and backend in proposed design

Process flow is for a Python-based Mapreduce and deep learning integrated big data security model system where MapReduce is applied over data, followed by K-Means clustering, and finally, a deep learning model using SVM for classification.

Data preparation and preprocessing

Step 1.1: Data collection: Collect and store data from various sources (e.g., databases, files, APIs).

Step 1.2: Data cleaning and preprocessing: Remove missing values, outliers, and irrelevant features. Normalize or standardize the data as needed.

MapReduce

Step 2.1: Implement map function: Define map function to process along with transform chunks of data.

Step 2.2: Implement Reduce Function: Define reduce function to aggregate and summarize the results from the map phase.

Step 2.3: Execute MapReduce Job: Apply the MapReduce job over the dataset to process the data in parallel.

KMeans clustering

Step 3.1: Define the KMeans Model: Set number of clusters k for K-means algorithm

Step 3.2: Fit the KMeans Model: Fit the KMeans model on the preprocessed data. Obtain cluster labels for each data point.

Step 3.3: Add Cluster Labels to Data: Append the cluster labels as a new feature to the original data.

Deep Learning Model with SVM for Classification

Step 4.1: Data Preparation: Reshape and prepare the data to fit the input requirements of the SVM model and deep learning model. Split data into training along with testing sets.

Step 4.2: Define the SVM Model Architecture: Specify the SVM layers, dense layers, and activation functions. Compile model with an appropriate loss function, optimizer, and metrics.

Step 4.3: Train the SVM Model: Train model using training dataset. Validate the model on the validation dataset.

Step 4.4: Evaluate the SVM Model: Evaluate model's performance on the test dataset. Calculate performance metrics such as accuracy, precision, recall, along with F1-score.

Step 4.5: Define the Deep Learning Architecture: Specify the layers and hyper parameters for deep learning model such as epoch, learning rate, batch size, optimization.

Step 4.6: Train the Deep Learning Model: Train model using training dataset. Validate the model on the validation dataset.
Step 4.7: Evaluate the accuracy of deep learning Model: Evaluate model's performance on the test dataset. Calculate performance metrics such as accuracy, precision, recall, along with F1-score.

Results and Visualization

Step 5.1: Visualize Clustering Results: Plot the clusters and visualize the distribution of data points.
Step 5.2: Visualize SVM Classification Results: Plot the training and validation loss/accuracy curves. Visualize the confusion matrix and other relevant performance metrics.
Step 5.3: Visualize Deep learning Classification Results: Plot the training and validation loss/accuracy curves. Visualize the confusion matrix and other relevant performance metrics.

Deployment and Monitoring

Step 6.1: Model Deployment: Deploy the trained SVM and deep learning model to a production environment for real-time classification.
Step 6.2: Model Monitoring and Maintenance: Monitor the model's performance in production. Update the model periodically with new data and retrain if necessary.

To find out where the system is falling short of its performance objectives, it is vital to analyze it using these measurements. The recommended architecture is a secure and scalable solution for real-world big data applications because it follows a systematic, step-by-step strategy to properly cluster, encrypt, and categorize large data.

V. RESULT AND DISCUSSION

This presents results of experiments conducted to evaluate performance along with security of the proposed secure clustering model. It includes a detailed analysis of the accuracy, processing speed, and efficiency of Hive, Pig, and Spark in handling the datasets. The chapter also compares the results of optimized and non-optimized datasets, highlighting the improvements achieved through the proposed approach. Overall, the research highlights the importance of efficient data processing tools in the Hadoop framework, particularly in cloud-based environments, where speed and accuracy are paramount. The comparative analysis between Hive, Pig, and the proposed Python script underscores the advancements made in data processing techniques, showcasing the potential for optimized methods to enhance big data security and performance.

Execution Times Taken By Hive And Pig With Mapreduce

Experimental pigs and beehives are common. These are used while undertaking. All of the aforementioned data on the hive and pigs' performance is saved in csv files. The time savings achieved by Hive were more noticeable as compared to Pig. Hive is more efficient than a pig when it comes to getting things done. This is in addition to the many lines that have been found in Pig. In contrast, the hive has used only a single line.

Table 2 Execution times taken by hive and pig with MapReduc.

S. No.	Time taken by PIG	Time taken by Hive
Query 1	12.975	7.572
Query 2	11.327	3.128
Query 3	16.782	1.659
Query 4	20.527	5.366

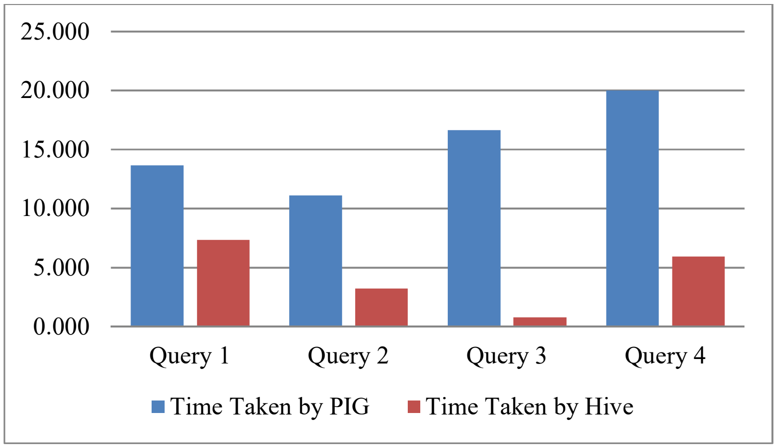


Fig 3 Execution time in case of hive and pig with MapReduce.

Using Spark, it has been executed in memory. This is why Hadoop MapReduce requires disc reading and disc writing capabilities. As a result, processing rates might vary greatly. Evidence suggests that Spark is a hundred times faster than MapReduce. This led to the observation of time required for Hive along with pig with Spark in the same architecture. They can see how long it takes a pig and a hive to get going in the table below. Currently, four questions are under review.

Table 3 Time required in case of hives and pig with spark.

Query	Time taken by PIG	Time taken by Hive
1	0.12	0.04
2	0.17	0.01
3	0.21	0.06
4	0.15	0.03

This is the graph they made to compare the hive's and PIG's performance.

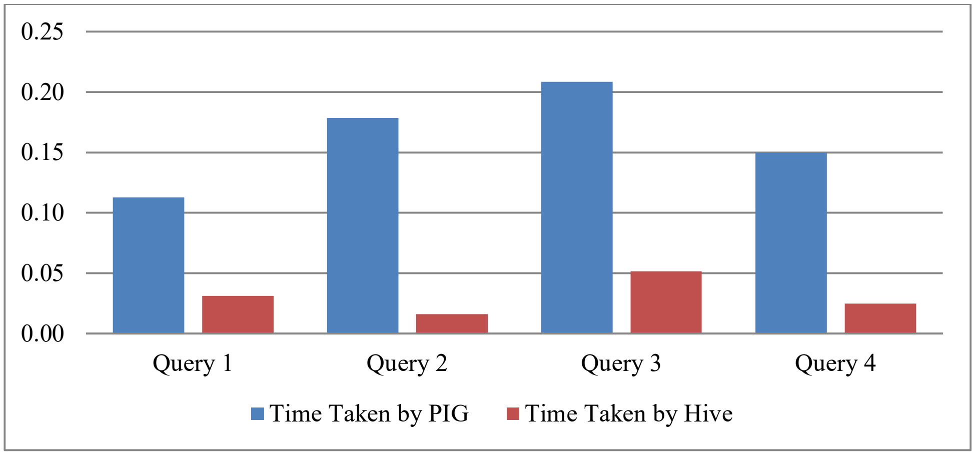


Fig 4 Execution by hive and pig with Spark

Comparative Analysis Of Performance

Following the hive simulation, tables containing pig and proposed Mapreduce and deep learning integrated big data security model where content were generated to facilitate the analysis.

Table 4 Hive, pig and Proposed Mapreduce and deep learning integrated big data security model content have been stored.

Cycle	Tim Taken by hive	Time taken by pig	Time taken by proposed Mapreduce and deep learning integrated big data security
1	0.218	0.405	0.045
2	0.182	0.531	0.058
3	0.188	0.284	0.122
4	0.117	0.308	0.101
5	0.172	0.463	0.117
6	0.184	0.418	0.122
7	0.126	0.374	0.089

Fig 5.3 Hive, pig and proposed work content have been stored

Confusion Matrix Of Non-Optimization Technique

Share your findings and assess the experiments' performance in terms of security and data transfer efficiency. To show how your strategy improves upon the status quo, you might look at comparable approaches or benchmarks.

Table 5 Confusion matrix of non-optimization technique.

	Cluster 1	Cluster 2	Cluster 3
Cluster 1	1900	71	60
Cluster 2	90	2590	85
Cluster 3	190	110	3470

Table 6 serves as the confusion matrix for the calculations of accuracy, precision, recall, along with f-score, which are all shown in table 5. The reliability is 92.93 percent utilising the raw data.

Table 6 Accuracy parameter of non-optimization.

Class	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score
1	2180	2031	95.2%	0.94	0.87	0.90
2	2771	2765	95.84%	0.94	0.93	0.94
3	3615	3770	94.81%	0.92	0.96	0.94

Confusion Matrix Of Optimization Technique

Table 7 displays confusion matrix for three-class non-optimization dataset. Each cluster ID has a slightly distinct frequency distribution.

Table 7 Confusion matrix of optimization technique.

	Cluster 1	Cluster 2	Cluster 3
Cluster 1	1911	70	50
Cluster 2	85	2600	80
Cluster 3	180	100	3490

Table 5.6 serves as the confusion matrix for the calculations of accuracy, precision, recall, along with f-score, which are all shown in table 7. The reliability of the raw data is 93.4 percent.

Table 8 Accuracy parameter of optimization.

Class	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score
1	2176	2031	95.51%	0.94	0.88	0.91
2	2770	2765	96.09%	0.94	0.94	0.94
3	3620	3770	95.21%	0.93	0.96	0.94

COMPARISON OF ACCURACY

Table 5.8 contains the overall accuracy in case of non-optimization dataset and optimization dataset.

Table 9 Overall accuracy.

Non optimized	Optimized
92.93	93.4

Table 9 shows a comparison of accuracy of Optimization dataset and non-Optimization dataset. Figure 5 displays these results.

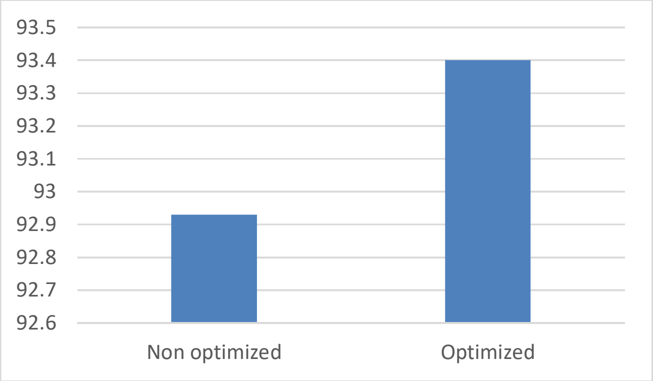


Fig 5 Comparison of overall Accuracy for optimized and non-optimized model.

Each data point inside a cluster should be relatively close to the cluster's centre, an idea captured by the k-means method. Methodology Here's how it works: Initially, we decide on k, the target number of clusters to discover in the data.

Simulation Of K-Mean Clustering

The k-means algorithm takes into account the realisation that nodes in a cluster should be too far from the cluster's epicentre. Step one is to determine how many clusters, or groups, should be extracted from the data, denoted by the variable k.

Table 10 Simulation of Kmean clustering.

Cluster id	Frequency
1	2032
2	3012
3	4089
4	2004
5	5234

Table 10 displays confusion matrix for non-optimization dataset, which consists of five classes. The frequencies of the various cluster IDs vary.

Table 11 Confusion matrix of non optimization technique.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Cluster 1	1827	72	61	39	33
Cluster 2	91	2598	84	94	145
Cluster 3	194	112	3476	98	209
Cluster 4	59	41	54	1738	112
Cluster 5	234	198	202	151	4449

Using table 11 as confusion matrix, the following metrics were calculated and shown in table 11: accuracy, precision, recall, along with f-score. Accuracy rate is 86.05% while working with the raw data.

Table 11 Accuracy parameter of non optimization.

Class	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score
1	2405	2032	95.22%	0.90	0.76	0.82
2	3021	3012	94.89%	0.86	0.86	0.86
3	3877	4089	93.81%	0.85	0.90	0.87
4	2120	2004	96.04%	0.87	0.82	0.84
5	4948	5234	92.16%	0.85	0.90	0.87

In Table 12 we can see confusion matrix for 5-class non-optimization dataset. The frequencies of the various cluster IDs vary.

Table 12 Confusion matrix of optimization technique.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Cluster 1	1827	72	61	39	33
Cluster 2	91	2598	84	94	145
Cluster 3	194	112	3476	98	209
Cluster 4	59	41	54	1738	112
Cluster 5	234	198	202	151	4449

Using table 12 as confusion matrix, following metrics were calculated and shown in table 13 in form of accuracy, precision, recall, along with f-score. With the raw data, the accuracy rate is above 92.2%.

Table 13 Accuracy parameter of optimization.

Class	n (truth)	n (classified)	Accuracy	Precision	Recall	F1 Score
1	2203	2032	97.37%	0.94	0.86	0.90
2	3017	3012	97.03%	0.92	0.92	0.92
3	3997	4089	96.57%	0.92	0.94	0.93
4	2072	2004	97.63%	0.92	0.89	0.90
5	5082	5234	95.81%	0.92	0.95	0.93

Comparison Of Accuracy

Accuracy of SVM model and deep learning model have been compared in present simulation.

Table 14 Overall accuracy.

Conventional SVM-only Model	Conventional MapReduce-only Model	Non-Optimized Classification Model	Proposed Mapreduce and deep learning integrated big data security model
81.23%	87.35%	91.12%	92.2%

Following figure 6 shows comparison of accuracy of SVM and deep learning model.

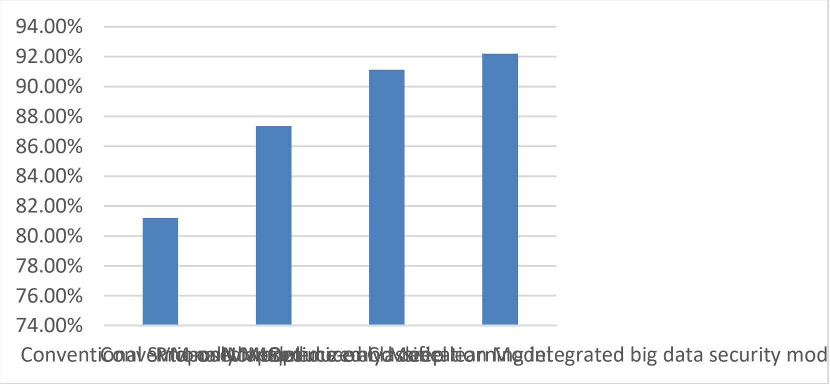


Fig 6 Accuracy comparison of SVM and Deep learning

VII. CONCLUSION

The study aims to provide a secure clustering approach for better protection of big data during data transfer across networks. It focuses on addressing security risks associated with structured and unstructured data and identifying factors influencing the performance of big data transmission. The main objectives of this work are to review the literature, investigate clustering and encryption methods for enhancing big data security during higher transmission, suggest a model that can secure data transmission using encryption and improve performance through clustering, and evaluate the suggested model in relation to current techniques in terms of security and performance. The proposed architecture consists of five layers: client-side, server-side, database, and third tier. The first layer checks data before being transmitted to server-side script for processing, the second layer stores data in a database after validation, and the third tier handles data storage and maintenance. The proposed model incorporates an optimization mechanism, yielding superior accuracy parameters compared to a non-optimized method. The study demonstrates superior performance in several accuracy criteria, including precision, recall, and f1-score. The study focuses on the implementation of a secure clustering paradigm for large data transmission security and performance. It involves installing a virtual machine, setting up the cloud era, loading a CSV file into HDFS, evaluating hive's efficiency, building a Python module for reading CSV files, applying MapReduce and Spark to datasets, and evaluating their performance. The data is then extracted and moved to a cloud storage system using Pig, Hadoop, and Hive. Python scripts are built using gedit, and three data retrieval approaches are used: Hive, MAP Reduce, and SPARK techniques. The research extends into a Cloudera-based system, seamlessly integrated with the Hadoop environment. The study compared the performance of hives and pigs using Hadoop frameworks, Spark, and a Python script. Results showed that Hive was more efficient than Pig, with a hundred times faster processing rate. Spark was found to be more efficient and useful for Hadoop. The secure clustering model improves massive data transfer accuracy and speed, with optimized datasets performing better. The report provides a solid basis for cloud data safety and efficiency.

VII.FUTURE SCOPE

The use of clustered transmission mechanisms has potential applications in healthcare, AI, and IoT. However, unmanaged data could have significant implications for large-scale data analysis. To enhance speed and maintain data security, a system integrates encryption and clustering techniques. Clustering is a fundamental technique for uncovering information in data exploration and plays a pivotal role in data mining. The efficacy of hierarchical clustering depends on the utilization of relevant values for column properties. The findings of this study pave the way for future research, particularly in optimizing big data processing within the Hadoop framework. Future research could focus on integrating Hive, Pig, and Python scripts to create more versatile and powerful data processing frameworks. Enhancing clustering algorithms to raise data classification and clustering accuracy could also be a focus. Secure clustering methods aim to improve data safety during network transfers, and future studies could investigate developing better encryption techniques specifically designed for big data clustering techniques. Building upon the architecture proposed in this work, more sophisticated models for large data management could be developed. Future research could also explore tying IoT devices with large data systems, focusing on the security, performance, and efficiency of handling big data in various contexts.

REFERENCES

1. Al-Sai, Z. A., Abdullah, R., & Husin, M. H. (2020). Critical success factors for big data: a systematic literature review. IEEE Access, 8, 118940-118956.
2. Arjunan, T. (2024). Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. International Journal for Research in Applied Science and Engineering Technology, 12(9), 10-22214.
3. Bag, S., Wood, L. C., Xu, L., Dhamija, P., & Kayikci, Y. (2020). Big data analytics as an operational excellence approach to enhance sustainable supply chain performance. Resources, conservation and recycling, 153, 104559.
4. Bante, P. M., & Rajeswari, K. (2017). Big Data Analytics Using Hadoop Map Reduce Framework and Data Migration Process. In 2017 International Conference on Computing, Communication, Control and Automation (pp. 1–5).
5. Barragán, D., & Manero, J. (2020). How Big Data and Artificial Intelligence Can Help Against COVID-19. IE Business School, 4–11. https://www.ie.edu/business-school/news-and-events/whats-going-on/big-data-artificial-intelligence-can-help-covid-19/
6. Batko, K., & Ślęzak, A. (2022). The use of Big Data Analytics in healthcare. Journal of big Data, 9(1), 3.
7. Belcastro, L., Cantini, R., Marozzo, F., Orsino, A., Talia, D., & Trunfio, P. (2022). Programming big data analysis: principles and solutions. Journal of Big Data, 9(1), 4.
8. Bhandarkar, M. (2010). MapReduce programming with apache Hadoop. In 2010 IEEE International Symposium on Parallel &amp; Distributed Processing (IPDPS). IEEE.
9. Bharati, T. S. (2020). Challenges, issues, security and privacy of big data. International Journal of Scientific and Technology Research, 9(2), 1482–1486.
10. Bobade, V. B. (2016). Survey paper on big data and Hadoop. Int. Res. J. Eng. Technol, 3(1), 861-863.
11. Chebbi, I., Boulila, W., Mellouli, N., Lamolle, M., & Farah, I. R. (2018). A comparison of big remote sensing data processing with Hadoop MapReduce and Spark. In 2018 4th International Conference on Advanced Technology & Signal Image Processing (ATSIP) (pp. 1–4).

12. Cheng, D., Zhou, X., Lama, P., Wu, J., & Jiang, C. (2017). Cross-Platform Resource Scheduling for Spark and MapReduce on YARN. IEEE Transactions on Computers, 66(8), 1341–1353.

13. Deutsch, E. W., Bandeira, N., Sharma, V., Perez-Riverol, Y., Carver, J. J., Kundu, D. J., ... & Vizcaíno, J. A. (2020). The ProteomeXchange consortium in 2020: enabling 'big data'approaches in proteomics. Nucleic acids research, 48(D1), D1145-D1152. https://doi.org/10.1093/nar/gkz984

14. Eman Meslhy, & Abd Elkader, Hatem & Eletriby, Sherif. (2013). Data Security Model for Cloud Computing. Journal of Communication and Computer 10 (2013) 1047-1062. 10. 1047-1062.

15. Fuad, A., Erwin, A., & Ipung, H. P. (2014). Processing performance on Apache Pig, Apache Hive and MySQL cluster. In Proceedings of International Conference on Information, Communication Technology and System (ICTS) 2014. IEEE. https://doi.org/10.1109/icts.2014.7010600

16. Garg, P., & Sharma, M. (2016). Study of Big Data Technology & Analysis. In Proceedings of GCRSTS-2016 (Global Challenges-Role of Sciences & Technology in Imparting Their Solutions) (pp. 23-24). T.I.T Bhiwani, India: Publisher.

17. Garg, V. (2015). Optimization of Multiple Queries for Big Data with Apache Hadoop/Hive. In 2015 International Conference on Computational Intelligence and Communication Networks (CICN). IEEE. https://doi.org/10.1109/cicn.2015.184

18. Guan, S., Zhang, C., Wang, Y., & Liu, W. (2024). Hadoop-based secure storage solution for big data in cloud computing environment. Digital Communications and Networks, 10(1), 227–236. https://doi.org/10.1016/j.dcan.2023.01.014

19. Haddad, O., Fkih, F., & Omri, M. N. (2024). An intelligent sentiment prediction approach in social networks based on batch and streaming big data analytics using deep learning. Social Network Analysis and Mining, 14(1), 150.

20. Hancock, J. T., & Khoshgoftaar, T. M. (2020). CatBoost for big data: an interdisciplinary review. Journal of big data, 7(1), 94.

21. Hasan, Md. M., Popp, J., & Oláh, J. (2020). Current landscape and influence of big data on finance. In Journal of Big Data (Vol. 7, Issue 1). Springer Science and Business Media LLC.

22. Hassani, A., Walton, S., Shah, N., Abuduweili, A., Li, J., & Shi, H. (2021). Escaping the big data paradigm with compact transformers. arXiv preprint arXiv:2104.05704.

23. Himeur, Y., et al. (2022). AI-big data analytics for building automation and management systems: a survey, actual challenges and future perspectives. Artificial Intelligence Review, 56(6), 4929–5021.

24. Huang, X., & Su, W. (2014). An Improved K-means Clustering Algorithm. In Journal of Networks (Vol. 9, Issue 1). Academy Publisher.

25. Jain, P., Gyanchandani, M., & Khare, N. (2019). Enhanced Secured Map Reduce layer for Big Data privacy and security. In Journal of Big Data (Vol. 6, Issue 1). Springer Science and Business Media LLC.

26. Jiang, Y., Wang, D., Song, B., & Du, X. (2024). A prototype-assisted clustered federated learning for big data security and privacy preservation. Future Generation Computer Systems, 161, 376-389.

27. Kala Karun, A., & Chitharanjan, K. (2013). A review on hadoop &amp;#x2014; HDFS infrastructure extensions. In 2013 IEEE Conference on Information & Communication Technologies (ICT). IEEE. https://doi.org/10.1109/cict.2013.6558077

28. Kar, A. K., & Dwivedi, Y. K. (2020). Theory building with big data-driven research – Moving away from the 'What' towards the 'Why.' International Journal of Information Management, 54(July), 102205. https://doi.org/10.1016/j.ijinfomgt.2020.102205

29. Kavitha, V., Sundaravadivazhagan, B., Karthikeyan, R., & Hemashree, P. (2024). Analysing the Application of a Smart City in Big Data Using K-Means Clustering Algorithm. In Handbook of Artificial Intelligence for Smart City Development (pp. 157-172). CRC Press.

30. Kumar, G., & Chelikani, A. (2011). Analysis of security issues in cloud-based e-learning. University of Borås/School of Business and IT.

31. Lawrance, J. U., Jesudhasan, J. V. N., & Thampiraj Rittammal, J. B. (2024). Parallel Fuzzy C-Means Clustering Based Big Data Anonymization Using Hadoop MapReduce. Wireless Personal Communications, 1-28.

32. Li, Youguo & Wu, Haiyan. (2012). A Clustering Method Based on K-Means Algorithm. Physics Procedia. 25. 1104-1109. 10.1016/j.phpro.2012.03.206.

33. Lv, Z., & Qiao, L. (2020). Analysis of healthcare big data. In Future Generation Computer Systems (Vol. 109, pp. 103–110). Elsevier BV. https://doi.org/10.1016/j.future.2020.03.039

34. M. Dhavapriya, N. Yasodha, & I. Introduction. (2016). Big Data Analytics: Challenges and Solutions Using Hadoop, Map Reduce and Big Table. International Journal of Computer Science Trends and Technology, 4(1), 5–14. https://www.ijcstjournal.org

35. Mahdi, M. A., Hosny, K. M., & Elhenawy, I. (2021). Scalable Clustering Algorithms for Big Data: A Review. In IEEE Access (Vol. 9, pp. 80015–80027). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2021.3084057

36. Meng, S., Dou, W., Zhang, X., & Chen, J. (2014). KASR: A Keyword-Aware Service Recommendation Method on MapReduce for Big Data Applications. In IEEE Transactions on Parallel and Distributed Systems (Vol. 25, Issue 12, pp. 3221–3231). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/tpds.2013.2297117

37. Merla, P. R., & Liang, Y. (2018). Data analysis using Hadoop MapReduce environment. In Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017 (pp. 4783–4785).

38. Misra, N. N., Dixit, Y., Al-Mallahi, A., Bhullar, M. S., Upadhyay, R., & Martynenko, A. (2020). IoT, big data, and artificial intelligence in agriculture and food industry. IEEE Internet of things Journal, 9(9), 6305-6324.

39. Mitra, A. (2024). Cellular automata-based MapReduce design: Migrating a big data processing model from Industry 4.0 to Industry 5.0. e-Prime-Advances in Electrical Engineering, Electronics and Energy, 8, 100603.

40. Mohammed, A. Q., & Bharati, R. (2018). An efficient technique to improve resources utilization for Hadoop MapReduce in heterogeneous system. In ICCT 2017 - International Conference on Intelligent Communication and Computational Technologies (pp. 12–16).

41. Monika Sharma, Dr. Sat Pal, "A Review on Storage and Large-Scale Processing of Data-Sets using MapReduce, Yarn, Spark, Avro, MongoDB" International Conference on SUSCOM-2019 Feb26-28,2019 at Amity University Rajasthan, Jaipur.

42. Monika Sharma, Dr. Sat Pal, "Analysis of Big Data techniques and Technologies using Hadoop Framework (MapReduce, Spark, Yarn, HBase)"- National Conference on" Challenges & Opportunities in Commerce, Management, Economics and Computer Science" at Baba MasthNath University,Asthal Bohar,Rohtak (March 28,2019).

43. Monika Sharma, Dr. Sat Pal, "Simulation of Performance Analysis of MongoDB, Pig, Hivstorage, MapReduce, Spark and Yarn" International Conference on SUSCOM-2019 Feb26-28,2019 at Amity University Rajasthan, Jaipur.

44. Narayan, S., Bailey, S., & Daga, A. (2012). Hadoop Acceleration in an OpenFlow-Based Cluster. In 2012 SC Companion: High Performance Computing, Networking Storage and Analysis. 2012 SC Companion: High Performance Computing, Networking, Storage and Analysis (SCC). IEEE. https://doi.org/10.1109/sc.companion.2012.76

45. Narayanan, U., Paul, V., & Joseph, S. (2022). A novel system architecture for secure authentication and data sharing in cloud enabled Big Data Environment. Journal of King

Saud University - Computer and Information Sciences, 34(6), 3121–3135. https://doi.org/10.1016/j.jksuci.2020.05.005

46. Nazir, S., Khan, S., Khan, H. U., Ali, S., Garcia-Magarino, I., Atan, R. B., & Nawaz, M. (2020). A Comprehensive Analysis of Healthcare Big Data Management, Analytics and Scientific Programming. In IEEE Access (Vol. 8, pp. 95714–95733). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2020.2995572
47. Nehe, N.S. (2016). Malware and Log file Analysis Using Hadoop and Map Reduce. International Journal of Engineering Development and Research, 4, 529-533.
48. Omran, M. G. H., Engelbrecht, A. P., & Salman, A. (2007). An overview of clustering methods. In Intelligent Data Analysis (Vol. 11, Issue 6, pp. 583–605). IOS Press. https://doi.org/10.3233/ida-2007-11602.
49. Osinga, S. A., Paudel, D., Mouzakitis, S. A., & Athanasiadis, I. N. (2022). Big data in agriculture: Between opportunity and solution. Agricultural Systems, 195, 103298.
50. Pal, A., & Agrawal, S. (2014). An experimental approach towards big data for analyzing memory utilization on a hadoop cluster using HDFS and MapReduce. In 2014 First International Conference on Networks &amp; Soft Computing (ICNSC2014). 2014 International Conference on Networks & Soft Computing (ICNSC). IEEE. https://doi.org/10.1109/cnsc.2014.6906718
51. Pandagale, A. A., & Surve, A. R. (2016). Big Data Analysis Using Hadoop Framework. International Journal of Computer Science Trends and Technology, 3(1), 87–91.
52. Panigrahi, S., & Kumar, S. M. (2016). A Survey on Social Data Processing Using Apache Hadoop, Map-Reduce. International Journal of Computer Science Trends and Technology, 2(2), 121–123.
53. Patil, P. S., & Phursule, R. (2014). Survey paper on big data processing and hadoop components. International Journal of Science and Research (IJSR), 3(10), 585-590.
54. Pramanik, P. K. D., Pal, S., & Mukhopadhyay, M. (2022). Healthcare big data: A comprehensive overview. Research anthology on big data analytics, architectures, and applications, 119-147.
55. Qi, C. (2020). Big data management in the mining industry. In International Journal of Minerals, Metallurgy and Materials (Vol. 27, Issue 2, pp. 131–139). Springer Science and Business Media LLC. https://doi.org/10.1007/s12613-019-1937-z
56. Rajeshkumar, K., Dhanasekaran, S., & Vasudevan, V. (2024). A novel three-factor authentication and optimal mapreduce frameworks for secure medical big data transmission over the cloud with shaxecc. Multimedia Tools and Applications. https://doi.org/10.1007/s11042-024-18147-6.a
57. Ramachandra, M. N., Srinivasa Rao, M., Lai, W. C., Parameshachari, B. D., Ananda Babu, J., & Hemalatha, K. L. (2022). An efficient and secure big data storage in cloud environment by using triple data encryption standard. Big Data and Cognitive Computing, 6(4), 101.
58. Rattanaopas, K., & Kaewkeeree, S. (2017). Improving Hadoop MapReduce performance with data compression: A study using wordcount job. In ECTI-CON 2017 - 2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (pp. 564–567).
59. Rawat, R., Oki, O. A., Sankaran, K. S., Olasupo, O., Ebong, G. N., & Ajagbe, S. A. (2023). A New Solution for Cyber Security in Big Data Using Machine Learning Approach. In Mobile Computing and Sustainable Informatics (pp. 495–505). Springer Nature Singapore. https://doi.org/10.1007/978-981-99-0835-6_35
60. Riaz, S., Khan, A. H., Haroon, M., Latif, S., & Bhatti, S. (2020). Big data security and privacy: Current challenges and future research perspective in cloud environment. Proceedings of 2020 International Conference on Information Management and Technology (ICIMTech 2020), 977–982. https://doi.org/10.1109/ICIMTech50083.2020.9211239

61. Roohi, F. (2013). Artificial neural network approach to clustering. Int. J. Eng. Sci.(IJES), 2(3), 33-38.
62. Rossi, R., & Hirama, K. (2022). Characterizing big data management. arXiv preprint arXiv:2201.05929.
63. S. Ikhlaq and B. Keswani, "Computation of Big Data in Hadoop and Cloud Environment," vol. 06, no. 01, pp. 31–39, 2016.
64. S. Y. Inamdar, A. H. Jadhav, R. B. Desai, P. S. Shinde, I. M. Ghadage, and A. A. Gaikwad, "Data Security in Hadoop Distributed File System," pp. 939–944, 2016.
65. Safa, M., Pandian, A., Gururaj, H. L., Ravi, V., & Krichen, M. (2023). Real time health care big data analytics model for improved QoS in cardiac disease prediction with IoT devices. Health and Technology, 13(3), 473–483. https://doi.org/10.1007/s12553-023-00747-1
66. Schintler, L. A., & McNeely, C. L. (Eds.). (2022). Encyclopedia of big data. Cham: Springer International Publishing.
67. Scott, S. L., Blocker, A. W., Bonassi, F. V., Chipman, H. A., George, E. I., & McCulloch, R. E. (2022). Bayes and big data: The consensus Monte Carlo algorithm. In Big Data and Information Theory (pp. 8-18). Routledge.
68. Sharma, M., & Garg, P. (2016). "Big Data" Analysis using Hadoop and MongoDB. International Journal of Modern Computer Science, 4(3). Retrieved from ISSN 2320-7868.
69. Sharma, M., & Pal, S. (2018). Big Data Analytic Techniques with Hadoop Framework for Health Care System: A Review. International Journal of Research and Analytical Reviews (IJRAR), 5(4). Retrieved from E-ISSN 2348-1269, P-ISSN 2349-5138.
70. Shi, Y., & Shi, Y. (2022). Big data and big data analytics. Advances in Big Data Analytics: Theory, Algorithms and Practices, 3-21.
71. Sowan, B., & Qattous, H. (2017). A Data Mining of Supervised learning Approach based on K-means Clustering. IJCSNS International Journal of Computer Science and Network Security, 17(1), 18–24.
72. Stephen, A., Arumugam, P., & Arumugam, C. (2024). An efficient deep learning with a big data-based cotton plant monitoring system. International Journal of Information Technology, 16(1), 145-151.
73. Stergiou, C. L., Plageras, A. P., Psannis, K. E., & Gupta, B. B. (2020). Secure Machine Learning Scenario from Big Data in Cloud Computing via Internet of Things Network. In B. B. Gupta (Ed.), Handbook of Computer Networks and Cyber Security (pp. 525–554). Springer International Publishing. https://doi.org/10.1007/978-3-030-22277-2_21
74. Sun, J., Gan, W., Chen, Z., Li, J., & Yu, P. S. (2022). Big data meets metaverse: A survey. arXiv preprint arXiv:2210.16282.
75. Sun, X., Zhao, L., Chen, J., Cai, Y., Wu, D., & Huang, J. Z. (2024). Non-MapReduce computing for intelligent big data analysis. Engineering Applications of Artificial Intelligence, 129, 107648.
76. Talukder, M. A., et al. (2024). Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. Journal of Big Data, 11(1). https://doi.org/10.1186/s40537-024-00886-w
77. Tang, L., Li, J., Du, H., Li, L., Wu, J., & Wang, S. (2022). Big data in forecasting research: a literature review. Big Data Research, 27, 100289.
78. Taylor, Ronald. (2010). An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics. BMC bioinformatics. 11 Suppl 12. S1. 10.1186/1471-2105-11-S12-S1.
79. Vasa, J., & Thakkar, A. (2022). Deep Learning: Differential Privacy Preservation in the Era of Big Data. Journal of Computer Information Systems, 63(3), 608–631.