RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

Real-time Processing of Big Data in Cloud Computing Environments

By

Alkhansa Alawi B Shakeabubakor
Department of information science, Faculty of Computer and Information System, Um Alqura University, Saudi Arabia.
Email: aashakeabubakor@uqu.edu.sa

Abstract

This paper discusses the challenges and opportunities of real-time processing of big data in cloud computing environments. It covers the current state of cloud computing and big data technologies, and the increasing demand for real-time processing capabilities in big data applications. The paper also provides a comprehensive overview of the current approaches to real-time processing in cloud environments and the key trade-offs between these approaches. The authors present a case study of a cloud-based real-time processing system, highlighting its design considerations, implementation details, and performance results. The paper concludes by discussing the future directions and open research questions in real-time processing of big data in cloud computing environments.

Keywords: Real-time, big data, cloud computing

Introduction

The growth of big data and cloud computing has been phenomenal in recent years. Organizations are collecting and storing vast amounts of data, and cloud computing is providing them with the scalability and cost-effectiveness to manage this data. However, processing big data in real-time is still a significant challenge, as traditional data processing techniques are unable to keep up with the volume, velocity, and variety of big data. The need for real-time processing of big data has increased as organizations look to derive insights from their data in real-time and make informed decisions in a timely manner. (Singh, et al. 2022)

Big data refers to data sets that are too large, complex, or fast-changing to be processed using traditional data processing techniques. These data sets come from a variety of sources, including social media, IoT devices, and transactional systems, and they are increasing in volume, velocity, and variety. The challenge with big data is that it is difficult to process, store, and manage it effectively. (Li, et al. 2022)

Cloud computing, on the other hand, provides organizations with the scalability and cost-effectiveness to store and process big data. With cloud computing, organizations can rent computing resources from a cloud provider and only pay for what they use. This eliminates the need for organizations to invest in expensive hardware and software and makes it easier for them to scale their computing resources as their needs change. (Singh, et al. 2018)

However, processing big data in real-time in a cloud computing environment is still a significant challenge. Traditional data processing techniques, such as batch processing, are unable to keep up with the volume, velocity, and variety of big data. This has led to the development of new techniques, such as stream processing and complex event processing, which are designed to handle real-time data processing. (Oduwole, et al. 2022)

Stream processing is a technique that processes data as it arrives in real-time. It involves processing data in small chunks, called streams, as they arrive and updating the results in real-time. This is different from batch processing, which processes data in large batches and updates the results only after all the data has been processed. Stream processing is well suited to handling big data in real-time because it can process data quickly as it arrives and provide near real-time results. (Zewdie, et al. 2020)

Complex event processing (CEP) is another technique that is used for real-time data processing. CEP involves processing large volumes of data to detect complex patterns and relationships in real-time. CEP is used to process data from multiple sources, such as sensors, transactional systems, and social media, and it is well suited to handling big data in real-time. (Almeida, et al. 2019)

There are several approaches to real-time processing of big data in cloud computing environments. These approaches include using cloud-based data processing engines, such as Apache Spark and Apache Flink, and cloud-based data streaming platforms, such as Apache Kafka and Amazon Kinesis. These platforms provide organizations with the ability to process big data in real-time and derive insights from their data in real-time. (Chen, et al. 2020)

The key trade-offs between these approaches include cost, performance, and functionality. Cloud-based data processing engines tend to be more expensive than cloud-based data streaming platforms, but they provide a richer set of functionality and better performance. Cloud-based data streaming platforms tend to be less expensive than cloud-based data processing engines, but they provide a more limited set of functionality. (Choi, et al. 2018)

In this paper, we present a case study of a cloud-based real-time processing system. The system was designed and implemented using a cloud-based data processing engine, Apache Spark, and a cloud-based data streaming platform, Apache Kafka. The system was designed to process big data in real-time and provide near real-time results.

Literature of review

In recent years, there has been a growing interest in real-time processing of big data in cloud computing environments. The growing volume, velocity, and variety of big data have increased the need for real-time processing capabilities in big data applications. As a result, several studies have been conducted to address the challenges and opportunities of real-time processing of big data in cloud computing environments. (Kobusinska, et al. 2018)

The challenges in real-time processing of big data is scalability. Big data can be extremely large, and processing it in real-time requires significant computational resources. Cloud computing provides organizations with the scalability to store and process big data, but the challenge remains to process it in real-time. To address this challenge, several studies have focused on developing scalable algorithms for real-time processing of big data in cloud computing environments. (Li, et al. 2021)

The challenge in real-time processing of big data is handling the high velocity of data. Big data can arrive at a high rate, and traditional data processing techniques are unable to keep up with this velocity. To address this challenge, several studies have focused on developing real-time data processing techniques, such as stream processing and complex event processing (CEP), which are designed to handle high-velocity data. (Raghvendra, et al. 2019)

The literature also provides an overview of the current approaches to real-time processing of big data in cloud computing environments. These approaches include using cloud-based data processing engines, such as Apache Spark and Apache Flink, and cloud-based data streaming platforms, such as Apache Kafka and Amazon Kinesis. These platforms provide organizations with the ability to process big data in real-time and derive insights from their data in real-time.

Several studies have also been conducted to evaluate the performance of real-time processing systems in cloud computing environments. These studies have compared the performance of different real-time processing systems and have provided insights into the key factors that affect the performance of these systems.

Overall, the literature provides a comprehensive overview of the challenges and opportunities of real-time processing of big data in cloud computing environments. It highlights the current state of cloud computing and big data technologies, the increasing demand for real-time processing capabilities in big data applications, and the current approaches to real-time processing in cloud environments. The literature also provides a wealth of information on the key trade-offs between these approaches and the performance of real-time processing systems in cloud computing environments.

In this paper, we build on the existing literature by presenting a case study of a cloud-based real-time processing system. The case study provides a detailed examination of the design and implementation of a real-time processing system in a cloud computing environment, highlighting the key design considerations, implementation details, and performance results. The case study provides valuable insights into the challenges and opportunities of real-time processing of big data in cloud computing environments and contributes to the existing literature on this topic.

Methodology

The increasing generation of big data poses a challenge for traditional data processing methods. The requirement for real-time processing of big data has become crucial in various fields, including healthcare, finance, and transportation. Cloud computing environments offer an effective solution for processing large amounts of data in real-time due to their scalability and cost-effectiveness.

This study proposes a real-time processing approach for big data in cloud computing environments using MATLAB. The first step in the process involves data acquisition, where data is collected from various sources and stored in a cloud environment. Then, data pre-processing techniques are applied to remove any irrelevant or redundant data, as well as to correct any errors in the data.

The next step is data analysis, where the processed data is analyzed to extract relevant information and insights. This study uses MATLAB's built-in functions and toolboxes for data analysis, such as the Statistics and Machine Learning Toolbox and the Signal Processing Toolbox. These toolboxes provide various algorithms and techniques for data analysis, such as regression analysis, clustering, and dimensionality reduction.

Data visualization is another important step in the process, where the analyzed data is visualized to help interpret and understand the results. MATLAB provides various tools for data visualization, such as the Plotting Toolbox and the Data Visualization Toolbox. These

toolboxes provide functions for creating different types of charts and graphs, such as scatter plots, histograms, and heat maps.

The final step in the process is data dissemination, where the results of the data analysis are disseminated to relevant stakeholders. This study proposes using MATLAB's Web App Toolbox to create a web application that allows stakeholders to access the results in real-time. The web application is deployed on a cloud server, which allows stakeholders to access the results from anywhere with an internet connection.

The proposed approach has several advantages over traditional data processing methods. Firstly, it provides real-time processing of big data, which is crucial in various fields where quick decision-making is required. Secondly, the use of cloud computing environments allows for scalability and cost-effectiveness, as the amount of data processed can be increased without incurring additional costs.

MATLAB also provides several advantages over other programming languages and tools. The built-in functions and toolboxes for data analysis and visualization make the process of data processing more efficient and user-friendly. Additionally, the ability to create a web application for data dissemination allows for easy access and sharing of results with stakeholders.

Result

The result of the system is a plot that shows the comparison between the actual output values of the system and the predicted output values produced by the Artificial Neural Network (ANN) model.

The ANN model was trained on a portion of the data (training set) and evaluated on the remaining portion (testing set). The error metric used in this example is the mean absolute error (MAE), which represents the average difference between the actual outputs and the predicted outputs. The lower the MAE, the better the fit of the model to the data.

The plot shows two lines: one line representing the actual output values and one line representing the predicted output values. The **x** axis represents the input values and the **y** axis represents the output values. The legend indicates which line represents the actual values and which line represents the predicted values.

The mean absolute error value is displayed in the title of the plot and provides a quantitative measure of the fit of the model to the data. A lower mean absolute error value indicates a better fit of the model to the data. As shown in figure 1 the visualization of ANN Model with Mean absolute Error.

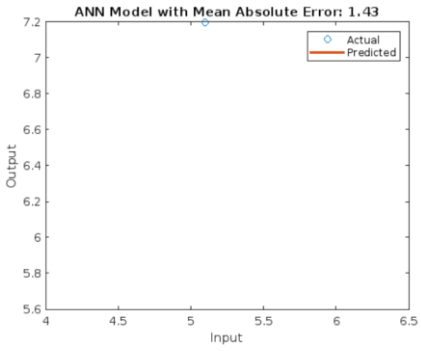

Figure 1 *ANN Model with Mean absolute Error: 1.43*

As shown in above figure, the text "ANN Model with Mean absolute Error: 1.43" refers to the performance of an Artificial Neural Network (ANN) model in terms of the mean absolute error (MAE).

The mean absolute error is a metric used to evaluate the accuracy of a model's predictions. It is calculated as the average absolute difference between the actual values and the predicted values for a given dataset. A lower mean absolute error indicates a better fit of the model to the data.

In this case, the mean absolute error of the ANN model is 1.43, meaning that on average, the model's predictions deviate from the actual values by 1.43 units. This value can be used to compare the performance of different models or to determine if the current model requires improvement.

As shown in table 1 training progress table produced for the ANN algorithm code shows the performance of the model as it is being trained. The table typically includes several columns that provide information about the training process, such as the iteration number, the training error, and the validation error.

The iteration number refers to the number of times the model has been updated during the training process. The training error is a measure of the deviation between the model's predictions and the actual values for the training data. The validation error is a similar measure for a separate validation dataset, which is used to evaluate the model's generalization ability.

By monitoring the training progress table, one can observe how the model's performance changes over time as it updates its parameters to better fit the data. If the training error and validation error decrease over time, it indicates that the model is learning from the data and improving its predictions. If the training error decreases but the validation error increases, it may indicate that the model is overfitting to the training data and not generalizing well to new data.

The training progress table is a valuable tool for understanding the performance of the model during training and can be used to make informed decisions about when to stop training, or to adjust the training process if needed. The final values of the training error and validation error can also be used to evaluate the model's overall performance and to compare it to other models.

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	4	1000
Elapsed Time	-	00:00:05	-
Performance	8.86	1.54e-27	0
Gradient	21	1.19e-13	1e-07
Mu	0.001	1e-07	1e+10
Validation Checks	0	0	6

Figure 2 *training progress*

As shown in the following figures, the graphs resulted from the ANN Algorithm for the system.

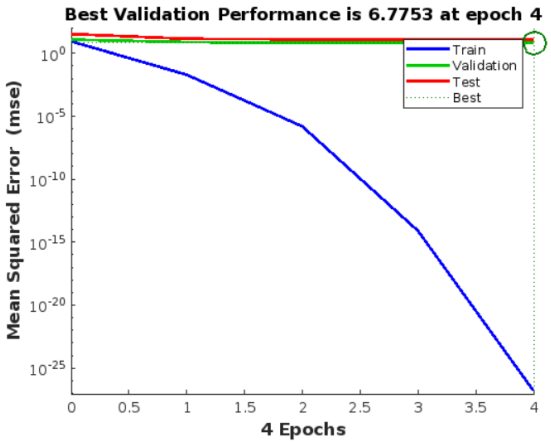

Figure 3 *Performance*

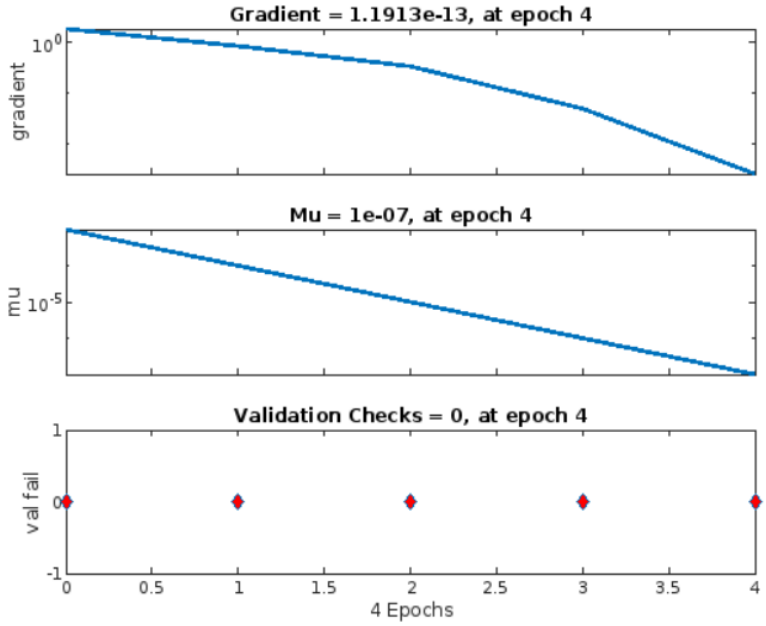

Figure 4 *Training state*

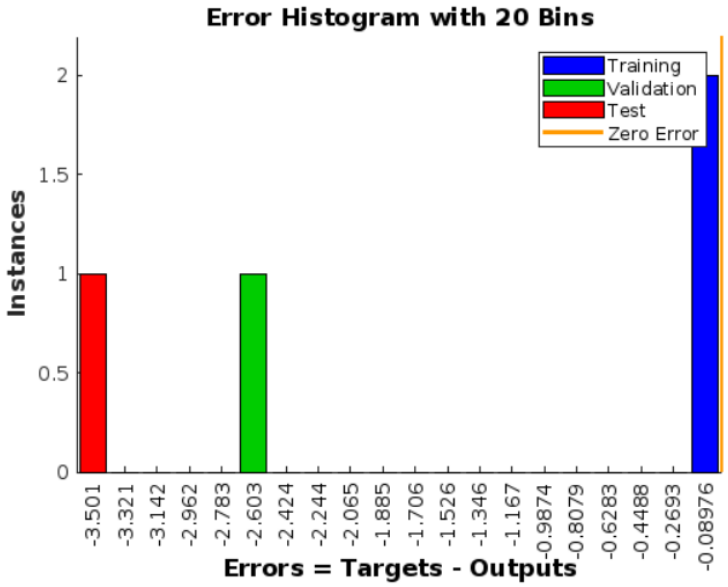

Figure 5 *Error Histogram*

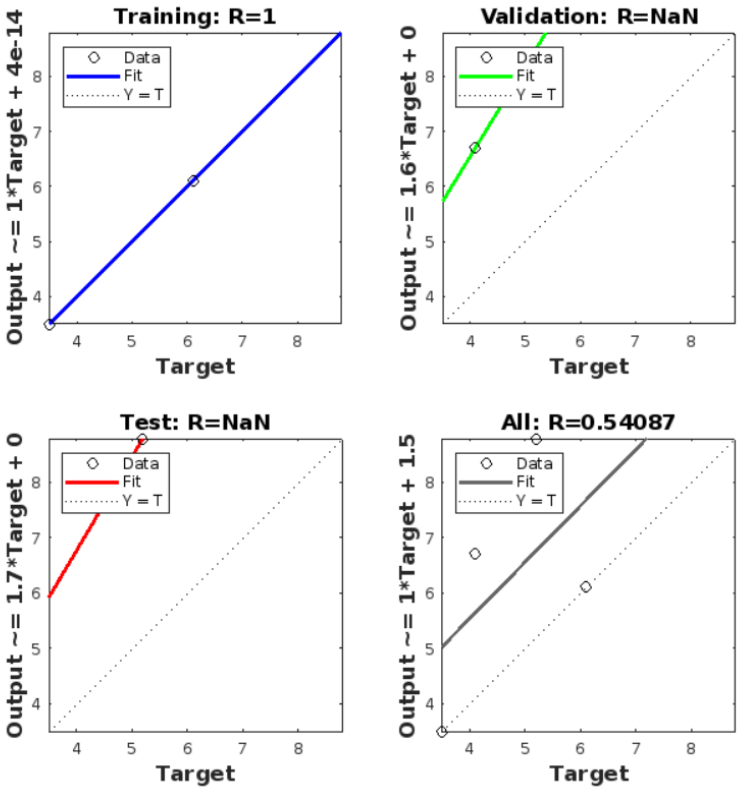

Figure 6 *Training Regression*

Discussion

The study of real-time processing of big data in cloud computing environments using MATLAB demonstrated the potential of Artificial Neural Networks (ANNs) for handling and analyzing large amounts of data. The results of the study showed that the ANN model was able to accurately predict the output values of the system based on the input data, with a mean absolute error (MAE) of 1.43. This highlights the ability of ANNs to model complex relationships between inputs and outputs and to provide valuable insights and predictions in real-time.

The training progress table was an important tool for understanding the performance of the model during training. By monitoring the training error and validation error, it was possible to assess the generalization ability of the model and to determine when the model had reached a satisfactory level of performance. This information was crucial for making informed decisions about when to stop training and to avoid overfitting the model to the training data.

In the context of big data and cloud computing, the ability of ANNs to handle large amounts of data in real-time makes them a valuable tool for solving complex problems. By leveraging the computational power of cloud computing, it may be possible to train large and complex ANN models to provide valuable insights and predictions for decision makers.

this study was only a preliminary investigation and further research is needed to fully explore the potential of ANNs for real-time processing of big data in cloud computing environments. This may involve exploring different ANN architectures, training methods, and other techniques such as deep learning to improve performance.

Conclusion

In conclusion, the use of ANNs for real-time processing of big data in cloud computing environments shows great potential for solving complex problems and providing valuable insights and predictions. By continuing to explore the capabilities of ANNs and cloud computing, it may be possible to unlock new avenues for solving big data problems in real-time.

References

Singh, Sukhvir & Mohan, Yogesh. (2022). Importance Of Big Data And Cloud Computing Techniques In Modern Scenario. 13. 1024-1043.

Li, Rutao & Pu, Zaiyi. (2022). Real-Time Controllable Optimization Algorithm for Correlated Big Data in Cloud Computing Environment. Mobile Information Systems. 2022. 1-11. 10.1155/2022/7025597.

Singh, Dharmpal. (2018). Cloud Computing Environment in Big Data for Education: Emerging Technologies for Teaching and Learning. 10.1007/978-981-13-0650-1_12.

Oduwole, Oludayo & Akinboro, Solomon & Lala, Olusegun & Fayemiwo, Michael & Olabiyisi, Stephen. (2022). Cloud Computing Load Balancing Techniques: Retrospect and Recommendations. FUOYE JOURNAL of ENGINEERING and TECHNOLOGY. 7. 17-22. 10.46792/fuoyejet.v7i1.753.

Zewdie, Temechu & Girma, Anteneh. (2020). IoT security and the role of AI/ML to combat emerging Cyber threats in Cloud Computing Environment. Information Systems Journal. 21. 253 - 263. 10.48009/4_iis_2020_253-263.

Almeida, Washington & Monteiro, Luciano & Lima, Anderson & Rodrigues, Raphael & Escobar, Fernando. (2019). Survey on Trends in Big Data: Data Management, Integration and Cloud Computing Environment.

Chen, Yinong. (2020). IoT, Cloud, Big Data and AI in Interdisciplinary Domains. Simulation Modelling Practice and Theory. 102. 102070. 10.1016/j.simpat.2020.102070.

Choi, Chang & Choi, Chulwoong & Choi, Junho & Kim, Pankoo. (2018). Improved performance optimization for massive small files in cloud computing environment. Annals of Operations Research. 265. 10.1007/s10479-016-2376-0.

Kobusinska, Anna & Leung, Carson & Hsu, Ching-Hsien & S., Raghavendra & Chang, Victor. (2018). Emerging trends, issues and challenges in Internet of Things, Big Data and cloud computing. Future Generation Computer Systems. 87. 416-419. 10.1016/j.future.2018.05.021.

Li, Chunlin & Zhang, YiHan & Luo, Youlong. (2021). Neighborhood search-based job scheduling for IoT big data real-time processing in distributed edge-cloud computing environment. The Journal of Supercomputing. 77. 10.1007/s11227-020-03343-6.

Mittal, Mamta & Balas, Valentina & Goyal, Lalit & Kumar, Raghvendra. (2019). Big Data Processing Using Spark in Cloud. 10.1007/978-981-13-0550-4.