

Adaptive Big Data Compression Engine Using AI-Powered Heuristics.

Amitha Mallikarjun Rajanal, Dr. Shankaragowda B B

P.G. Student, Department of Master of Computer Applications, BIET, Davangere-577004.

amitarajanal2003@gmail.com, Davangere, Karnataka, India

Associate Professor & HOD Department of Master of Computer Applications, BIET, Davangere-577004.

bsgowda2007@gmail.com, Davangere, Karnataka, India

ABSTRACT:

The amount of data produced in modern applications is enormous in the span of every second. Right from data in IoT and finance transactions to cloud apps and media the growth in data exceeds the capabilities of the normal storage and communication system. Normal methods of data management through compression are not competent in managing varying data at high speeds. That is because of the rigidity of data patterns and channel capacity.

The proposed research work includes the use of an Adaptive Big Data Compression Engine that is powered with AI technology with the aim of efficiently compressing big data. The proposed system has the ability to automatically dissect the incoming data learn and then adapt to the compression technique. The proposed research work includes the compression algorithms of Huffman Coding LZ77 compression and delta encoding algorithms along with AI models.

The engine works in an auto-sustaining mode-it learns from the decisions it has made and improves on them through continuous feedback cycles. It continuously analyzes critical performance factors such as the compression latency entropy ratio and resource allocation regarding new blocks of into the system.

.KEYWORDS: Compression Engine, AI-Powered, Heuristics, AI- driven Selector, Redundancy, Time-Series, Self Learning Modules.

I INTRODUCTION

The Adaptive Big Data Compression Engine has been designed to meet a real-world problem posed by huge and constantly shifting data with by many organizations. The traditional contraction threshold techniques impose a fixed system on the large data set irrespective of the huge variation shown by the attributes of the lines or data into gobebts.

This proposed system is totally a different approach. This system is actually a intelligent middleware service and acts as an intermediary layer in between directors and consumers. Now this system actually needs to analyze the entered data and choose a suitable contraction approach based on patterns linked to entered data. The patterns could be reiteration of position of noise and distribution values for entropy.

The machine makes this happen through the function of artificial intelligence in heuristic searches. The algorithms involved are not determined or set by humans but rather determined in the education of the figures and the approach is prophetic with regards to the most fashionable contraction algorithm that can be employed hence the function. Literacy processes make the machine smarter with time.

II LITERATURE REVIEW

The rapid growth of big data has created major challenges in storage, data transmission, and processing speed. Traditional data compression techniques such as Huffman coding, LZW, and BZIP2 have been widely used for many years to reduce data size. However, these traditional methods use fixed algorithms and do not adapt to different types of data automatically. Because of this limitation, they are not very efficient when dealing with modern big data environments that contain different types of data such as text, images, logs, and sensor data.

Several researchers have studied data compression techniques and found that no single compression algorithm performs best for all types of data. Some algorithms work better for text data, while others are suitable for multimedia or binary data. Traditional compression systems require manual selection of algorithms, which is time-consuming and not suitable for real-time big data processing. These systems also cannot change the compression strategy automatically during execution, which reduces efficiency when handling mixed or streaming data.

III EXISTING SYSTEM

The existing data compression algorithms are fixed and predefined which renders them less appropriate for Big Data in terms of adaptability and variability when compared to the required capabilities for Big Data.

Fundamental Dependencies on Existing Systems
The major dependencies on existing systems include:

- **Static Algorithms:** These employ fixed mathematical models including Huffman Coding (statistical modeling) LZWGZIP (dictionary-based models) and BZIP2

- **Manual Configuration:** Optimizing compression is a tedious and unscalable task that requires human interaction to determine the optimal compression algorithm and parameters that need to be used.

- **Less Adaptability:** These systems will not undergo strategy shifts during the middle of the execution process or even within the data blocks

- **Poor Performance on Mixed Data:** It will not be possible to find the algorithm that performs the best on log files image files sensor files and binary files by combining all of them in different streams.

- **Too Much Computational Overhead:** High ratio algorithms require an unusually high amount of computational overhead.

IV PROPOSED SYSTEM

Your suggested system makes use of the machine learning and Gen AI to develop a dynamic data-aware compression engine that strives to optimize in real time the compromise ratio for both processing speed and resource expenses.

1. **Machine Learning-based Data Pattern Recognition:** This is the heart of the Data Classification Module. Mixed Data For individual ML models it tracks qualities of the statistical and structural parcels of each piece of data which form a point vector (e.g. high entropy high redundancy certain structure).

2. **AI-Powered Heuristic Selection (Gen AILLM):** The point vector is used as the input in the AI Heuristic Selector. This predicts the optimal contraction technique of the algorithm block size and the parameters of the garbling before the actual contraction happens through the use of the pre-trained model Gen AILLM.

3. **Scalable Architecture :** The system is designed on fabrics similar to Spark icing that all ways Bracket Selection and Compression are done efficiently on a pall or distributed terrain.

SYSTEM ARCHITECTURE



Figure: shows that Architecture Diagram

V EXPERIMENTAL RESULTS

The experimental results of the Adaptive Big Data Compression Engine were evaluated by testing the system with different types of datasets such as text files, CSV files, IoT sensor data, and multimedia data. The main objective of the experiment was to measure compression ratio, compression time, CPU usage, and overall system performance when using the adaptive AI-based compression engine.

During the experiment, datasets of different sizes were uploaded into the system through the data upload module. The system first analyzed the dataset and identified the data type using the data classification module. After classification, the AI heuristic selector automatically selected the most suitable compression algorithm such as Huffman Coding, LZ77, Brotli, or Zstandard. The compression engine then compressed the data and stored the compressed output along with metadata.

The results showed that the adaptive compression engine performed better than traditional static compression methods. The system achieved an average compression reduction of around 40% to 60% depending on the type of data.



Figure: The figure shows the user interface of an Adaptive Compression Engine, a web-based application designed for analyzing and compressing large datasets using AI-powered heuristics.

Product Name	Column	Category	Type	Recommended AI Recommendation	Savings
Category	Category	Categorical		Dictionary Encoding	5.87 KB (40.0%)
Price	Price	Numbers		Group/Zip	3.21 KB (54.0%)
Stock	Stock	Numbers		Integer Packing	2.21 KB (60.0%)
ReleaseDate	ReleaseDate	Text		Delta Encoding	3.01 KB (50.0%)
InActive	InActive	Boolean		Boolean Packing	4.25 KB (50.0%)

Figure: The figure presents the column-wise analysis section of the Adaptive Compression Engine, which provides detailed insights into how each column of the dataset is optimized for compression.

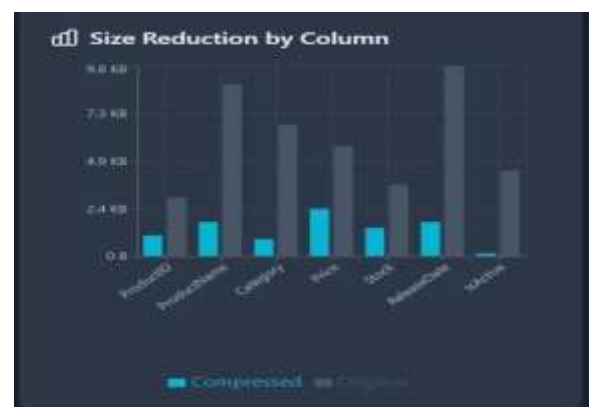


Figure: The figure illustrates a bar chart titled “size reduction by column”, which compares the original and compressed size of each column in the dataset.

VI CONCLUSION

The Adaptive Big Data Compression Engine powered by AI-driven Heuristics represents a significant advancement in the field of intelligent data processing and storage optimization. As the volume and velocity of digital information continue to grow exponentially traditional compression techniques struggle to deliver performance scalability and adaptability. This project demonstrates that by integrating machine-learning heuristics content-aware classification and high-performance compression algorithms it is possible to overcome these limitations and build a system that is faster smarter and more efficient than conventional solutions.

The engine begins with a very simple compression engine and progresses to an engine that is involved not only in aiding the process of decomposition for data but is also involved in determining what type of data is acquired and accordingly which algorithm to use. The feasibility tests for this engine asserted that this engine is but also very viable in the present situation with regards to networking with regards to data transmission with regards to IoT Cloud and Big Data. The final phase of this project also approved this engine.

If the above-mentioned technology is integrated with the cost factor optimum of the storage and traffic of data then it will be utilized in the optimum operation of other systems and will enable an organization to perform tasks in a data-intensive fashion. The implication arising from the emergence of the need with the current project to fill in the gap between the requirements of intelligent compression from the raw unstructured big data to the intelligent compression is that a brand-new paradigm with the AI-related data engineering systems has been developed.

VII REFERENCES

1. Buchanan-Gidumal J. (2022). Impact of Artificial Intelligence in Travel Tourism and Hospitality. Springer Handbooks.

This book highlights real-world applications of AI-driven decision systems. It supports the project's use of recommendation-style heuristics and adaptive learning models in big data compression.

2. Goodfellow I. Bengio Y. & Courville A. (2016). Deep Learning. MIT Press.

This text provides machine learning theoretical underpinning's methods used in the compression engine including neural architectures that guide predictive and adaptive heuristic design.

3. Sommerville I. (2015). Software Engineering (10th ed.). Pearson.

This book serves as the main reference for the system's development life-cycle modular architecture documentation standards testing strategy and quality engineering practices used in building the compression engine.

4. Norman D. (2013). The Design of Everyday Things. Basic Books.

This reference supports the design choices for the system's interface covering usability user-concentric workflows and intuitive control mechanisms for handling large-scale data operations.

5. Newman S. (2015). Building Microservices: Designing Fine-Grained Systems. O'ReillyMedia.

This book provides the architectural guidance for designing a scalable micro services-based back-end that supports high-volume data processing and distributed compression workflows.