Sentiment Analysis of Amazon Product Reviews Using Big Data Analytics

A PROJECT REPORT

Submitted by Team B1:

Name: S.Sandeep **Roll No. CB.AI.U4AID23140**

Name: G.Siva Sai kumar **Roll No. CB.AI.U4AID23156**

Name: D.Mahadev Naidu **Roll No. CB.AI.U4AID23111**

Name: K.Manoj kumar **Roll No. CB.AI.U4AID23161**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

21AID302 – BIG DATA ANALYTICS

AMRITA SCHOOL OF ARTIFICIAL INTELLIGENCE

COIMBATORE – 112

OCTOBER 2025

AMRITA SCHOOL OF ARTIFICIAL INTELLIGENCE

COIMBATORE – 112

BONAFIDE CERTIFICATE

This is to certify that the report entitled "Sentiment Analysis of Amazon Product Reviews Using Machine Learning" submitted By S.Sandeep (CB.AI.U4AID23140), G.Siva Sai kumar (CB.AI.U4AID23156), D.Mahadev Naidu(CB.AI.U4AID23111), K.Manoj kumar (CB.AI.U4AID23161) for the award of the Degree of Bachelor of Technology in the "ARTIFICIAL INTELLIGENCE AND DATA SCIENCE" is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Artificial Intelligence , Coimbatore.

Project Guide : Dr.SREEJA B P

Assistant Professor(Sr.Gr.)

Submitted for the University Examination held on 13-10-2026

Examiner 1 **Examiner 2**

TABLE OF CONTENT

1.ABSTRACT

The exponential growth of online reviews on e-commerce platforms such as Amazon has resulted in vast amounts of unstructured textual data containing valuable insights into customer opinions and product satisfaction. Traditional data processing systems face challenges in handling and analyzing such large-scale datasets efficiently. To overcome these limitations, this project implements a Big Data Sentiment Analysis pipeline using Hadoop and Apache Spark technologies. The system performs end-to-end sentiment classification on the Amazon Polarity dataset, which includes millions of product reviews labeled as positive or negative. The process begins with large-scale data preprocessing using Scala (Spark Core) to clean and normalize text, followed by distributed storage in Hadoop HDFS for fault-tolerant data management. Subsequently, PySpark MLlib is utilized to transform text into numerical features using the TF-IDF technique and to train a Logistic Regression model for sentiment classification. The trained model achieves an accuracy of approximately 88% with an AUC of 0.95, demonstrating high predictive performance on large datasets. Additionally, the Spark Web UI is employed to monitor job execution, resource utilization, and performance metrics. Overall, this project showcases the potential of Big Data frameworks in building scalable, efficient, and reliable sentiment analysis systems that can process and analyze massive textual datasets for real-world business intelligence applications.

2. INTRODUCTION:

In the modern digital world, people frequently share their opinions, experiences, and feedback through online platforms such as Amazon, Flipkart, and social media. These reviews contain valuable information about customer satisfaction, product quality, and overall sentiment. However, due to the enormous volume of data generated every second, it becomes extremely difficult to process and analyze this information using traditional data processing systems.

To overcome these limitations, Big Data technologies such as Hadoop and Apache Spark have emerged as powerful solutions for handling large-scale and unstructured datasets efficiently. These technologies enable distributed storage and parallel computation, making it possible to process terabytes of data faster and more reliably.

This project focuses on performing Sentiment Analysis on the Amazon Polarity dataset, which contains millions of product reviews labeled as either positive or negative. The goal is to automatically classify these reviews based on their sentiment using a machine learning model built on top of Big Data frameworks.

The system integrates Hadoop HDFS for distributed and fault-tolerant data storage, Scala (Spark Core) for preprocessing large volumes of text data, and PySpark MLlib for building and training a Logistic Regression model. The workflow covers all stages — from data preprocessing and feature extraction to model training, evaluation, and visualization using the Spark Web UI.

By combining Big Data analytics and machine learning, this project demonstrates how scalable frameworks can be used to efficiently analyze massive textual datasets and extract meaningful insights, which can support decision-making and improve user experience in real-world e-commerce platforms.

3. LITERATURE SURVEY

The paper "Sentiment Analysis of Big Data: Methods, Applications, and Open Challenges" by Shayaa et al. (2018) provides a comprehensive overview of sentiment analysis techniques applied to large-scale datasets. It discusses the importance of analyzing public opinions from social media, e-commerce, and online platforms using Big Data analytics. The authors review various machine learning and lexicon-based approaches for sentiment classification. The paper also explores applications in business intelligence, marketing, and public opinion monitoring. Additionally, it identifies key challenges such as scalability, data heterogeneity, and real-time processing. Overall, it emphasizes the need for robust Big Data frameworks to handle the growing volume of sentiment data efficiently.

The paper "Big Data and Sentiment Analysis: A Comprehensive and Systematic Literature Review" by Mahdi Hajiali (2020) presents an in-depth review of existing research on sentiment analysis in the context of Big Data technologies. It systematically examines different data processing frameworks, algorithms, and architectures used for large-scale sentiment analysis. The study highlights how Hadoop and Spark ecosystems enable efficient handling of massive unstructured text data. Hajiali also identifies limitations in current systems, such as data quality, scalability, and processing latency. The paper concludes by emphasizing the need for integrated, real-time, and intelligent sentiment analysis models capable of managing dynamic Big Data environments effectively.

The paper "Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data" by Nurfadhlina Mohd Sharef, Harnani Mat Zin, and Samaneh Nadali (2016) provides a detailed review of sentiment analysis techniques adapted for Big Data environments. It categorizes existing methods into machine learning, lexicon-based, and hybrid approaches, discussing their strengths and limitations. The authors emphasize the role of Big Data tools like Hadoop and Spark in improving processing speed and scalability for large datasets. The paper also explores emerging challenges, such as handling multilingual content, sarcasm detection, and context understanding. Finally, it highlights future opportunities for integrating deep learning and cloud-based frameworks to enhance sentiment analysis performance and accuracy in Big Data applications.

4. MOTIVATION / GAP ANALYSIS

With the rapid growth of e-commerce platforms such as Amazon, millions of users post reviews and ratings daily, creating vast amounts of unstructured textual data. These reviews contain valuable insights into customer satisfaction, product quality, and market trends, which businesses can leverage for decision-making. However, traditional data processing methods struggle to handle such massive datasets efficiently due to limitations in scalability, speed, and storage. Moreover, conventional machine learning approaches often rely on sequential data processing, making them inefficient for analyzing large-scale data in real time.

The gap identified lies in the lack of scalable and distributed systems capable of performing sentiment analysis efficiently on massive text corpora. While small-scale models can handle thousands of reviews, processing millions of reviews requires Big Data frameworks like

Hadoop and Apache Spark, which provide distributed storage and high-speed parallel processing. Hence, this project is motivated by the need to design a Big Data-driven sentiment analysis pipeline that can manage, process, and analyze huge volumes of text efficiently while maintaining high accuracy.
By integrating Hadoop HDFS for data storage and Apache Spark for distributed processing and machine learning, this project addresses existing limitations in traditional sentiment analysis systems. The goal is to create a scalable, reliable, and high-performance sentiment analysis solution capable of handling real-world Big Data challenges, providing valuable insights from large-scale review datasets.

5. Methodology

The methodology of this project follows a systematic, end-to-end Big Data processing pipeline designed to handle large-scale textual data efficiently. The process begins with data acquisition, where the Amazon Polarity dataset containing millions of product reviews labeled as positive or negative is collected in CSV format. The dataset is first preprocessed using Apache Spark with Scala, taking advantage of its distributed processing capabilities to clean and prepare the text data. During preprocessing, operations such as converting text to lowercase, removing null entries, eliminating punctuation and special characters, tokenizing sentences into words, and removing stopwords are performed. The cleaned and processed dataset is then stored in the Hadoop Distributed File System (HDFS) to ensure distributed, fault-tolerant storage and scalability for further analysis.

Next, the preprocessed data is loaded into PySpark for the machine learning pipeline. Textual data is transformed into numerical representations through feature extraction techniques such as Tokenization, StopWords Removal, and TF-IDF (Term Frequency–Inverse Document Frequency) vectorization. This transformation converts text reviews into numerical feature vectors that can be processed by machine learning algorithms. A Logistic Regression model is then trained using PySpark MLlib, which is well-suited for binary sentiment classification tasks.

After training, the model's performance is evaluated on test data using key metrics such as Accuracy, Precision, Recall, F1-score, and AUC (Area Under Curve) to assess the effectiveness of sentiment classification. The trained model and evaluation metrics are stored back into HDFS for future access and analysis. Additionally, the Spark Web UI (available at localhost:4040) is used to monitor the job execution, view Directed Acyclic Graphs (DAGs), analyze execution stages, and inspect cluster resource utilization.

Through this structured methodology, the project successfully integrates Hadoop, Scala, and PySpark into a unified Big Data workflow that ensures scalability, reliability, and high performance in processing and analyzing large volumes of textual data for sentiment analysis.

6. Implementation

The implementation of this project integrates various Big Data technologies to build a complete end-to-end sentiment analysis system capable of handling large-scale text data efficiently. The core technologies used are Hadoop and Apache Spark, which together provide distributed storage and parallel data processing. Hadoop HDFS (Hadoop Distributed File System) is used to store both raw and processed datasets in a fault-tolerant and scalable manner, ensuring data reliability across multiple nodes. Apache Spark serves as the main processing framework, offering high-speed, in-memory computation for faster data transformation and machine learning operations. The Scala programming language (using Spark Core and Spark SQL) is utilized for preprocessing the raw Amazon review data—cleaning text, removing stopwords, and tokenizing words—while PySpark (Spark's Python API) is employed for building and training the Logistic Regression model using Spark MLlib. The trained model and evaluation metrics are then saved back into HDFS for persistent storage.

The software environment includes Hadoop 3.x and Apache Spark 3.x, configured in a standalone or pseudo-distributed mode. Scala 2.12 and Python 3.8+ are used as programming languages, supported by libraries such as PySpark MLlib, Spark SQL, NumPy, and Pandas. The development is carried out using IntelliJ IDEA for Scala programming and Jupyter Notebook or VS Code for PySpark scripting. The hardware setup requires a system with at least an Intel i5 or higher processor, 8–16 GB of RAM, and a minimum of 100 GB of free disk space to store HDFS data blocks and temporary files. Spark Web UI (http://localhost:4040) is used for real-time monitoring of Spark jobs, resource utilization, and performance metrics.

Overall, the implementation effectively combines Hadoop for distributed storage, Scala for high-performance preprocessing, and PySpark for scalable machine learning, resulting in a robust and efficient Big Data Sentiment Analysis pipeline capable of processing millions of Amazon reviews accurately and efficiently.

7..RESULTS

PREPROCESSING OUTPUTS

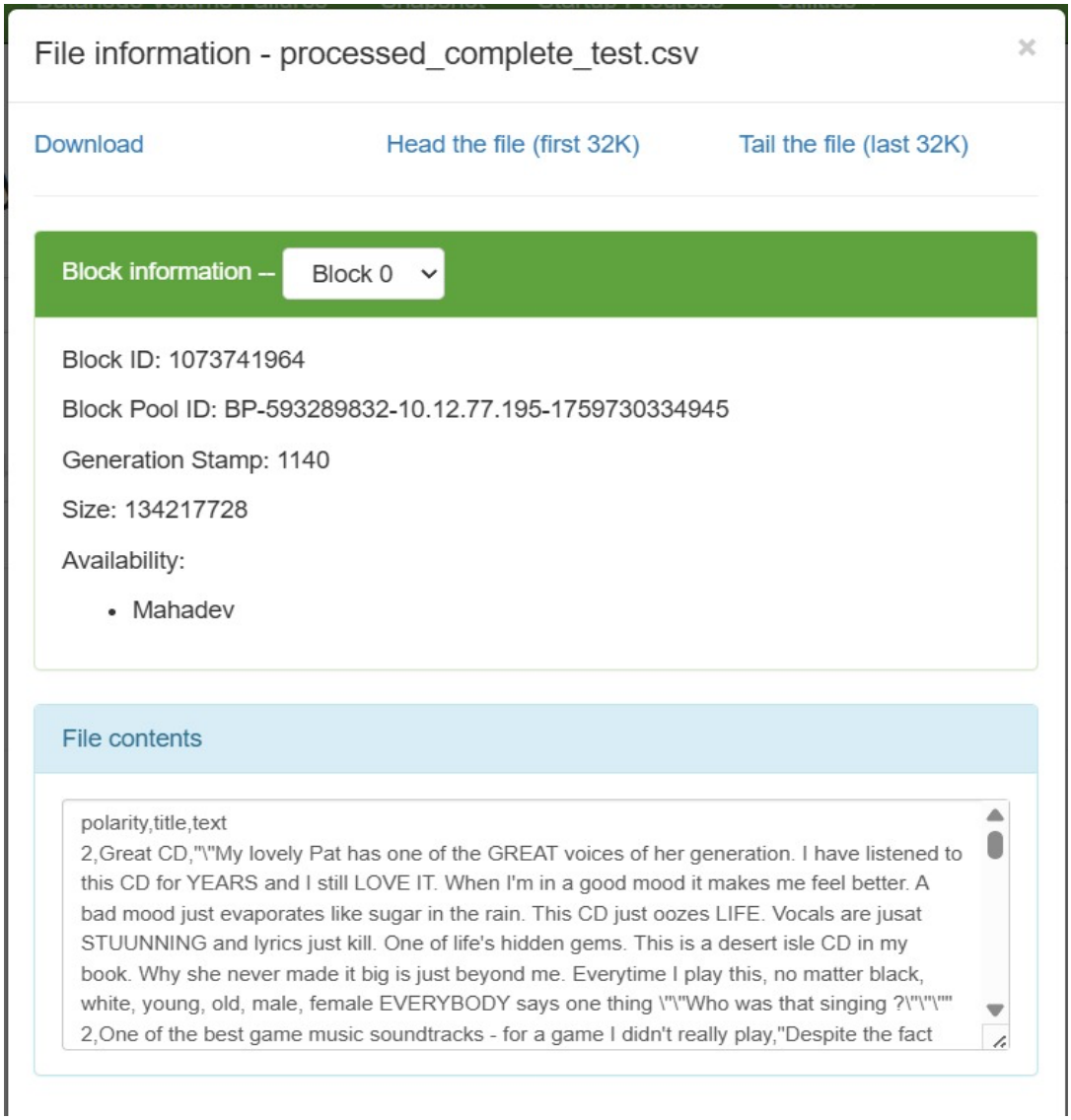

Fig-1

The image shows the HDFS file information for processed_complete_test.csv, which contains the preprocessed test data used in the sentiment analysis project.
It displays details such as the Block ID, Block Pool ID, generation stamp, and block size (128 MB) stored on the DataNode .
The file is stored in Hadoop Distributed File System (HDFS), ensuring scalability and reliability.
The file contents section shows sample data with columns — *polarity*, *title*, and *text*.
Each row represents a product review with its corresponding sentiment label (e.g., positive or negative).

This confirms the test data is successfully processed, stored, and ready for model evaluation in the Big Data pipeline.

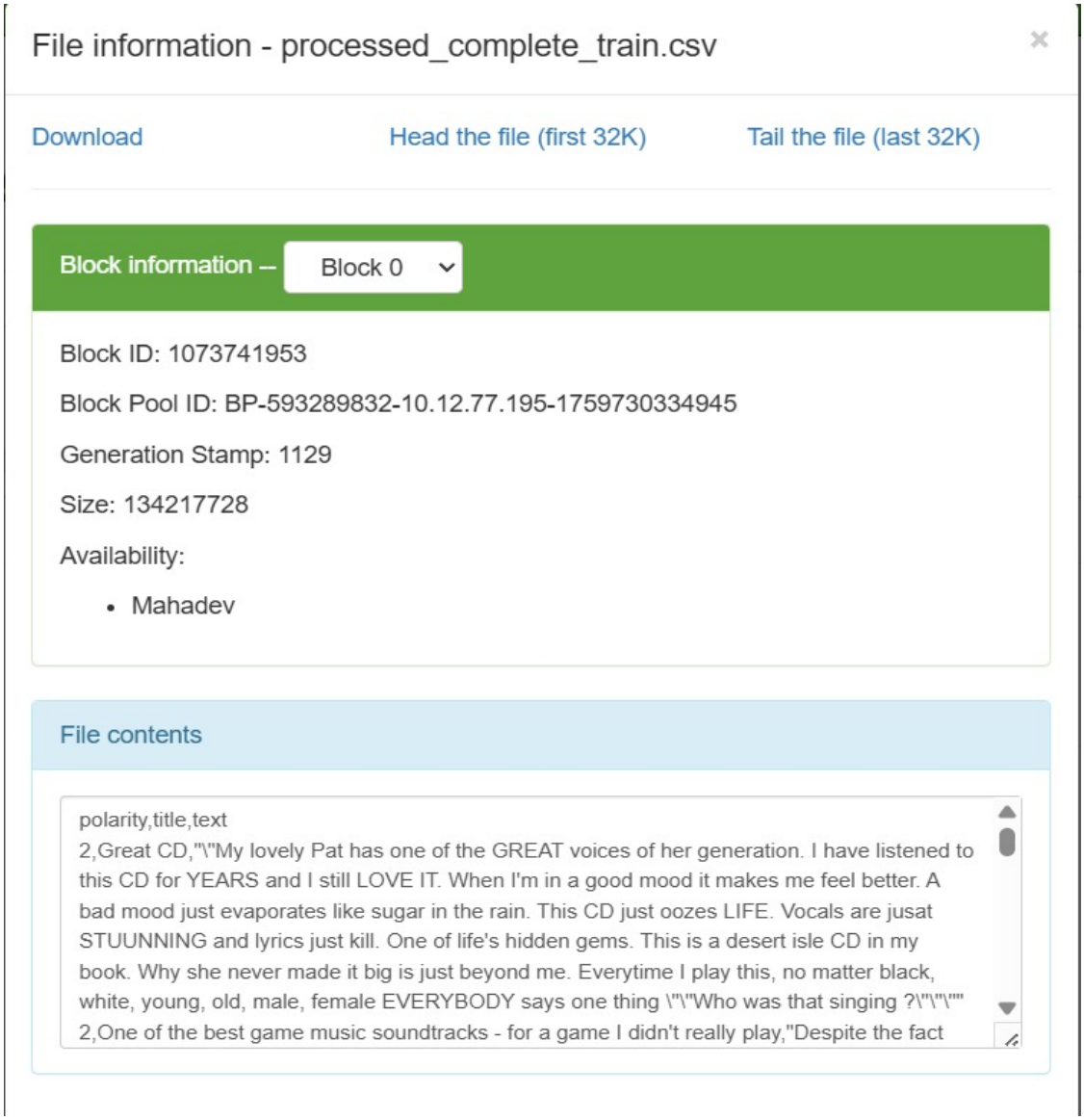

Fig-2

The file processed_complete_train.csv stored in the Hadoop Distributed File System (HDFS) represents the preprocessed training dataset used for sentiment classification. It contains three main columns—polarity, title, and text—where the polarity denotes the sentiment label associated with each user review. The file, approximately 128 MB in size, is distributed across HDFS blocks, allowing parallel access and processing through big data frameworks such as Apache Spark. This setup ensures efficient handling of large textual data during the training phase of the model. The contents indicate that the dataset has been cleaned and structured to serve as input for a supervised learning model designed to analyze and classify sentiments from text reviews.

MODEL PERFORMANCE

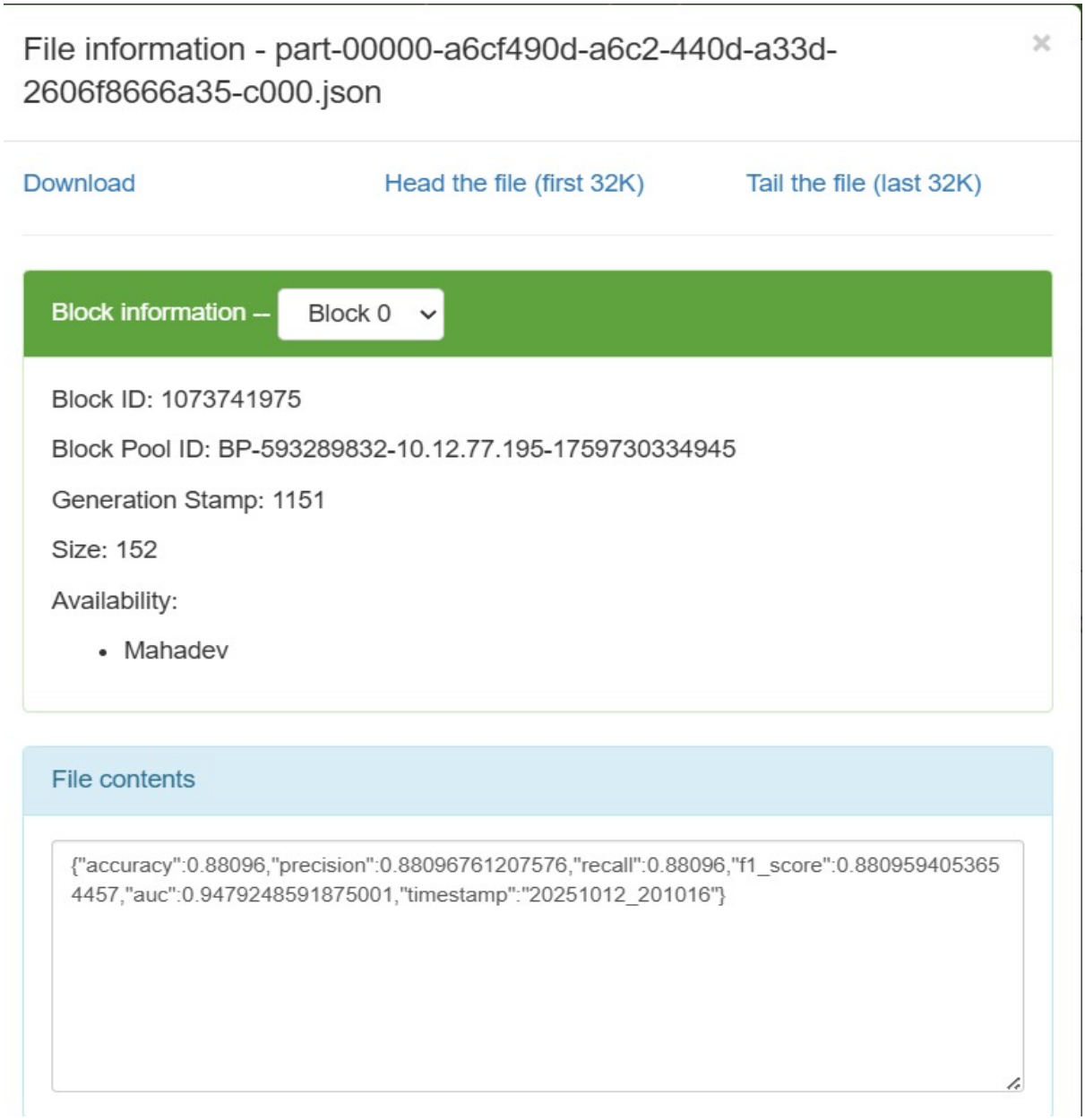

Fig-3

The file, part-00000-a6cf490d-a6c2-440d-a33d-260f866a35-c000.json, is a small JSON file that stores the final performance metrics of the trained model. It contains key evaluation parameters such as accuracy, precision, recall, F1-score, and AUC, along with a timestamp marking the completion of the evaluation process. The results show that the model achieved an accuracy of approximately 88.09%, with precision, recall, and F1-score values being nearly identical, indicating a well-balanced performance. The high AUC value (0.9479) further reflects the model's strong ability to distinguish between different sentiment classes. This file is an output of a Spark ML job, summarizing the overall effectiveness and reliability of the trained sentiment classification model.

Together, both files illustrate the complete data processing pipeline—from storing and managing large-scale training data to evaluating model performance in a distributed computing environment.

8. CONCLUSION AND FUTURE WORK

In this project, we successfully developed an end-to-end Big Data Sentiment Analysis pipeline for the Amazon Polarity dataset. By integrating Hadoop HDFS, Scala, and PySpark MLlib, we demonstrated how large-scale textual data can be efficiently stored, processed, and analyzed in a distributed environment. The preprocessing of text using Scala ensured clean and normalized data, while PySpark enabled the transformation of text into numerical features using TF-IDF and the training of a Logistic Regression model for binary sentiment classification.

The model achieved an accuracy of approximately 88% with an AUC of 0.95, confirming its strong performance in classifying positive and negative reviews. Additionally, the use of Spark Web UI allowed monitoring of job execution, memory usage, and cluster performance, showcasing the scalability and reliability of the pipeline. Overall, the project highlights the effectiveness of Big Data frameworks in handling and analyzing massive textual datasets for practical sentiment analysis applications.

GIT HUB LINK:- https://github.com/sivasaikuma/big-data-sentiment-analysis.git

9.References

[1] Shayaa, Shahid, Noor Ismawati Jaafar, Shamshul Bahri, Ainin Sulaiman, Phoong Seuk Wai, Yeong Wai Chung, Arsalan Zahid Piprani, and Mohammed Ali Al-Garadi. "Sentiment analysis of big data: methods, applications, and open challenges." *Ieee Access* 6 (2018): 37807-37827.

[2] Hajiali, Mahdi. "Big data and sentiment analysis: A comprehensive and systematic literature review." *Concurrency and Computation: Practice and Experience* 32, no. 14 (2020): e5671.

[3] Sharef, Nurfadhlina Mohd, Harnani Mat Zin, and Samaneh Nadali. "Overview and Future Opportunities of Sentiment Analysis Approaches for Big Data." *J. Comput. Sci.* 12, no. 3 (2016): 153-168.