Telecom Big Data Analytics Using PySpark

N. Mohith Vijay Sai[1] , M. Dimple Riya[1] , M. Giri Priya[1] , K.Hanumanthu[1] , G. M. Padmaja[2]

[1] students, email ids of students [3]
[2] Assistant Professor, padmaja.gmp@gmail.com
Department of Computer Science and Engineering,
SRK Institute of Technology, Andhra Pradesh, India

Abstract : Telecom operators generate massive volumes of Call Detail Records (CDR) on a daily basis, capturing detailed information about voice calls, short message services, and data usage. Processing and analyzing such large-scale telecom data using traditional single-node systems is inefficient and time-consuming. This paper presents a scalable Telecom CDR Analysis System implemented using Apache Spark (PySpark) to efficiently process and analyze large datasets. The proposed system analyzes approximately 1 GB of CDR data to derive insights related to revenue generation, network quality, customer usage behavior, and abnormal activity detection. By leveraging distributed processing, in-memory computation, and parallel execution, the system achieves improved performance, scalability, and fault tolerance. The results demonstrate that Spark-based analytics significantly outperform conventional approaches and provide timely, actionable insights for telecom operators.

IndexTerms - Call Detail Records, Big Data Analytics, Apache Spark, PySpark, Telecom Analytics, Distributed Processing

I. INTRODUCTION

Telecom operators generate massive volumes of Call Detail Records (CDR) on a daily basis, capturing detailed information about voice calls, short message services, and data usage. Processing and analyzing such large-scale telecom data using traditional single-node systems is inefficient and time-consuming. This paper presents a scalable Telecom CDR Analysis System implemented using Apache Spark (PySpark) to efficiently process and analyze large datasets. The proposed system analyzes approximately 1 GB of CDR data to derive insights related to revenue generation, network quality, customer usage behaviour, and abnormal activity detection. By leveraging distributed processing, in-memory computation, and parallel execution, the system achieves improved performance, scalability, and fault tolerance. The results demonstrate that Spark-based analytics significantly outperform conventional approaches and provide timely, actionable insights for telecom operators.

The exponential growth of mobile communication services has resulted in the generation of enormous volumes of telecom data. Every call, message, or internet session produces a Call Detail Record (CDR) containing information such as caller number, call duration, data usage, charges, and timestamp. These records play a crucial role in billing, revenue assurance, network optimization, fraud detection, and customer behaviour analysis.

Traditional data processing techniques and relational database systems are not designed to handle the volume, velocity, and variety of telecom CDR data. As the dataset size increases to gigabytes or terabytes, these systems suffer from performance bottlenecks and limited scalability. Consequently, there is a strong need for distributed big data processing frameworks that can efficiently manage and analyze telecom datasets.

Apache Spark is a widely used big data processing engine that supports in-memory computation and parallel execution across distributed clusters. In this work, Apache Spark (PySpark) is utilized to design and implement a Telecom CDR Analysis System capable of processing large datasets efficiently. The system provides insights into revenue trends, network performance, customer usage patterns, and abnormal behaviour, thereby supporting informed decision-making for telecom operators.

II. RELATED WORK

Several research efforts and practical systems have been proposed over the years to analyze telecom data and other large-scale datasets. This section discusses the major categories of related work in a structured, heading-wise manner to clearly position the proposed Telecom CDR Analysis System.

A. Apache Spark: A Unified Engine for Big Data Processing – Zaharia et al. (2016)

This paper introduces Apache Spark as a general-purpose distributed computing engine designed to overcome the performance limitations of Hadoop MapReduce .The authors propose an in-memory cluster computing model where intermediate data is stored in RAM instead of disk. Spark introduces Resilient Distributed Datasets (RDDs) and later DataFrames to support fault tolerance and parallel execution.

B. Machine Learning for Churn Prediction in Telecom – Bifet et al. (2019)

This research focuses on predicting customer churn using machine learning models trained on telecom datasets .The authors apply classification algorithms such as Random Forest, Logistic Regression, and Gradient Boosting on customer usage data to predict churn probability.

C. MapReduce: Simplified Data Processing on Large Clusters – Dean & Ghemawat (2004)
This foundational paper introduced the MapReduce programming model for distributed computing.
Tasks are divided into:
- Map phase – data transformation
- Reduce phase – aggregation

Google's infrastructure handled job scheduling, fault tolerance, and data distribution.

D. Telecom Analytics Survey – IEEE Communications Magazine (2020)
This survey reviews big data analytics techniques applied in the telecom industry.
- Network traffic analysis
- Customer segmentation
- Fraud detection
- Revenue optimization
- Quality of Service (QoS) monitoring

E. Apache Spark Documentation – Apache Software Foundation
Official documentation describing Spark's architecture, APIs, and optimization mechanisms.
- Catalyst query optimizer
- Tungsten execution engine
- DataFrame API
- Window functions
- Approximate aggregations
- Parquet integration

These features are critical to achieving performance and scalability.

Relevance to Our work:

Our project proposes PySpark as the core processing engine. Concepts such as DataFrames, lazy evaluation, caching, partitioning, and fault tolerance are directly adopted from this framework to efficiently analyze telecom CDR data. Our system includes a churn risk proxy based on call drop rates and revenue. While simpler than ML-based models, it provides a scalable baseline that can be extended to machine learning in future work .Spark, which our system uses, was developed to overcome MapReduce's limitations. Our project benefits from Spark's faster, memory-based execution. Our project directly implements several analytics tasks:
- ARPU calculation
- Network quality analysis
- User behaviour segmentation
- Operator-wise performance evaluation

From the related work, it is evident that while existing research addresses individual aspects such as distributed processing, churn prediction, and telecom KPIs, there is limited work on integrated end-to-end analytics pipelines using modern big data frameworks. The proposed system bridges this gap by combining scalable CDR ingestion, multi-dimensional KPI computation, optimization techniques, and analytical storage into a unified PySpark-based architecture.

III. PROPOSED WORK

The proposed Telecom Call Detail Record Analysis System is designed to efficiently analyze large-scale telecom datasets and extract meaningful insights. Unlike traditional telecom analytics tools that operate on limited data volumes, the proposed system leverages Apache Spark to handle large datasets with improved performance and scalability.
The system begins by ingesting raw CDR data in CSV format and loading it into Spark DataFrames using a predefined schema. Data preprocessing is performed to handle missing values, remove duplicates, and convert data into appropriate formats. Feature extraction techniques are applied to derive attributes such as year and month from timestamps, enabling time-based analysis.
The core analytics module performs multiple analyses, including revenue calculation per operator, network quality assessment through call-drop analysis, customer usage profiling, and abnormal behaviour detection. These analyses help telecom operators understand network performance, optimize resources, and detect potential issues such as fraud or network congestion.
The proposed system follows a layered architecture to ensure modularity and scalability. The Data Ingestion Layer handles the loading of raw CDR files into Spark. The Processing Layer performs data cleaning, transformation, and feature extraction. The Analytics Layer executes revenue analysis, usage profiling, and abnormal behaviour detection using Spark SQL and DataFrame operations. Finally, the Output Layer stores the processed results and generates reports for decision-making.
Advantages of proposed System Architecture :
- Scalable distributed processing using Apache Spark
- Faster execution through in-memory computation
- Efficient handling of large telecom datasets
- Improved network quality monitoring
- Supports extension to real-time analytics

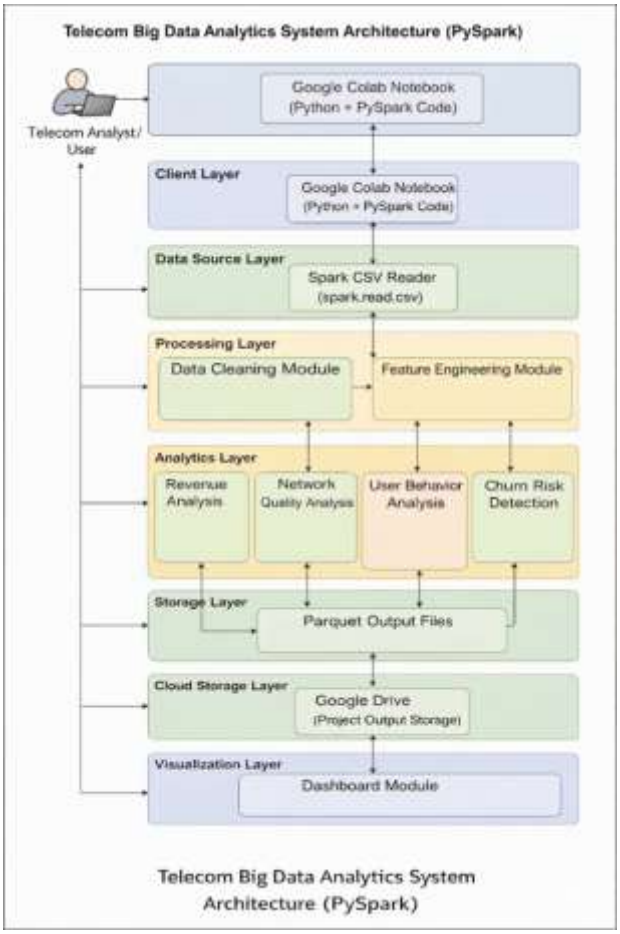

Fig3.1 : Architecture of Telecom Big Data Analytics System using Pyspark

IV. IMPLEMENTATION

CDR Data Preprocessing
Objective:
Clean and prepare raw CDR data for analysis.
Execution Logic
Load raw CDR data → remove duplicates → handle missing values → convert data types.
Output
Preprocessed CDR dataset ready for analytics.
Analysis 1: Revenue Calculation
Objective:
Compute total revenue per telecom operator.
Computation Logic
Revenue = Σ (charge)
Grouped by operator.
The Revenue is calculated for 4 types of SIMs that are VI, Airtel, Jio, BSNL .The Output will be in the form of total revenue ,number of users ,average charge , maximum charge, minimum charge, Average revenue per person, Monthly Revenue ,Yearly Revenue,

Analysis 2: Call Drop Rate Analysis
Objective:
Measure network quality using call drop statistics.
Call Drop Rate = (Number of dropped calls / Total calls) × 100
Call Drop Rate is calculated for 4 types of SIMs and also the states that are in the dataset also with the yearly drop rates in the addition to it .The States in the dataset are Delhi, Telangana, Tamil Nadu, Maharashtra, Karnataka, Andhra Pradesh.
Analysis 3: Customer Usage Profiling
Objective:
Identify heavy users based on call duration and data usage.
Logic
Aggregate duration and data usage per subscriber and rank users.
In this category there are analytics like peak hours, Heavy users, Top users, Also according to their usage they are divided into LOW, MEDIUM, HIGH categories.
Analysis 4: Abnormal Behaviour Detection

Objective:
Detect unusual usage patterns indicating potential fraud or network issues.
Decision Rule
If usage exceeds predefined threshold → mark as abnormal.

- Scalable distributed architecture
- High-performance in-memory processing
- Fault-tolerant and reliable
- Supports multiple telecom analytics tasks
- Cost-effective and extensible design

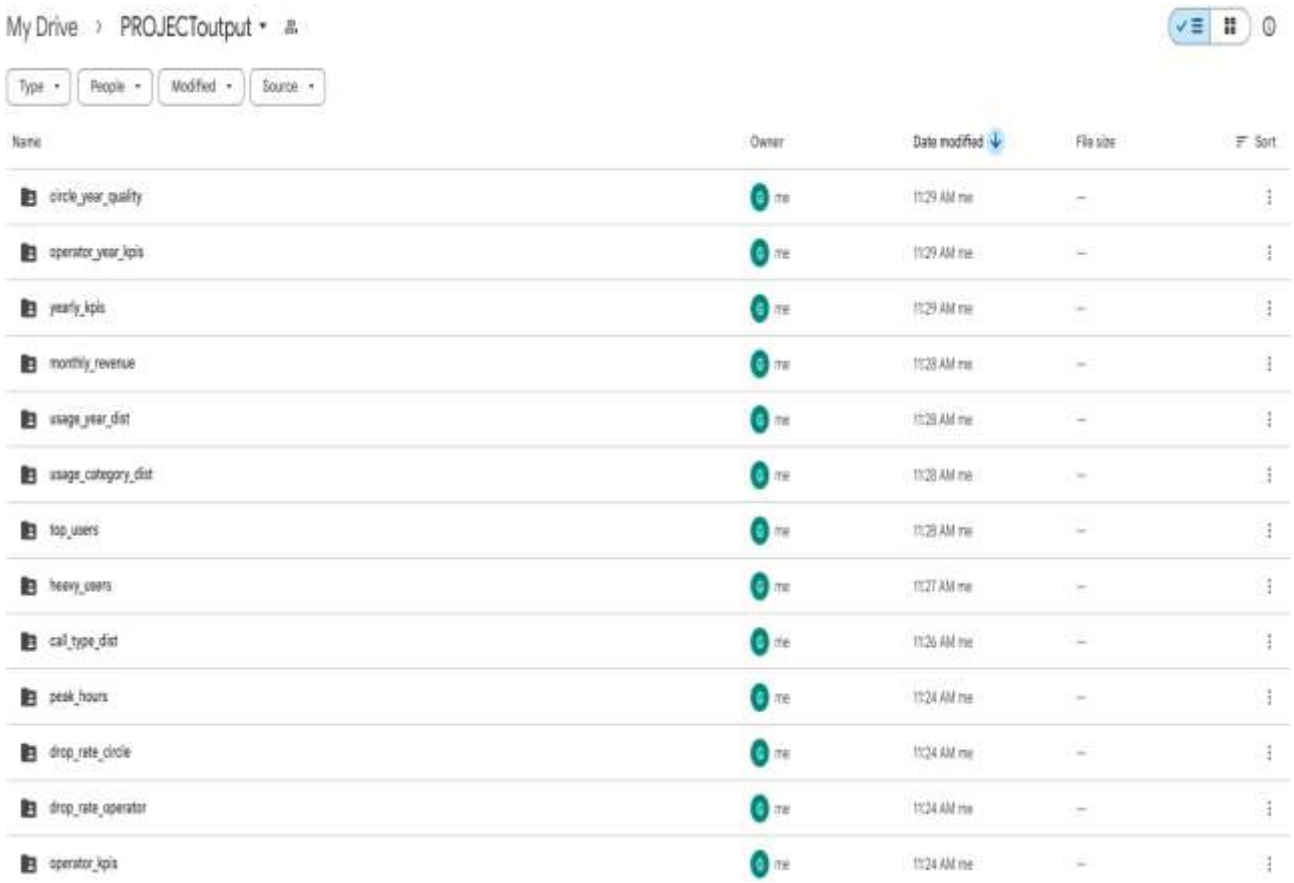

Fig 4.1: Generated Output Folders for Telecom CDR Analysis

Fig 4.2: One of the compressed folder

V. RESULTS

Operator-wise Revenue Dashboard:

Year-wise analysis of total telecom revenue showing a decline after 2020 followed by gradual stabilization, highlighting revenue fluctuations and the need for data-driven optimization.

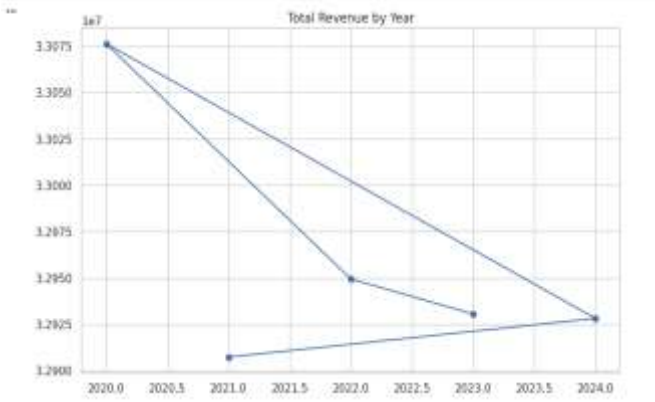

FIG 5.1 : Operator-wise Revenue Dashboard

Monthly Revenue Trend:

Monthly revenue trends over five years showing consistent seasonal patterns with recurring revenue dips and peaks, highlighting predictable customer usage behaviour in telecom networks.

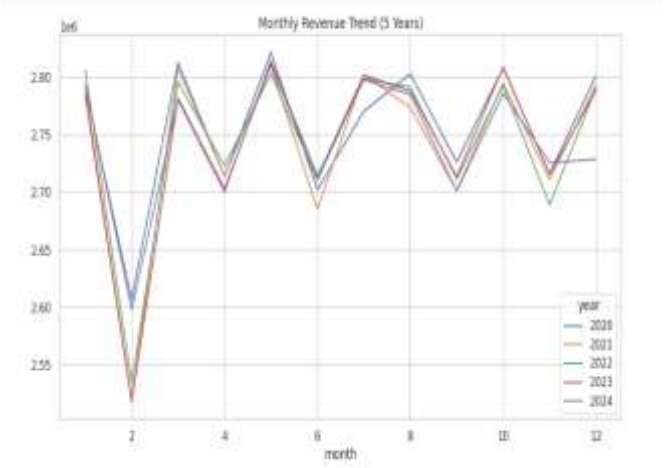

FIG 5.2: Monthly Revenue Trend

Operator-wise Revenue Comparison:

Year-wise comparison of total revenue across major telecom operators, showing stable performance with minor variations in a competitive market.

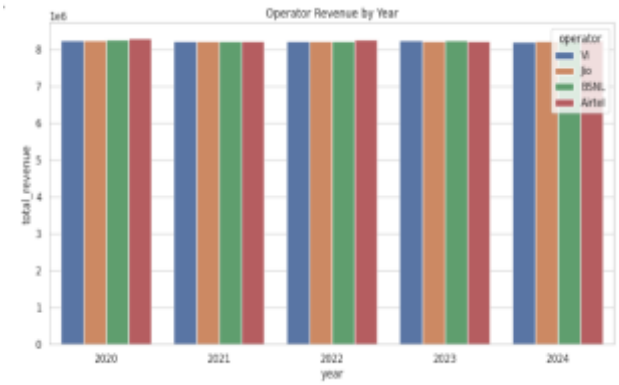

FIG 5.3: Operator-wise Revenue Comparison

Circle-wise Call Drop Rate Analysis:
This graph compares call drop rates across telecom circles over five years and shows stable network performance with minor regional variations.

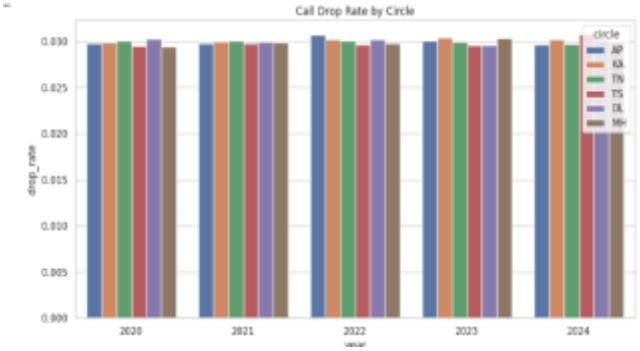

FIG 5.4: Circle-wise Call Drop Rate Analysis

Usage Category Distribution Over Years:
Year-wise usage category distribution showing dominance of low usage customers and stable usage behaviour over time.

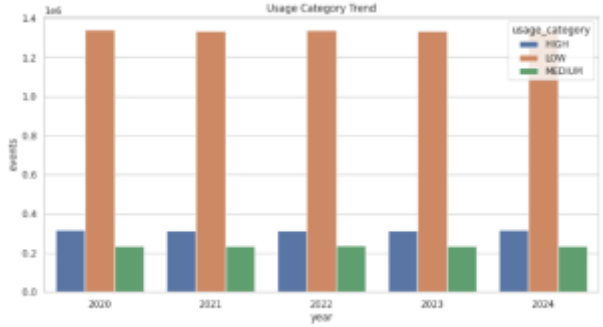

FIG 5.5 : Usage Category Distribution Over Years

ARPU Trend by Operator:
This graph shows the year-wise trend of Average Revenue Per User (ARPU) for different telecom operators from 2020 to 2024. It highlights variations in revenue generated per customer across operators, reflecting differences in pricing strategies and customer usage behavior. The trend helps identify operators with better revenue efficiency over time.

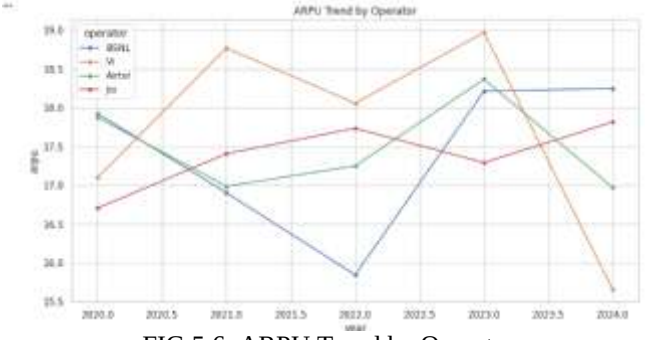

FIG 5.6: ARPU Trend by Operator

Revenue Market Share of Telecom Operators:
Revenue market share distribution among telecom operators showing equal contribution from each operator

FIG 5.7 : Revenue Market Share of Telecom Operators

Monthly Event Volume Pattern Analysis:
This heatmap shows month-wise and year-wise variations in telecom event volumes, highlighting seasonal and consistent usage patterns

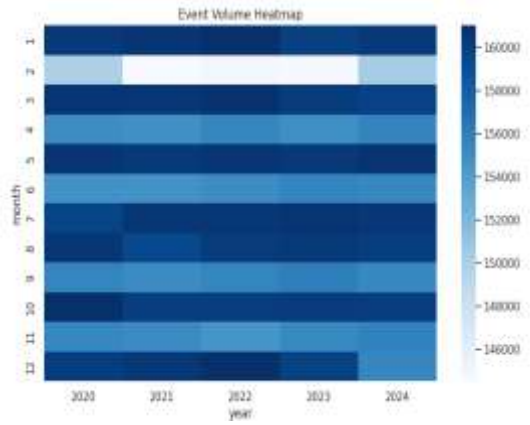

FIG 5.8: Monthly Event Volume Pattern Analysis

Average Charge Distribution:
It compares the average charges (avg_charge) among four mobile network operators: BSNL, Vi, Airtel, and Jio.

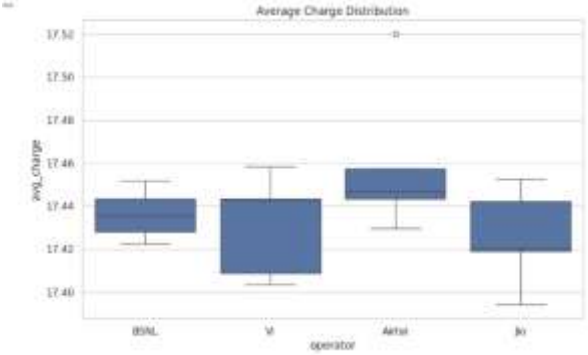

FIG 5.9: Average Charge Distribution

Chunk Risk Distribution:
The chart highlights that while most customers are stable, the high-risk group is significant enough to warrant attention. Proactively addressing their concerns could reduce churn and improve long-term profitability.

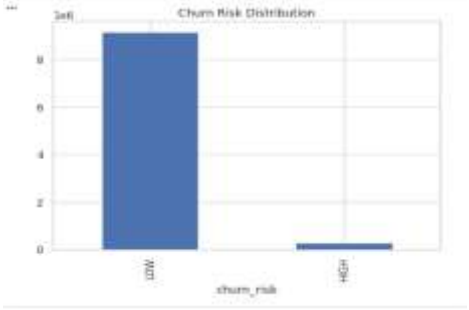

FIG 5.10 : Chunk Risk Distribution

Top Revenue Customers:
This chart is a Bar Chart showing the *Top 10 Revenue Customers*. Each bar represents one customer and the height shows how much they spent. All the bars are nearly equal, close to 140 units, meaning each of these top customers contributes almost the same high amount of revenue. In short, the business has a balanced group of valuable customers who are all important for maintaining steady income.

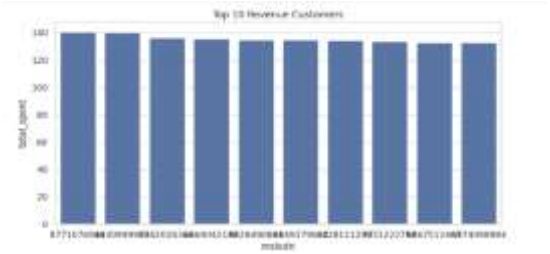

FIG 5.11 : Top Revenue Customers

VI. CONCLUSION

This paper presented a scalable Big Data analytics framework for analyzing telecom Call Detail Records (CDRs) using Apache Spark (PySpark). The proposed system efficiently processes large volumes of telecom data and extracts meaningful insights related to revenue generation, network quality, customer usage behavior, and abnormal activity detection. By leveraging distributed processing and in-memory computation, the system overcomes the limitations of traditional data processing techniques and significantly improves analytical performance.

The experimental analysis demonstrates that Apache Spark provides a reliable and high-performance platform for large-scale telecom data analytics. The modular and fault-tolerant system architecture ensures scalability, flexibility, and efficient resource utilization. Overall, the proposed framework offers a practical and effective solution for real-world telecom data analysis and supports data-driven decision-making for telecom operators.

Future Scope:

Although the proposed system effectively analyzes large-scale telecom Call Detail Records using Apache Spark, several enhancements can be explored in future work. Machine learning and deep learning models can be integrated to enable predictive analytics such as customer churn prediction, fraud detection, and demand forecasting. These models can improve decision-making by learning patterns from historical CDR data.
The system can be extended to support real-time analytics by incorporating Spark Streaming, enabling continuous monitoring of live telecom data streams for instant detection of network issues and abnormal behavior. Integration with advanced visualization dashboards can further enhance interpretability and support interactive data exploration for telecom operators.
Additionally, the framework can be scaled to process terabytes of data in a multi-node cluster environment and integrated with cloud platforms to improve flexibility and resource management. Future studies may also focus on incorporating additional data sources such as social media or network sensor data to provide more comprehensive telecom intelligence.

I. ACKNOWLEDGMENT

The authors would like to express their sincere gratitude **to G. M. Padmaja, Assistant Professor**, Department of Computer Science and Engineering, **SRK Institute of Technology**, for her valuable guidance, continuous support, and encouragement throughout the development of this project. Her insights and suggestions greatly contributed to the successful completion of this work.
The authors also extend their heartfelt thanks to the faculty members of the Department of Computer Science and Engineering for their support and for providing the necessary resources to carry out this research. Finally, the authors would like to thank their institution, **SRK Institute of Technology**, Andhra Pradesh, for providing a conducive environment and infrastructure to successfully complete this project.

REFERENCES

[1] M. Zaharia et al., “Apache Spark: A Unified Engine for Big Data Processing,” Communications of the ACM, 2016.

[2] Machine Learning for Churn Prediction in Telecom – Bifet et al. (2019)

[3] J. Dean and S. Ghemawat, “MapReduce: Simplified Data Processing on Large Clusters,” OSDI, 2004.

[4] Telecom Analytics Survey – IEEE Communications Magazine (2020)

[5] Apache Spark Documentation – Apache Software Foundation