TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

A
Final Report
On

Tuberculosis Monitoring and Analysis using Big Data Stack

SUBMITTED BY:
Nishchal Pokhrel : 080MSDSA016
Rupak Khatiwada: 080MSDSA017
Sabal Thapa : 080MSDSA018
Suman Lamichhane : 080MSDSA021

SUMBITTED TO:
Basanta Joshi, PhD

September 18, 2024

ACKNOWLEDGMENT

We are thankful to **Dr. Basanta Joshi, PhD** for the opportunity to propose the plan of this project. We are also grateful to our teachers and friends who provided us valuable suggestions and insights that have been very helpful for the development of this project.
We extend our deep gratitude to **National Tuberculosis Control Center (NTCC)** for providing us with the valuable datasets required to realize this project.

Nishchal Pokhrel (080MSDSA016.nishchal@pcampus.edu.np)
Rupak Khatiwada (080MSDSA017.rupak@pcampus.edu.np)
Sabal Thapa (080MSDSA018.sabal@pcampus.edu.np)
Suman Lamichhane (080MSDSA021.suman@pcampus.edu.np)

ABSTRACT

This project presents a robust system to analyze the raw Tuberculosis (TB) using the Big Data Pipeline and use the result to monitor TB cases in Nepal while also helping the stakeholders to plan the data-driven interventions throughout the country. The main objective of this effort is to explore the utility of a big data approach in the TB data analysis.

Summary of the procedures, results, databases and the methodologies are presented in this report. Advantages and limitations of the proposed systems are explained finally.

The TB data received from NTCC is ingested into the Hadoop Distributed File System (HDFS) through Apache Flume. Apache Spark is used for data processing and analysis. Finally, New SQL instance, HIVE is used to integrate with the Apache Superset for the visualization of the underlying patterns and analysis of the TB data.

The project aims to realize an efficient data analysis framework to reduce the TB cases in Nepal through efficient dissection of the underlying patterns hidden in the collected data through Health Management Information System (HMIS). Although, the project doesn't completely replace the existing analytical strategies, it is the first step towards that goal.

ABBREVIATIONS

NTCC	National Tuberculosis Control Center
NTP	National Tuberculosis Program
TB	Tuberculosis
HIV	Human Immunodeficiency Virus
ISO	International Organization for Standardization
WHO	World Health Organization
HMIS	Health Management Information System
HDFS	Hadoop Distributed File System
DHIS2	District Health Information Software 2

LIST OF FIGURES

List of Figures

Contents

1 INTRODUCTION

1.1 Background

Tuberculosis (TB) still remains one of the top killers in the Healthcare context and ranks among the top 10 deadliest disease and a major Public Health problem in Nepal. According to the latest Survey **Prevalence Survey 2020** ,there are about $1,17,000$ people living with active tuberculosis in Nepal, with more than 69,000 new cases and death toll reaching approximately 17000 every year**WHO**; **2023** and only 37447 cases has been identified **TB Fact Sheet**; **2079/80**, and nearly 46% of the cases go missing, a major setback for the National Tuberculosis Program (NTP).
With this project, we aim to provide deeper explanation on the potential gaps of TB case notification, giving further insights for the targeted interventions and resource allocation to diagnose and notify prevalent TB cases in Nepal. This will help devise actionable plans and helps to achieve the EndTB targets that the Government of Nepal has vouched for.

1.2 Statement of the Problem

Nepal hopes to end the TB Epidemic by 2035 and eliminate TB by 2050 **TB Fact Sheet**; **2079/80**, however, it looks far fetched, given the present TB case notification rate. The need to change the current interventions and introduce newer innovative approach is crucial to achieve the desired target. The situation is critical and it is imperative to design, endorse and implement the newer and scientific interventions backed by data analysis to increase the TB diagnosis and plan effective strategies to control it. There is a need for a robust data analysis framework that can integrate and analyze TB data from multiple sources to identify vulnerable populations, detect geographical clusters of cases, and uncover trends and patterns that can inform targeted public health interventions.

2 Theoretical Background

Some of the tools and services used in the Big Data pipeline is briefly described in the subsequent sections.

2.1 Apache Flume

Apache Flume is a distributed, reliable, and available service designed for efficiently collecting, aggregating, and moving large amounts of log data from various sources to a centralized data store. Flume's architecture consists of sources to collect data, channels to buffer it, and sinks to deliver it to destinations, providing a flexible and extensible solution for data ingestion needs.

2.2 Apache Spark

Apache Spark is an open-source, distributed computing system designed for fast processing of large-scale data. It provides an interface for programming entire clusters with implicit data parallelism and fault tolerance. Spark's in-memory processing capabilities make it significantly faster than traditional disk-based Hadoop MapReduce jobs. It supports a wide range of applications, including batch processing, interactive querying, real-time analytics, machine learning, and graph processing, through its core components and libraries such as Spark SQL, Spark Streaming, MLlib, and GraphX. Spark's flexibility and scalability have made it a popular choice for big data analytics and processing.

2.3 Hadoop Distributed File System (HDFS)

Apache Hadoop Distributed File System (HDFS) is a scalable and reliable storage system designed for storing large datasets across multiple machines. It provides high-throughput access to data, ensuring fault tolerance through data replication across nodes. Optimized for large-scale batch processing, HDFS works closely with computational frameworks like Apache MapReduce to enhance data locality and performance.

2.4 Apache Hive

Apache Hive is a data warehouse infrastructure built on top of Hadoop that provides tools for querying and managing large datasets using a SQL-like language called HiveQL. It enables users to perform data analysis and aggregation on large volumes of data stored in Hadoop's HDFS or other compatible storage systems. Hive translates HiveQL queries into low-level MapReduce jobs, making it easier to interact with big data without needing to write complex MapReduce code. It supports a variety of data formats, including text files, RCFile, ORC, and Parquet, allowing for flexible data storage and retrieval. Hive's architecture also supports user-defined functions and custom input/output formats, enhancing its extensibility and integration with other data processing tools.

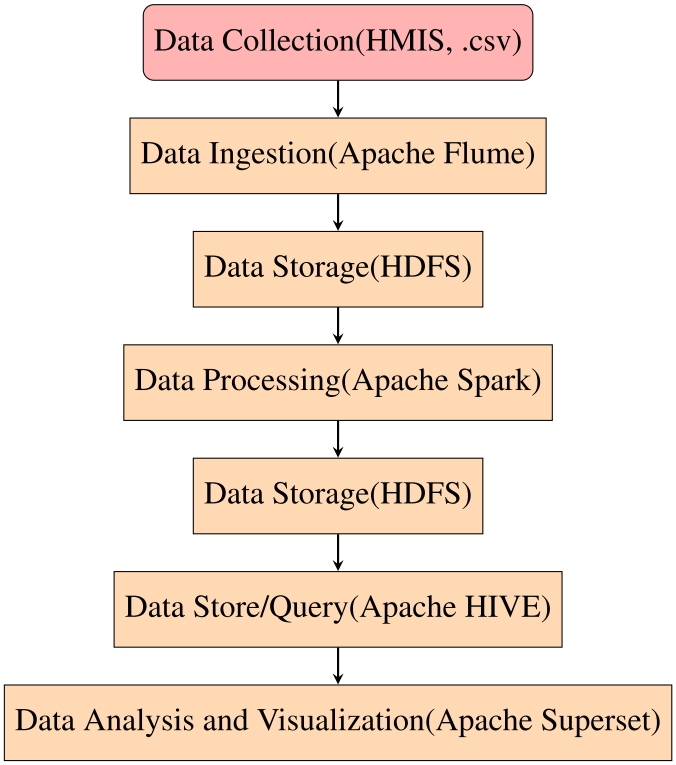
2.5 Apache Superset

Apache Superset is an open-source data exploration and visualization platform designed for creating interactive dashboards and data visualizations. It provides a user-friendly interface for querying and analyzing large datasets from various data sources, supporting a wide range of visualizations such as charts, maps, and graphs. Superset's flexibility, scalability, and ease of integration with big data tools make it a powerful choice for data analysts and business intelligence applications.

3 Methodology

Flow of project methods is shown below and methodologies incorporated are briefly described here. It contains 6 main components.

Data Processing Pipeline Flowchart


Data Collection(HMIS, .csv)
Data Ingestion(Apache Flume)
Data Storage(HDFS)
Data Processing(Apache Spark)
Data Storage(HDFS)
Data Store/Query(Apache HIVE)
Data Analysis and Visualization(Apache Superset)

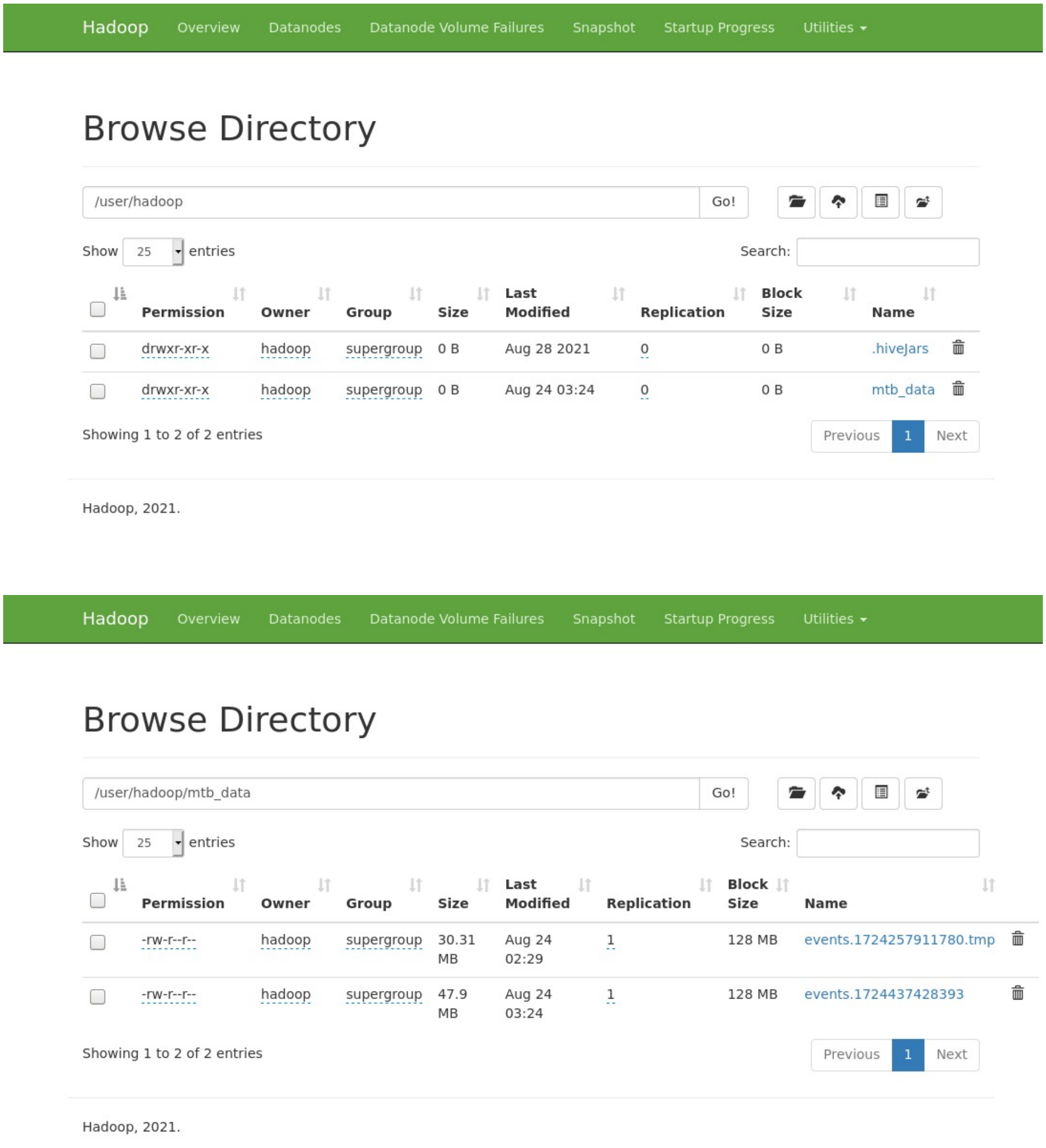
3.1 Data Collection

HMIS (Hospital Management Information System) is a systematic approach used to collect, manage, and analyze health-related data to improve healthcare delivery and decision-making in Nepal. The HMIS is typically managed by the Ministry of Health and Population (MoHP) in Nepal, in collaboration with other health organizations and international partners. National Tuberculosis Control Center (NTCC) has provided us with the data from the Dhis2 platform to use in our project. Due to confidentiality, patient name has been removed. The data is in the .csv format. The outline of the data provided is shown in the **Appendix** A.

3.2 Data Ingestion

After collection of data from NTCC, Apache flume is used for data ingestion into the Hadoop Distributed File System (HDFS). The configuration of Flume is shown in the **Appendix** B. The process for data ingestion using Apache Flume is given below :

- Configuration was done to specify source, sink and channels to buffer the data for ingestion.
- Data was ingested into the HDFS.
- Flume agent monitors the input folder (project_mtb) and ingests the data into the HDFS automatically in folder (mtb_data) for further processing. Refer to Figure 1



Browse Directory

/user/hadoop

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	drwxr-xr-x	hadoop	supergroup	0 B	Aug 28 2021	0	0 B	.hiveJars
□	drwxr-xr-x	hadoop	supergroup	0 B	Aug 24 03:24	0	0 B	mtb_data

Showing 1 to 2 of 2 entries

Hadoop, 2021.

Browse Directory

/user/hadoop/mtb_data

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	-rw-r--r--	hadoop	supergroup	30.31 MB	Aug 24 02:29	1	128 MB	events.1724257911780.tmp
□	-rw-r--r--	hadoop	supergroup	47.9 MB	Aug 24 03:24	1	128 MB	events.1724437428393

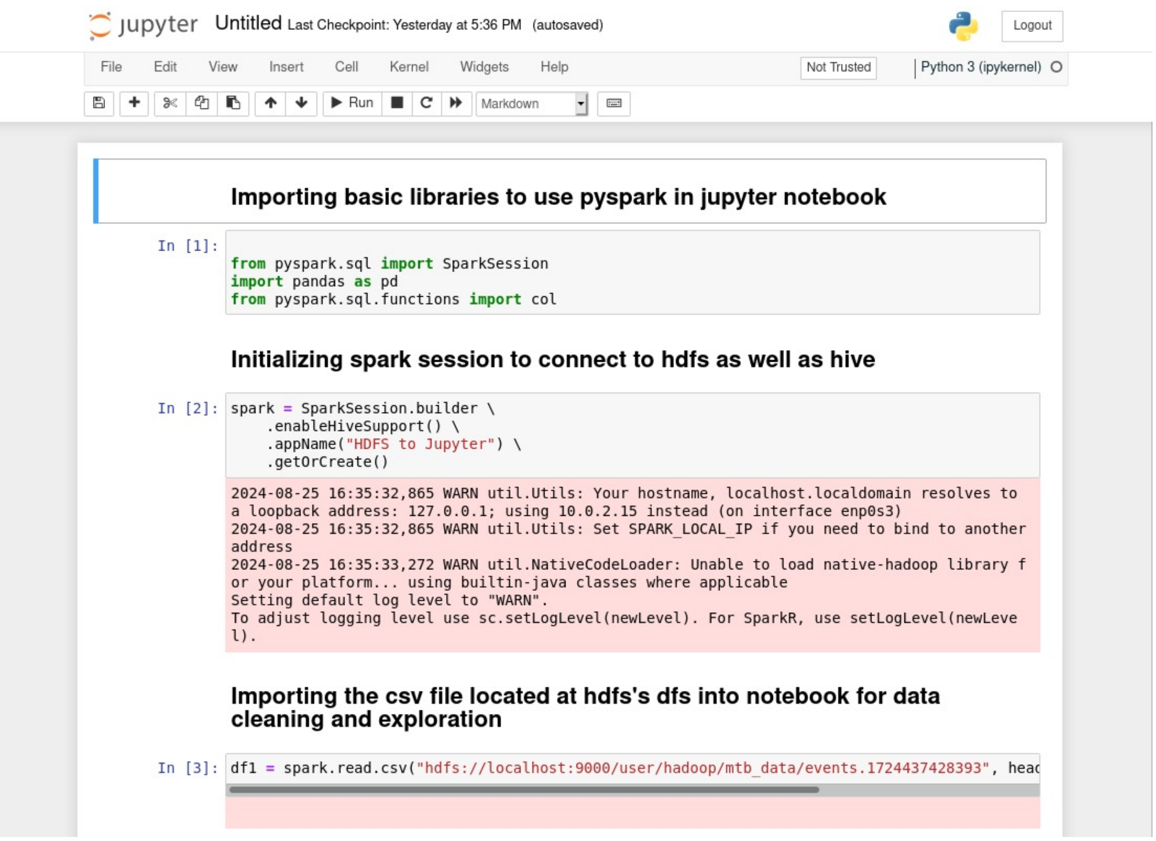
Showing 1 to 2 of 2 entries

Hadoop, 2021.

Figure 1: Flume Agent Monitors continuously and ingests data

3.3 Data Processing

The ingested data from the HDFS was then cleaned and processed using Apache Spark. Python programming language was used and for the processing tasks. The final output was saved in the HDFS again. Refer to the complete PySpark code in the **Appendix** C. See below figure 2 for a snapshot.



Importing basic libraries to use pyspark in jupyter notebook

```
In [1]: from pyspark.sql import SparkSession
        import pandas as pd
        from pyspark.sql.functions import col
```

Initializing spark session to connect to hdfs as well as hive

```
In [2]: spark = SparkSession.builder \
            .enableHiveSupport() \
            .appName("HDFS to Jupyter") \
            .getOrCreate()
```

```
2024-08-25 16:35:32,865 WARN util.Utils: Your hostname, localhost.localdomain resolves to
a loopback address: 127.0.0.1; using 10.0.2.15 instead (on interface enp0s3)
2024-08-25 16:35:32,865 WARN util.Utils: Set SPARK_LOCAL_IP if you need to bind to another
address
2024-08-25 16:35:33,272 WARN util.NativeCodeLoader: Unable to load native-hadoop library f
or your platform... using builtin-java classes where applicable
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLeve
l).
```

Importing the csv file located at hdfs's dfs into notebook for data cleaning and exploration

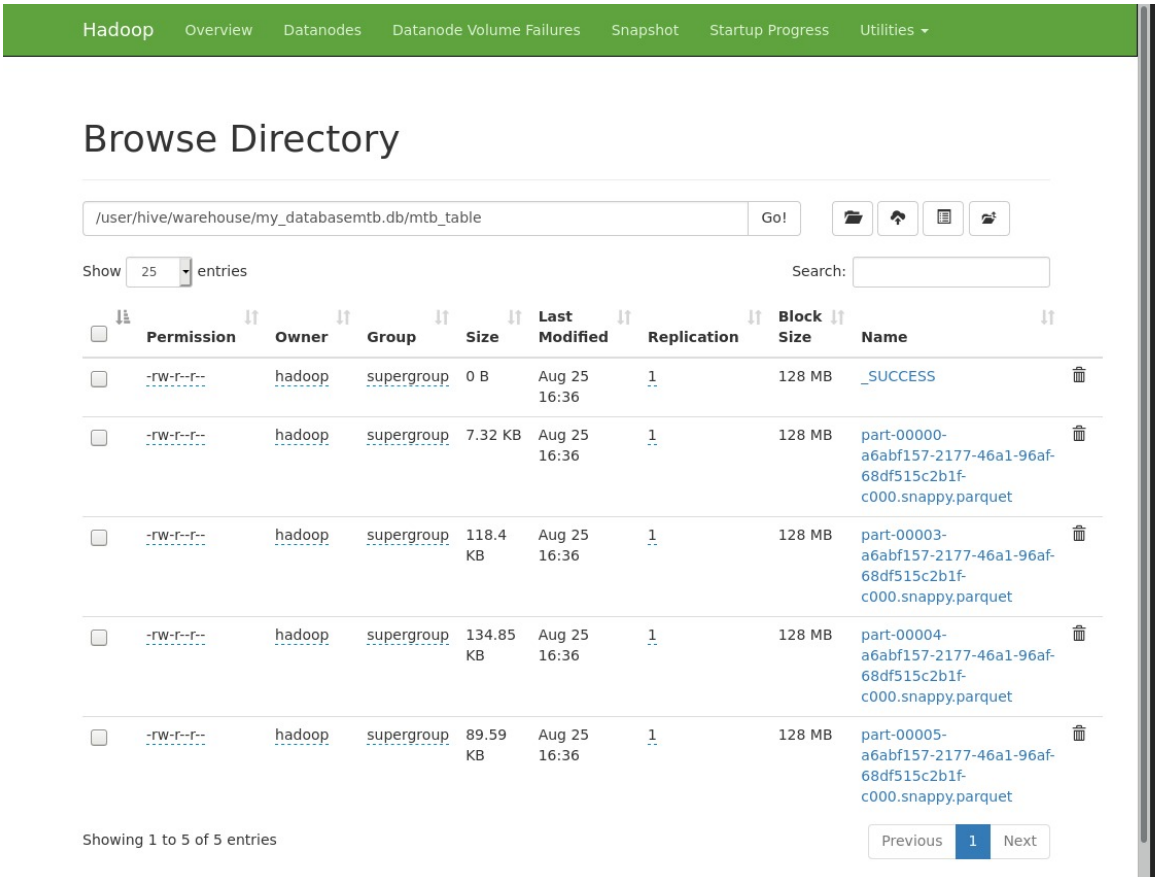
```
In [3]: df1 = spark.read.csv("hdfs://localhost:9000/user/hadoop/mtb_data/events.1724437428393", heac
```

Figure 2: Pyspark Data Processing

3.4 Data Storage

HDFS is a primary data storage for data in our project. HDFS is primarily used to:

- Store raw data and processed data.
- Load processed data into the data warehouse. Apache Hive has been used for its seamless integration with the Apache Superset for the data visualization. Refer to figure 3.



Browse Directory

/user/hive/warehouse/my_databasemtb.db/mtb_table

Show 25 entries

Search:

□	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
□	-rw-r--r--	hadoop	supergroup	0 B	Aug 25 16:36	1	128 MB	_SUCCESS
□	-rw-r--r--	hadoop	supergroup	7.32 KB	Aug 25 16:36	1	128 MB	part-00000-a6abf157-2177-46a1-96af-68df515c2b1f-c000.snappy.parquet
□	-rw-r--r--	hadoop	supergroup	118.4 KB	Aug 25 16:36	1	128 MB	part-00003-a6abf157-2177-46a1-96af-68df515c2b1f-c000.snappy.parquet
□	-rw-r--r--	hadoop	supergroup	134.85 KB	Aug 25 16:36	1	128 MB	part-00004-a6abf157-2177-46a1-96af-68df515c2b1f-c000.snappy.parquet
□	-rw-r--r--	hadoop	supergroup	89.59 KB	Aug 25 16:36	1	128 MB	part-00005-a6abf157-2177-46a1-96af-68df515c2b1f-c000.snappy.parquet

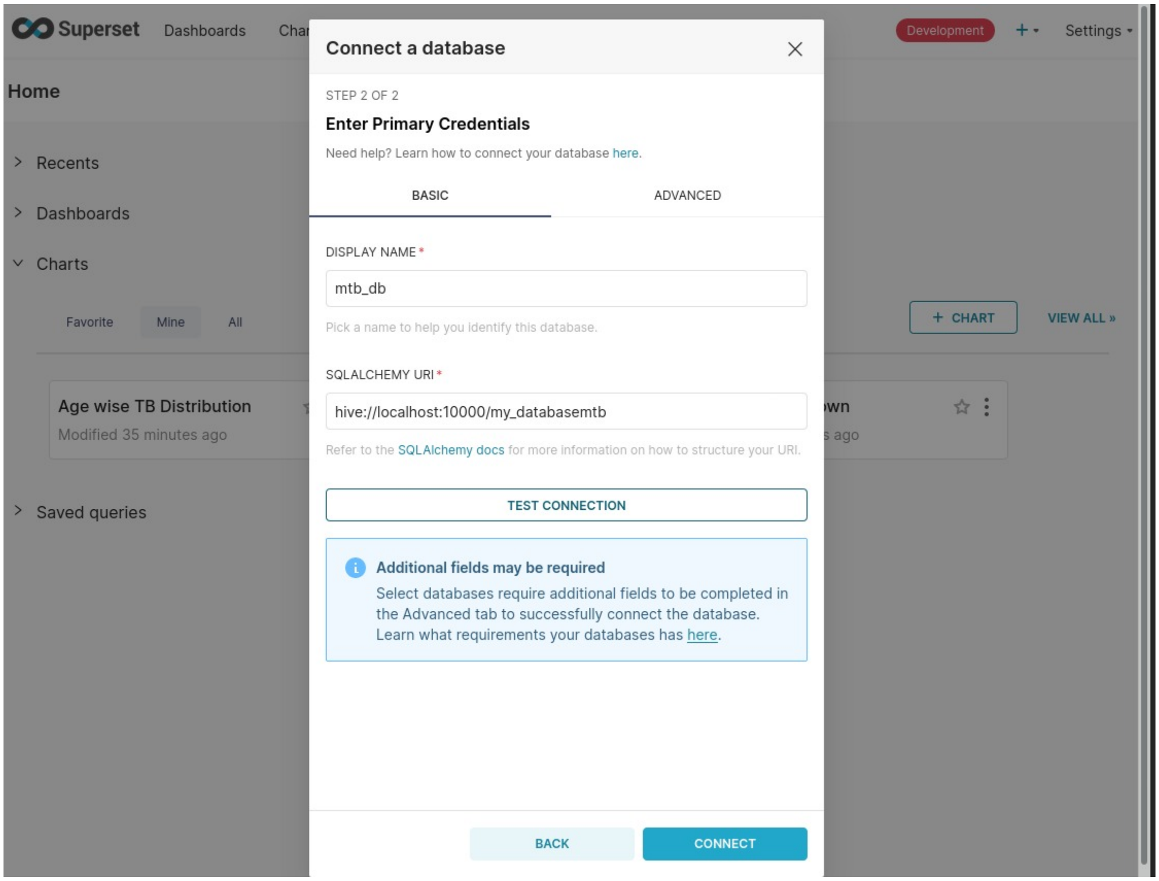
Showing 1 to 5 of 5 entries

Figure 3: Hive Table Import Successful

3.5 Data Analysis and Visualization

After the processed data has been loaded into the HIVE database and appropriate tables has been formed, it is connected with the Apache Superset with SQL Alchemy URL. Following processes has been performed for Superset Visualization:

- Load processed data from storage (Apache Hive). Refer to Figure 4.



Connect a database

STEP 2 OF 2

Enter Primary Credentials

Need help? Learn how to connect your database here.

DISPLAY NAME *

mtb_db

Pick a name to help you identify this database.

SQLALCHEMY URI *

hive://localhost:10000/my_databasemtb

Refer to the SQLAlchemy docs for more information on how to structure your URI.

Additional fields may be required

Select databases require additional fields to be completed in the Advanced tab to successfully connect the database. Learn what requirements your databases has here.

Figure 4: Connecting Hive DB with Superset

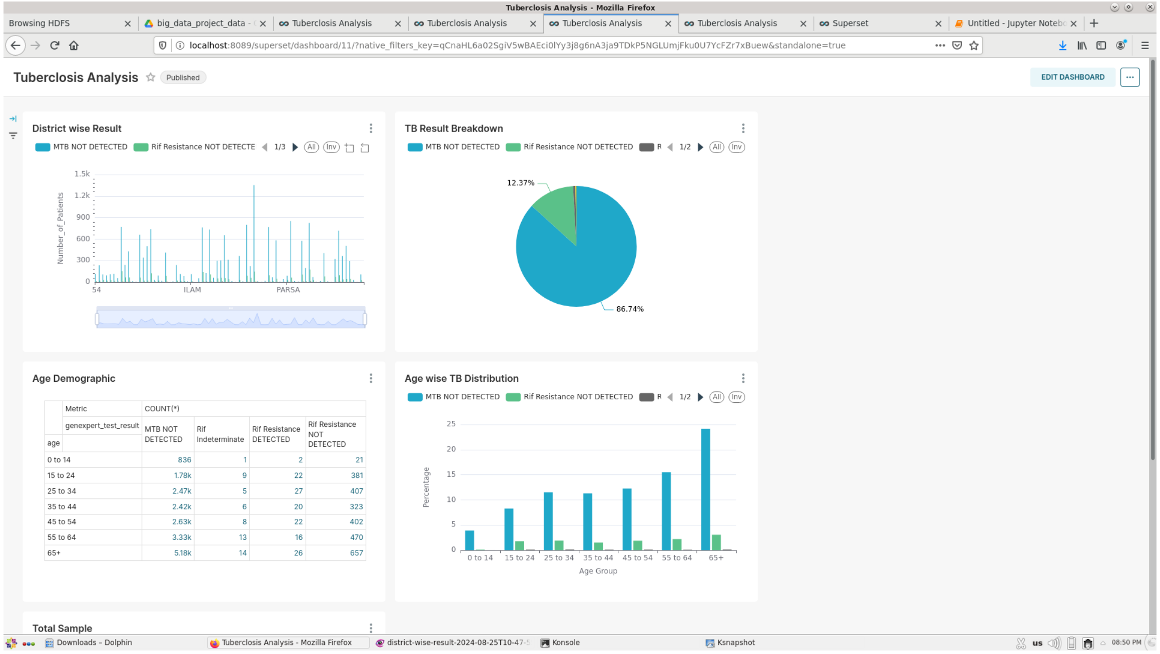
- Perform exploratory data analysis (EDA) and generate visualizations.
- Create dashboards to monitor key metrics.

4 Results and Interpretation

We obtained raw unprocessed data from National Tuberculosis Control Center (NTCC) and the data was processed and analysed.

4.1 Results

- **Dashboard** : The complex data is transformed into visual formats such as bargraphs, histograms, pie charts etc making it easier to grasp patters, trends and outcomes quickly.



	Metric	COUNT(*)			
	genexpert_test_result	MTB NOT DETECTED	Rif Indeterminate	Rif Resistance DETECTED	Rif Resistance NOT DETECTED
age					
0 to 14		836	1	2	21
15 to 24		1.78k	9	22	381
25 to 34		2.47k	5	27	407
35 to 44		2.42k	6	20	323
45 to 54		2.63k	8	22	402
55 to 64		3.33k	13	16	470
65+		5.18k	14	26	657

Figure 5: Superset Dashboard

- **Support to NTP**: The project is aimed to aid the policy makers and authorities to design, endorse and implement targeted interventions to increase the case notification and help to overall improve the public health.
- **Geographical Distribution**: Through this analysis, one can comprehend the TB disease distribution scenario in Nepal and helps to find the hotspots.
- **Vulnerable Population**: Identify population that are at higher risk of contracting TB disease on the basis of age, socio-economic status, lifestyles etc.

  Some of the snapshots are given below:

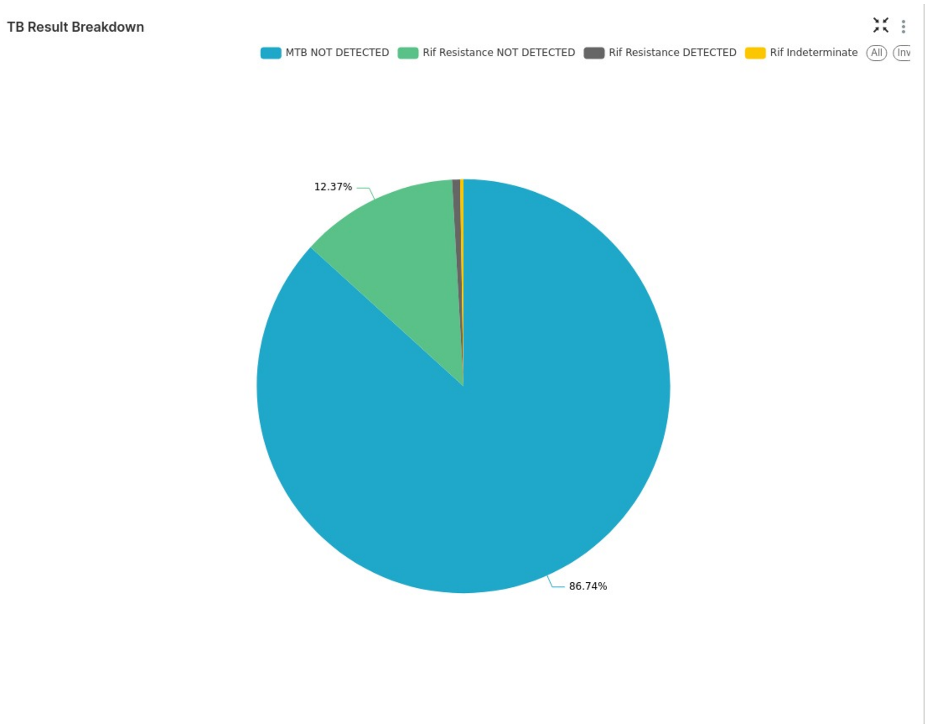


Figure 6: Overall TB Result Breakdown

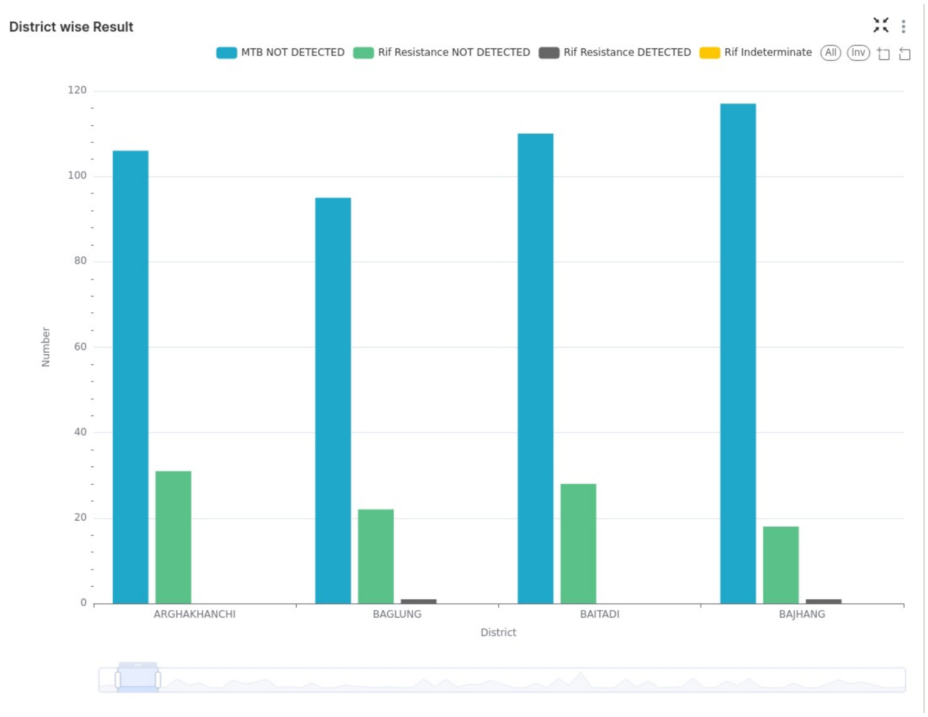


Figure 7: Geographical Breakdown of TB Cases

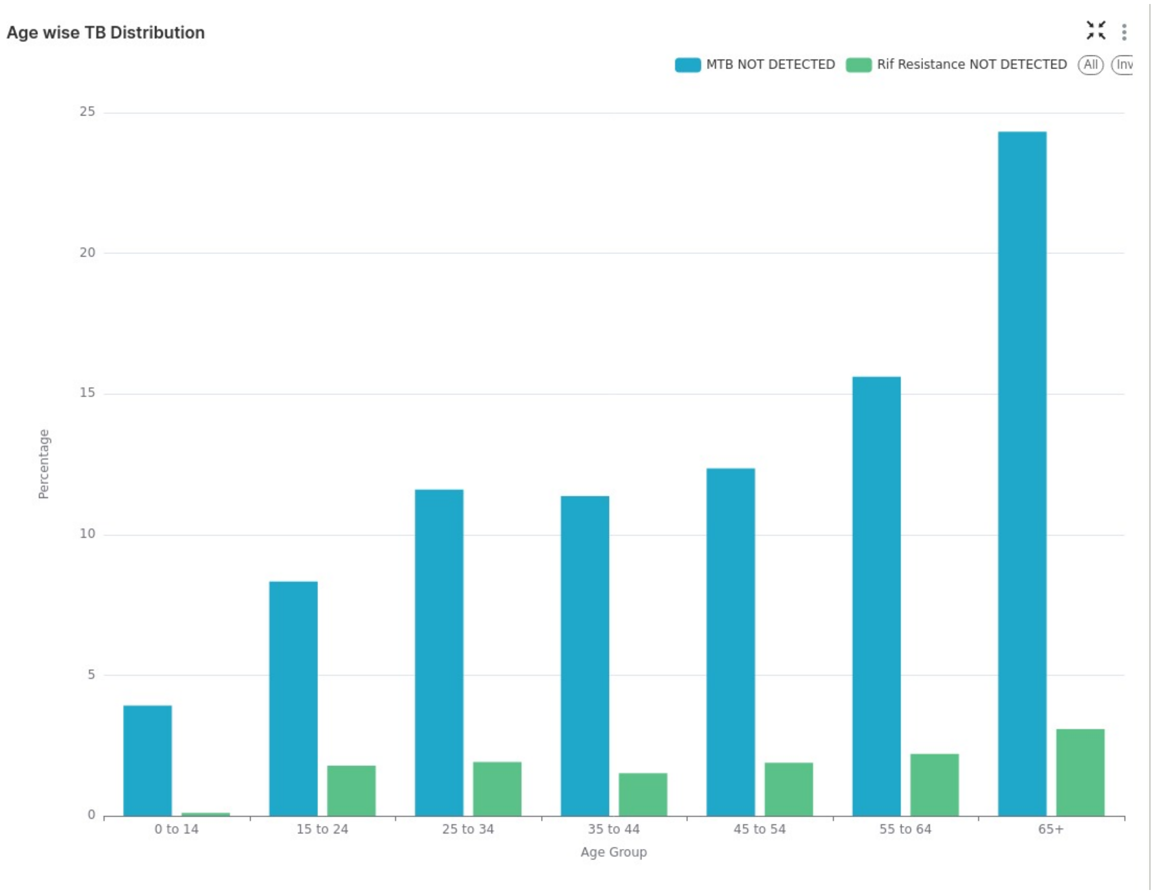


Figure 8: Age-Wise Distribution of TB Cases

4.2 Interpretation

The analysis and visualizations provided in this report offer valuable insights into the current state of tuberculosis (TB) in Nepal. From the geographical breakdown, we have identified Darchula as the highest cases of TB in percentage. The age-wise distribution reveals that older people with age 65+ are more vulnerable to TB, enabling targeted health campaigns and resource allocation. Additionally, the overall breakdown of TB cases provides a clear picture of the current burden of the disease, highlighting the need for continued efforts to improve TB case detection and notification rates. These insights are crucial for refining public health strategies and achieving national TB control goals.

5 Limitation of the Project

The TB monitoring and analysis framework are constrained by following limitations:

1. The project is primarily focused on analyzing data from the District Health Information Software 2 (DHIS2) provided by NTCC, and any data that are not included in DHIS2 is not taken into consideration.

2. The project is designed for batch data processing and not real-time data.

3. Due to lack of patient data like symptoms, past history, lifestyles, no predictive analysis has been performed.

6 Future Enhancements

1. The project can be expanded to real-time data processing and analysis.

2. With additional dimension in our data, predictive model can be incorporated to predict the probability of TB infection prior to laboratory testing.

3. This project is realized on a single machine. The system can be deployed in distributed environment to handle larger datasets, improve processing speed, and ensure scalability for real-time analysis across multiple nodes or cloud infrastructure.

7 Conclusion

”Tuberculosis Monitoring and Analysis using Big Data Pipeline” demonstrates the potential of leveraging big data technologies to address critical public health issues like tuberculosis (TB) in Nepal. By utilizing a comprehensive data pipeline that integrates tools such as Apache Flume, Apache Spark, HDFS, Apache Hive, and Apache Superset, we were able to process and analyze TB data efficiently. The resulting dashboards and visualizations offer actionable insights that can aid policymakers and healthcare professionals in targeting interventions, identifying vulnerable populations, and improving TB case notification rates.

While the project has limitations, such as focusing on batch processing and reliance on data from the DHIS2 platform, it lays a foundation for future enhancements, including real-time data processing and predictive modeling. The scalability and flexibility of the system make it a promising approach for expanding its scope to include more complex datasets and broader public health applications.

In conclusion, this project represents a significant step towards utilizing big data for public health surveillance and decision-making in Nepal. By providing deeper insights into TB trends and hotspots, it supports the National Tuberculosis Program’s efforts to reduce TB incidence and move closer to achieving the EndTB targets. Continued development and deployment of such data-driven systems will be crucial in tackling TB and other public health challenges in the years to come.

References

[1] National Tuberculosis Control Center, *NATIONAL TUBERCULOSIS PREVALENCE SURVEY REPORT*, 2020. https://nepalntp.gov.np/wp-content/uploads/2021/03/NTPS-Report-Bodypages.pdf

[2] National Tuberculosis Control Center, *TUBERCULOSIS PROFILE 2079/80 (2022/23)* https://nepalntp.gov.np/wp-content/uploads/2024/04/Latest-TB-profile-for-update-in-NTCC-website.pdf

[3] World Health Organization, *Transforming Drug-Resistant TB Treatment in Nepal*, https://www.who.int/nepal/news/detail/22-12-2023-transforming-drug-resistant-tb-treatment-in-nepal, Accessed: 2024-07-18.

[4] Nathan Marz, James Warren, *Big Data*, Manning Publications, 2015.

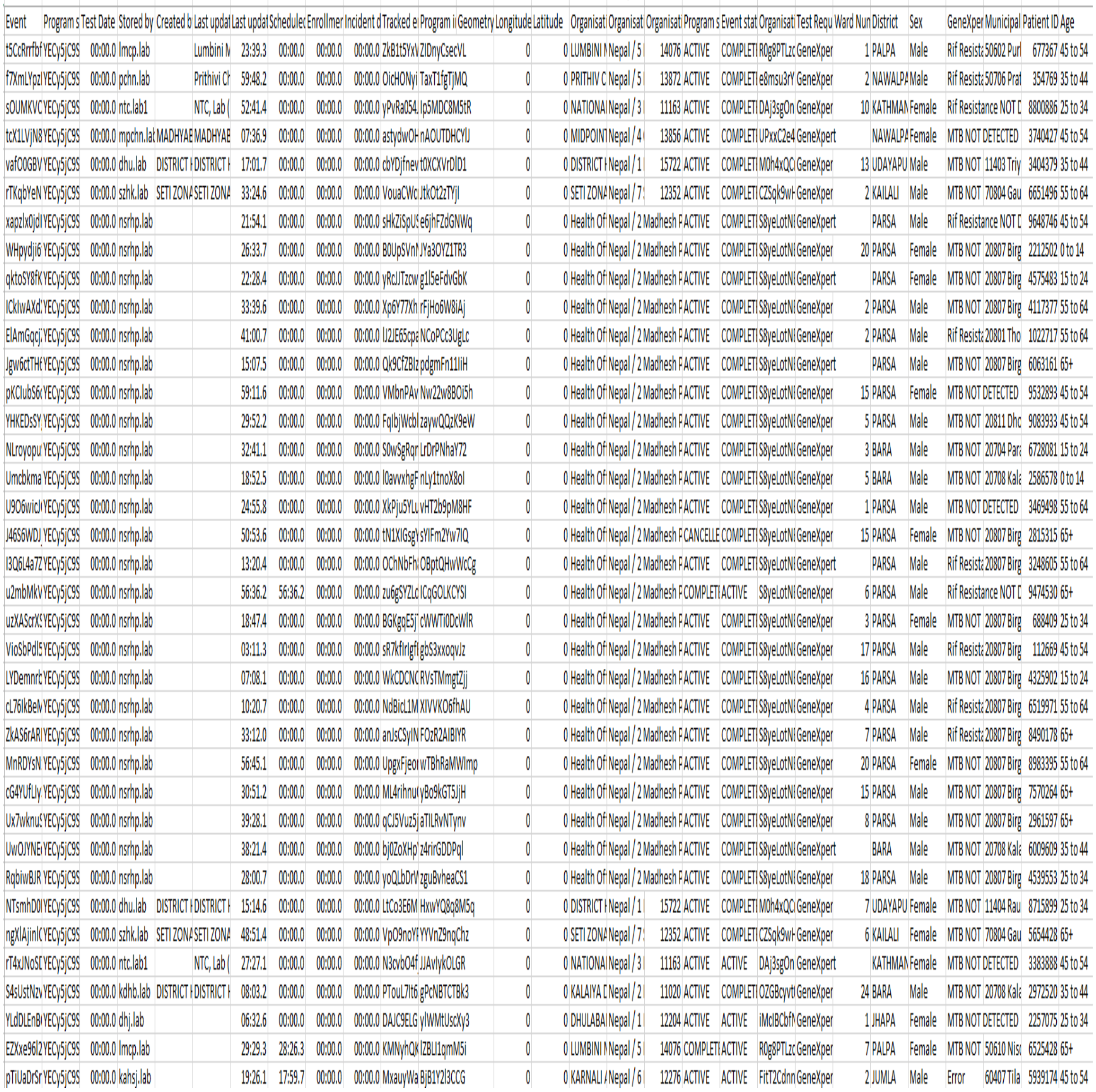
[5] Holden Karau, Andy Konwinski, Patrick Wendell & Matei Zaharia, *Learning Spark*,O'Reilly Media, Inc., 2015.

[6] Tom White, *Hadoop, The Definitive Guide*, Fourth, O'Reilly Media, Inc., April 2015.

8 APPENDIX

A Data Review

Snapshot of the data obtained from National Tuberculosis Control Center is given below :



Event	Program s	Test Date	Stored by	Created b	Last updat	Last updat	Schedulec	Enrollmer	Incident d	Tracked e	Program i	Geometry	Longitude	Latitude	Organisati	Organisati	Organisati	Program s	Event stat	Organisati	Test Requ	Ward Nun	District	Sex	GeneXper	Municipal	Patient ID	Age
t5CcRrrfbf	YECy5jC9S	00:00.0	lmcp.lab		Lumbini M	23:39.3	00:00.0	00:00.0	00:00.0	ZkB1t5YxV	ZIDnyCsecVL		0	0	LUMBINI N	Nepal / 5 I	14076	ACTIVE	COMPLETI	R0g8PTLzc	GeneXper	1	PALPA	Male	Rif Resista	50602 Purl	677367	45 to 54
f7XmLYpzl	YECy5jC9S	00:00.0	pchn.lab		Prithivi Ch	59:48.2	00:00.0	00:00.0	00:00.0	OicHONyi	TaxT1fgTjMQ		0	0	PRITHIV C	Nepal / 5 I	13872	ACTIVE	COMPLETI	e8msu3rY	GeneXper	2	NAWALPA	Male	Rif Resista	50706 Prat	354769	35 to 44
sOUMKVC	YECy5jC9S	00:00.0	ntc.lab1		NTC, Lab (	52:41.4	00:00.0	00:00.0	00:00.0	yPvRa054	Ip5MDC8M5tR		0	0	NATIONAI	Nepal / 3 I	11163	ACTIVE	COMPLETI	DAj3sgOn	GeneXper	10	KATHMAN	Female	Rif Resistance NOT D		8800886	25 to 34
tcX1LVjN8	YECy5jC9S	00:00.0	mpchn.lak	MADHYAE	MADHYAE	07:36.9	00:00.0	00:00.0	00:00.0	astydwOH	nAOUTDHCYlJ		0	0	MIDPOINT	Nepal / 4 (	13856	ACTIVE	COMPLETI	UPxxC2e4	GeneXpert		NAWALPA	Female	MTB NOT DETECTED		3740427	45 to 54
vafO0GBV	YECy5jC9S	00:00.0	dhu.lab	DISTRICT H	DISTRICT H	17:01.7	00:00.0	00:00.0	00:00.0	cbYDjfnev	t0XCXVrDlD1		0	0	DISTRICT H	Nepal / 1 I	15722	ACTIVE	COMPLETI	M0h4xQC	GeneXper	13	UDAYAPU	Male	MTB NOT	11403 Triy	3404379	35 to 44
rTKqbYeN	YECy5jC9S	00:00.0	szhk.lab	SETI ZONA	SETI ZONA	33:24.6	00:00.0	00:00.0	00:00.0	VouaCWc	JtkOt2zTYjI		0	0	SETI ZONA	Nepal / 7	12352	ACTIVE	COMPLETI	CZSqk9wH	GeneXper	2	KAILALI	Male	MTB NOT	70804 Gau	6651496	55 to 64
xapzlx0jdl	YECy5jC9S	00:00.0	nsrhp.lab			21:54.1	00:00.0	00:00.0	00:00.0	sHkZiSpUS	e6jhFZdGNWq		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXpert		PARSA	Male	Rif Resistance NOT D		9648746	45 to 54
WHpydji6	YECy5jC9S	00:00.0	nsrhp.lab			26:33.7	00:00.0	00:00.0	00:00.0	B0UpSVnN	JYa3OYZ1TR3		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	20	PARSA	Female	MTB NOT	20807 Birg	2212502	0 to 14
qktoSY8fK	YECy5jC9S	00:00.0	nsrhp.lab			22:28.4	00:00.0	00:00.0	00:00.0	yRcJJTzcw	g1l5eFdvGbK		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXpert		PARSA	Female	MTB NOT	20807 Birg	4575483	15 to 24
ICkIwAXd	YECy5jC9S	00:00.0	nsrhp.lab			33:39.6	00:00.0	00:00.0	00:00.0	Xp6Y77Xh	rFjHo6W8iAj		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	2	PARSA	Male	MTB NOT	20807 Birg	4117377	55 to 64
ElAmGqcj	YECy5jC9S	00:00.0	nsrhp.lab			41:00.7	00:00.0	00:00.0	00:00.0	lJ2JE65cpa	NCoPCc3UgLc		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	2	PARSA	Male	Rif Resista	20801 Tho	1022717	55 to 64
Jgw6ctTH6	YECy5jC9S	00:00.0	nsrhp.lab			15:07.5	00:00.0	00:00.0	00:00.0	Qk9CfZBIz	pdgmFn11IiH		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXpert		PARSA	Male	MTB NOT	20807 Birg	6063161	65+
pKClubS6	YECy5jC9S	00:00.0	nsrhp.lab			59:11.6	00:00.0	00:00.0	00:00.0	VMbnPAv	Nw22w8BOi5h		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	15	PARSA	Female	MTB NOT DETECTED		9532893	45 to 54
YHKEDsSY	YECy5jC9S	00:00.0	nsrhp.lab			29:52.2	00:00.0	00:00.0	00:00.0	FqIbjWcbl	zaywQQzK9eW		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	5	PARSA	Male	MTB NOT	20811 Dhc	9083933	45 to 54
NLroyopu	YECy5jC9S	00:00.0	nsrhp.lab			32:41.1	00:00.0	00:00.0	00:00.0	S0wSgRqr	LrDrPNhaY72		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	3	BARA	Male	MTB NOT	20704 Para	6728081	15 to 24
Umcbkma	YECy5jC9S	00:00.0	nsrhp.lab			18:52.5	00:00.0	00:00.0	00:00.0	l0avvxhgF	nLy1tnoX8oI		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	5	BARA	Male	MTB NOT	20708 Kala	2586578	0 to 14
U9O6wicJ	YECy5jC9S	00:00.0	nsrhp.lab			24:55.8	00:00.0	00:00.0	00:00.0	XkPju5YLu	vHT2b9pM8HF		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	1	PARSA	Male	MTB NOT DETECTED		3469498	55 to 64
J46S6WDJ	YECy5jC9S	00:00.0	nsrhp.lab			50:53.6	00:00.0	00:00.0	00:00.0	tN1XIGsgY	sYIFm2Yw7lQ		0	0	Health Of	Nepal / 2	Madhesh P	CANCELLE	COMPLETI	S8yeLotN	GeneXper	15	PARSA	Female	MTB NOT	20807 Birg	2815315	65+
I3Q6L4a7Z	YECy5jC9S	00:00.0	nsrhp.lab			13:20.4	00:00.0	00:00.0	00:00.0	OChNbFh	OBptQHwWcCg		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXpert		PARSA	Male	Rif Resista	20807 Birg	3248605	55 to 64
u2mbMkV	YECy5jC9S	00:00.0	nsrhp.lab			56:36.2	56:36.2	00:00.0	00:00.0	zu6gSYZLc	ICqGOLKCYSI		0	0	Health Of	Nepal / 2	Madhesh P	COMPLETI	ACTIVE	S8yeLotN	GeneXper	6	PARSA	Male	Rif Resistance NOT D		9474530	65+
uzXAScrX	YECy5jC9S	00:00.0	nsrhp.lab			18:47.4	00:00.0	00:00.0	00:00.0	BGKgqE5j	cWWTi0DcWlR		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	3	PARSA	Female	MTB NOT	20807 Birg	688409	25 to 34
VioSbPdl5	YECy5jC9S	00:00.0	nsrhp.lab			03:11.3	00:00.0	00:00.0	00:00.0	sR7kfIrlgf	gbS3xxoqvJz		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	17	PARSA	Male	Rif Resista	20807 Birg	112669	45 to 54
LYDemnrb	YECy5jC9S	00:00.0	nsrhp.lab			07:08.1	00:00.0	00:00.0	00:00.0	WkCDCNC	RVsTMmgtZjj		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	16	PARSA	Male	MTB NOT	20807 Birg	4325902	15 to 24
cL76lkBeM	YECy5jC9S	00:00.0	nsrhp.lab			10:20.7	00:00.0	00:00.0	00:00.0	NdBicL1M	XIVVKO6fhAU		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	4	PARSA	Male	Rif Resista	20807 Birg	6519971	55 to 64
ZkAS6rARI	YECy5jC9S	00:00.0	nsrhp.lab			33:12.0	00:00.0	00:00.0	00:00.0	anJsCSyIN	FOzR2AIBIYR		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	7	PARSA	Male	Rif Resista	20807 Birg	8490178	65+
MnRDYsN	YECy5jC9S	00:00.0	nsrhp.lab			56:45.1	00:00.0	00:00.0	00:00.0	UpgxFjeor	wTBhRaMWImp		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	20	PARSA	Female	MTB NOT	20807 Birg	8983395	55 to 64
cG4YUfLly	YECy5jC9S	00:00.0	nsrhp.lab			30:51.2	00:00.0	00:00.0	00:00.0	ML4rihnu	yBo9kGT5JjH		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	15	PARSA	Male	MTB NOT	20807 Birg	7570264	65+
Ux7wknuS	YECy5jC9S	00:00.0	nsrhp.lab			39:28.1	00:00.0	00:00.0	00:00.0	qCJ5Vuz5j	aTILRvNTynv		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	8	PARSA	Male	MTB NOT	20807 Birg	2961597	65+
UwOJYNE	YECy5jC9S	00:00.0	nsrhp.lab			38:21.4	00:00.0	00:00.0	00:00.0	bj0ZoXHp	z4rirGDDPql		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXpert		BARA	Male	MTB NOT	20708 Kala	6009609	35 to 44
RqbiwBJR	YECy5jC9S	00:00.0	nsrhp.lab			28:00.7	00:00.0	00:00.0	00:00.0	yoQLbDrV	zguBvheaCS1		0	0	Health Of	Nepal / 2	Madhesh P	ACTIVE	COMPLETI	S8yeLotN	GeneXper	18	PARSA	Male	MTB NOT	20807 Birg	4539553	25 to 34
NTsmhD0I	YECy5jC9S	00:00.0	dhu.lab	DISTRICT H	DISTRICT H	15:14.6	00:00.0	00:00.0	00:00.0	LtCo3E6M	HxwYQ8q8M5q		0	0	DISTRICT H	Nepal / 1 I	15722	ACTIVE	COMPLETI	M0h4xQC	GeneXper	7	UDAYAPU	Female	MTB NOT	11404 Rau	8715899	25 to 34
ngXlAjinl	YECy5jC9S	00:00.0	szhk.lab	SETI ZONA	SETI ZONA	48:51.4	00:00.0	00:00.0	00:00.0	VpO9noYF	YYVnZ9nqChz		0	0	SETI ZONA	Nepal / 7	12352	ACTIVE	COMPLETI	CZSqk9wH	GeneXper	6	KAILALI	Female	MTB NOT	70804 Gau	5654428	65+
rT4xJNoSD	YECy5jC9S	00:00.0	ntc.lab1		NTC, Lab (	27:27.1	00:00.0	00:00.0	00:00.0	N3cvbO4f	JJAvlykOLGR		0	0	NATIONAI	Nepal / 3 I	11163	ACTIVE	ACTIVE	DAj3sgOn	GeneXpert		KATHMAN	Female	MTB NOT DETECTED		3383888	45 to 54
S4sUstNzv	YECy5jC9S	00:00.0	kdhb.lab	DISTRICT H	DISTRICT H	08:03.2	00:00.0	00:00.0	00:00.0	PTouL7It6	gPcNBTCTBk3		0	0	KALAIYA D	Nepal / 2 I	11020	ACTIVE	COMPLETI	OZGBcyvt	GeneXper	24	BARA	Male	MTB NOT	20708 Kala	2972520	35 to 44
YLdDLEnB	YECy5jC9S	00:00.0	dhj.lab			06:32.6	00:00.0	00:00.0	00:00.0	DAJC9ELG	ylWMtUscXy3		0	0	DHULABAI	Nepal / 1 I	12204	ACTIVE	ACTIVE	iMclBCbfN	GeneXper	1	JHAPA	Female	MTB NOT DETECTED		2257075	25 to 34
EZXxe96l2	YECy5jC9S	00:00.0	lmcp.lab			29:29.3	28:26.3	00:00.0	00:00.0	KMNyhQK	lZBLI1qmM5i		0	0	LUMBINI N	Nepal / 5 I	14076	COMPLETI	ACTIVE	R0g8PTLzc	GeneXper	7	PALPA	Female	MTB NOT	50610 Nisc	6525428	65+
pTiUaDrSr	YECy5jC9S	00:00.0	kahsj.lab			19:26.1	17:59.7	00:00.0	00:00.0	MxauyWa	BjB1Y2l3CCG		0	0	KARNALI /	Nepal / 6 I	12276	ACTIVE	ACTIVE	FitT2Cdnn	GeneXper	2	JUMLA	Male	Error	60407 Tila	5939174	45 to 54

B Flume Configuration

Configuration file of the flume located at $FLUME_HOME/conf/project.conf needs to be created (if needed) and configure as per follows:

```
# Define the agent name
agent2.sources = source1
agent2.channels = channel1
agent2.sinks = sink1

# Define the source
agent2.sources.source1.type = spooldir
agent2.sources.source1.spoolDir = /home/hadoop/Documents/project_mtb
agent2.sources.source1.fileHeader = true
agent2.sources.source1.fileHeaderKey = file
agent2.sources.source1.batchSize= 1000
agent2.sources.sources.batchDelay = 1000

# Define the channel
agent2.channels.channel1.type = memory
agent2.channels.channel1.capacity = 10000
agent2.channels.channel1.transactionCapacity = 1000

# Define the sink
agent2.sinks.sink1.type = hdfs
agent2.sinks.sink1.hdfs.path = hdfs://localhost:9000/user/hadoop/mtb_data
agent2.sinks.sink1.hdfs.filePrefix = events
agent2.sinks.sink1.hdfs.rollInterval =60
agent2.sinks.sink1.hdfs.rollSize = 0
agent2.sinks.sink1.hdfs.rollCount = 0
agent2.sinks.sink1.hdfs.fileType = DataStream
agent2.sinks.sink1.hdfs.writeFormat = Text
agent2.sinks.sink1.hdfs.batchSize = 100

# Bind the source and sink to the channel
agent2.sources.source1.channels = channel1
agent2.sinks.sink1.channel = channel1
```

To run the flume agent, put the following command:

```
./bin/flume-ng agent --name agent2 --conf ./conf
--conf-file ./conf/project.conf -Dflume.root.logger=INFO,console
```

C Spark Processing

Spark code for pre-processing and cleaning of data can be found on below link:
https://drive.google.com/file/d/15UIYmWRyOsdKTKjkR6H5GAEzm98IyHgs/view