CPSC 5330 01 - Big Data Analytics
Lab 03
Sqoop and Hive

Fariha Shah

April 26, 2025

1 How much time you spent on the assignment

I spent approximately 15-20 hours completing this lab. I have spend major hours on dealing with working on the Hive script and integration with MapReduce output.

2 If parts of the assignments are not fully working, which parts and what the problem(s) are

- First MapReduce Job (term-count-doc-and-term): Successfully processes the input text corpus and outputs tuples in the format (document, term, count), representing the frequency of each term in each document.
- Second MapReduce Job (term-count-doc): Accurately aggregates term counts for each document to generate (document, total_term_count) tuples, providing the basis for term frequency calculations.
- Hive Script:Correctly performs the join between the term-level and document-level datasets and calculates the term frequency as a percentage for each term within a document.
- Main Shell Script: Efficiently coordinates the entire pipeline—from data upload to HDFS, executing MapReduce jobs, running Hive queries, and retrieving the final output—resulting in the expected output format.

3 Were there aspects of the assignment that were particularly challenging? Particularly confusing?

- Designing the multi-stage data pipeline required careful planning to ensure smooth data flow between stages. Understanding how the output of one MapReduce job served as the input for the next took time to conceptualize and implement correctly.
- Handling custom keys in Hadoop Streaming was difficult. Properly using the -D stream.num.map.output.key.fields=2 flag to group by both document ID and term involved additional research and multiple rounds of testing.

- Integrating Hive with MapReduce output was also tricky. It took several attempts to correctly define external tables and configure paths so that Hive could read the output format produced by MapReduce.
- Managing file paths across HDFS and the local system proved to be more complicated than expected. Ensuring the main shell script handled all file transfers and directory setups correctly required extensive troubleshooting and attention to detail.

4 What were the main learning take-aways from this lab – that is, did it introduce particular concepts or techniques that might help you as an analyst or engineer in the future?

- MapReduce Pipeline Design: Designing a multi-stage data processing pipeline taught me how to decompose complex analytical workflows into manageable and modular stages with clear dependencies.
- Hadoop Streaming with Python: Implementing mappers and reducers in Python and integrating them using Hadoop Streaming provided hands-on experience with distributed computing, without requiring deep familiarity with Java.
- Hive for Distributed Analytics: Leveraging Hive to join MapReduce outputs and perform aggregations helped me understand the power of SQL-like tools in simplifying large-scale data transformations.
- File Path and Environment Management: Coordinating data between the local filesystem and HDFS enhanced my awareness of data flow in distributed environments and taught me how to manage paths, permissions, and transfer operations effectively.
- Pipeline Orchestration with Shell Scripting: Building a master script that orchestrates multiple stages of data processing reinforced my skills in automation and fault-tolerant pipeline design.
- Term Frequency Calculation: Implementing the term frequency metric across documents deepened my understanding of a core concept in information retrieval and text mining, commonly used in applications like search engines.