# Practical Work 5: The Longest Path

Group 16

December 16, 2025

## 1 Overview

The goal of this practical work is to utilize the MapReduce programming model to identify the longest file path within a large dataset. This simulates a distributed operation where finding a global maximum requires aggregating local maximums from distributed nodes.

## 2 Implementation Strategy

We utilized Python's **multiprocessing** library to create a realistic simulation of a distributed environment.

### 2.1 Why this approach?

- **Efficiency:** Instead of sending every single path to the Reducer (which would cause network congestion), each Mapper performs a "Local Reduce." It calculates the longest path within its assigned chunk and transmits only that single candidate to the Reducer.

- **Parallelism:** The dataset is split into chunks, and multiple worker processes scan these chunks simultaneously.

## 3 System Architecture

### 3.1 Data Flow Diagram

The figure below demonstrates how the system filters data. Unlike Word Count which expands data, Longest Path reduces data volume at every step.

### 3.2 Mapper Logic

The mapper iterates through its assigned list of strings and keeps only the longest one.

```
def map_worker ( path_chunk ) :
    if not path_chunk :
        return None
    # Efficiently find the longest string in the list
    local_longest = max ( path_chunk , key=len )
    return ( "MAX" , local_longest )
```

Listing 1: Mapper Function

### 3.3 Reducer Logic

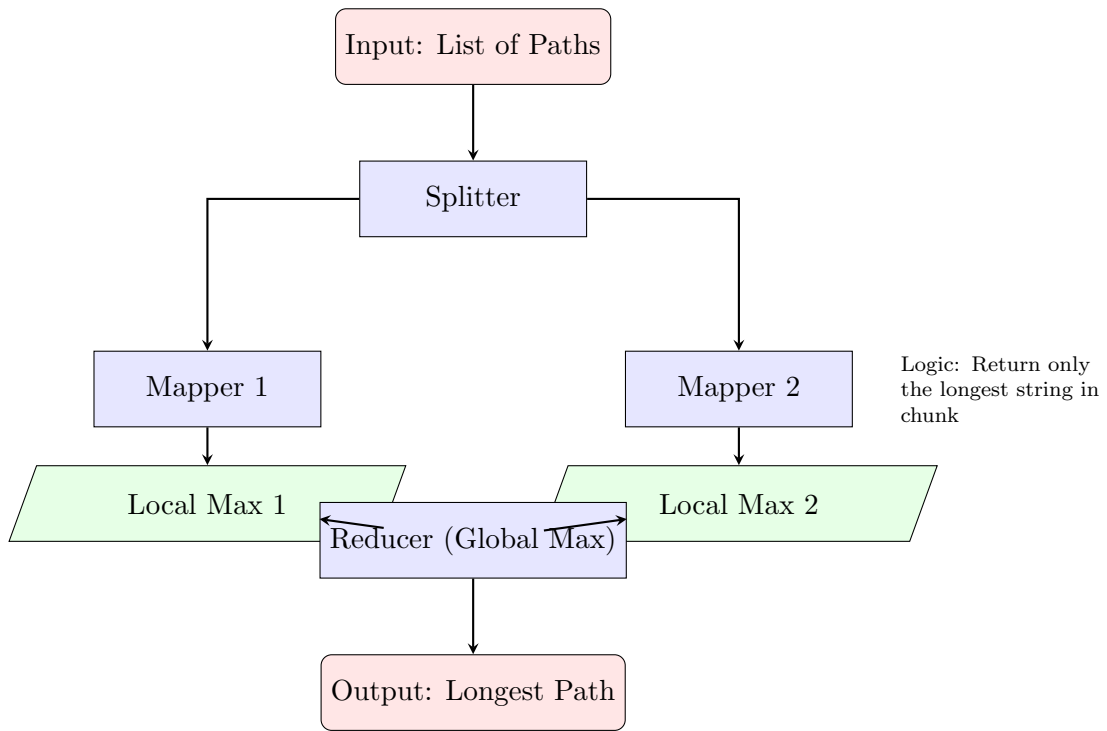The reducer receives a list of "Local Maximums" from all mappers and selects the winner.

Figure 1: MapReduce Data Flow for Longest Path

```python
def reduce_worker(item):
    key, paths = item
    # Compare local maximums to find global maximum
    global_longest = max(paths, key=len)
    return (key, global_longest)
```

Listing 2: Reducer Function

# 4    Roles and Responsibilities

| Member | Task | Details |
|--------|------|---------|
| Member 1 | Driver Code | Implemented file reading, data splitting, and process pool management. |
| Member 2 | Map/Reduce Logic | Implemented the max-length logic and optimized the local-reduce strategy. |
| Member 3 | Documentation | Created the architecture diagram and compiled the LaTeX report. |

Table 1: Group Task Allocation