Longest Path: Mapper and Reducer

Nguyen Thanh Nam

1 Overview

The objective of this project is to determine the file path(s) with the maximum depth in terms of folder count and slash (/) count. The process follows a **MapReduce** paradigm where the input consists of multiple text files, each containing paths from different systems.

2 Mapper Function

The **Mapper** processes each input file line by line to count:

- The number of slashes (/) in each path.
- The number of components (folders and files) in each path.

Example Input

A sample input file `laptop1.txt`:

```
/Users/Desktop/new folder/hello_world.txt
/Users/Document/probability
/Users/Music/Touhou
```

Mapper Logic

For each line, the Mapper performs:

- Count the number of slashes (/).
- Split the path by / and count the components.

Visualization of Mapper

```
Input: /Users/Desktop/new folder/hello_world.txt
Mapper Output: (slashes = 4, components = 4)

Input: /Users/Document/probability
Mapper Output: (slashes = 3, components = 3)

Input: /Users/Music/Touhou
Mapper Output: (slashes = 3, components = 3)
```

3 Reducer Function

The **Reducer** aggregates the outputs from all Mappers to identify:

- The path(s) with the maximum number of slashes (`/`).
- The path(s) with the maximum number of components.

Reducer Logic

The Reducer compares each path's slash count and component count:

- If a path has more slashes than the current maximum, it becomes the new longest path.
- If a path has equal slashes but more components, it updates the longest path.

Visualization of Reducer

```
Mapper Outputs:
    Path1: slashes = 4, components = 4
    Path2: slashes = 3, components = 3
    Path3: slashes = 3, components = 3

Reducer Final Output:
    Longest Path: /Users/Desktop/new folder/hello_world.txt
    Slash Count: 4
    Component Count: 4
```

4 Who Does What?

1. **Mapper:**
    - Reads the input line by line.
    - Counts the number of slashes (`/`).
    - Counts the number of components (folders and files).
    - Emits intermediate outputs with slash and component counts.
2. **Reducer:**
    - Receives intermediate outputs from the Mapper.
    - Finds the path(s) with the maximum slash count.
    - Updates paths if a larger component count is found.

5 Conclusion

The program efficiently uses the MapReduce approach to process multiple files containing paths. By breaking the problem into Mapper and Reducer tasks, the program ensures scalability and accurate identification of the longest paths.