COMP9313 Assignment-2

Amritesh Singh
z5211987

Documentation

The Scala program has the following process:

- Initially, the input text file is read and the blank lines are filtered out using the filter function. Subsequently, the **URL** and the **payload size** for each HTTP request is obtained by splitting each line at commas.
- Currently, the payload size has different units (such as **B**, **KB**, **MB**). The payload size for each request is converted to bytes (**B**) by separating the numerical part in the payload size string and multiplying it by **1024** if the original payload string contains **"K"**, multiplying by **1024 * 1024** if the original payload string contains **"M"**, or multiplying by **1** if the original payload string contains neither **"K"** nor **"M"** (and hence is already in the required bytes form). The obtained payload size is stored as a **Long** object.
- 4 computations are required in the assignment:
    - **Minimum payload:** This is computed by reducing by key on the tuple **(x, y)**, where **x** and **y** are values of different key-value pairs having the same key, to the one with the smaller value from **x** and **y**.
    - **Maximum payload:** This is computed by reducing by key on the tuple **(x, y)**, where **x** and **y** are values of different key-value pairs having the same key, to the one with the larger value from **x** and **y**.
    - **Mean of payload:** This is computed by reducing by key on the tuple **((x1, x2), (y1, y2))**, where **x1** and **x2** correspond to the payload size for that request, and **y1** and **y2** correspond to **1**, to the tuple **((x1 + y1), (x2 + y2))**, i.e., it results in the **<sum, count>** pair. Subsequently, the mean is obtained by computing **(sum / count)** for each key.
    - **Variance of payload:** This is computed by reducing by key on the tuple **((x1, x2), (y1, y2))**, where **x1** and **x2** correspond to the square of the difference between the payload size and the mean payload for that request, i.e., **(current payload size - mean payload) * (current payload size - mean payload),** and **y1** and **y2** correspond to **1**, to the tuple **((x1 + y1), (x2 + y2))**, i.e., it results in the **<square_difference, count>** pair. Subsequently, the variance is obtained by computing **(square_difference / count)** for each key.
- The 4 computations are merged together into a single RDD using the join function and converted to the desired output format by mapping the RDD to an RDD containing the URL string and all the merged computations (currently stored as a single tuple) in the form of a single string, separated by commas and with the unit **(B)** added to the end of each computation.
- Finally, the resultant RDD is written to the target output file.