Assignment No. 03

Performance	Understanding	Regularity	Total	Dated Sign of Subject Teacher
03	01	01	05	

Date of Performance: ..... **Date of Completion**: .....

Title Integrate the Python and Hadoop and perform the following operation on forest fire datasets.

Objectives:

To understand and apply the Analytical concept of Big data using Python.

Problem Statement:

Integrate the Python and Hadoop and perform the following operation on forest fire datasets

a. Data analysis using MapReduce in PyHadoop
b. Data mining in Hive

Outcomes:

Students will be able to,

1. Apply the Analytical concept of Big data using Python.

Software and Hardware requirements:

1. Software: Ubuntu OS, Anaconda, Jupyter Notebook, Hadoop 2.0
2. Hardware: Processor, Ethernet Connection or WiFi, RAM 1GB, HDD, Sound Card, camera, microphone (depending upon website selection)

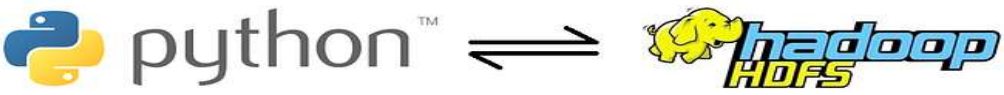
Theory:

Integration of Python with Hadoop:

Hadoop:

Hadoop is the best solution for storing and processing Big Data because Hadoop stores huge files in the form of (HDFS) Hadoop distributed file system without specifying any schema. It is highly scalable as any number of nodes can be added to enhance performance. In Hadoop data is highly available if there is any hardware failure also takes place.

Toady Data Scientist's first choice of language is Python and Hadoop provide Python APIs that provides processing of the Big Data and also allows easy access to Big data platforms.

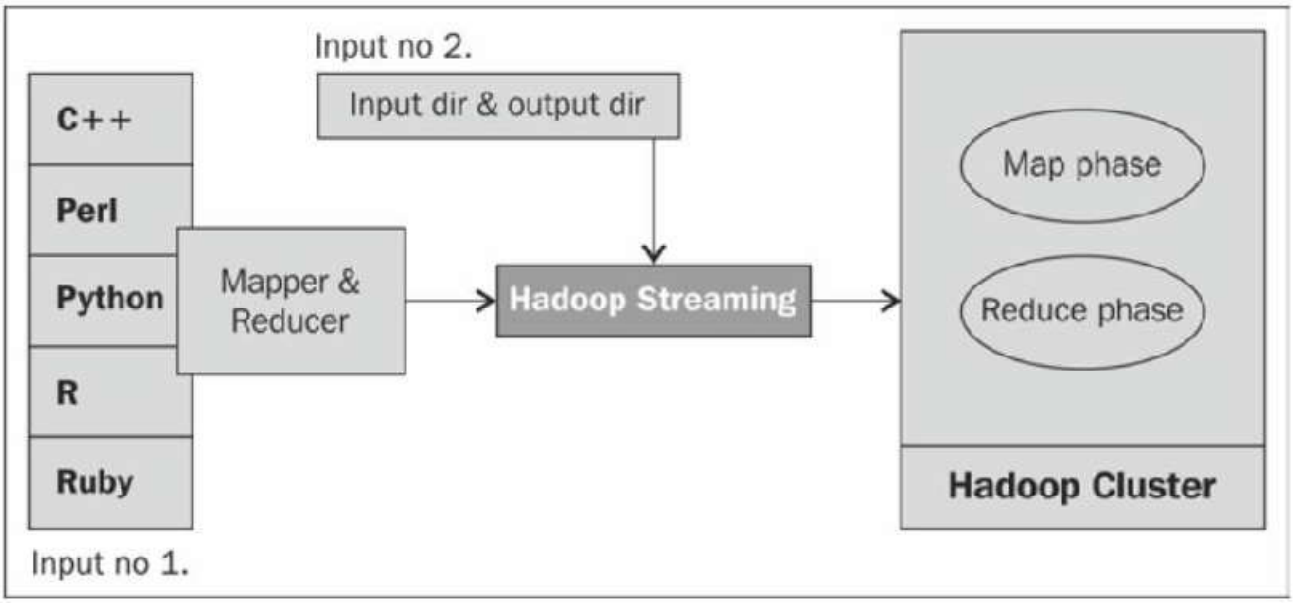


Hadoop Streaming:

Hadoop Streaming uses UNIX standard streams as the interface between Hadoop and your program so you can write MapReduce program in any language which can write to standard output and read standard input. Hadoop offers a lot of methods to help non-Java development.

The primary mechanisms are Hadoop Pipes which gives a native C++ interface to Hadoop and Hadoop Streaming which permits any program that uses standard input and output to be used for map tasks and reduce tasks.

With this utility, one can create and run Map Reduce jobs with any executable or script as the mapper and/or the reducer.

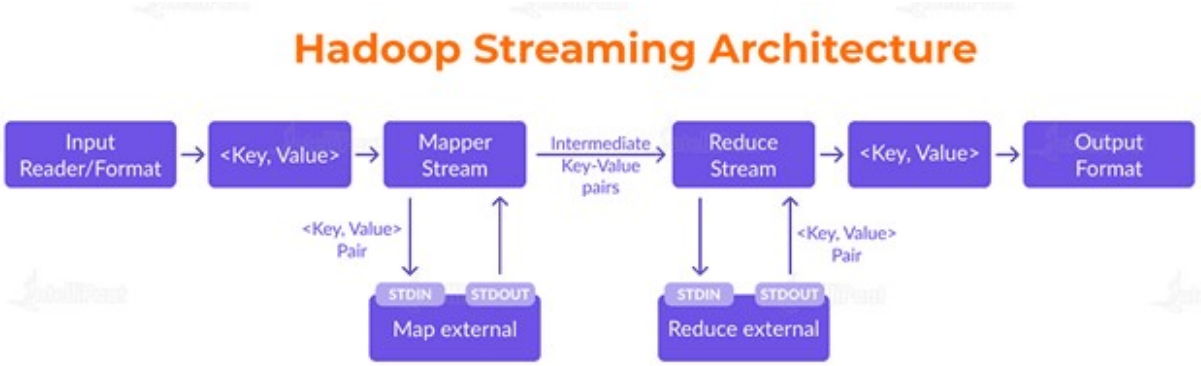


Features of Hadoop Streaming

Some of the key features associated with Hadoop Streaming are as follows:

- Hadoop Streaming is a part of the Hadoop Distribution System.
- It facilitates ease of writing Map Reduce programs and codes.
- Hadoop Streaming supports almost all types of programming languages such as Python, C++, Ruby, Perl etc.
- The entire Hadoop Streaming framework runs on Java. However, the codes might be written in different languages as mentioned in the above point.
- The Hadoop Streaming process uses Unix Streams that act as an interface between Hadoop and Map Reduce programs.
- Hadoop Streaming uses various Streaming Command Options and the two mandatory ones are – -input directoryname or filename and -output directoryname

Hadoop Streaming architecture



As it can be clearly seen in the diagram above that there are almost 8 key parts in a Hadoop Streaming Architecture. They are :

- Input Reader/Format
- Key Value
- Mapper Stream
- Key-Value Pairs
- Reduce Stream
- Output Format
- Map External
- Reduce External

The involvement of these components will be discussed in detail when we explain the working of the Hadoop streaming. However, to precisely summarize the Hadoop Streaming Architecture, the starting point of the entire process is when the Mapper reads the input value from the Input Reader Format. Once the input data is read, it is mapped by the Mapper as per the logic given in the code. It then passes through the Reducer stream and the data is transferred to the output after data aggregation is done. A more detailed description is given in the below section on the working of the Hadoop Streaming.

How does Hadoop Streaming Work?

- Input is read from standard input and the output is emitted to standard output by Mapper and the Reducer. The utility creates a Map/Reduce job, submits the job to an appropriate cluster, and monitors the progress of the job until completion.
- Every mapper task will launch the script as a separate process when the mapper is initialized after a script is specified for mappers. Mapper task inputs are converted into lines and fed to the standard input and Line oriented outputs are collected from the standard output of the procedure Mapper and every line is changed into a key, value pair which is collected as the outcome of the mapper.
- Each reducer task will launch the script as a separate process and then the reducer is initialized after a script is specified for reducers. As the reducer task runs, reducer task input

key/value pairs are converted into lines and fed to the standard input (STDIN) of the process.

- Each line of the line-oriented outputs is converted into a key/value pair after it is collected from the standard output (STDOUT) of the process, which is then collected as the output of the reducer.

Hadoop Streaming using Python

Hadoop Streaming supports any programming language that can read from standard input and write to standard output. For Hadoop streaming, one must consider the word-count problem. Codes are written for the mapper and the reducer in python script to be run under Hadoop.

Conclusion:

We studied how Python can become a good and efficient tool for Big Data Processing also. We can integrate all the Big Data tools with Python which makes data processing easier and faster. Python has become a suitable choice not only for Data Science but also for Big Data processing.

FAQ:

1. How does Hadoop Streaming Work?
2. What are the different key parts of Hadoop streaming architecture?
3. Explain Hadoop streaming using python.
4. What are the different features of Hadoop streaming?