BIG DATA ANALYTICS

Kafka and Spark Streaming

Homework 9

1. What is Apache Spark Streaming?

Apache Spark Streaming is a scalable fault-tolerant streaming processing system that natively supports both batch and streaming workloads.

Spark Streaming is an extension of the core Spark API that allows data engineers and data scientists to process real-time data from various sources including (but not limited to) Kafka, Flume, and Amazon Kinesis.

2. Describe how Spark Streaming processes data?

The Receiver is implemented using the Kafka high-level consumer API. As with all receivers, the data received from Kafka through a Receiver is stored in Spark executors, and then jobs launched by Spark Streaming processes the data.

3. What are DStreams?

Discretized Stream or DStream is the basic abstraction provided by Spark Streaming. It represents a continuous stream of data, either the input data stream received from source, or the processed data stream generated by transforming the input stream.

4. What is a StreamingContext object?

Public class StreamingContext extends Object implements Logging. Main entry point for Spark Streaming functionality. It provides methods used to create DStreams from various input sources.

It can be either created by providing a Spark master URL and an appName, or from a org. apache.

5. What are some of the common transformations on DStreams supported by Spark Streaming?

Different transformations in DStream in Apache Spark Streaming are:

- map - transforms RDD of one type to another. ...
- flatten map - similar to map but the output is flattenend, ie. ...
- filtering - returns an RDD composed with elements satisfying given predicate. ...
- glom - the result of it are RDDs composed by coalescing of all elements within each partition into an array.

6. What are the output operations that can be performed on DStreams?

Output Operation	**Meaning**
print()	Prints first ten elements of every batch of data in a DStream on the driver node running the streaming application. This is useful for development and debugging. Python API This is called **pprint()** in the Python API.
saveAsTextFiles(*prefix*, [*suffix*])	Save this DStream's contents as a text files. The file name at each batch interval is generated based on *prefix* and *suffix*: *"prefix-TIME_IN_MS[.suffix]"*.
saveAsObjectFiles(*prefix*, [*suffix*])	Save this DStream's contents as a SequenceFile of serialized Java objects. The file name at each batch interval is generated based on *prefix* and *suffix*: *"prefix-TIME_IN_MS[.suffix]"*. Python API This is not available in the Python API.
saveAsHadoopFiles(*prefix*, [*suffix*])	Save this DStream's contents as a Hadoop file. The file name at each batch interval is generated based on *prefix* and *suffix*: *"prefix-TIME_IN_MS[.suffix]"*.

	Python API This is not available in the Python API.
foreachRDD(*func*)	The most generic output operator that applies a function, *func*, to each RDD generated from the stream. This function should push the data in each RDD to a external system, like saving the RDD to files, or writing it over the network to a database. Note that the function *func* is executed in the driver process running the streaming application, and will usually have RDD actions in it that will force the computation of the streaming RDDs.