CS4225/CS5425 BIG DATA SYSTEMS FOR DATA SCIENCE

Tutorial 2: NoSQL and Spark

1. MapReduce/Hadoop

a) MapReduce and the Google File System (GFS) were designed to work well together. What important optimization in MapReduce is enabled by having GFS expose block replica locations via an API?

Answer: The MapReduce scheduler can arrange for Map tasks to execute on the same node that stores the data, avoiding a copy across the network.

b) List two features that are originally designed for relational databases and are now integrated into the MapReduce/Hadoop software stack.

Answer:

- High-level languages
- Column stores

2. Spark

a) What are the advantages of using schema in Spark?

Answer:

- Consistency. Applying schema omits the need to infer the data types from the table or DataFrame which may not be correct.
- Schemas define a contract for data, providing the same logical views of tables on different systems.

b) List three of the many common development features or considerations between relational databases and Spark.

Answer:

- schema
- query optimization
- high level languages

c) In HDFS, each chunk is replicated for three times by default. In contrast, in Spark, RDD uses lineage for reliability. What is a major problem if Spark also uses replications for reliability?

Answer: Consumes a lot of memory

d) Is it true that in the Spark runtime, RDD cannot reside in the hard disk?

Answer: False. RDD can also be in the disk if out of memory.

3. NoSQL

NoSQL databases have been a hot research topic.

The following questions relate to the trade-offs between relational and NoSQL systems. A more detailed discussion can be found in this paper (not required reading for the class, but still a useful summary if you are interested):

Rick Cattell. 2011. Scalable SQL and NoSQL data stores. SIGMOD Rec. 39, 4 (May 2011), 12-27.

a) Compare ACID and BASE. Why do NoSQL systems choose BASE?

Answer:

BASE = Basically Available, Soft state, Eventually consistent
ACID = Atomicity, Consistency, Isolation, and Durability

The idea is that by giving up ACID constraints, one can improve performance.

b) Why do we need specialized engines (e.g. document stores) in NoSQL systems, as compared to relational databases?

Answer: Performance and flexibility.

c) What is a practical reason to prefer horizonal scalability over vertical scalability?

Answer: Less expensive, using commodity servers.

d) In the paper, they have shared suitable applications for key-value stores and document stores:

Application of key-value store:

As an example, suppose you have a web application that does many RDBMS queries to create a tailored page when a user logs in. Suppose it takes several seconds to execute those queries, and the user's data is rarely changed, or you know when it changes because updates go through the same interface. Then you might want to store the user's tailored page as a single object in a key-value store, represented in a manner that's efficient to send in response to browser requests, and index these objects by user ID. If you store these objects persistently, then you may be able to avoid many RDBMS queries, reconstructing the objects only when a user's data is updated.

Application of document store:

A good example application for a document store would be one with multiple different kinds of objects (say, in a Department of Motor Vehicles application, with vehicles and drivers), where you need to look up objects based on multiple fields (say, a driver's name, license number, owned vehicle, or birth date).

Discuss some factors that make these applications suitable for key-value stores and document stores respectively.

Answer:

Key-value store:

- Improves scalability and efficiency – writing or reading user pages is faster.
- No need for complex queries or based on the content of user pages – just reads and writes.
- May be acceptable for user pages to be slightly stale – then eventual consistency is acceptable

Document store:

- Flexible schema may be beneficial (e.g. special types of vehicles may require different sets of fields)
- Unlike key-value stores, document stores allow for queries based on fields of a document