CS4225/CS5425 BIG DATA SYSTEMS FOR DATA SCIENCE

Tutorial 2: NoSQL and Spark

1. NoSQL

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:

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

Answer:

c) 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:

2. Spark

a) Why Spark is more suitable for iterative processing compared to Hadoop?

Answer:

b) In the below spark code block, please indicate which lines are transformation and which lines are action. For transformation, please also indicate whether it is a narrow transformation or wide transformation.

```
1  df1 = spark.range(2, 10000000, 2)
2  df2 = spark.range(2, 10000000, 4)
3  df3 = df1.join(df2, ["id"])
4  df3.count()
```

Answer:

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:

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

Answer:

e) Explain how the following program can be sped up.

```
1   lines = spark.textFile("hdfs://log")
2   errors = lines.filter(lambda s: s.startswith("INFO"))
3   info = errors.map(lambda s: s.split("\t")[2])
4   info.filter(lambda s: "hadoop" in s).count()
5   info.filter(lambda s: "spark" in s).count()
```

Answer: