

Apache Spark

Nidhin

Problem

- Data is growing faster than processing speeds
- Only solution is to parallelize on large clusters

What is Spark

- open source cluster computing framework
- 'successor' of Hadoop
- developed in AMPLab in UC Berkeley

History

- 2002: MapReduce @Google
- 2004: MapReduce [paper](#)
- 2006: Hadoop @ Yahoo
- 2009: Amazon EMR
- 2010 Spark [paper](#)
- 2014: Apache Spark (top level)

Hadoop vs Spark

- only way to reuse data, is to write to disk
 - Logistic Regression: 3 sec vs 76 sec
 - K-Means: 33 sec vs 106 sec
- interactively explore data

Spark Components

- Spark Core
- Spark SQL
- Spark Streaming
- Mlib
- GraphX

Resilient Distributed Dataset (RDD)

- fundamental unit of data in spark
- immutable resilient distributed collection
- can be in disk or memory
- knows how it was created

Analyzing Github Event Data

- what are the most common words used in the github commits (public repos)

Data

- file for every hour in 2015
 - 2015-01-01-00.json.gz
 - 2015-01-01-01.json.gz
 - ...
 - 2015-07-23-11.json.gz

Data

- file content: each line is a json
 - {..."type":"PushEvent"....}
 - {..."type":"CommitCommentEvent":..."body":"my super awesome commit"...}

Approach (each file)

- read the file
- filter lines that are commit events
- extract the actual commit message
- split the commit message to words
- get frequency of word
- combine results
- sort in descending order

Conclusion

- Spark is awesome ?

Resources

- Moocs

- [Introduction to Big Data using Apache Spark](#)
 - [Scalable Machine Learning](#)

- Papers:

- [Spark - Lightning-Fast Cluster Computing](#)
 - [Spark SQL: Relational Data Processing in Spark](#)
 - [Resilient Distributed Datasets: A Fault-Tolerant Abstraction for In-Memory Cluster Computing](#)