



IIT KHARAGPUR
IIT GANDHINAGAR



NPTEL ONLINE
CERTIFICATION COURSES

Scalable Data Science

Lecture 18: Spark

Sourangshu Bhattacharya
Computer Science and Engineering
IIT KHARAGPUR

In the previous lectures:

- Outline:
 - Scala
 - Spark
 - Motivation
 - RDD
 - Actions and transformations
 - Examples:



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Sourangshu Bhattacharya
Computer Science and Engg.

In this Lecture:

- Outline:
 - Spark: Programming examples:
 - Alternating least squares
 - User log mining
 - Partitioning
 - Accumulators
 - Scheduling of tasks



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Sourangshu Bhattacharya
Computer Science and Engg.

Example: Alternating Least squares



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Collaborative filtering

Predict movie ratings for a set of users based on their past ratings of other movies

$$R = \begin{pmatrix} 1 & ? & ? & 4 & 5 & ? & 3 \\ ? & ? & 3 & 5 & ? & ? & 3 \\ 5 & ? & 5 & ? & ? & ? & 1 \\ 4 & ? & ? & ? & ? & 2 & ? \end{pmatrix}$$

← Movies →

↑
Users
↓



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Matrix Factorization

Model R as product of user and movie matrices
A and B of dimensions $U \times K$ and $M \times K$

$$R = A B^T$$

Problem: given subset of R, optimize A and B



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

Alternating Least Squares

Start with random A and B

Repeat:

1. Fixing B, optimize A to minimize error on scores in R
2. Fixing A, optimize B to minimize error on scores in R



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Naïve Spark ALS

```
val R = readRatingsMatrix(...)

var A = (0 until U).map(i => Vector.random(K))
var B = (0 until M).map(i => Vector.random(K))

for (i <- 1 to ITERATIONS) {
    A = spark.parallelize(0 until U, numSlices)
        .map(i => updateUser(i, B, R))
        .toArray()
    B = spark.parallelize(0 until M, numSlices)
        .map(i => updateMovie(i, A, R))
        .toArray()
}
```



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

Efficient Spark ALS

```
val R = spark.broadcast(readRatingsMatrix(...))

var A = (0 until U).map(i => Vector.random(K))
var B = (0 until M).map(i => Vector.random(K))

for (i <- 1 to ITERATIONS) {
    A = spark.parallelize(0 until U, numSlices)
        .map(i => updateUser(i, B, R.value))
        .toArray()
    B = spark.parallelize(0 until M, numSlices)
        .map(i => updateMovie(i, A, R.value))
        .toArray()
}
```

Solution:
mark R as
“broadcast
variable”



IIT KHARAGPUR



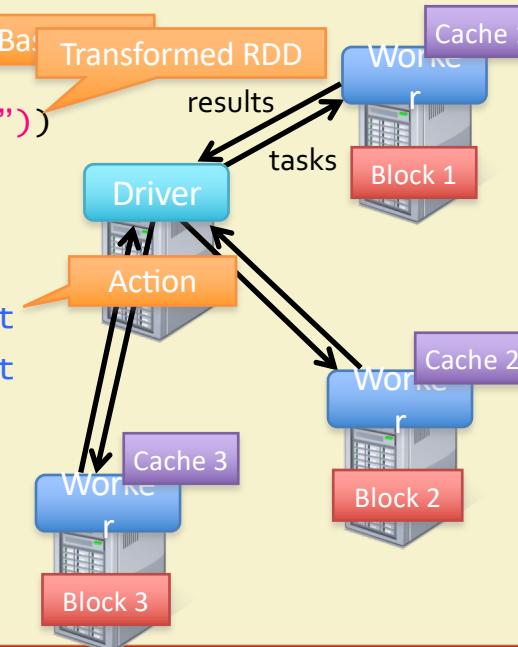
NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Example: Log Mining

Load error messages from a log into memory, then interactively search for various patterns

```
lines = spark.textFile("hdfs://...")  
errors = lines.filter(_.startsWith("ERROR"))  
messages = errors.map(_.split('\t')(2))  
cachedMsgs = messages.cache()  
  
cachedMsgs.filter(_.contains("foo")).count  
cachedMsgs.filter(_.contains("bar")).count  
....
```

Result: scaled to 1 TB data in 5-7 sec
(vs 170 sec for on-disk data)



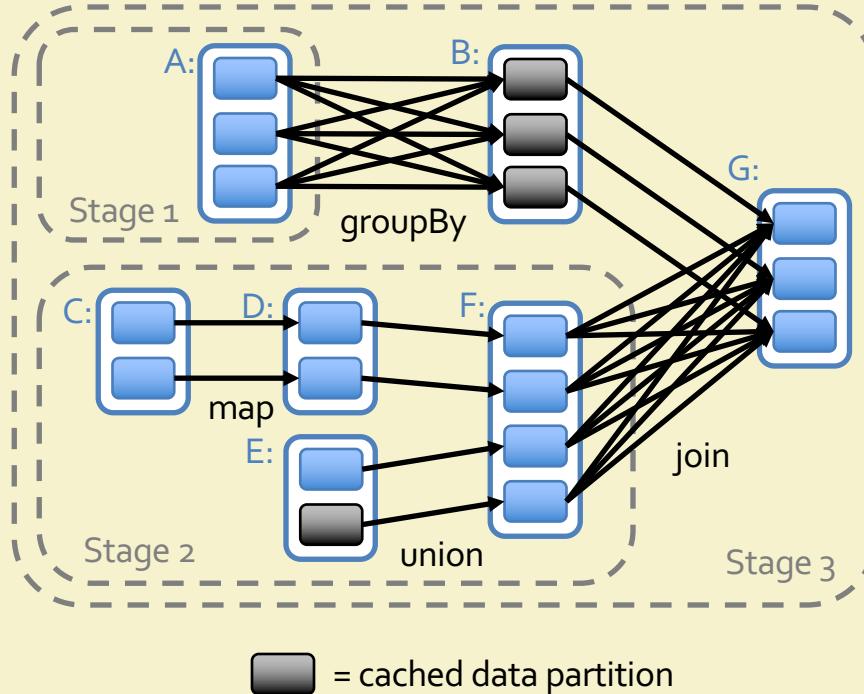
Spark Scheduler

Dryad-like DAGs

Pipelines functions
within a stage

Cache-aware work
reuse & locality

Partitioning-aware
to avoid shuffles



User Log Mining

```
val userData = sc.sequenceFile[UserID, UserInfo]("hdfs://...").persist()

def processNewLogs(logFileName: String) {

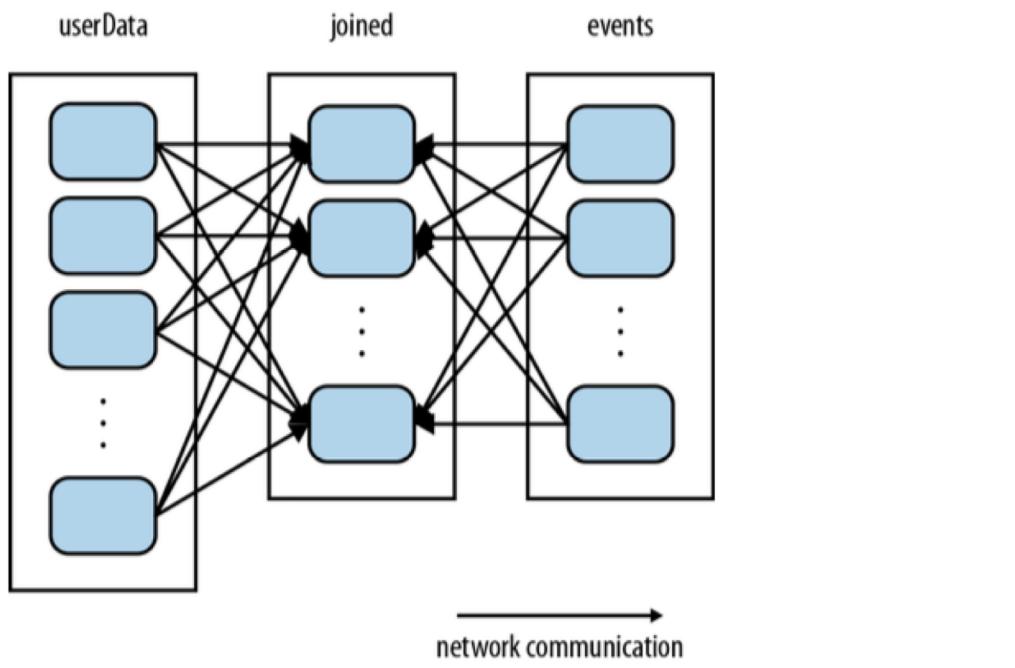
    val events = sc.sequenceFile[UserID, LinkInfo](logFileName)

    val joined = userData.join(events) // RDD of (UserID, (UserInfo, LinkInfo)) pairs

    val offTopicVisits = joined.filter {
        case (userId, (userInfo, linkInfo)) => // Expand the tuple into its components
                                                userInfo.topics.contains(linkInfo.topic)
    }.count()

    println("Number of visits to non-subscribed topics: " + offTopicVisits)
}
```

User Log Mining



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

User Log Mining

```
val userData = sc.sequenceFile[UserID, UserInfo]("hdfs://...")  
.partitionBy(new HashPartitioner(100)) // Create 100 partitions  
.persist()  
  
def processNewLogs(logFileName: String) {  
  
    val events = sc.sequenceFile[UserID, LinkInfo](logFileName)  
  
    val joined = userData.join(events) // RDD of (UserID, (UserInfo, LinkInfo)) pairs  
  
    val offTopicVisits = joined.filter {  
        case (userId, (userInfo, linkInfo)) =>  
            // Expand the tuple into its components  
            userInfo.topics.contains(linkInfo.topic)  
    }.count()  
    println("Number of visits to non-subscribed topics: " + offTopicVisits)  
}
```

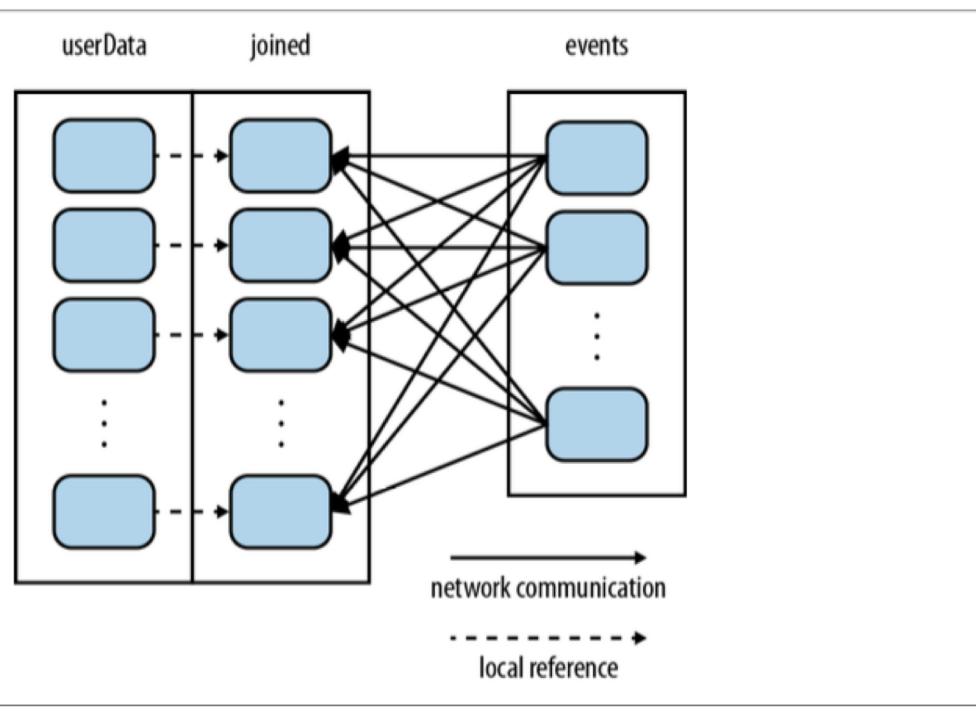


IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

User Log Mining



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Partitioning

- ❑ Operations **benefitting** from partitioning:

cogroup(), groupWith(), join(), leftOuterJoin(), rightOuter Join(), groupByKey(), reduceByKey(), combineByKey(), and lookup().

- ❑ Operations **affecting** partitioning:

cogroup(), groupWith(), join(), leftOuterJoin(), rightOuter Join(), groupByKey(), reduceByKey(), combineByKey(), partitionBy(), sort()

mapValues() (if the parent RDD has a partitioner),

flatMapValues() (if parent has a partitioner)

filter() (if parent has a partitioner).



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Page Rank (Revisited)

```
val links = sc.objectFile[(String, Seq[String])]("links") .  
partitionBy(new HashPartitioner(100)).persist()  
  
var ranks = links.mapValues(v => 1.0)  
  
for(i<-0 until 10) {  
    val contributions = links.join(ranks).flatMap {  
        case (pageId, (nbh, rank)) => nbh.map(dest => (dest, rank / nbh.size))  
    }  
    ranks = contributions.reduceByKey((x, y) => x + y).  
mapValues(v => 0.15 + 0.85*v)  
}  
ranks.saveAsTextFile("ranks")
```



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

Accumulators

```
val sc = new SparkContext(...) val file = sc.textFile("file.txt")

val blankLines = sc.accumulator(0)
// Create an Accumulator[Int] initialized to 0
val callSigns = file.flatMap(
  line => { if (line == "") {
    blankLines += 1           // Add to the accumulator
  }
  line.split(" ") })
callSigns.saveAsTextFile("output.txt")

println("Blank lines: " + blankLines.value)
```

Physical Execution Plan

- ❑ User code defines a DAG (directed acyclic graph) of RDDs
 - ❑ Operations on RDDs create new RDDs that refer back to their parents, thereby creating a graph.
- ❑ Actions force translation of the DAG to an execution plan
 - ❑ When you call an action on an RDD, its parents must be computed. That job will have one or more stages, with tasks for each partition. Each stage will correspond to one or more RDDs in the DAG. A single stage can correspond to multiple RDDs due to pipelining.
- ❑ Tasks are scheduled and executed on a cluster
 - ❑ Stages are processed in order, with individual tasks launching to compute segments of the RDD. Once the final stage is finished in a job, the action is complete.



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Tasks

- Each task internally performs the following steps:
 - ❑ Fetching its input, either from data storage (if the RDD is an input RDD), an existing RDD (if the stage is based on already cached data), or shuffle outputs.
 - ❑ Performing the operation necessary to compute RDD(s) that it represents. For instance, executing filter() or map() functions on the input data, or performing grouping or reduction.
 - ❑ Writing output to a shuffle, to external storage, or back to the driver (if it is the final RDD of an action such as count()).



IIT KHARAGPUR



NPTEL ONLINE
CERTIFICATION COURSES

Conclusion:

- We have seen:
 - Spark: Programming examples:
 - Alternating least squares
 - User log mining
 - Partitioning
 - Accumulators
 - Scheduling of tasks



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Sourangshu Bhattacharya
Computer Science and Engg.

References:

- Learning Spark: Lightning-Fast Big Data Analysis. Holden Karau, Andy Konwinski, Patrick Wendell, Matei Zaharia. O Reilly Press 2015.
- Any book on scala and spark.



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Sourangshu Bhattacharya
Computer Science and Engg.

Thank You!!



IIT KHARAGPUR



NPTEL
NPTEL ONLINE
CERTIFICATION COURSES

Sourangshu Bhattacharya
Computer Science and Engg.