CS. 6240-02 HW4 Vikas Janardhanan

Design Discussion (20 points total)

Describe the steps taken by Spark to execute your source code. In particular, for each method invocation of your Scala Spark program, give a brief high-level description of how Spark processes the data. (10 points)

Compare the Hadoop MapReduce and Spark implementations of PageRank. (10 points)

```
// Input: Takes the bz compressed file as input
// Output: PairRDD - (PageName:String,AdjList:List[String])
// Parses each line in the bz file and converts it to a pageName and the corresponding
// Adjacency list representation
1. val nodeGraph = sc.textFile(args(0)+"/*.bz2").
2.  map(line => WikiParser.parseLine(line)).//reads each line in bz file
3.  filter(pageInfo => pageInfo!=null).
4. map(pageInfo => pageInfo.split("~~")).//splits to seperate to pagename, outlink node list
5.  map(splitpageInfo => (splitpageInfo(0),// converts the comma seperated outlinks to list of
strings representing adjList
6. if(splitpageInfo.size>1)
7.  splitpageInfo(1).split(", ").toList // for pages that have outgoing links
8.  else
9.  List[String]())). // for pages that dont have outgoing links
10.  map(pageNode => List((pageNode._1,pageNode._2)) ++ pageNode._2.
11.    map(adjNode => (adjNode,List[String]()))). // add dangling nodes to the graph Nodes list
with  empty AdjList
12.  flatMap(pageNode => pageNode).
13.  reduceByKey((x,y) => (x++y)).// multiple entries for same pageName collapsed to one
14.  persist()
```

Analysis:

Line 1-3 : Reads all the lines in each bz2 file in input path and parses each line using the WikiParser and filters out results from parser to remove null.
Output: RDD: Strings
Line 4: Splits each of the pageNode details using "~~" to separate out the page name and outlinks from that node(adjacency nodes)
Output: RDD: Array[Strings]
Line 5-9: Now the split is converted to pageNode,adjList representation and condition is imposed to take care of sinks as well as nodes which have outgoing links
Output: Pair RDD: (String,List[String])

Line 10-11: Handles dangling nodes and emits entries for same with empty AdjList and combines them as a list
Output: Pair RDD: List[(String, List[String])]
Line 12: Flatens the RDD
Output: Pair RDD: (String,List[String])
Line 13: Reduces multiple entries for same pageName into one which represents (PageName,adjList)
Output: Pair RDD: (String,List[String])
Line 14: Save the RDD in memory enabling them to be reused across parallel operations.

MapReduce Version:
Above functionality is implemented in the MapReduce code in the Preprocessing job: Bz2ParserMapper,Bz2ParserReducer

```
// total node count in the webgraph
15.  val totPages = nodeGraph.count()

// Input: PairRDD - (PageName:String,AdjList:List[String])
// Output: PairRDD - (PageName:String,PageRank:Double)
// Creates default initial page rank for each node in webgraph
16. val pageRanks = nodeGraph.map(pageNode => (pageNode._1,1.0/totPages))

// Input: PairRDD - (PageName:String,AdjList:List[String])
// Output: PairRDD - (PageName:String,(AdjList:List[String],PageRank:Double))
// Performs equijoin to generate record corresponding to default pagerank and adjList for each pageName
17. var nodeGraphWithRank = nodeGraph.join(pageRanks)
```

Analysis:
Line 15: Counts the total number of pages in the graph
MapReduce version: This is aggregated and stored at the preprocessing job and later reused at the other jobs.

Line 16: Default pageRank for each pageNode calculated as 1.0/totalPages
MapReduce version: During the first iteration, the PageRankMapper sets this.

Line 17: Join operation to create records to relate each pageName with its initial pageRank value.
Output: PairRDD – (String,(List[String],Double))
MapReduce version:
This happens at the end of preprocessing(where pageNames with pageRank set as 0) and at the end of each pageRank computation job,(where pageNames with new PageRank set)

```
18. for(i <- 1 to 10){

   // Input: PairRDD - (PageName:String,(AdjList:List[String],PageRank:Double))
   // Output: Double
   // Does filter to get all dangling nodes and then accumulates their pageRank to calculate
   // delta for the current iteration
19.   val delta = nodeGraphWithRank.filter(pageNode => pageNode._2._1.length==0)
 20.    .reduce((totDelta,pageNode) =>
21.(totDelta._1,(totDelta._2._1,totDelta._2._2+pageNode._2._2)))._2._2

   // Input: PairRDD - (PageName:String,(AdjList:List[String],PageRank:Double))
   // Output: PairRDD - (PageName:String,PageRank:Double)
   // Total inlink pageRank contribution to a PageNode accumulated and produced as
(pageName,total inlink pageRank
   // contribution to pageName)
 22.   val pageRanks = nodeGraphWithRank.values.
 23.   map(adjListPageRank => adjListPageRank._1.
 24.     map(pageNode => (pageNode,adjListPageRank._2/adjListPageRank._1.size))).
 25.    flatMap(pageNode => pageNode).
 26.    reduceByKey((x,y) => x+y)

   // Input: PairRDD - (PageName:String, AdjList:List[String]),
   //       PairRDD - PairRDD - (PageName:String,PageRank:Double)
   // Output: PairRDD - (PageName:String, PageRank:Double)
   // Calculate the pageRank associated with each PageNode for this iteration by taking into
   // account the dangling node contribution and incoming link contribution to pageNode under
   // consideration
 27.   nodeGraphWithRank = nodeGraph.leftOuterJoin(pageRanks).
 28.    map(u => {
 29.     (u._1, (u._2._1, u._2._2 match {
 30.       case None => (alpha/totPages) + ((oneMinusAlpha) * delta/totPages) //PAge with no
inlink
31.       case Some (x:Double) => (alpha/totPages) + (oneMinusAlpha*((delta/totPages) + x)) //
Page with inlink contribution
```

```
32.      }))
33.    })

34.  }

 // Input: PairRDD -(PageName:String, (AdjList:List[String],PageRank:Double))
 // Output: Array - (PageRank:Double, PageName:String)
 // Converts into a structure as required and outputs the top 100 pageNode records based on
 // pageRank
 35. val result = nodeGraphWithRank.
 36.   map(pageNode => {(pageNode._2._2,pageNode._1)}).
 37.   top(100)

38.  sc.parallelize(result,1).saveAsTextFile(args(1))
39. }
```

Analysis:
Line 19: Calculates the delta for current iteration. Filters to get pageNodes which has empty size of adjList
Output: Pair RDD: (String,(List[String],Double)

Line 20-21: The pageRank values are accumulated to calculate the total delta contribution
Output: Pair RDD: (String,(List[String],Double)) -> Double
MapReduce version:
The contribution of delta from dangling nodes is accumulated at PageRankReducer and updated in the global counter. This dangling node contribution is used in the next iteration's PageRankMapper to calculate pageRank.

Line 22: Runs for each (AdjList:List[String], PageRank:Double)
Line 23-24: Each node in adjacency list sends the contribution that it receives.
Output: RDD: List[(String,Double)]

Line 25- Flattens the list structure
Output: Pair RDD – (String, Double)

Line 26: Multiple contributions to same node is aggregated by using the pageName key as reduce key.
Output: Pair RDD –(String, Double)

MapReduce version:
Line 22-25: Done in PageRankMapper map task
Line 26: Done in PageRankReducer reducer task
Line 27: Does leftouterjoin with the pageRank to get the information with respect to each pageName and its pageRank. To account for pages that don't have any incoming links we do a left outer join since it wont have entry in pageRanks
Output: pair RDD – (String, (List[String],Double))

Line 28-34: Calculates the new page rank value for each of the nodes. Takes into consideration two cases: None -> To account for nodes which have no inlinks and Some-> to account for pages which have inlink contribution towards it.
Output: Pair RDD –(String,(List[String],Double)

MapReduce version:
Delta contribution for dangling node correction of ith run is computed at the PageRankReducer reduce task and updated to global counter which is used at the i+1 th PageRankMapper map task where updated pageRank incorporating the delta contribution is computed.

New pageRank computation happens in the PageRankReduce reduce task using incoming contribution to the node. Every node and its incoming contribution is emitted at the PageRankMapper map task.

```
// Input: PairRDD –(PageName:String, (AdjList:List[String],PageRank:Double))
// Output: Array – (PageRank:Double, PageName:String)
// Converts into a structure as required and outputs the top 100 pageNode records
based on
// pageRank
35. val result = nodeGraphWithRank.
36.  map(pageNode => {(pageNode._2._2,pageNode._1)}).
37. top(100)

38. sc.parallelize(result,1).saveAsTextFile(args(1))
```

Line 35,36 – Modifies the order of records in a format suitable for topK calculation
Output: Pair RDD –(Double,String)
Line 37: Calculates the local top 100 and then computes the global top 100 from that.
Line 38 – Writes the final output to disk.

MapReduce Version:

CS. 6240-02 HW4 Vikas Janardhanan

TopKMapper map task and TopKReducer Reduce task performs the same job of local top 100 and global top 100 at the reducer computation.

Below methods are provided by Spark to implement various functionalities:

Map : Map function iterates over each element from the RDD and applies some function on each of the element which creates the contents of the new RDD

FlatMap: Applies a function that returns a sequence for each element in the list and hence flattens the result into the original result.

Reduce: It accepts a function with two arguments which returns the result as a single element

Reducebykey: Its same as reduce but happens on each distinct key. So the number of elements produced will be equal to the number of distinct keys

Filter: Takes a function which has one parameter which gets each element from the RDD. Filter returns Boolean after evaluating a function to indicate if the element is to be kept in the resulting RDD or not.

• Discuss the advantages and shortcomings of the different approaches. This could include, but is not limited to, expressiveness and flexibility of API, applicability to PageRank, available optimizations, memory and disk data footprint, and source code verbosity.

1. The whole task of generating pageRank which included running multiple mapreduce jobs in the Hadoop environment is simplified in spark into just around 38 lines of code which indicates the power of Spark as a programming language well suited to tasks like these
2. In MapReduce using Hadoop, the dangling node handling and its delta contribution of ith run, is incorporated into pageRank during the i+1th run. However, with Spark Dangling node contribution can be calculated before reduce job.
3. MapReduce starts a new JVM for each task which takes some time to initialize that includes loading JARs,JITing, parsing XML configuration etc.

Spark on the other hand keeps an executor JVM running on each node and hence provides better performance as it avoids unnessary load times.

4. Processing data: Since spark can cache partial or complete data in memory, it can avoid a lot of disk I/O. On the other hand, since MapReduce persists full dataset to HDFS after running each job. This is more expensive because it results in results in three times(due to replication) the size of dataset in disk I/O and similar increase in network I/O as a result of the same.
   Spark however, takes a holistic view of pipeline of operations. When the output of one operation needs to be fed into another operation, Spark passes the data directly without writing to persistent storage.
5. Spark has better optimization due to use of DAG based processing engine.It can optimize and perform computation in a single stage where in mapreduce, the same would have taken place in multiple stages.
6. Also, due to DAG based processing engine, Spark also avoids unwanted reducer tasks

Performance Comparison (12 points total) Run your program in Elastic MapReduce (EMR) on the four provided bz2 files, which comprise the full English Wikipedia data set from 2006, using the following two configurations: • 6 m4.large machines (1 master and 5 workers) • 11 m4.large machines (1 master and 10 workers) Report for both configurations the Spark execution time. For comparison, also include the total execution time (from pre-processing to top-k) of the corresponding Hadoop executions from Assignment 3. (4 points)

	EMR 6 Machine(ms)	EMR 11 Machine(ms)
Spark	2814381	1494000
MapReduce	2917979	1999488

Discuss which system is faster and briefly explain what could be the main reason for this performance difference. (4 points)

Spark is much faster. In terms of algorithm, the logic performed by both versions is the same except that the dangling node contribution is handled by spark during the current iteration, as compared to MapReduce in Hadoop where dangling node contribution from ith run is used during i+1th run.

But the main reason why spark is better is because of how the system works under the hood:

CS. 6240-02 HW4 Vikas Janardhanan

1. Processing data: Since spark can cache partial or complete data in memory, it can avoid a lot of disk I/O. On the other hand, since MapReduce persists full dataset to HDFS after running each job. This is more expensive because it results in results in three times(due to replication) the size of dataset in disk I/O and similar increase in network I/O as a result of the same. Spark however, takes a holistic view of pipeline of operations. When the output of one operation needs to be fed into another operation, Spark passes the data directly without writing to persistent storage.
2. Spark has better optimization due to use of DAG based processing engine.It can optimize and perform computation in a single stage where in mapreduce, the same would have taken place in multiple stages.
3. Also, due to DAG based processing engine, Spark also avoids unwanted reducer tasks
4. MapReduce starts a new JVM for each task which takes some time to initialize that includes loading JARs,JITing, parsing XML configuration etc. Spark on the other hand keeps an executor JVM running on each node and hence provides better performance as it avoids unnessary load times.

Report the top-100 Wikipedia pages with the highest PageRanks, along with their PageRank values, sorted from highest to lowest, for both the simple and full datasets, from both the Spark and MapReduce execution. Are the results the same? If not, try to find possible explanations. (4 points)

The PageRank results observed are same in both the versions. However, there is a slight difference in the pageRank value because of the below reason:
The Dangling node contribution to the pagerank in ith run is calculated in the reduce phase and updated to global counter and in the i+1th run, it's used to update the pageRank resulting from dangling nodes. So after the last run, the dangling node contribution for that run is updated to global counter but not added to pageRank calculation of the nodes, hence this contribution is lost in case of the Hadoop implementation.

However, in the case of the spark implementation, the dangling node contribution for the current run is calculated before the reduce phase and hence is incorporated into the pageRank computation in the current run. Hence there is no loss of pageRank after iiterations.

CS. 6240-02 HW4 Vikas Janardhanan

Top-100 Results for Spark:
Local:
(0.006268262079422903,United_States_09d4)
(0.004752925651663405,Wikimedia_Commons_7b57)
(0.0038820830791587396,Country)
(0.00267763949005994,England)
(0.00260685204215944,United_Kingdom_5ad7)
(0.0026025866760897346,Europe)
(0.0025800718211158383,Water)
(0.0025361667093965145,Germany)
(0.002511005318307044,France)
(0.0024540946934262686,Animal)
(0.0024209955637570176,Earth)
(0.002357188856916948,City)
(0.002007228437722828,Week)
(0.0019207074411200856,Asia)
(0.0018678519236924747,Sunday)
(0.0018585003126376514,Wiktionary)
(0.0018407354138432587,Monday)
(0.001835229786866236,Money)
(0.0018225276952226052,Wednesday)
(0.0018102092804128489,Plant)
(0.001778053236719982,Friday)
(0.0017602592546616617,Computer)
(0.001758340297136833,Saturday)
(0.001746304355422851,English_language)
(0.0017357191264349941,Thursday)
(0.0017232944612568138,Tuesday)
(0.00171377591640442,Italy)
(0.0017032005827194263,Government)
(0.0017016985990959987,India)
(0.0015872815219805333,Number)
(0.001558476480081002,Spain)
(0.0015143848821488875,Japan)
(0.0014977586737395846,Canada)
(0.0014701832794905244,Day)
(0.0014445754534188207,People)

(0.001417123178485724 2,Human)
(0.0013745140467073375,Wikimedia_Foundation_83d9)
(0.001365939939167297,Australia)
(0.001365505639614912 6,China)
(0.0013330361169103808,Energy)
(0.0013165263130523455,Food)
(0.0012932737014161185,Sun)
(0.0012906864348722,Science)
(0.0012758619110732738,Mathematics)
(0.0012472048845615182,index)
(0.0012255116379364071,Television)
(0.001188597631189134,Capital_(city))
(0.0011817627345243 64,Russia)
(0.0011634823096139234,State)
(0.0011571083709808068,Music)
(0.0011349190516769287,Year)
(0.0011118098670028008,Greece)
(0.0011084419898384937,Language)
(0.0011054136172328208,Scotland)
(0.0010817279846470234,Metal)
(0.0010724538303110648,Wikipedia)
(0.0010611310344791013,Greek_language)
(0.001056525628519365,2004)
(0.0010307339000191 39,Planet)
(0.0010254766058030392,Sound)
(0.0010222548404020997,Religion)
(0.0010202595604546 6,London)
(9.907897019959893E-4,Africa)
(9.555984391172881E-4,20th_century)
(9.490638415110773E-4,Law)
(9.435252509644823E-4,Geography)
(9.370763607187824E-4,Liquid)
(9.367445379889459E-4,19th_century)
(9.246934303337297E-4,World)
(9.228039937696954E-4,Poland)
(9.127625824779857E-4,Scientist)
(9.103721224399922E-4,Society)

(8.778491911297393E-4,Latin)
(8.777601463480544E-4,Atom)
(8.759419632775697E-4,History)
(8.689008813834748E-4,Sweden)
(8.681683457472488E-4,War)
(8.64680411263722E-4,Light)
(8.581913913912013E-4,Netherlands)
(8.494797122620919E-4,Culture)
(8.399317320858321E-4,Building)
(8.235468196323817E-4,God)
(8.216698391470171E-4,Turkey)
(8.163021946764355E-4,Plural)
(8.133630565582855E-4,Information)
(8.055473764137156E-4,Centuries)
(7.928525045745685E-4,Chemical_element)
(7.908141953201536E-4,Portugal)
(7.870761352325709E-4,Inhabitant)
(7.777398953081298E-4,Denmark)
(7.75360065686562E-4,Capital_city)
(7.708899281389988E-4,Austria)
(7.589071127376724E-4,Cyprus)
(7.570695631356184E-4,Species)
(7.559989237421153E-4,Ocean)
(7.551197162145131E-4,Book)
(7.534113103962682E-4,Disease)
(7.52869639796831E-4,North_America_e7c4)
(7.505074858768424E-4,University)
(7.481623158660931E-4,Biology)

6 Machines EMR:
(0.002882660103856217,United_States_09d4)
(0.002578497367000244,2006)
(0.0013708401711878062,United_Kingdom_5ad7)
(0.0011888528908954909,2005)
(9.451296001063609E-4,Biography)
(8.962201733987506E-4,Canada)
(8.904855222492458E-4,England)

(8.810956932863831E-4,France)
(8.280172576009193E-4,2004)
(7.570130035043359E-4,Germany)
(7.332469481166557E-4,Australia)
(7.18050055315927E-4,Geographic_coordinate_system)
(6.668430754552268E-4,2003)
(6.463289311811408E-4,India)
(6.406942242984752E-4,Japan)
(5.378130474020753E-4,Italy)
(5.353568495441892E-4,2001)
(5.290731868620813E-4,2002)
(5.240808518165254E-4,Internet_Movie_Database_7ea7)
(5.09863848072464E-4,Europe)
(5.010563963308715E-4,2000)
(4.83079420028856E-4,World_War_II_d045)
(4.662585387385535E-4,London)
(4.490846192006855E-4,Population_density)
(4.435162847752467E-4,Record_label)
(4.4287519496137985E-4,1999)
(4.397181918727361E-4,English_language)
(4.395443164973752E-4,Spain)
(4.1483531175826873E-4,Russia)
(4.1192436210405587E-4,Race_(United_States_Census)_a07d)
(4.05284906741803E-4,Wiktionary)
(3.8598312060265217E-4,Wikimedia_Commons_7b57)
(3.8282795989750607E-4,1998)
(3.7344901889252484E-4,Music_genre)
(3.6518623802572785E-4,1997)
(3.593922595129753E-4,Scotland)
(3.5902154792803606E-4,New_York_City_1428)
(3.502583175010795E-4,Football_(soccer))
(3.426084070012259E-4,1996)
(3.376927373883108E-4,Sweden)
(3.3704910641610246E-4,Television)
(3.2523713708594126E-4,Square_mile)
(3.2450484348489834E-4,Census)
(3.2268733928410536E-4,1995)

(3.199984130805184E-4,California)
(3.1591359850808497E-4,China)
(3.111460771214885E-4,Netherlands)
(3.1009409760585646E-4,New_Zealand_2311)
(3.0807719698716163E-4,1994)
(2.9394515721342966E-4,1991)
(2.9129353778678383E-4,1993)
(2.8955029703323613E-4,1990)
(2.8766964014626125E-4,New_York_3da4)
(2.874417567824211E-4,Public_domain)
(2.791972611907569E-4,1992)
(2.7709385232491845E-4,United_States_Census_Bureau_2c85)
(2.767422171279525E-4,Film)
(2.747044314743817E-4,Actor)
(2.742168317933441E-4,Scientific_classification)
(2.719345001694367E-4,Norway)
(2.705140804957328E-4,Ireland)
(2.687750958819775E-4,Population)
(2.6826248931454154E-4,Poland)
(2.6180650970769717E-4,1989)
(2.5575570935395246E-4,1980)
(2.5550249470367E-4,January_1)
(2.540247363139217E-4,Marriage)
(2.5353040576733685E-4,Brazil)
(2.520195040358151E-4,Mexico)
(2.517970055424863E-4,Latin)
(2.4996426121655856E-4,Politician)
(2.48792226802344E-4,1986)
(2.426889925671344E-4,1985)
(2.4241657542496025E-4,1979)
(2.417805214695013E-4,1982)
(2.415706053625445E-4,1981)
(2.415506038058859E-4,French_language)
(2.3963093603840955E-4,Per_capita_income)
(2.3933247137452853E-4,1974)
(2.3818243404850892E-4,Album)
(2.3734520588733076E-4,Switzerland)

(2.3709191033745997E-4,1984)
(2.368872460299237E-4,1987)
(2.3684417278114833E-4,South_Africa_1287)
(2.3680078377405706E-4,1983)
(2.3447757734346185E-4,Record_producer)
(2.3303693895980934E-4,1970)
(2.3146167218874057E-4,1988)
(2.3033684046043477E-4,1976)
(2.2783649166699537E-4,Km²)
(2.2767249845499004E-4,1975)
(2.2468131276483954E-4,Paris)
(2.2451996330240072E-4,1969)
(2.2416835181288586E-4,Greece)
(2.2334759300397211E-4,1945)
(2.2306589635477891E-4,1972)
(2.2185620732801642E-4,Personal_name)
(2.21246946745586E-4,1977)
(2.2046077066534817E-4,Soviet_Union_ad1f)
(2.201825163708337E-4,1978)

11 Machines EMR
(0.002882660103856217,United_States_09d4)
(0.0025784973670002435,2006)
(0.0013708401711878062,United_Kingdom_5ad7)
(0.001188852890895491,2005)
(9.451296001063613E-4,Biography)
(8.962201733987504E-4,Canada)
(8.904855222492456E-4,England)
(8.810956932863829E-4,France)
(8.280172576009193E-4,2004)
(7.570130035043359E-4,Germany)
(7.332469481166558E-4,Australia)
(7.18050055315927E-4,Geographic_coordinate_system)
(6.66843075455227E-4,2003)
(6.46328931181141E-4,India)
(6.406942242984751E-4,Japan)
(5.378130474020754E-4,Italy)

(5.35356849544189E-4,2001)
(5.290731868620814E-4,2002)
(5.240808518165252E-4,Internet_Movie_Database_7ea7)
(5.098638480724641E-4,Europe)
(5.010563963308713E-4,2000)
(4.830794200288561E-4,World_War_II_d045)
(4.6625853873855344E-4,London)
(4.490846192006859E-4,Population_density)
(4.4351628477524665E-4,Record_label)
(4.4287519496137996E-4,1999)
(4.39718191872736E-4,English_language)
(4.3954431649737496E-4,Spain)
(4.148353117582687E-4,Russia)
(4.119243621040558E-4,Race_(United_States_Census)_a07d)
(4.052849067418031E-4,Wiktionary)
(3.8598312060265217E-4,Wikimedia_Commons_7b57)
(3.8282795989750623E-4,1998)
(3.734490188925248E-4,Music_genre)
(3.6518623802572785E-4,1997)
(3.593922595129753E-4,Scotland)
(3.590215479280361E-4,New_York_City_1428)
(3.502583175010795E-4,Football_(soccer))
(3.4260840700122574E-4,1996)
(3.3769273738831063E-4,Sweden)
(3.3704910641610235E-4,Television)
(3.252371370859413E-4,Square_mile)
(3.245048434848986E-4,Census)
(3.226873392841053E-4,1995)
(3.199984130805183E-4,California)
(3.159135985080847E-4,China)
(3.1114607712148844E-4,Netherlands)
(3.100940976058565E-4,New_Zealand_2311)
(3.080771969871618E-4,1994)
(2.939451572134296E-4,1991)
(2.912935377867839E-4,1993)
(2.895502970332362E-4,1990)
(2.876964014626114E-4,New_York_3da4)

(2.87441756782421E-4,Public_domain)
(2.791972611907572E-4,1992)
(2.7709385232491845E-4,United_States_Census_Bureau_2c85)
(2.767422171279525E-4,Film)
(2.747044314743817E-4,Actor)
(2.742168317933442E-4,Scientific_classification)
(2.7193450016943667E-4,Norway)
(2.70514080495733E-4,Ireland)
(2.6877509588197756E-4,Population)
(2.6826248931454154E-4,Poland)
(2.6180650970769717E-4,1989)
(2.5575570935395246E-4,1980)
(2.5550249470366996E-4,January_1)
(2.540247363139216E-4,Marriage)
(2.535304057673366E-4,Brazil)
(2.5201950403581523E-4,Mexico)
(2.5179700554248634E-4,Latin)
(2.499642612165585E-4,Politician)
(2.487922268023441E-4,1986)
(2.426889925671343E-4,1985)
(2.424165754249602E-4,1979)
(2.417805214695013E-4,1982)
(2.415706053625445E-4,1981)
(2.4155060380588581E-4,French_language)
(2.396309360384096E-4,Per_capita_income)
(2.3933247137452853E-4,1974)
(2.3818243404850892E-4,Album)
(2.373452058873308E-4,Switzerland)
(2.370919103374599E-4,1984)
(2.3688724602992237E-4,1987)
(2.3684417278114835E-4,South_Africa_1287)
(2.3680078377405722E-4,1983)
(2.3447757734346188E-4,Record_producer)
(2.330369389598093E-4,1970)
(2.3146167218874062E-4,1988)
(2.3033684046043488E-4,1976)
(2.278364916669953 7E-4,Km²)

(2.2767249845499017E-4,1975)
(2.2468131276483954E-4,Paris)
(2.2451996330240064E-4,1969)
(2.241683518128858E-4,Greece)
(2.2334759300397217E-4,1945)
(2.2306589635477897E-4,1972)
(2.2185620732801647E-4,Personal_name)
(2.2124694674558606E-4,1977)
(2.204607706653483E-4,Soviet_Union_ad1f)
(2.2018251637083381E-4,1978)

Top-100 Results for Hadoop MapReduce:
Local:
United_States_09d4:0.006262544724786
Wikimedia_Commons_7b57:0.004747208297026
Country:0.003876365724522
England:0.002671922135423
United_Kingdom_5ad7:0.002601134687522
Europe:0.002596869321453
Water:0.002574354466479
Germany:0.002530449354759
France:0.002505287963670
Animal:0.002448377338789
Earth:0.002415278209120
City:0.002351471502280
Week:0.002001511083086
Asia:0.001914990086483
Sunday:0.001862134569055
Wiktionary:0.001852782958001
Monday:0.001835018059206
Money:0.001829512432229
Wednesday:0.001816810340586
Plant:0.001804491925776
Friday:0.001772335882083
Computer:0.001754541900025
Saturday:0.001752622942500

English_language:0.001740587000786
Thursday:0.001730001771798
Tuesday:0.001717577106620
Italy:0.001708058561767
Government:0.001697483228082
India:0.001695981244459
Number:0.001581564167344
Spain:0.001552759125444
Japan:0.001508667527512
Canada:0.001492041319103
Day:0.001464465924854
People:0.001438858098782
Human:0.001411405823849
Wikimedia_Foundation_83d9:0.001368796692070
Australia:0.001360222584530
China:0.001359788284978
Energy:0.001327318762273
Food:0.001310808958415
Sun:0.001287556346779
Science:0.001284969080235
Mathematics:0.001270144556436
index:0.001241487529925
Television:0.001219794283299
Capital_(city):0.001182880276552
Russia:0.001176045379887
State:0.001157764954977
Music:0.001151391016344
Year:0.001129201697040
Greece:0.001106092512366
Language:0.001102724635201
Scotland:0.001099696262596
Metal:0.001076010630010
Wikipedia:0.001066736475674
Greek_language:0.001055413679842
2004:0.001050808273882
Planet:0.001025016545382
Sound:0.001019759251166

Religion:0.001016537485765
London:0.001014542205818
Africa:0.000985072347359
20th_century:0.000949881084480
Law:0.000943346486874
Geography:0.000937807896327
Liquid:0.000931359006082
19th_century:0.000931027183352
World:0.000918976075697
Poland:0.000917086639133
Scientist:0.000907045227841
Society:0.000904654767803
Latin:0.000872131836493
Atom:0.000872042791711
History:0.000870224608641
Sweden:0.000863183526746
War:0.000862450991110
Light:0.000858963056627
Netherlands:0.000852474036754
Culture:0.000843762357625
Building:0.000834214377449
God:0.000817829464995
Turkey:0.000815952484510
Plural:0.000810584840039
Information:0.000807645701921
Centuries:0.000799830021777
Chemical_element:0.000787135149938
Portugal:0.000785096840683
Inhabitant:0.000781358780596
Denmark:0.000772022540671
Capital_city:0.000769642711050
Austria:0.000765172573502
Cyprus:0.000753189758101
Species:0.000751352208499
Ocean:0.000750281569105
Book:0.000749402361577
Disease:0.000747693955759

North_America_e7c4:0.000747152285160
University:0.000744790131240
Biology:0.000742444961229

6 Machines EMR
United_States_09d4:0.002905049211098
2006:0.002601052663040
United_Kingdom_5ad7:0.001381933693276
2005:0.001198785776077
Biography:0.000950231857881
Canada:0.000902948043433
England:0.000897252898679
France:0.000888489576305
2004:0.000834786716495
Germany:0.000763110031137
Australia:0.000738825188152
Geographic_coordinate_system:0.000722554706546
2003:0.000672261798953
India:0.000651145680121
Japan:0.000645777402587
Italy:0.000542419130970
2001:0.000539561851684
2002:0.000533306832033
Internet_Movie_Database_7ea7:0.000527718734330
Europe:0.000514002813042
2000:0.000505009259219
World_War_II_d045:0.000487160678123
London:0.000470074453236
Population_density:0.000452275208234
1999:0.000446418899664
Record_label:0.000446183915032
English_language:0.000443664387416
Spain:0.000443156662935
Russia:0.000418420387229
Race_(United_States_Census)_a07d:0.000415283406358
Wiktionary:0.000408740523612
Wikimedia_Commons_7b57:0.000389606462805

1998:0.000385864293375
Music_genre:0.000375615703614
1997:0.000368118107863
Scotland:0.000362208528710
New_York_City_1428:0.000362021171877
Football_(soccer):0.000352446564407
1996:0.000345343839368
Sweden:0.000340214714714
Television:0.000339458759153
Square_mile:0.000327662198420
Census:0.000326717713307
1995:0.000325277961537
California:0.000322410941768
China:0.000318512347458
Netherlands:0.000313758387111
New_Zealand_2311:0.000312482578150
1994:0.000310555561147
1991:0.000296362568883
1993:0.000293635132755
1990:0.000291916513006
New_York_3da4:0.000289886879812
Public_domain:0.000289559817245
1992:0.000281442926449
United_States_Census_Bureau_2c85:0.000279107103114
Film:0.000278708552637
Actor:0.000276463548893
Scientific_classification:0.000275890010209
Norway:0.000273797725988
Ireland:0.000272592027011
Population:0.000270709495661
Poland:0.000270300703246
1989:0.000263930437892
1980:0.000257874272306
January_1:0.000257872304183
Marriage:0.000255737339646
Brazil:0.000255431810114
Latin:0.000254142613717

Mexico:0.000254010007714
Politician:0.000251315058529
1986:0.000250838427877
1985:0.000244661664647
1979:0.000244439542430
French_language:0.000243774554375
1982:0.000243751608508
1981:0.000243553693248
1974:0.000241356433487
Per_capita_income:0.000241205949684
Album:0.000239460974004
Switzerland:0.000239276855584
1984:0.000239006414395
1987:0.000238787242833
South_Africa_1287:0.000238736833875
1983:0.000238701883728
Record_producer:0.000235823714677
1970:0.000235026354548
1988:0.000233317564855
1976:0.000232281228604
1975:0.000229592645040
Km²:0.000229295667798
Paris:0.000226542759783
1969:0.000226441217981
Greece:0.000226040283926
1945:0.000225328429128
1972:0.000224947052252
1977:0.000223085419689
Personal_name:0.000222994194712
Soviet_Union_ad1f:0.000222497758946
1978:0.000221991918240

11 Machines EMR
United_States_09d4:0.002905032006329
2006:0.002601076649573
United_Kingdom_5ad7:0.001381924062984
2005:0.001198770556559

Biography:0.000950232035161
Canada:0.000902946609550
England:0.000897257267659
France:0.000888500227283
2004:0.000834763436315
Germany:0.000763160669759
Australia:0.000738818201905
Geographic_coordinate_system:0.000722725288700
2003:0.000672244999079
India:0.000651145556541
Japan:0.000645777074573
Italy:0.000542382834957
2001:0.000539560091809
2002:0.000533306438610
Internet_Movie_Database_7ea7:0.000527705682315
Europe:0.000514040822245
2000:0.000505010268098
World_War_II_d045:0.000487161249831
London:0.000470090427904
Population_density:0.000452296840027
1999:0.000446416487551
Record_label:0.000446173252360
English_language:0.000443684693668
Spain:0.000443189041423
Russia:0.000418420842199
Race_(United_States_Census)_a07d:0.000415281053465
Wiktionary:0.000408755453552
Wikimedia_Commons_7b57:0.000389619070516
1998:0.000385867406202
Music_genre:0.000375614670938
1997:0.000368117601535
Scotland:0.000362215204190
New_York_City_1428:0.000362018605227
Football_(soccer):0.000352449597580
1996:0.000345348468695
Sweden:0.000340213229927
Television:0.000339457888664

Square_mile:0.000327665338053
Census:0.000326718057923
1995:0.000325278095442
California:0.000322410990844
China:0.000318508037310
Netherlands:0.000313766166194
New_Zealand_2311:0.000312485381252
1994:0.000310560968765
1991:0.000296372652449
1993:0.000293615205418
1990:0.000291919248116
New_York_3da4:0.000289889335582
Public_domain:0.000289550296507
1992:0.000281438708945
United_States_Census_Bureau_2c85:0.000279105730636
Film:0.000278702873905
Actor:0.000276459971619
Scientific_classification:0.000275889670506
Norway:0.000273898756834
Ireland:0.000272612850648
Population:0.000270725056028
Poland:0.000270336802547
1989:0.000263938335896
1980:0.000257873237556
January_1:0.000257860779852
Marriage:0.000255736025673
Brazil:0.000255457249352
Latin:0.000254143994131
Mexico:0.000254010212922
Politician:0.000251314347330
1986:0.000250838418147
1985:0.000244660491629
1979:0.000244437006268
French_language:0.000243775547458
1982:0.000243752808624
1981:0.000243552526072
1974:0.000241356353959

Per_capita_income:0.000241204517792
Album:0.000239468213543
Switzerland:0.000239262314441
1984:0.000239000443780
1987:0.000238793966442
South_Africa_1287:0.000238737715177
1983:0.000238712317022
Record_producer:0.000235816191851
1970:0.000235026298389
1988:0.000233317490668
1976:0.000232279330600
1975:0.000229593276736
Km²:0.000229303435496
Paris:0.000226542542286
1969:0.000226444508921
Greece:0.000226042703453
1945:0.000225328294457
1972:0.000224957020473
1977:0.000223085854211
Personal_name:0.000222993533765
Soviet_Union_ad1f:0.000222504696273
1978:0.000221989954110