LSI - Project 2: Fast Convergence PageRank in Hadoop

Deployment Steps

Step 1: S3 Bucket Setup:

- Authenticate and hit the URL ("https://console.aws.amazon.com/s3/home")
- Setup the input/output folder in S3
  i. Create a new Bucket
     a. Click on 'Create Bucket'
     b. Enter a unique name for the 'Bucket Name'
     c. Click 'Create'
  ii. Enter the bucket and create the folders
     a. Create folders 'input' , 'output' and 'log'
     b. Upload 'BlockerPageRank.jar' in the Bucket next to the other folders

Step 2: Mapreduce Setup

- Authenticate and hit the URL ("https://console.aws.amazon.com/elasticmapreduce/home")
- Click on 'Create New Job Flow'
  a. Provide a Job Flow Name ,
     Hadoop Version -> Amazon Distribution
     Create a Job flow -> Run your own application
     Job type -> Custom JAR
  b. Specify the JAR location : <S3bucket location>/<jar name>
     JAR Arguments : job.BlockedPR <S3 input bucket location>
     <S3 output bucket location> ….....from STEP 1
  c. Specify the number of instances desired. Click continue
  d. Set Amazon S3 Log Path : <S3 Log Bucket locaton>.....from STEP 1
     Select Enable Debugging.
     Click Continue. Click Continue
  e. Confirm the configurations and click 'Create Job Flow'

Preprocessing:

Reject Edges:

The random subset of the edges are taken using the Net ID : hpp8

```
// compute filter parameters for netid hpp8
double fromNetID = 0.8;
double rejectMin = 0.792
double rejectLimit = 0.802;
```

File Formats:

The input file is preprocessed as follows

<From Node> <To Node>

➔to

<From Node> <To Node> <degree> < PageRank>

The degree of each node is computed and the page rank is set as 1/N

Format of Data

Input File format (Mapper):

Each line represents the edge and provides the information of the node

<From node> <To node> <degree> <Page Rank>

In case of nodes with no outedges, we create an entry of the format

<From node> <NONE> <degree> <Page Rank>

Input File format (Reducer):

Intermediate file between Map and Reduce: The values of a line are delimited by '_'

<From node>_<To node>_<degree>_<Page Rank>

Features Implemented

- Node by Node computation of Page Rank
- Blocked computation of Page Rank
- Jacobi vs. Gauss-Seidel
- Random Block Partition

Structure of Solution

Simple computation of PageRank:

In this method, the reducer key corresponds to the individual node of the graph. The convergence of this method is relatively slow.

Logic:

Mapper:

<;u->v, outdeg(u), PR(u)>

--> <u; u , v, outdeg(u), PR(u)>

--> <v; u , v, outdeg(u), PR(u)>

Reducer:

(v; { < u → v, outdeg(u), PR(u) > ∀ u → v } ∪

{ < v → w, outdeg(v), PR(v) > ∀ (w | v → w) })

Compute : PR(v) = (1-d) /N + d ∑ PR(u)/outdeg(u).....d = 0.85

Emit :<;v->w, outdeg(v), PR(v)>

Blocked Computation of PageRank:

In this method the reducer key corresponds to the block which the node belongs. The nodes are partitioned in a fashion that reduces inter-block edges.

Logic:

Mapper:

<;u->v, outdeg(u), PR(u)>

--> <Block(u); u , v, outdeg(u), PR(u)>

--> <Block(v); u , v, outdeg(u), PR(u)>

Reducer:

(Block(v); { < u → v, outdeg(u), PR(u) >| ∀ u → v , v ∈ Block(v)} ∪

{ < v → w, outdeg(v), PR(v) > | ∀ (w | v → w), v ∈ Block(v)})

Compute :

while(average_residue<0.001)

{

$PR_T(v) = (1-d) /N + d \sum PR_{T-1}(u)/outdeg(u).....d = 0.85$

average_ residue = [∑ | $PR_T(v)$ - $PR_{T-1}(v)$ | / $PR_T(v)$] / SizeOFBlock .....∀ v

}

Emit: <;v->w, outdeg(v), $PR_T(v)$>

Gauss-Seidel Computation of PageRank:
In this method the reducer key corresponds to the block which the node belongs. The nodes are partitioned in a fashion that reduces inter-block edges.

Logic:

Compute :

while(average_residue<0.001)
{

$PR_T(v) = (1-d) /N + d \sum PR_T(u)/outdeg(u)$……...d = 0.85

average_ residue = [$\sum$ | $PR_T(v) - PR_{T-1}(v)$ | / $PR_T(v)$] / SizeOFBlock ..∀ v

}

Classes

UnlockedPR.java:
Is the main file that is run for the MapReduce task(Node by Node). It has the global state i.e. the average number of iterations and average residue for each task. The convergence criteria are specified here.
UnblockedPRMap.java:
Has the Mapper logic for Node by Node computation for PageRank
UnlockedPRRReduce.java:
Has the Mapper logic for Node by Node computation for PageRank

BlockedPR.java:
Is the main file that is run for the MapReduce task(Blocked). It has the global state i.e. the average number of iterations and average residue for each task. The convergence criteria is specified here.
BlockedPRMap.java:
Has the Mapper logic for block computation for PageRank
BlockedPRRReduce.java:
Has the Reducer logic for block computation for PageRank
Block.java:
Contains the logic for calculating the block number for a node

Node.java:
It is wrapper for a node in the graph. Contains the node number and arraylists for outgoing/incoming edges.

Results:

Simple computation of PageRank:

Average Residue:

Pass	Average Residue
1	2.338458
2	0.332288
3	0.192037
4	0.094273
5	0.062793
6	0.033937
7	0.027224

Blocked Computation of PageRank:

Average Residue:

Pass	Average Residue
1	2.814155
2	0.038009
3	0.023898
4	0.00977
5	0.003781
6	0.000884

Harsh Panchal(hpp8), Jayanth Parayil Kumarji(jp972), Supriya Singh(ss2895)

Average Iterations:

Pass	Average Iterations
1	17.5
2	7.17
3	5.86764
4	3.89706
5	2.52941
6	1.33822

Page Rank Values:

Node	Page Rank	Node	Page Rank	Node	Page Rank	Node	Page Rank
10327	1.87E-06	181513	3.53E-07	353644	3.77E-07	524509	0.000982353
20372	5.19E-07	191624	5.28E-06	363928	3.61E-07	534708	2.54E-05
30628	3.06E-07	202003	4.12E-06	374235	3.36E-06	545087	0.002572951
40644	2.81E-07	212382	2.62E-07	384553	4.00E-07	555466	0.001175638
50461	3.00E-07	222761	0.001008524	394928	6.42E-07	565845	1.64E-06
60840	2.19E-07	232592	8.37E-05	404711	2.46E-07	576224	1.09E-06
70590	3.06E-07	242877	5.35E-07	414616	3.12E-07	586603	9.72E-07
80117	2.19E-07	252937	2.19E-07	424746	1.53E-06	596584	4.00E-06
90496	0.000863231	263148	2.98E-07	434706	2.78E-06	606366	4.28E-07
100500	2.57E-07	273209	2.19E-07	444488	3.84E-06	616147	6.40E-07
110566	2.03E-06	283472	2.71E-07	454284	4.98E-07	626447	1.70E-05
120944	4.78E-07	293254	2.62E-07	464397	1.13E-05	636239	1.07E-06
130998	2.35E-07	303042	4.96E-06	474195	9.62E-07	646021	3.12E-07
140573	6.29E-07	313369	9.04E-07	484049	5.82E-07	655803	2.19E-07
150952	2.19E-07	323521	2.19E-07	493967	2.19E-07	665665	9.88E-07
161331	2.19E-07	333882	5.16E-07	503751	6.43E-06	675447	1.10E-06
171153	5.62E-07	343662	2.52E-07	514130	6.18E-07	685229	3.58E-07

Gauss-Seidel Computation of PageRank:

Average Residue:

Pass	Average Residue
1	2.814802
2	0.038895
3	0.025097
4	0.010899
5	0.004798
6	0.001867
7	7.63E-04

Average Iterations:

Pass	Average Iterations
1	9.61764
2	5.26470
3	4.52941
4	3.27941
5	2.32352
6	1.64705
7	1.26470

Page Rank Values:

Node	Page Rank	Node	Page Rank	Node	Page Rank	Node	Page Rank
10327	1.87E-06	181513	3.53E-07	353644	3.77E-07	524509	0.000982353
20372	5.19E-07	191624	5.28E-06	363928	3.61E-07	534708	2.54E-05
30628	3.06E-07	202003	4.12E-06	374235	3.36E-06	545087	0.002572951
40644	2.81E-07	212382	2.62E-07	384553	4.00E-07	555466	0.001175638
50461	3.00E-07	222761	0.001008524	394928	6.42E-07	565845	1.64E-06
60840	2.19E-07	232592	8.37E-05	404711	2.46E-07	576224	1.09E-06
70590	3.06E-07	242877	5.35E-07	414616	3.12E-07	586603	9.72E-07
80117	2.19E-07	252937	2.19E-07	424746	1.53E-06	596584	4.00E-06
90496	0.000863231	263148	2.98E-07	434706	2.78E-06	606366	4.28E-07
100500	2.57E-07	273209	2.19E-07	444488	3.84E-06	616147	6.40E-07
110566	2.03E-06	283472	2.71E-07	454284	4.98E-07	626447	1.70E-05
120944	4.78E-07	293254	2.62E-07	464397	1.13E-05	636239	1.07E-06
130998	2.35E-07	303042	4.96E-06	474195	9.62E-07	646021	3.12E-07
140573	6.29E-07	313369	9.04E-07	484049	5.82E-07	655803	2.19E-07
150952	2.19E-07	323521	2.19E-07	493967	2.19E-07	665665	9.88E-07
161331	2.19E-07	333882	5.16E-07	503751	6.43E-06	675447	1.10E-06
171153	5.62E-07	343662	2.52E-07	514130	6.18E-07	685229	3.58E-07

Random Partition:

Average Residue:

Pass	Average Residue
1	2.33908
2	0.32229
3	0.191197
4	0.093665
5	0.061965
6	0.033462
7	0.02675
.	.
.	.
21	9.82E-04

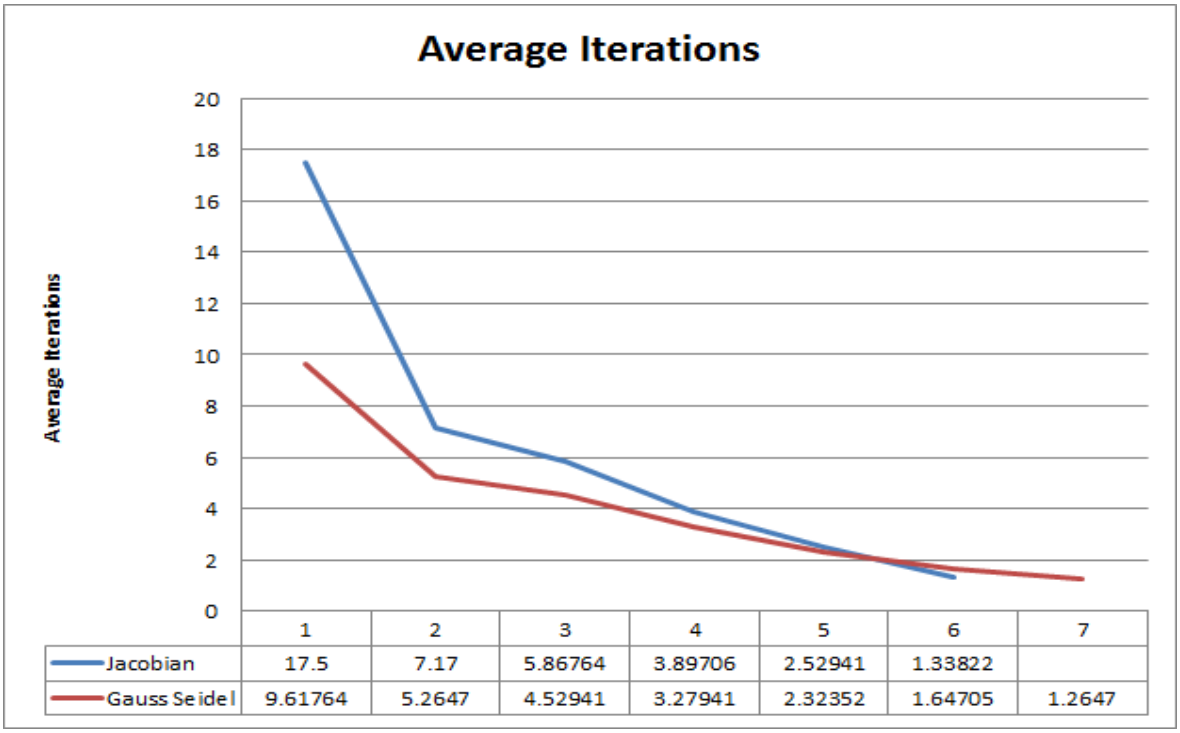
Average Iteration:

Pass	**Average Iteration**
1	3
2	2.705882
3	2.014706
4	2
5	2
6	2
7	2
.	
.	
21	1.838235

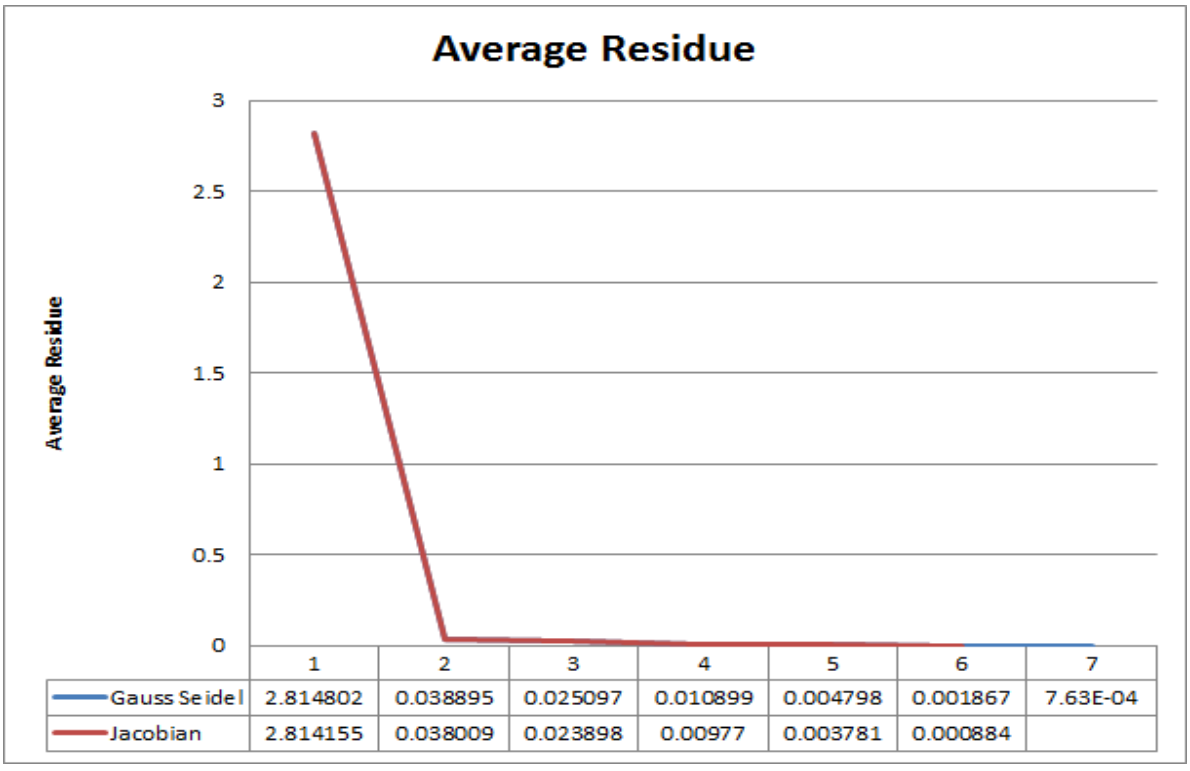
Page Rank Values:

Node	PageRank	Node	PageRank	Node	PageRank	Node	PageRank
10327	1.87E-06	181513	4.02E-07	353644	4.48E-07	524509	0.001079
20372	5.15E-07	191624	7.26E-06	363928	3.56E-07	534708	1.26E-05
30628	3.08E-07	202003	3.95E-06	374235	2.57E-06	545087	0.002738
40644	2.81E-07	212382	2.64E-07	384553	4.21E-07	555466	0.00142
50461	2.98E-07	222761	0.001207	394928	5.41E-07	565845	1.74E-06
60840	2.19E-07	232592	0.000102	404711	2.40E-07	576224	1.16E-06
70590	3.09E-07	242877	3.92E-07	414616	3.14E-07	586603	1.02E-06
80117	2.19E-07	252937	2.19E-07	424746	2.11E-06	596584	3.94E-06
90496	0.000952	263148	3.00E-07	434706	2.99E-06	606366	4.31E-07
100500	2.59E-07	273209	2.19E-07	444488	3.73E-06	616147	6.13E-07
110566	1.87E-06	283472	2.71E-07	454284	5.46E-07	626447	6.98E-06
120944	4.33E-07	293254	2.64E-07	464397	1.19E-05	636239	1.12E-06
130998	2.30E-07	303042	6.34E-06	474195	1.12E-06	646021	3.12E-07
140573	6.55E-07	313369	8.43E-07	484049	6.60E-07	655803	2.19E-07
150952	2.19E-07	323521	2.19E-07	493967	2.19E-07	665665	9.84E-07
161331	2.19E-07	333882	7.07E-07	503751	4.71E-06	675447	1.08E-06
171153	5.65E-07	343662	2.61E-07	514130	7.00E-07	685229	3.85E-07

Analysis(Jacobi vs. Gauss-Seidel):



	1	2	3	4	5	6	7
Jacobian	17.5	7.17	5.86764	3.89706	2.52941	1.33822	
Gauss Seidel	9.61764	5.2647	4.52941	3.27941	2.32352	1.64705	1.2647



	1	2	3	4	5	6	7
Gauss Seidel	2.814802	0.038895	0.025097	0.010899	0.004798	0.001867	7.63E-04
Jacobian	2.814155	0.038009	0.023898	0.00977	0.003781	0.000884	