# 

Finding most influential papers based on DBLP Citation Network

**May 08,2016**

**University of Illinois at Springfield**

### 

### Project Team:

### Anusha Valiveti(avali2)

### Sharmilee Rajkumar Rajan(srakj2)

**Table of Contents:**

Introduction ………………………………………………………………………..3

Methodology ………………………………………………………………………3

Results and Conclusion …………………………………………...……………….5

References …………………………………………………………………………6

**Introduction:**

The aim of the project is to find the top 10 influential papers of the DBLP Citation network. The Weighted page rank algorithm is used in order to find the most influential papers.

The Page rank algorithm is one of the most widely used ranking algorithm. Page rank algorithm states that if a page has more or important links to it, its links to other pages also become important. This page rank algorithm is applicable to all the linked graphs. The Page rank algorithm takes incoming links of a page into account and propagates the ranking through links.

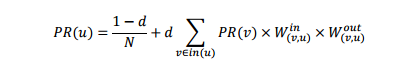
**Methodology:**

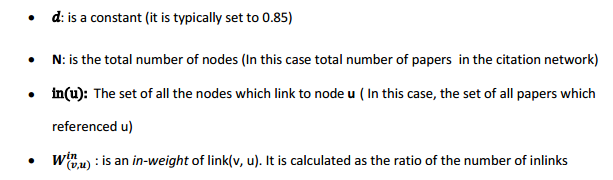
In this project to extract the data from the DBLP citation network to the required format we used Map reduce programming where the output from the map reduce program goes to hive for the computation of page rank.

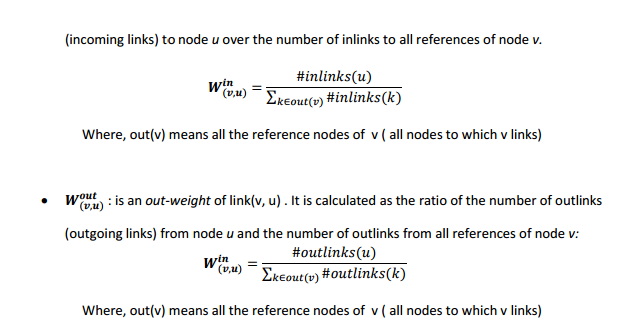
As the first step, the mapper class receives the data from the DBLP Citation network, we used custom input format class to read the data. The mapper class checks if there are any papers which doesn’t have at least one reference paper and removes them. All the valid papers and its references along with the page title goes to the reducer class where the reducer gets a list of referenced papers of a single paper. Reducer performs the summation of all the reference papers and emits them. Here the key to the Reducer class is the page id and the title of the page.

The hive query takes the input from the Map reduce program and generates a link graph where the data is in the form of page id, reference id. The page rank is calculated for 10 iteration in this project using hive queries and joins. The whole dataset is executed in the local mode in UIS cluster.

The formulae to compute the pagerank is :

Where 

****

****

**Results and Conclusion:**

The input to the project is taken from DBLP citation network.

<https://aminer.org/billboard/citation>

and the file is DBLP-Citation-network V7

Explanation of the project on sample input data.

For better understanding the whole project, we created sample data to explain each and every step of the map reduce programming and hive queries.

The sample input data to the map reduce program is:



This sample data has different varieties of papers like

* A paper which has no reference papers.
* A paper with number of incoming links as zero.
* A paper with outgoing links as zero, etc.

**Explanation of Map & Reduce Programs:**

In the input data, the paper id starts with ‘#index’, paper title starts with #\* and reference id starts with ‘#%’.

In this sample data, the paper id 2 has no reference id’s hence it is filtered from the remaining records in the mapper class. The mapper class emits the page id with page title as key and reference id as value.

**Example:**

key=101 Performance limitations of the Java core libraries

Value=102

In Reducer class gets the list of reference id’s of a paper id as input. Here, reference id’s belongs to the same paper id is aggregated together to form a string in which the reference id’s are separated by ‘,’.

**Example:**

Reducer emits:

Key=101 Performance limitations of the Java core libraries

Value=102,103,105

Here is the output file generated from the map reduce program

****

**Hive queries Explanation with Examples:**

Here is the document explaining the hive queries and results of the queries on sample data.



**References:**

<https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF>

http://people.cis.ksu.edu/~halmohri/files/weightedPageRank.pdf