International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 12
April 2018

# An Effective Algorithm designed for Ranking Research Papers based on the Citation Network

C.B.Rajasekhar Reddy & S.M.Ad.Norullabaig (M.Tech)
*MCA 6th sem, regno:0515122. Rims college, Tirupati

## Abstract

*In this paper we introduce a novel and efficient approach to detect and rank topics in a large corpus of research papers. With rapidly growing size of academic literature, the problem of topic detection and topic ranking has become a challenging task. We present a unique approach that uses* closed frequent keyword-*set to form topics. We devise a modified* time independent Page Rank *algorithm that assigns an authoritative score to each topic by considering the sub-graph in which the topic appears, producing a ranked list of topics. The use of citation network and the introduction of time invariance in the topic ranking algorithm reveal very interesting results. Our approach also provides a clustering technique for the research papers using topics as similarity measure. We extend our algorithms to study various aspects of topic evolution which gives interesting insight into trends in research areas over time. Our algorithms also detect hot topics and landmark topics over the years. We test our algorithms on the* DBLP *dataset and show that our algorithms are fast, effective and scalable. We have introduced a new metric in the algorithm which takes into account the time factor in ranking there search papers to reduce the bias against the recent papers which get less time for being studied and consequently cited by the researchers as compared to the older papers. Often a researcher is more interested in finding the top conferences in a particular year rather than the overall conference ranking.*

## INTRODUCTION

Thousands of research papers are published every year and these papers span various fields of research. For a new researcher, it becomes a very difficult task to go through the entire repository of research papers in order to determine the *important* ones. The term *important* is subjective but it can be assured that a research paper that is popular will be *important* in most cases. There can be several ways of determining whether a research paper is *important* depending on

the field of work, conference of publication, etc.

In this paper we propose an efficient method to rank the research papers from various fields of research published in various conferences over the years. Our method is based on the citation network of the research papers. Research papers cite other research papers from which they derive inspiration and there exists a well connected graph structure among the network of research papers. The importance of a research paper is directly proportional to the number of research papers that cite it. We have used this concept in our algorithm. Often a researcher might be more interested in knowing about the *important* conferences and authors from the field of his/her research. Using the research

paper scores as determined by our algorithm, we formulate the conference scores. A conference is as good as the research paper that it publishes. We have used this notion to rank the conferences based on the quality of research papers published by them. For an author, not only the quality of research paper published by him/her but also the quality of conference in which he/she publishes the paper is important. So for the author scores, we use both the research paper scores and the conference scores for all the research papers published by the authors.

A system ranking research papers, conferences and authors can have various applications. Some straightforward uses can be:

• Looking up the quality of a given research paper

• Comparing two or more research papers

• Comparing the research papers published in a given year

• Comparing the research papers published at a conference

• Comparing two or more conferences

• Comparing the quality of a conference as years pass

• Comparing two or more authors

• Comparing the research papers by a particular author

## RELATED WORK

A page with a high Page Rank means that there are many pages pointing to it, or that a page with high Page Rank is pointing to it. Intuitively, pages that are highly cited are worth browsing, and pages that are cited by the high Page Rank pages are also worth reading. Page Rank handles both cases

recursively by propagating weights through the link structure of the web (Maslov & Redner, 2008). Page 4 of 19 The damping factor d is the probability that the random surfer will follow a link on the existing webpage (Brin & Page, 1998). The random surfer, however, still has a (1-d) chance to start a completely new page. A high damping factor means that the random surfer has a high chance of following the internal link, and a low chance of clicking a new external random page. Boldi, Santini, and Vigna (2005) provided the mathematical analysis of Page Rank when the damping factor d changes, finding that contrary to popular belief, for real-world graphs, values of damping factor d close to 1 do not give a more meaningful ranking than other high damping factors. A low damping factor means that every node has more or less the same chance (probability roughly equals to 1/N where N is number of nodes in the graph) to get clicked by the random surfer. The choice of damping factor is empirical and it is set to be 0.85 in Google Page Rank, giving the speedy convergence of the power method (Brin & Page, 1998). Chen, Xie, Maslov, and Redner (2007) explain that the reason Page Rank sets d to 0.85 is based on the observation that a typical web surfer follows the order of six hyperlinks (coincident to the six degrees of separation in social network analysis). It implies that roughly five-sixths of the time a random surfer follows the links on the webpage, while one-sixth of the time this random surfer will go to a completely new page.

There has been considerable work in the field of academic research. Most of the work is on researcher profiling, which aim at ranking the authors. The work by Quinkun Zhao *et al.* [9] is one such work, which studies the relationship between authors using community mining techniques. Other work is *ArnetMiner* [8] which ranks the authors on *h*-index [5] and conferences on impact factor Our main focus here is to rank the research papers. The rankings for conferences and authors are derived from the research paper ranks. Most work on academic research uses number of citations as the metric. However, it is quite intuitive that this metric ignores the importance of the quality of citations, taking into consideration only the quantity of citations. The metrics like *h*-index, *g*-index and impact factor are based on the number of citations, and hence would not give correct results in all cases. We use a modified version of the Page Rank algorithm [7], which considers the quality of

the citing paper to rank the paper being cited. There has been some work on academic research using Page Rank algorithm like the work by Ying Ding *etal.* [3], where author co-citation network is used to rank the authors.

## PROPOSED APPRAOCH

Our method is entirely based on the graph structure of the citation network formed by the research papers.

**Definition 1.** Citation Graph: *We define the citation graph G = (V, E) comprising a set V of nodes, which each node Ni representing a research paper Ri and a set E of directed edges, with each edge Eij directed from the citing node Ni to the cited node Nj.*

We will build the citation network defined as a graph, with each research paper representing a node and the citations representing the edges in the graph, the edges being directed ones, directed from the citing node to the cited node. Each node has several attributes viz. research paper title, author(s), year of publication and conference of publication.

*A. The Ranking Algorithm*

The algorithm for ranking the research papers based on citation network uses the two types of edges in a graph:

**Definition 2.** Out links*: From a given node N, link all the nodes Ni that the node N cites.*

**Definition 3.** In links: *To a given node N, link all the nodes Nj that cite the node N.*

These *out links* and *in links* will be used while

Calculating the authoritative score [6] for each node. The algorithm for ranking the research papers consists of two parts:

• Creating the *out links* and the *In links*.

• Using modified iterative Page Rank algorithm to

Calculate the authoritative score for each node.

The above two parts are implemented by the Following two procedures.

**Procedure 1:** Create Out links and In links

**Require:** Paper Citation *PC*

1: Paper Outlinks *PO* = *PC*
2: **for each** paper *P* **in** *PO*
3: Citations List *CL* = *PC*[*P*]
4: Outlinks Count *OC*[*P*] = *CL*.**length()**
5: **for each** citation *C* **in** *CL*
6: **if** *C* **in** *PI*
7: **append** *P* **to** *PI*[*C*]
8: **else**
9: *PI*[*C*] = [*P*]

The procedure 1 uses paper citations and create a data structure mapping each paper to its inlinks and outlinks. In the procedure, we first determine the list of all the papers a paper cites (*Step 3*). Corresponding to each paper, the number of outlinks is stored in OC (*Step 4*).

Then from s*tep 5* to s*tep 9*, we form the data structure to store the inlinks corresponding to each paper.

**Procedure 2:** Modified Iterative PageRank Algorithm

**Require:** Citation Network *CN*, Paper Year *PY*, Outlinks Count *OC*, Paper Inlinks *PI*, Damping Factor θ

1: **Initialize** Paper Rank *PR* **to** 1.0 **for each** paper *R*
2: **while** true
3: *flag* = true
4: **for each** paper *R* **in** *PR*
5: Current Score *CS* = *PR*[*R*]
6: **if** *R* **in** *PI*
7: Inlinks List *IL* = *PI*[*R*]
8: New Score *NS* = 0.0
9: **for each** inlink *I* **in** *IL*
10: **if** *I* **in** *PR*
11: *NS* += *PR*[*I*]/*OC*[*I*]
12: *NS* = (1-θ) + θ**NS*
13: **if** *CS* **is not equal to** *NS*
14: *flag* = false
15: Updated Paper Rank *UPR*[*R*] = *NS*
16: **if** *flag* **is equal to** true
17: **break**
18: **copy** *UPR* **to** *PR*
19: **clear** *UPR*
20: Maximum Score *MS* = **Maximum Score in** *PR*
21: **for each** paper *R* **in** *PR*
22: *PR*[*R*] /= *MS*

The procedure 2 is essentially a modified version of the iterative PageRank Algorithm.

The iterative Page Rank algorithm starts with initializing all the candidates to a constant value, generally *unity* and then it iteratively modifies each candidate's score depending on the score of the candidates that point towards it. It stops when all the candidate scores converge, i.e. become constant.

*Step 1* initializes the score of each paper to unity.

*Step 2* to *19* is the iterative calculation of the authoritative score for each paper. The iteration stops when there is no change in the score of any paper during the iteration. This is signified by no change in value of *flag* set to *true* in

*step 3; flag* is set to *false* in *step 13* and 14 if there is a change in the score of any paper during the update step.

From *step 4* to *15*, for each paper, a new authoritative score is calculated based on the scores of the inlinks in the previous iteration (*Step 9* to *11*). The Page Rank algorithm is based on the fact that the quality of a node is equivalent to the summation of the qualities of the nodes that point to it. In this case, quality refers to the score of the research paper. Now if a research paper cites more than one research paper, it is obvious that it has drawn inspiration from various sources and hence its effect on the score of the paper it cites should diminish by a factor equal to the number of paper it cites. This fact is used in *step 11* by dividing the inlink score by the number outlinks of the inlink. *Step 12* modifies the score calculated above to incorporate the damping factor. The use of damping factor is required to prevent the scores of research papers that do not have any inlinks from falling to zero. The value of damping factor is set to *0.85* [1].

The formula for calculating the score at a given iteration is:

$$NS = 0.15 + 0.85 * \Sigma (PR[I]/OC[I])$$

In *step 20* to *22*, we normalize the scores to scale down the scores within the range *[0,1]*. Thus, finally we get the authoritative scores of each research paper based on the score of the research papers that have cited it.

The above algorithm is fairly sufficient for ranking the research papers. But there is one important aspect that has not been considered yet – *time*. It is quite obvious how *time* is a very important factor when it comes to ranking the research papers. The older research papers obviously have more time to be studied by the researchers all over the world and consequently be cited in

various research papers. The newer ones fall behind in this aspect. To make our algorithm time-independent, we need to have some normalization of the scores based on time.

*B. Time-independent Scoring*

To make the algorithm time-independent, we need a time-dependent metric that can be used in our formula for the authoritative score in the algorithm. The metric we propose to use is the *Average Number of Citations per Paper* in a year, i.e. the average of the total number of citations of the research papers published in each year. This metric is suitable for normalization with respect to time as it captures the fact that an older paper has more time to be cited by researchers in comparison to the recent papers. We need to modify the formula for calculation of *NS* in the procedure 2, to incorporate time-independence. For each research paper, we have the year of its publication. Using the year of publication of all the research papers, we will pre-compute the total number of citations for each year and the number of research papers published in each year. Using them, we will determine the average number of citations per paper for each year. The following pseudo-code needs to be added to the procedure 2, before *step 1*.

0a: **for each** paper *R* **in** *PY*
b: year *Y* = *PY*[*R*]
c: Year Citation Count *YCC*[*Y*] += *OC*[*R*]
d: Year Paper Count *YPC*[*Y*] += 1
e: **for each** year *Y* **in** *YCC*
f: Average Year Citations Count *AYCC*[*Y*] = *YCC*[*Y*]/*YPC*[*Y*]

Once we have the average number of citations per year, we need to modify the formula for calculating the authoritative score. We will simply divide the *NS* by the average number of citations in the year of publication of the paper concerned. The change required in procedure 2 is in *step 12:*

12a: year *Y* = *PY*[*R*]
b: *NS* = (1-θ) + θ**NS*/*AYCC*[*Y*]

*C. Ranking Conferences*

The rank of a conference depends on the quality of research papers it publishes. This is the key idea behind our algorithm for ranking conferences. Using the authoritative scores for the research papers, we will rank the various conferences by calculating a cumulative authoritative score for each conference. The score for a conference will

essentially be the average score of all the research papers published in that conference. The formula for the score of a conference is:

**Conference Score *CS* = Σ(Paper Score *PS*)/( Number of Papers published in the Conference *NPC*[*C*])**

An extension of the ranking of the conferences is ranking the conferences year-wise and not as a whole. Forexample, we might want to know whether *VLDB1990* was better than *SIGMOD1992* in terms of the quality of research papers published or was it vice-versa or we might want to know how the quality of research papers published in *KDD* is varying over the years. For this, we need to make a small change in the way the conference score is being calculated. Earlier we used just the conference name as the key for scoring the conference score. Now we concatenate the year of publication to the conference of publication to the conference name and use this concatenated term is the key. Thus we have keys based on year-wise granularity like *KDD1999, ICDE1991, etc.*

The formula for the score of the conference remains the same:

**Conference Yearwise Score *CYS* = Σ(Paper Score *PS*)/( Number of Papers published in the Conference *NPCY*[*C*][Y])**

*D. Ranking Authors*

The rank of an author depends not only on the quality of research papers but also the conference he/she publishes his/her papers in. For ranking authors, thus we will use the scores of the research papers published by the author as well as the score of the conference in which the paper was published. This would incorporate the fact that publication in a comparatively better conference is more difficult. The authoritative score for the authors thus is essentially a weighted mean of the scores of the research paper the author has published, with the score of the conference as the weight.

The formula for the score of an author is:

**Author Score = Σ(Paper Score *PS* * Conference Score *CS*)/(Number of Papers published by the Author *NPA[A]***

## RESULTS AND DISCUSSION

*A. Dataset*

For building the citation network, we used the *DBLP XML Records* available at *http://dblp.uni-trier.de/xml/* [2]. The DBLP dataset contains information about various research papers from various fields published over the years. This information includes the name of the research paper, its author(s), the year of publication, the conference of publication and the list of research papers the given research paper cites. We built the citation network defined as a graph, with each research paper representing a node and the citations representing the edges in the graph, the edges being directed ones, directed from the citing node to the cited node. Each node is uniquely represented by the unique key from *DBLP* and the node has several attributes viz. research paper title, author(s), year of publication and conference of publication. It is to be noted that the *DBLP* dataset that we used contained citation information till the year *2010* only.

*B. Data Pre-processing*

The *DBLP* dataset also contains information that is not useful in our algorithms. We need pre-process the dataset to extract only the information that we will use in our algorithms. In data pre-processing, we extracted all the titles, author(s), conference of publication, year of publication and citations.

*C. Results*

We implemented our algorithms on the *DBLP* dataset and discovered various interesting results. The *DBLP* dataset contains *16,32,442* research papers from the various fields of research and published in various conferences all over the world. The iterative modified PageRank algorithm converges in *212* iterations for the *DBLP* dataset in the time dependent domain and in *32* iterations in the time independent domain. This shows that the time independent algorithm converges faster than the time dependent algorithm. This is because the authoritative scores of the papers are more uniform in the time independent domain due to the normalization by *Average Number of Citations per Paper*. The ranked list of research papers, conferences and authors produced by our experiments were verified and found to be reasonably good by the field experts.

*C.1. Ranking Research Papers*

The following tables show the top 10 research papers as determined by our algorithm for ranking the research papers, in time-dependent and time-independent domains.

TABLE I. TOP 10 PAPERS IN TIME-DEPENDENT DOMAIN

| Title | Score |
|---|---|
| Computers and Intractability: A Guide to the Theory of NP-Completeness. | 1.00000000 |
| A Relational Model of Data for Large Shared Data Banks. | 0.71658114 |
| Communicating Sequential Processes. | 0.51063789 |
| Introduction to Algorithms. | 0.48190250 |
| Compilers: Principles, Techniques, and Tools. | 0.48052828 |
| Smalltalk-80: The Language and Its Implementation | 0.45773251 |
| Introduction to Modern Information Retrieval. | 0.44686714 |
| Report on the algorithmic language ALGOL 60. | 0.42833770 |
| A Characterization of Ten Hidden-Surface Algorithms. | 0.41820052 |
| Graph-Based Algorithms for Boolean Function Manipulation. | 0.41524671 |

TABLE VI. TOP10 AUTHORS IN TIME-DEPENDENT DOMAIN

| Rank | Author Name | Score |
|---|---|---|
| 1 | A. Nico Habermann | 0.34606894 |
| 2 | E. F. Codd | 0.33449698 |
| 3 | Van Jacobson | 0.15132072 |
| 4 | William E. Lorensen | 0.13856569 |
| 5 | Kapali P. Eswaran | 0.11449051 |
| 6 | Irving L. Traiger | 0.10304509 |
| 7 | Ivan E. Sutherland | 0.09953599 |
| 8 | David D. Clark | 0.09351475 |
| 9 | Alan J. Perlis | 0.07604195 |
| 10 | Jack B. Dennis | 0.07403101 |

## APPLICATIONS

As mentioned earlier, our algorithms have a variety of applications in various fields of research, using the scores and ranks calculated by our algorithms.

*A. Ranking Keywords (Theme Mining)*

An important application of such a system is in the area of theme mining. Given a research paper, we can very well find out the keywords that the research paper is based on. Thus we can cluster research papers on the basis of their keywords.

Now we simply need to iterate over the keywords and find the average score of the research papers that contain that keyword. This would be the authoritative score for that keyword.

***Keyword Score*** *KS* = Σ***(Paper Score PS)/(number of research paper based on the keyword NP[K])***

Using a technique similar to that used for ranking conferences on year wise granularity, we can rank the keywords on year wise granularity as well.

***Keyword Yearwise Score KSY[Y]*** = Σ***(Paper Score PS)/(number of research paper published in a given year based on the keyword NPY[K][Y])***

Now having the *keyword Year wise Score*, we can select saythe top ten keywords for

each year and call that the major theme of research for that year.

*B. Speculative Ranking*

For a newly published research paper, we can devise a formula based on the citation network of the author(s) and the authoritative score of the conference in which the paper is published. Thus, we can incrementally add newly published research papers to the ranking system.

## CONCLUSIONS AND FUTURE WORK

The purpose of our work was to propose a way to identify the key research papers, conferences and authors from various fields of research. We have proposed a new and efficient method to achieve this cause. We introduced the time-independent algorithm for ranking, which as shown gives fairly satisfactory results. We analyzed the results and did a comparative study based on the ranking obtained by our algorithms.

The above work has a host of new areas of implementation. As mentioned above, we would like to implement theme mining on the keywords based on the ranking of the research papers. This can be extended to keywords in conferences, which could help new researchers to identify the conferences of his/her interest. Similarly we can also extend this work to keywords by authors.

Another future work can be the implementation of our work in recommendation systems. We can have a system identifying co-authors and recommending topics
of interest, etc.

## REFERENCES

[1] S. Brin, and L. Page, "The Anatomy of a Large- Scale Hypertextual Web Search Engine," Proceedings of the 7th international conference on World Wide Web (WWW), 1998.

[2] The DBLP Computer Science Bibliography. http://dblp.uni-trier.de/

[3] Y. Ding, E. Yan, A. Frazho, J. Caverlee, "PageRank for Ranking Authors in Co-citation Networks," Journal of the American Society for Information Science and Technology, 2009.

[4] E. Garfield, "The History and Meaning of the Journal Impact Factor," Journal of the American Medical Association, 2006.

[5] J. E. Hirsch, "An Index to Quantify an Individual's Scientific Research Output,"

Proceedings of National Academy of Sciences, 2005.

[6] J. Klienberg, "Authoritative sources in a hyperlinked environment," Proceeding of the 9th Annual ACM- SIAM Symposium on Discrete Algorithms, 1998.

[7] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Technical Report, Stanford InfoLab, 1999.

[8] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, Z. Su, "Arnetminer: Extraction and Mining of Academic
Social Networks," Proceedings of 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008.

[9] Q. Zhao, S. S. Bhowmick, X. Zheng, "Characterizing and Predicting Community Members from Evolutionary and Heterogeneous Networks," Proceedings of the 17th Conference on Knowledge Management, 2008