ISSN 2395-8618

A Multi-Entity Page Rank Algorithm

Chandramouli Shama Sastry, Darshan S Jagaluru and Kavi Mahesh

***Abstract*—We propose a generic multi-entity page rank algorithm for ranking a set of related entities of more than one type. The algorithm takes into account not only the mutual endorsements among entities of the same type but also the influences of other types of entities on the ranks of all entities involved. A key idea of our algorithm is the separation of prime and non-prime entities to structure the iterative evolution of the ranks and matrices involved. We illustrate the working of the proposed algorithm in the domain of concurrently ranking research papers, their authors and the affiliated universities.**

Index Terms— Multi Entity Page Rank, Mathematical Model, Evolving Stochastic Matrix, Prime Entity

I. Introduction

In this paper, we propose a mathematical model which can be used for ranking multiple interacting entities. It is widely accepted that page rank algorithm gives a meaningful and practical ranking order among a network of mutually related entities. However, the original page rank algorithm is designed for homogeneous entities and more often than not one finds it useful to rank sets of interacting or dependent entities of more than one kind in a system. The concept of ranking multiple entities at once is not entirely new and has been explored earlier, especially in ranking authors, papers and journals in a single system. However, the modifications to the page rank algorithm suggested in previous work were application specific and, as such are not readily suited to other applications. We present a generic algorithm which can be adapted to any specific application domain. We also illustrate the working of the algorithm with suitable examples and show mathematically how our algorithm is different from previous modifications. The main intuition behind the mathematical model is that tighter coupling between various entities in the ranking algorithm gives us better ranking orders. Our model ensures this by having an evolving stochastic matrix that changes iteratively along with the rank vectors of the entities.

Let's begin with an example to illustrate the intuition and motivation behind the algorithm for ranking multiple entities concurrently. Consider the problem of ranking universities purely from an academic perspective based on the research output of the universities. One would consider multiple factors for ranking: professors who work there, the research work they publish, the citations they obtain, and so on. We now face the related problem of ranking professors across universities by perhaps considering quite the same factors like the university where they work, their publications, citations, and so on. What we see here are a set of related entities whose ranks depend on each other: universities, professors, publications and perhaps others such as journals, conferences and publishers. In order to model this, we'll need a multi-entity ranking algorithm.

The outline of this paper is as follows: a brief review of previous work; a generic mathematical model for multi-entity page ranking algorithm; an example of how we can use the model for ranking authors and papers along with results in brief; a mathematical exposition of the internals of the algorithm.

II. Previous Work

Page Rank algorithm, developed by Sergey Brin and Larry Page was designed to rank web pages [1]. It is based on the random surfer model which allows a surfer to jump to a random page without necessarily following the "out-links" of the current web page. It is designed such that web pages with higher numbers of in-links or higher quality in-links, or both, are assigned higher ranks. This is a recursive algorithm that can be realized by a series of matrix multiplications. We present the core ideas and notations of the Page Rank algorithm briefly for the reader's convenience:

Let N be the number of pages to be ranked. We have a row normalized matrix H representing the directed graph of N nodes (i.e., pages) where the edges denote the hyperlinks between the pages (or any other such relation among the entities). The stochastic matrix G is then constructed as:

$$G = d \times H + \frac{1-d}{N} \times E \qquad (1)$$

where d is the damping factor describing the probability of jumps from one node to another and E is a matrix of all 1's. This matrix G should be interpreted as the matrix showing how a user may navigate from one page to another on the web. A user can either jump to a page following the links on that page or go directly to a random page. d indicates the probability that he

Acknowledgement: This work is supported in part by the World Bank and Government of India research grant under the TEQIP programme (subcomponent 1.2.1) to the Centre for Knowledge Analytics and Ontological Engineering (KAnOE), http://kanoe.org at PES University, Bangalore, India. The authors are grateful to Dr. I. K. Ravichandra Rao.

Chandramouli S Sastry and Darshan S Jagaluru are Undergraduate Students at PES University, Bangalore. Dr. Kavi Mahesh is the Dean of Research, Director of KAnOE and Professor of Computer Science at PES University.

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

will continue following the links mentioned on the page. Empirically, d is usually set to a value of 0.85. Page ranks are computed by the formula:

$$R = R_0 \times G^n \tag{2}$$

where R is the vector containing scores (i.e., ranks) of the nodes, R_0 is the vector describing initial assignment of ranks and n is the number of iterations until convergence. R_0 is generally constructed by assuming that all nodes have equal scores. Finally, after a sufficient number of iterations n, nodes which have in-links from other high scoring nodes or have a lot of in-links or both, get higher scores.

Modifications suggested in previous work to adopt the page rank algorithm for multiple interacting entities have focused mainly on the ranking of authors and papers in a network [2,3,4,5,6]. We consider two main papers out of those. Zhou, Orshanskiy, Zha and Giles [6] have suggested an algorithm based on Page Rank that considers three networks – social network connecting authors, citation network connecting publications and the authorship network connecting the authors with the papers. In their algorithm, the ranks of the authors and papers are first independently computed and then coupled using intra-class or inter-class walks. That is, in terms of the original Page Rank algorithm, if the random surfer was at an author node then he could randomly jump to another author node or to a paper node. This captures the interdependency between the final ranks of authors and papers to some extent. Yan, Ding and Sugimoto [5] have also taken a similar approach where they rank journals, authors and papers together. In this work, however, the ranks of the papers, authors and journals are computed simultaneously. They create a stochastic network of papers and then create inter-entity walks between papers and journals and papers and authors. Interestingly, the stochastic matrix is dynamically updated as the ranks of authors or journals change. The formula used to update the matrix is:

$$\bar{\bar{M}} = d\bar{M} + (1-d)ve^T \tag{3}$$

where $\bar{\bar{M}}$ is the stochastic matrix, $\bar{M}$ is the adjacency matrix and e is a vector of 1s. The rationale followed is that, users don't navigate towards all papers equally; rather they jump towards papers published by reputed authors or journals or both. The vector v captures the impact of the score of the journal and the author as a metric for the probabilities of jumping. In fact, the probabilities of jumping towards a paper changes as the ranks of authors or journals change and this is reflected in the stochastic matrix. That is, higher the author score or journal score, greater the probability of a random surfer jumping towards that paper. However, this may lead to a false ranking order as average papers written by good authors get a higher score irrespective of the citations obtained by them. This is because the score assigned depends on the contents of the stochastic matrix.

We can further illustrate this drawback as follows: consider a paper that is written by very good authors and published in a reputed journal but has not yet received any cites. When we do a stochastic matrix update, its cell contents get updated and logically, all papers start giving credit to this paper (which could sometimes be even greater than the credit that those papers are giving to actually cited papers). Further, the whole process being recursive, the paper score boosts the author and journal scores which further boost the citing strength. As such, papers which may not deserve high scores end up getting them. Nevertheless, changing the matrix dynamically has its own advantages, primarily because we cannot decouple the ranking of any one of the entities from the others given their interdependencies. In our model, we evolve the stochastic matrix by considering who is citing (which, in fact is the main idea of Page Rank) rather than who is getting cited and the resulting probabilities of jumping to any node are more likely to reflect the reality of the application domain.

III. Test Data

We use a publication and citation data set that we extracted from Google Scholar to illustrate and test our algorithm. We chose all papers belonging to the subject of "Web Semantics" published between 2013-2014. The details of the dataset are as shown in Table 1.

Table 1. Data about Papers in Web Semantics in 2013-14.

Number of unique Authors	1,801
Number of Papers	1,124
Number of Citation edges (excluding self-cites)	8,294
Number of Citation edges (including self-cites)	10,192

IV. Mathematical Model

A. *Notation:*

- E=Set of all entity classes considered for ranking.
- N_i = Number of instances of entity type i.
- T_{ij} denotes the j^{th} instance of entity type i.
- O_{ij}, the inter-entity relations, where i and j are instances of different entity classes and the order of the matrix is $N_i \times N_j$. For example, i could be paper and j could be organization. These matrices need not be Boolean; they could represent real numbers as well. If we considered professors and universities, a professor can be related to multiple universities in terms of: where he studied, where he works full-time, where he works as visiting professor, and so on and we could quantify these using non-negative real valued numbers.
- L_i, the intra-entity relations, where i is the type of entity. L_i is a square matrix of order $N_i \times N_i$. These could have different semantics depending upon type of entity.

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

- R_i − rank vector of order $1 \times N_i$ denoting scores of all instances of the i^{th} entity.

B. Prime entity

The first step is to choose an entity type P from the set E and designate it as the prime entity. The remaining entities are referred to as non-prime entities. NP_i refers to the i^{th} non-prime entity. The organization and interaction between prime and non-prime entities are as shown in Figure 1.

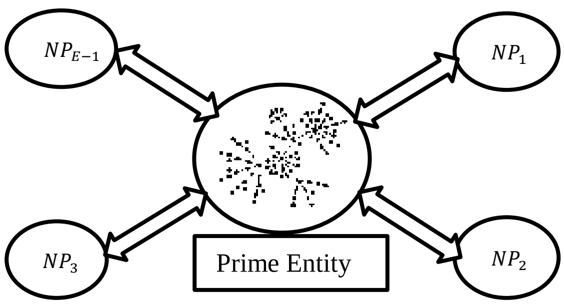


Fig. 1. Prime and Non-Prime Entities.

Prime entities are linked to one another by directed or undirected edges in a graph, whereas the non-prime entities are connected only to the prime entities. There are no edges among the non-prime entities. Prime entities serve to link the various non-prime entities. Also, the ranks of non-prime entities are influenced by the ranks of the prime entities and vice-versa.

For example, in the university ranking problem, we could consider the prime entity to be published papers. In the resulting model, papers will have a directed link between them which could denote, for example, the citation relationship. If we chose university to be prime entity, on the other hand, we could either consider an undirected graph denoting collaboration or consider a directed edge between two universities to denote that a professor who obtained his PhD from one of those universities works for the other. Choosing the prime entity determines the semantics of ranking in the chosen domain.

C. Representing the graphs

The graph connecting instances of prime entity is represented by a square matrix of order $N_P \times N_P$; this matrix is referred to as H in the future discussions. H is got by row-normalizing the matrix L_P. We then obtain the stochastic matrix G from H using (1) as described in original page rank algorithm.

In order to represent graphs linking the prime entities with the non-prime entities, we introduce:

For all $i \in E - \{P\}$,

Define $M_i = H \times O_{Pi}$ (4)

The matrix M_i quantifies the incoming links to the instances of non-prime entity i from the instances of prime entity P. Though both M_i and O_{Pi} denote and quantify the relationship between prime entity P and non-prime entity i, O_{Pi} denotes the direct relationship and M_i denotes the aggregate relationship.

In our example of university ranking, we could consider the "belongs to" relation for the inter-entity matrices between university and paper and between professor and paper. These two matrices on being multiplied with paper citation matrix would give us two matrices, each representing and quantifying the citation relationship between papers and professors and between papers and organizations thereby attributing credit to the entities.

D. Intra and Inter Entity walk

Using the matrices defined above, we can define the score of the non-prime entities in terms of the score of the prime entity as:

$$\forall\ i \in E - \{P\}, R_i = R_P \times M_i \quad (5)$$

We can also define the scores of the prime entity using the notion of page-rank (ignoring iteration numbers) as,

$$R_P = R_P \times G \quad (6)$$

R_P is initialized according to the original page rank algorithm as $R_P = \{1/N_P\ , for\ all\ T_{Pi}\ \}$.

Thus, we observe that the prime entities participate in the intra-class stochastic random walk and each of the matrices M_i help in inter-entity walk. It may be noted that this step is application independent. Continuing with our example, we can see that the model enables us to define the ranks of professors and organizations using those of papers that have cited them.

E. Building Recurrences:

This is the last and most important step of the mathematical model. Here, we modify the score vector of the prime entity based on the scores of the non-prime entities and the prime entities themselves. We use the notion of ownership and collaboration when making this modification. Note that the scores due to incoming links are already accommodated by the equations of the previous step. The general method for the modification is:

$$R_P = \alpha_0 R_P + \sum_{\forall\ i \in NP} \alpha_i R_i I_i + \beta R_P X \quad (7)$$

Where, I_i =Influence matrix$(N_i \times N_P)$ determining influence of the score of the i^{th} non-prime entity on prime entity and $X =$ collaboration matrix for determining collaboration score of a

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

given instance of prime entity using scores of other instances of prime entity using the notion of "collaboration" and

$$\sum_{i=0} \alpha_i + \beta = 1 \quad (8)$$

In this step, the influence matrix for each of the non-prime entities should be defined. Consider a non-prime entity W. The influence matrix I_W should define the influence that the ranks of instances of type W have on each of the instances of the prime entity. For example, assume that there are k instances of type W which influence the ranks of a certain instance T_{Pi} of the prime entity. Then, the i^{th} column of I_W should indicate the share of each of these k instances in influencing the score of T_{Pi} (other entries in the column being 0). These matrices are column stochastic. In our example, the ranks of the papers are influenced by the scores of the participating authors and organizations. Better the scores of professors and organizations, better the scores of the papers.

Collaboration matrix is used to bring in the notion of collaboration between one or more non-prime entity types with respect to a given prime entity instance. As described above, the scores of other instances of prime entity are used for factoring this in. However, in order to introduce this notion, the shares of each of the instances of the prime entity whose scores are to be considered should be based on non-prime entities. The description of this matrix is similar to that of the influence matrix with the exception that it is a square matrix since the influencing and influenced instances are of the same type. In our example, we could introduce this notion by considering the relative success of other papers when certain combinations of organizations and professors work on them. That is, any work done by a certain collaboration of organizations and professors could be as good as another. For example, works which are brought out by the collaboration of Google and Stanford involving some group of researchers from academia and industry may be at least as good as each other. In our adaptation of this model to the problem of ranking papers and authors, we show a method of including the impact of collaboration.

The factor β can be made zero if it doesn't make sense to include the collaboration factor between the non-prime entities. We have included it in the model so that certain applications could benefit by the use of this. However, if any of the αs are made zero, then the corresponding entities are effectively decoupled from the whole system. That is, the prime entity ranks are not affected by their ranks and we can compute the ranks of those instances separately. This defeats the whole purpose of multi-entity page ranking. Thus, we should not make any of these α's zero. Note that α_0 defines what percentage of the prime entity score is defined through in-links and the other α's define what percentage is defined by each of the other entities. Hence, higher the α_0, better the ranking order as a large percentage of the scores is defined through endorsements. These matrices can be static or dynamic. We give an example of one static and one dynamic in the example application that we describe later.

F. *Putting it all together: pseudo-code*

The scores of all the prime entities are initialized as $1/n_p$ as described in the model. The first step of the repeat-until loop involves computing the scores of the paper ranks using intra-entity stochastic walk among the network of prime entities. Following this, we compute the ranks of all the non-prime entities in terms of the prime entity. Having done this, we perform the modification step where we modify R_P to factor in the impacts of the scores of non-prime entities on the prime entity. We then normalize R_p in order to prevent arithmetic overflow. The convergence of the algorithm is then computed which determines the termination condition. The following algorithm assumes that the matrices and parameters are set as described in the preceding section.

Input:
- E − Set of entities
- P − Prime Entity
- NP − Set of non-prime entities
- Matrix G – The stochastic matrix
- Matrices M_i − Matrices for inter-entity walk
- Matrix X − Collaboration matrix
- Matrix I_i − Influence Matrices
- $\alpha_0, \alpha_i s \; and \; \beta$

Output:
- Ranking order of each of the entities

Procedure MERank:
Begin
    $R_P = [1/n_P \;\; ,1/n_P \,, \ldots ,1/n_P \,]$
    $\epsilon \;= 10^{-15}$
    Repeat:
        For every i ϵ NP
            $R_i = R_P \times M_i$
        End
        $R_P^{prev} = R_P$
        $R_P = R_P \times G$
        $R_P = \alpha_0 \times R_P + \beta \times R_P \times X$
        For every i ϵ NP
            $R_P = R_P + \alpha_i \times R_i$
        End
        Normalize R_P
        convergence = calc_convergence(R_P^{prev},R_P)
    Until convergence $< \epsilon$
    Return $\{R_i \; \forall \; i \; \epsilon E\}$
End

We can re-write (7) considering iteration coefficients as:

$$R_P^j = \alpha_0 R_P^j + \sum_{\forall \; i \; \epsilon \; NP} \alpha_i R_i^j I_i + \beta R_P^j X \quad (7a)$$

In terms of R_P, we can re-write (7a) using (5) as

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

$$R_P^j = \alpha_0 R_P^j + R_P^{j-1} \sum_{\forall\, i \,\epsilon\, NP} \alpha_i M_i I_i + \beta R_P^j X \qquad (7b)$$

Observe that we use R_P^{j-1} while computing influences and R_P^j while computing collaboration impacts. The rationale is explained in the mathematical exposition.

V. Example Application Showing Ranking of Authors and Papers

A. Choosing Prime Entity

Prime Entity = Papers; Non-Prime Entities = Authors. Let entity 1 refer to Authors and entity 2 refer to Papers.

B. Representing the graphs

Intra-entity walk.

We use the paper citation graph for this. The adjacency matrix L_2 is defined as:

$$L_2[i,j] = \begin{cases} = 1.0 \text{ if paper } i \text{ cites paper } j \\ = 0.2 \text{ if papers } i \text{ and } j \text{ have at least 1 common author (self-citation)} \\ = 0 \text{ if paper } i \text{ does not cite paper } j \end{cases}$$

We can use domain-specific metrics to get better results. For example, here, we have considered the strength of a self-citation to be lower than that of a normal cite. We get the matrix H by row normalizing L_2. We constructed stochastic matrix G using damping factor value of 0.85.

Inter-entity walk

We define the matrix M_1 as the product of H and O_{21}, where O_{21} captures the ownership relation between authors and papers (order $= N_2 \times N_1$). The ownership matrix can be a real-valued matrix as well. In this example, we discriminated between the ownership influence of first author and later authors using the following idea: If author a is the k^{th} author of paper b, then $O_{21}[b,a] = \frac{2\times(n-k+1)}{n(n+1)}$, where n is the total number of authors of paper b [8].

C. Building Recurrences

Intra and inter entity walk step is common to all applications and we omit their details here.

Influence Matrix I_1

We used the transpose of the ownership matrix O_{21} as the influence matrix I_1. For this application, we chose to make it static for all iterations.

Collaboration Matrix X

For quantifying the score of collaboration of non-prime entities (authors), we developed the following formulation. For ease of explanation, consider the papers which have at least 1 common author as *co-papers* of each other. For any given paper, we partition the set of co-papers into 3 sets based on how many common authors are present [8]: A_1 the set of papers having exactly 1 common author, A_2 papers having exactly 2 common authors and A_3 papers having 3 or more common authors. For a paper B:

$$A_1 = \{\text{B and co-papers of B with 1 common author}\}$$
$$A_2 = \{\text{B and co-papers of B with 2 common authors}\}$$
$$A_3 = \{\text{B and co-papers of B with 3 or more common authors}\}$$

$Collab_{Score}[B] = s_1 \text{ x } \max(A_1) + s_2 \text{ x } \max(A_2) + s_3 \text{ x } \max(A_3)$, where $\max(A_i)$ = highest score of all papers belonging to set A_i.

The weights s_1, s_2 and s_3 are chosen such that $s_1 < s_2 < s_3$ *and* $s_1 + s_2 + s_3 = 1.0$. This ensures that the major part of the $Collab_{Score}$ is determined by the best paper in the set where most authors of the paper have collaborated. If, in case the paper B is itself the best paper that they have produced (in all the three sets), then the $Collab_{Score}$ will be same as the paper's own score, thereby ensuring that the value of such papers is not diminished, i.e., the minimum score of the $Collab_{Score}$ is same as the score of the paper. Table 2 shows values of s_1, s_2 and s_3 for different numbers of authors.

Table 2. Values of s1, s2 and s3

Number of authors	***s1***	***s2***	***s3***
1 author	1.0	0.0	0.0
2 authors	0.25	0.75	0.0
3 or more authors	0.0625	0.1875	0.75

These computations can be represented in matrix form as: $X = C_1 \times S_1 + C_2 \times S_2 + C_3 \times S_3$, where

$$C_x[i,j] = \begin{cases} = 1 \text{ if paper } i \text{ is the best paper of paper } j \text{ in the set } Ax(x = 1, 2 \text{ or } 3) \\ = 0 \text{ otherwise} \end{cases}$$

$$S_x[i,j] = \begin{cases} 0, \text{ if } i \neq j \\ s_x \text{ from Table 4 based on \# of auth of paper } i. \end{cases}$$

Note that this matrix changes in every iteration as the scores of the papers change. The changes in the entries are proportional to the convergence. The influence matrix I_1 was static whereas this matrix is dynamic.

D. Weights

As mentioned earlier, it is preferable to have α_0 greater than α_is and β to ensure that in-links dominate over ownership. In this application, we can say we prefer citations over authorship, i.e. higher score of papers shouldn't be attributed to quality of authors but rather to the quality of citations received by the papers. Secondly, we chose a higher value for β than α_1 as we

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

wanted to give a higher weight to the collaboration factor than ownership. The values that we chose are:

$$\alpha_0 = 0.7\,, \alpha_1 = 0.1,\ \ \beta = 0.2$$

Results and qualitative analysis are provided in section VII.

VI. Algorithm Internals: Mathematical Exposition

The following exposition gives us an insight into the internals of the algorithm showing the evolution of the matrix. The equation for modification as explained in (7b) considering $(j+1)^{th}$ iteration is:

$$R_P^{j+1} = \alpha_0 R_P^{j+1} + R_P^j \sum_{\forall\, i\, \epsilon\, NP} \alpha_i M_i I_i + \beta R_P^{j+1} X$$

From (4), we can rewrite the above as (considering iteration numbers as well):

$$R_P^{j+1} = \alpha_0 R_P^{j+1} + R_P^j H \sum_{\forall\, i\, \epsilon\, NP} \alpha_i \mathrm{O}_{\mathrm{Pi}} I_i + \beta R_P^{j+1} X$$

Using (6), we can substitute R_p^{j+1} on RHS as $R_P^{j+1} = R_P^j \times G$:

$$R_P^{j+1} = R_P^j \times \left(\alpha_0 G + \mathrm{H} \sum_{\forall\, i\, \epsilon\, NP} \alpha_i O_{Pi} I_i + \beta G X \right)$$

Note that if we had used R_P^{j+1} instead of R_P^j for computing influences, we'd have had a H^2 term associated with the second term; this causes R_i to be ahead by one time-step. Hence, we use R_P^j for computation of R_is. We can simplify this using (1) as:

$$R_P^{j+1} = R_P^j \times \left(d \times \left(\alpha_0 H + H \sum_{\forall\, i\, \epsilon\, NP} \alpha_i O_{Pi} I_i + \beta H X \right) + \frac{1-d}{N} \alpha_0 E + \frac{1-d}{N} \beta E X \right) \qquad (9)$$

We can rewrite this as:

$$R_P^{j+1} = R_P^j \times \left(dH' + \frac{1-d}{N} (\alpha_0 + \beta) E \right) \qquad (10)$$

by replacing $\alpha_0 H + H \sum_{\forall\, i\, \epsilon\, NP} \alpha_i O_{Pi} I_i + \beta H X$ with H' and simplifying $E \times X$ as E because matrix X is column-stochastic i.e., sum of all columns of X is 1. Thus, we can now see how the stochastic matrix evolves as the ranks change. We also see that the probabilities of jumping to any of the nodes is equal$\left(= \frac{1-d}{N} (\alpha_0 + \beta) \right)$. The matrix H' of this iteration $(j+1)$ will be the H of next iteration $(j+2)$. Also, by substituting (9) in (5), we can see that the non-prime entities get their ranks based on all the other entities.

VII. Results

We executed our algorithm on different configurations of the parameters α_0, $\alpha_1$and β. We describe four important configurations here. The results in the form of author ranks and paper ranks are shown in Tables 3 and 4 respectively for the four cases:

Case 1: The configuration followed in this case is α_0=1, α_1=0 and β=0. This configuration is same as computing the scores of papers using normal page rank algorithm and then computing the scores of the authors using these. We can see that the ranks of authors and papers are linearly related.

Case 2: The configuration followed in this case is α_0=0.7, α_1=0.3 and β=0. This configuration considers the scores of the owning authors along with the citations a paper has got for computing the ranks. We observe that top six authors remain in the same positions. However, authors like LK, NH and MH have got a higher rank than previous configuration. The reason is they've been cited by many papers authored by ACNN. However, in case 1, he didn't receive much credit because these citations were considered as self-citations. But, in this case, even though the citations were considered self-citations, each of the citing paper's score itself was enhanced by ACNN's and his co-authors' scores, which caused them to move higher up in the ranking order. Also, paper B which is cited by paper A, got 1 rank lower than paper C, which is cited by many more papers having ACNN as author.

Case 3: The configuration followed in this case is α_0=0.7, α_1=0 and β=0.3. This configuration considers the collaboration scores of every paper along with the paper's own score. Here, there's no direct coupling between author and paper scores. However, the collaboration matrix brings in the coupling between the two entities. At the very first look, we find that the top ranking author is now IHW instead of ACNN. IHW's paper D has been cited by ACNN and friends who have maintained a consistent top record in all their papers. Hence, he's got a higher score. We find that this ranking order is little too strict given the strict nature of the technique used for computing the collaboration scores.

Case 4: The configuration followed in this case is α_0=0.7, α_1=0.1 and β=0.2. In this case, we combine the good features of case 2 and case 3. We chose to give a little higher share to the collaboration scores of the paper than the owning author scores as we get a more meaningful ranking order using this.

We analyzed the results qualitatively and saw how our algorithm could factor in collaboration and dependence between inter-entity scores into ranking. For a more complete evaluation, we computed h-index - a popular metric used for the

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

purposes of ranking in the problem domain chosen - of all the authors within our data set and compared it with the author ranking our algorithm produces. We generated the score-distribution graphs for both page rank and h-index for comparison. While we plotted the h-index scores as they were, we scaled the page rank scores 100 times and considered top 80 percentile scores for ease of visualization. Figure 2 shows the distributions of h-index and page-rank when applied on our data set.

Table 3. Top 24 Author Ranks for the Four Cases

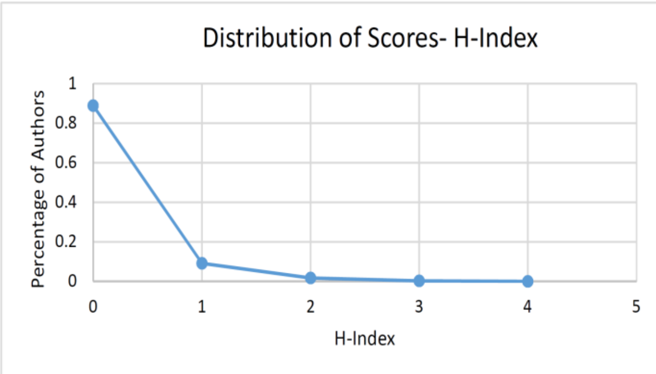
	Author	Case 1 Rank	Case 2 Rank	Case 3 Rank	Case 4 Rank		Author	Case 1 Rank	Case 2 Rank	Case 3 Rank	Case 4 Rank
1	ACNN	1	1	7	1	18	MAM	18		20	
2	MV	2	2	13	2	19	CG	19		21	
3	SA	3	3	8	6	20	MS	20	22	18	22
4	JL	4	4	9	7	21	RN	21	16	11	17
5	AZ	5	5	10	8	22	DMH	22	24	3	9
6	KL	6	6	12	5	23	RB	23		4	10
7	IHW	7	10	1	3	24	HH	24		5	11
8	DM	8	11	2	4	25	CU		13		23
9	LK	9	7		13	26	PC		14		24
10	NH	10	8		14	27	VL		15		
11	MH	11	9		15	28	DG		17		
12	VC	12	12		16	29	H		21		21
13	BH	13	18	14	18	30	AM		23		
14	WR	14	19	15	19	31	PM			6	12
15	JN	15	20	16	20	32	GDG			22	
16	MH2	16		17		33	OM			23	
17	JDF	17		19		34	DC			24	

Table 4. Top 10 Ranks of Papers in the Four Cases

Paper	Authors	Case 1 Rank	Case 2 Rank	Case 3 Rank	Case 4 Rank
A	SA, JL, AZ, ACNN	1	1	3	1
B	MV	2	3		5
C	ACNN, KL	3	2	4	3
D	DM, IHW	4	6	2	4
E	ACNN	5	4		6
F	ACNN	6	5		7
G	ACNN, LK, NH, MH	7	8		8
H	ACNN, KL, VC	8	9		9
I	JDF, MAMP, CG	9		7	
J	DC, GDG, DL, ML	10		5	
K	DG, ACNN		7		
L	RB, HH, DMH, PM		10	1	2
M	BH, WR, JN, MH2			6	10
N	MS			8	
O	PC, DS, SP, TC			9	
P	OM			10	

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618



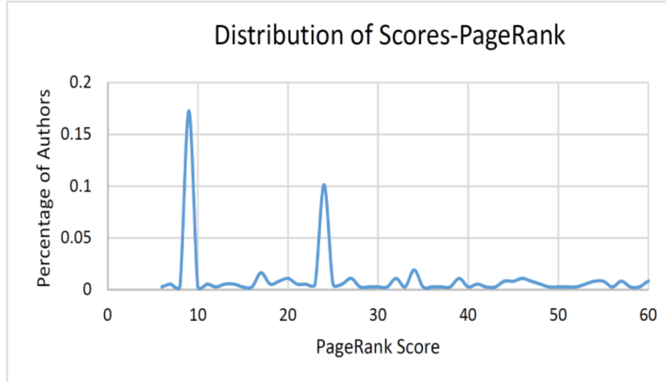


Fig. 2. Distributions of h-index and multi-entity page-rank scores

A clear observation is that h-index maps all 1,801 authors to just 5 distinct h-index scores, whereas page-rank assigns a larger distribution of scores to the authors. This can be attributed to the fact that the proposed model can account for factors like collaboration and inter-entity dependence and give a finer score allocation and method or ranking. We also report a 75% increase in the number of distinct scores when multi entity page rank algorithm is used in comparison to the original page rank (got by using the configuration defined in Case 1). From the h-index distribution, we can infer that the data set under consideration has relatively newer authors and their papers. Thus, the model which we have proposed considers multiple factors and is capable of producing a finer and practical ranking order even in cases where h-index fails to provide a clear distinction among the authors.

The results show that our multi-entity page-rank algorithm, while accounting for a number of factors as against conventional page rank algorithm, can produce practical and meaningful ranking orders of multiple related entities.

VIII. Discussion

As we had earlier mentioned, the semantics of the ranking is defined by the way we choose the prime entity and the relations between them. For example, in the university ranking problem, if we choose the prime entity to be papers and consider the citation graph, the results are purely in the academic perspective. That is, works which are well-cited have a positive influence on the university ranks and once, universities which have produced more novel works get a higher rank. The same logic holds for professors as well. However, if we considered a directed graph of universities, where the links indicate that professors who've obtained PhDs from one university serve for the other, universities professors from good universities or both, will get a higher rank. This graph can be inverted to mean that universities producing lot of good professors get a good rank. As another example, we could also consider the undirected graph denoting collaboration between the professors (or universities) as the prime entity graph. Yan and Ding in their work [7] have reported good results by applying page rank on an undirected graph of authors denoting co-authorship**.**

Given the variety of choices that one could make, the success of the algorithm depends on how the influence matrices and collaboration matrix are defined in consideration of the available data and the semantics of the domain.

Conclusion

In this paper, we have proposed a generic mathematical model of a multi-entity ranking algorithm which employs basic concepts of page rank, whose parameters can be set appropriately depending on the application. We've also provided a mathematical derivation showing the internals of the algorithm which is important in designing a successful ranking model. We have given examples throughout the paper illustrating the use of various parameters, which aid in designing a model which is well-suited to the application at hand.

References

[1] Brin, S., and Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer networks and ISDN systems*, 30(1), 107-117.

[2] Laure Soulier, Lamjed Ben Jabeur, Lynda Tamine and Wahiba Bahsoun (2012). On Ranking Relevant Entities in Heterogeneous Networks Using a Language-Based Model. *Journal of the American Society for Information Science and Technology*. Volume 64, Issue 3, pages 500-515.

[3] Ming Zhang, Sheng Feng, Jian Tang, Bolanle Ojokoh and Guojun Liu (2011). Co-Ranking Multiple Entities in a Heterogeneous Network: Integrating Temporal Factor and Users' Bookmarks. *Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation: 13th International Conference on Asia-Pacific Digital Libraries*, (ICADL 2011), pages 202-211.

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.

ISSN 2395-8618

[4] Tehmina Amjad, Ying Ding, Ali Daud, Jian Xu, Vincent Malic(2015). Topic-based Heterogeneous Rank. *Scientometrics*. Volume 104, Issue 1, pages 313-334.

[5] Yan, Erija, Ding, Ying and Sugimoto, Cassidy R. (2011). P-Rank: An indicator measuring prestige in heterogeneous scholarly networks. *Journal of the American Society for Information Science and Technology*. Volume 62, Issue 3, pages 467–477.

[6] Zhou, Ding, Orshanskiy, Sergey, Zha, Hongyuan and Giles, C. Lee (2007). Co-Ranking Authors and Documents in a Heterogeneous Network. *IEEE International Conference on Data Mining* (ICDM 2007), pages 739-744.

[7] Yan, Erija, and Ding, Ying (2011). Discovering author impact: A PageRank perspective. *Information processing & management*, 47(1), 125-134.

[8] Chandramouli Shama Sastry, Darshan S. Jagaluru, and Kavi Mahesh (2016) Author ranking in multi-author collaborative networks. *COLLNET Journal of Scientometrics and Information Management*, 10(1), 21-40.

IMPORTANT: This is a pre-print version as provided by the authors, not yet processed by the journal staff. This file will be replaced when formatting is finished.