AuthorRank: A New Scheme for Identifying Field-Specific Key Researchers

Ya-Chen Chang
Institute Center for Smart and Sustainable Systems (iSmart)
Masdar Institute of Science and Technology
Abu Dhabi, UAE
ychang@masdar.ac.ae

Zeyar Aung
Institute Center for Smart and Sustainable Systems (iSmart)
Masdar Institute of Science and Technology
Abu Dhabi, UAE
zaung@masdar.ac.ae

Abstract

When navigating into a new research field, it is important to identify papers with greatest impact and prominent authors which we can refer to. This work is motivated by the need to identify key authors in research fields. Traditional indices such as *h*-index only show the overall performance of an author. However, researchers generally contribute to more than one fields of research in their career, which makes it impractical to use *h*-index for identifying a key researcher in a research field. In this paper we propose a new PageRank-based scheme named “AuthorRank” for identifying key researchers in a specific field. We show that the proposed ranking system performs better than *h*-index does.

Keywords

Author Ranking, Publication Ranking, IEEE Xplore, PageRank.

AuthorRank: A New Scheme for Identifying Field-Specific Key Researchers

1. Introduction

With the flourishing development of the Internet and web technology, nowadays people tend to search for information from the Internet. Since a large number of resources would be matched to a given query, ranking technique is crucial for all kinds of search engines. When it comes to academic search, it has some characteristics which are different from web page search. One of the characteristics that we should be considering when implementing ranking techniques is that a paper can only cite papers published earlier than it, and the citations could not be modified after publishing. This causes latest papers hardly get high ranking in citation-based ranking system. Another characteristic is that except for citations, there are various information could be considered using as ranking factors, such as titles, abstracts, reference, authors, journals and vice versa.

Several algorithms have been proposed to improve the way paper is ranked. Some of them are inspired by the famous PageRank algorithm (Page et al., 1999), while others consider properties like author-paper relationship (Nie et al., 2005; Yan and Lee, 2007). However, currently neither search engines nor the online databases in the library could tell which author is more important from the others in a specific field. There are works been done to propose new algorithm for author ranking in either micro-blog (Kong and Feng, 2011) or question answer portal (Chen and Nayak, 2008), but little work has been done in the academic fields.

Identifying key authors in research fields help researchers to find important paper more quickly and give researchers a general sense on which opinion leader in that field is. Therefore, it is crucial to find out proper methodology for ranking authors in the target field for academic communities. In this paper, a PageRank-based technique is introduced. The core idea is that an author's rank should be calculated from both the impact of his paper and the order listed in the author list.

The paper is organized as follows: related works on this topic are presented in Section 2. In Section 3 we proposed a field-specific author ranking algorithm, and details of the implementation are described in 4. In Section 5 we present the experiment result and evaluate its performance. Section 6 concludes the paper with future work.

2. Related Work

There are a lot of similarities between ranking web pages, ranking researchers in academic field and ranking key opinion leaders in social network. All of them have citation (retweet) relationship which can be transformed into a directed graph. Link analysis algorithms like PageRank and Hypertext Introduced Topic Selection (HITS) (Kleinberg, 1999) are the most popular algorithm for ranking webpages. These algorithms are later used for identifying the key opinion leader in social networking or academic paper ranking. For example, IP-Influence (Romero et al., 2011), considering passivity of users, is a HITS-based approaches, while the variants of PageRank, such as TunkRank (Tunkelang, 2009) and TwitterRank (Weng et al., 2010) are based on a user graph which is constructed according to following relationships in twitter. In addition to user graph, some researches use a user-tweet graph, which emphasizes real interaction between users and tweets, like TURank (Yamaguchi et al., 2010) and topic-specific author ranking algorithm (Kong and Feng, 2011). Link analysis also works well on expertise analysis if users in a question answering system behave properly (Chen and Nayak, 2008).

Since author ranking in academic communities is based on the contribution of their published papers, it would be helpful for examining paper ranking algorithm to get some idea for building the author ranking algorithm. There have been several algorithms consider not only the number of citations but also other factors which could be used to measure the quality of the paper. For example, Authority-Based Ranking (Hristidis et al., 2008) determines ranking by simultaneously taking citations, authors, publication venues, and relevance to queries into account. Both PopRank (Nie et al., 2005) and Browsing-Based Model (Chen and Nayak, 2008) utilize the author-paper relationship. The difference between them is that PopRank considers the publication venue-paper relationship apart from citations and the author-paper relationship. To give recent papers a fair credit, there are researchers defined the age damping factor, which consists of decay time and the age of paper, for the papers in their proposed algorithm (Hwang et al., 2010). CiteRank modifies PageRank algorithm by initially distributing random surfers exponentially with age, in favor of more recent publications (Walker et al., 2007).

To the best of our knowledge, there is only little work done on author ranking in the academic communities, especially for identifying key authors in a specific field. So far, the most famous method for measuring the productivity and impact of the publication works of a scholar is *h*-index. The basic idea of *h*-index is that a scientist has index *h* if *h* of his or her N_p papers have at least *h* citations each, and the other (N_p - *h*) papers have fewer than *h* citations each (Hirsch, 2005). In recent studies, *h*-index has been proved to be valid (Bornmann and Daniel, 2006; Bornmann et al., 2008). However, *h*-index has several shortcomings, such as its weakness to differentiate between significant works in the past (but not anymore) and the works which are trendy (Sidiropoulos et al., 2007) and its tendency to put newcomers at a disadvantage (Cronin and Meho, 2006; Glanzel, 2006). Since *h*-index is more about the productiveness of a scientist, there are several other indices like *a*-index (Jin et al., 2007), *m*-index (Bornmann et al., 2008) and *hw*-index (Egghe and Rousseau, 2008) proposed to address the peer assessment. Therefore, some researchers recommend a combination of indices be used for evaluative purposes (Bornmann et al., 2008; Jin et al., 2007; Liu and Rousseau, 2007).

However, *h*-index and its variants all overlook the impact of the order of the author in the author list and none of them can be used to find important author in a specific field. To solve these problems, Osaka et al. (2012) proposed a matrix called Author and Paper Matrix (APM). The APM is shown as a direct graph which consists of a matrix of nodes. Each node, generated from a paper, represents the binding of the paper and an author of that paper. The co-author and related-paper are factors to determine edges and weight of edges between nodes. The lead author is considered to be most influential while the last author is deemed to be the one responsible for the paper. The weaknesses of this research are that the result of the ranking algorithm depends the search result of third-party searching engines, and they did not consider the time factor, such that paper publish long time ago tend to have higher score then recent published papers. In this work we tried to solve the above problem be introducing a normalization of time factor.

The lack of related research motivates us to develop a new technique for finding out the author ranking in a research field. In our approach, we consider both the impact of the paper and the order of the author to get an accurate author ranking. Since link analysis has good performance on author ranking in both social networking and question answering portal, we first use PageRank algorithm to address the citation relationship in order to measure the impact of the paper. Then, we add a parameter to take time effect into account because paper can only cite papers published earlier than it, which would underrate the value of recent papers. The author ranking is based on the paper rankings, and each author would gain different scores depending on their order in each paper. With this, we developed a field-specific author ranking algorithms that could extract key authors from a selected research field.

3. Model and Methodology

The core idea of our methodology is that an author's ranking should be calculated from both the impact of his paper and the order listed in the author list. In the following subsections we address the details of the methodology.

3.1 Ranking Algorithm

To generate a list of important authors in the target field, it is important to figure out a reasonable method to extract all the related papers in a field for further process. One way to classify the topic of the paper is by keywords. Therefore, in this study, we first choose a target field, and then we select a set of keywords which best describe the target field. By filtering the papers which contains the selected keywords, we can generate a list of paper related to the target field.

Next, we rank the selected papers by their relative importance. In the field of paper ranking, most of the proposed algorithms consider many other factors such as author and journal or time. For example, Walker et al. (2007) proposed a technique which considers the publication time of a paper as a factor. However, these methods often require additional information such as conference and journal ranking, making it harder to implement. In our work we use a modified-PageRank algorithm as the ranking algorithm. Since the citation relationship of the papers can be viewed as a directed graph, we can apply PageRank algorithm easily without extra information other than citation relationship. The original PageRank algorithm is described as follows:

$$PageRank_i = 1 - d + d \sum_{j \in B_i} \frac{PageRank_j}{O_j} \text{ , where:}$$

- $i = 1, 2 \ldots, n$: n is the total number of papers in the dataset.
- d: a damping factor which can be set between 0 and 1. It is usually set to 0.85 according to the researchers who proposed PageRank algorithm (Brin and Page, 1998).
- N: the total number of papers in the network.
- $j = 1, 2 \ldots, n$: n is the total number of papers citing paper i.
- B_i: the set of papers citing paper i.
- O_j: the number of outgoing links from paper j.

However, the publication time of a paper greatly affects the number of citation a paper has. A paper published earlier tends to have more citation than paper published recently. In order to compensate for this effect, we divide the original PageRank value by a time factor in deriving a publication rank (PubRank):

$$PubRank_i = \frac{PageRank_i}{\log(year_c - year_i)} \text{ , where:}$$

- $i = 1, 2 \ldots, n$: n is the total number of papers in the dataset.
- $PubRank_i$: paper $i's$ paper ranking.
- $PageRank_i$: a value counted by PageRank algorithm for paper i.
- $year_c$: current year (e.g., 2015)
- $year_i$: paper $i's$ publication year.

In PubRank, each paper $i's$ ranking is its PageRank value divided by a logarithm value of current year minus paper $i's$ publication year. By dividing the time factor, we get a hopefully time-independent ranking of the papers. In our experiment this method generates good results in that we find important papers published recently with rank higher than those published very long ago.

After calculating the PubRank of the select papers of the target field, we can calculate a rank for the authors involved in these papers. The AuthorRank algorithm is describes as follows:

$$AuthorRank_a = \sum_{p \in paper(a)} PubRank_p \cdot w_{p,r} \text{ , where:}$$

- $a = 1, 2 \ldots, n$: n is the total number of authors in the specific topic.
- $paper(a)$: author $a's$ number of papers in the specific topic.
- $p = 1, 2 \ldots, n$: n is author $a's$ total number of papers in the specific topic.
- $PubRank_p$: a value counted by PubRank algorithm for paper p.
- $w_{p,r}$: author $a's$ weight in paper p, which $\sum w_{p,r} = 1$. The value of the weight depends on author $a's$ order r of the authorship in paper p. There are more details about $w_{p,r}$ in the next section.

In AuthorRank, we first multiply paper $p's$ PubRank value with author $a's$ weight in paper p to get a value representing author $a's$ score for paper p, and then we sum all values getting from author a to get author $a's$ total score. The final score represent the author ranking of a specific author in a target field.

Summarizing the algorithm, the algorithm is composed of three parts:

- Extract related papers for a specific field using a group of keywords which best describes a field.
- Calculate the modified-PageRank on the extracted papers.
- Weighted-sum the PageRank of all the papers an author published. A weight is determined by the order of the author in the author list.

3.2 Keyword Selection

The task of identifying keywords which best describe a target field is a critical problem in that it seriously affect the field of the selected paper. In this work we either manually select the keywords by ourselves or use the top keywords provided by websites such as Google Scholar or Microsoft Academic Search. However, it is possible to develop a clustering-based algorithm for identifying top keywords for describing a field. Also keywords usually have synonyms, i.e. two different keywords conveying the same concept. When selecting keywords, we should also include the keywords with similar meaning, or else we may miss papers that use different keywords than those we selected.

4. Implementation

4.1 Dataset Attributes

In our experiments, we used IEEE Xplore Digital Library data, which was crawled in April 2014. The crawler, written in Python, sends a POST request to IEEE Xplore Digital Library in order to get the BibTex and citation information. The data downloaded from webpages would be first parsed to JSON format, and then would be stored in MongoDB. Note that the citation relationship in the database forms a strongly connected component (SCC), so all the cited paper can be found in the same database. The basic information of the dataset is shown in Table 1.

No. Papers	No. Authors	No. Keywords
1,428,568	1,023,544	799,888

Table 1: Characteristics of IEEE Xplore Digital Library (as of April 2014).

In order to have a better understanding for the data, Figure 1 shows the distribution of the number of times a paper is cited. As shown in the figure, most of papers have been cited at least one time. The average number of citations per paper is 4.42. Figure 2 shows the distribution of the number of authors per paper. Most of papers have two authors. The average number of authors per paper is approximately 3.

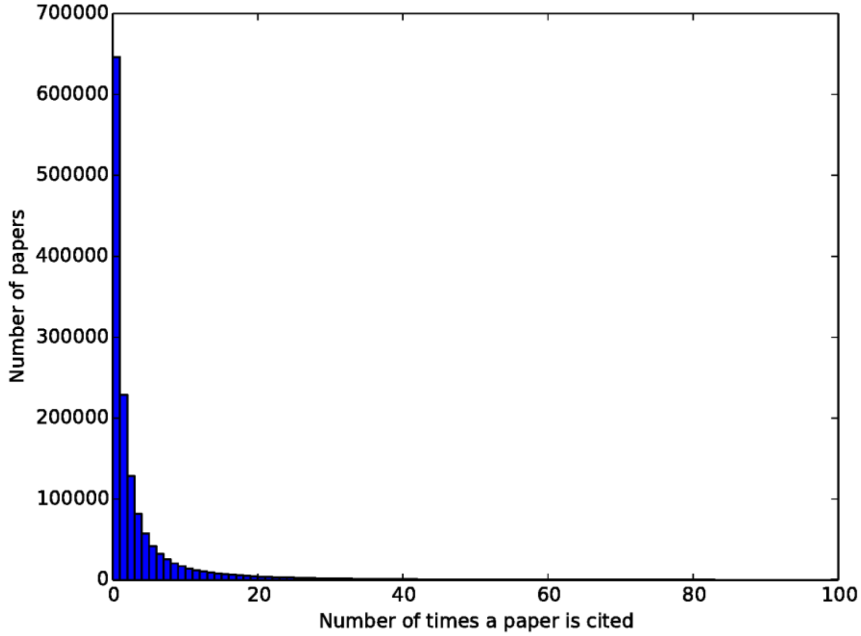

Figure 1: The distribution of the number of times a paper is cited.

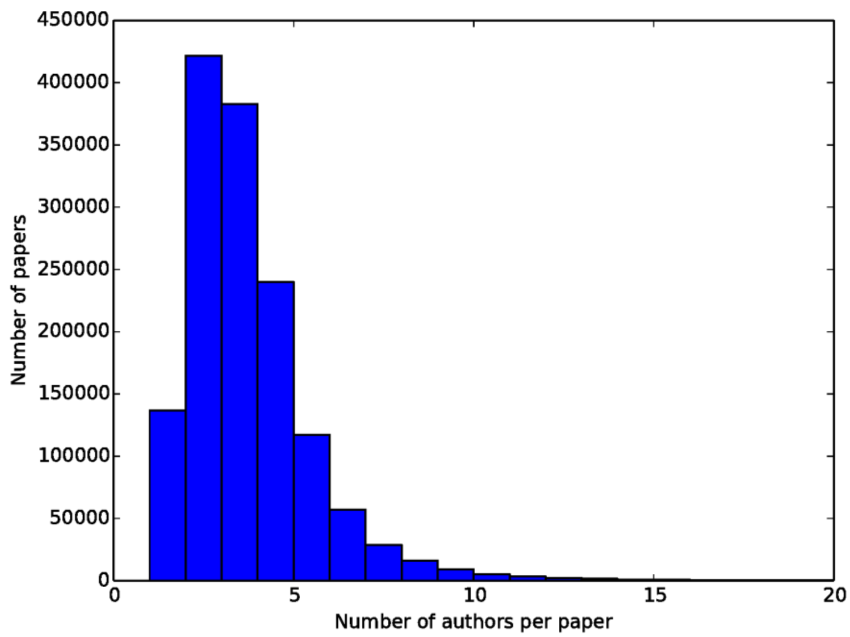

Figure 2: The distribution of the number of authors per paper.

4.2 Keyword and Parameter Selection

Since our purpose is to find out a list of key authors in a specific field, we selected three fields of study including Computer Vision, Operating System and Networking to measure the author ranking algorithm. A set of keywords defined as being relevant to a field of study is generate from Microsoft Academic Search, which has a page listing top keywords in a specific field of study. We collect top 13 keywords for each field. The top 13 keywords in three fields are shown in Table 2. These keywords are used to select papers for each field. We implemented all the ranking algorithms to these papers to generate a list of important author in the target field.

Operating Systems	Networking	Computer Vision
Operating Systems	Computer Network	Computer Vision
Storage System	Ad Hoc Network	Three Dimensional
Distributed System	Sensor Network	Feature Extraction
File System	Wireless Network	Image Features
Distributed File System	Wireless Communication	Image Processing
Virtual Machine	Communication System	Image Sequence
Virtual Memory	Peer To Peer	Image Segmentation
Shared Memory	Network Topology	Image Registration
Fault Tolerant	Quality of Service	Face Recognition
Real Time	Signal To Noise Ratio	Object Recognition
Scheduling Algorithm	Congestion Control	Object Detection
Energy Efficient	Fading Channel	Edge Detection
Priority	Routing Protocol	Optical Flow

Table 2: The top 13 keywords in three selected fields.

	1^{st}author	2^{nd}author	3^{rd}author	4^{th}author	n^{th}author
1 author	1.0	-	-	-	-
2 authors	0.7	0.3	-	-	-
3 authors	0.6	0.3	0.1	-	-
4 authors	0.45	0.3	0.15	0.1	-
n authors	0.45	0.3	0.15	0.05	$0.05/(n-4)$

Table 3: The weighting factor table of the authorship.

Table 3 shows the weighting factor of the authorship. The weight is determined by the order of the author in the author list. We experimented on several combinations and finally get a reasonable weight for each author. Finally, we compare our result with h-index to test the performance of the author ranking algorithm.

4.3 Program Implementation

Since we are using MongoDB as our data store, all the data processing steps are written in Javascript with MongoDB's Javascript shell. The PageRank algorithm is implemented in the MapReduce (Dean and Ghemawat, 2008) model, which is natively supported by MongoDB. Since PageRank is an iterative algorithm that needs to be run iteratively until the result converges. We implement each iteration as a MapReduce task and run it iteratively until the L2-

norm of the difference between current and previous iteration is smaller than some ϵ value. In our implementation, ϵ is selected to be 0.001, where most PageRank can converge within 20 iterations.

5. Performance

We perform field specific author rank on three fields including “Computer Vision”, “Operating System”, and “Networking”. The reason for choosing system-related papers is because that IEEE Xplore contains mostly such papers. After calculate the ranking, we can generate a list of top 10 authors in each field, which is shown in Table 4.

Operating Systems	Networking	Computer Vision
Shin, K.G.	Beaulieu, N.C.	Chellappa, R.
Kim, K.H.	Giannakis, G.B.	Jain, A.K.
Stankovic, J.A.	Alouini, M.-S.	Zhang, D.
Baruah, S.	Jie Wu	Huang, T.S.
Tei-Wei Kuo	Zorzi, M.	Fessler, J.A.
Buttazzo, G.	Hanzo, L.	Kanade, T.
Kwei-Jay Lin	Yang Xiao	Kumar, A.
Son, S.H.	Xin Wang	Pitas, I.
Iyer, R.K.	Poor, H.V.	Bovik, A.C.
Jie Wu	Win, M.Z.	Aggarwal, J.K.
Burns, A.	Akyildiz, I.F.	Medioni, G.
Anderson, J.H.	Boukerche, A.	Unser, M.
Blough, D.M.	Niyato, D.	Bruzzone, L.
Lui Sha	Boche, H.	Ahuja, N.
Kopetz, H.	Lie-Liang Yang	Nayar, S.K.
Schmidt, D.C.	Gerla, M.	Chein-I Chang
Pradhan, D.K.	Wei Wang	Sarkar, S.
Ahmad, I.	Xuemin Shen	Bhanu, B.
Raynal, M.	Shamai, S.	Gamba, P.
Baruah, S.K.	Kumar, A.	Xiaoou Tang

Table 4: The top 20 authors in selected field generated by field-specific AuthorRank.

It is very hard to create a ground truth for author rank, since the relative importance of author is subjective. However, looking at the result we can still gain insights on the accuracy of the proposed technique. We can see from the table that results are quite accurate. For example, in the Operating System field, Shin, K.G. is known for his contribution in the real-time system domain; Baruah, S. is well known for his work in real-time system; Tei-Wei Kuo is also known for his work in real-time databases; Lui Sha is well-known for his Priority Ceiling Protocol; and other listed here are also well known for each of their contribution in the operating system field.

The ranking in the Computer Vision field, Chellappa, R. is famous for his work in computer vision and pattern recognition, Jain, A.K. is also very well known for his work in computer vision. Besides, Iyer, R. K. and Boukerche, A. were elevated to IEEE fellows in 2015, which indicates newcomers are not put at a disadvantage as *h*-index.

Looking at the result generated by our algorithm, we find that the results are quite accurate. Since the selection of keywords affects that extracted papers and thus seriously affect the final score of the author, more keyword combination can be tested. Also the weight assignment shown in Table 3 requires more tuning to achieve moderate values.

Operating Systems	Networking	Computer Vision
Wei Wang	Wei Wang	Wei Wang
Itoh, T.	Itoh, T.	Itoh, T.
Kumar, A.	Kumar, A.	Kumar, A.
Tanaka, T.	Tanaka, T.	Tanaka, T.
Wei Zhang	Wei Zhang	Wei Zhang
Fukuda, T.	Fukuda, T.	Fukuda, T.
Sato, T.	Sato, T.	Sato, T.
Suzuki, T.	Suzuki, T.	Suzuki, T.
Wei Li	Wei Li	Wei Li
Poor, H.V.	Poor, H.V.	Poor, H.V.
Sato, K.	Sato, K.	Sato, K.
Tanaka, K.	Tanaka, K.	Tanaka, K.
Hanzo, L.	Hanzo, L.	Hanzo, L.
Watanabe, K.	Watanabe, K.	Watanabe, K.
Jun Wang	Jun Wang	Jun Wang
Wang, J.	Wang, J.	Wang, J.
Mittra, R.	Mittra, R.	Mittra, R.
Lei Wang	Lei Wang	Lei Wang
Das, S.	Das, S.	Das, S.
Watanabe, T.	Watanabe, T.	Watanabe, T.

Table 5: The top 20 authors in selected field generated by *h*-index.

To illustrate how *h*-index performs badly on identifying important authors in a target field, we sort the list of field specific authors generated with our technique with regard to their *h*-index. The result is show in Table 5. As the reader may point out, the *h*-index rank of the three fields is complete the same. This is because if an author published a paper with one of the keywords we select, the author is included in that field. Most authors publish a large amount of papers, and they usually also cross over fields. In our case, these 20 authors all published some papers including keywords in all three fields; this is why they are all selected. The experiment shows that *h*-index is completely useless in identifying key authors in a field.

Note that in Table 5, Wei Wang appears to have the highest *h*-index in our system. Digging in to the raw data we found that it is because Wei Wang is a very common name, in the first 10 papers we looked at with this author's name on it, they are all different individuals. We did not remove it from the list because we want to show the calculated data in their original form. When counting *h*-index, it is hard to rule out with certainty that papers by a different scientist of the same name are entering into the calculation. For this reason, Bornmann and Daniel (2007) recommend calculating the *h*-index on the basis of a complete list of publications that is authorized by the scientist himself or herself, which is one of the disadvantages of *h*-index.

6. Conclusion and Future Work

In this paper we proposed a PageRank-based technique for identifying author ranking in a specific research field. The main difference from *h*-index is that not only can we specify a research field, but we also take the authors' order in a paper into consideration. The resulting ranking system shows much better performance than *h*-index.

A key step in our methodology is to extract field related papers from the paper database. When extracting field related paper, we need to identify a group of keywords which best describes the field. In this work we use the top-*K* keywords listed in Microsoft Academic Search system. However, the listing of Microsoft Academic Search may be inaccurate. Furthermore, keywords with similar meanings may not be selected in this way. In our future work, we should investigate clustering related algorithms and stemming algorithms to automatic group keywords with similar meaning together, so we will not miss any related paper in the filtering step. Also, a new method is required to identify a subset of keywords which can best describe the target research field.

References

Bornmann, L. and Daniel, H.-D. (2006). Selecting scientific excellence through committee peer review - a citation analysis of publications previously published to approval or rejection of post-doctoral research fellowship applicants. *Scientometrics*, 68(3), 427–440.

Bornmann, L. and Daniel, H.-D. (2007). What do we know about the h index? *Journal of the American Society for Information Science and Technology*, 58(9),1381–1385.

Bornmann, L., Mutz, R., and Daniel, H.-D. (2008). Are there better indices for evaluation purposes than the h index? a comparison of nine different variants of the h index using data from biomedicine. *Journal of the American Society for Information Science and Technology*, 59(5), 830–837.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7), 107–117.

Chen, L. and Nayak, R. (2008). Expertise analysis in a question answer portal for author ranking. *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, 1, 134–140.

Cronin, B. and Meho, L. (2006). Using the h-index to rank influential information scientistss. *Journal of the American Society for Information Science and Technology*, 57(9), 1275–1278.

Dean, J. and Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1), 107–113.

Egghe, L. and Rousseau, R. (2008). An h-index weighted by citation impact. *Information Processing & Management*, 44(2), 770–780.

Glanzel, W. (2006). On the opportunities and limitations of the h-index. *Science Focus*, 1(1), 10–11.

Hirsch, J. E. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National academy of Sciences of the United States of America*, 102(46), 16569–16572.

Hristidis, V., Hwang, H., and Papakonstantinou, Y. (2008). Authority-based keyword search in databases. *ACM Transactions on Database Systems*, 33(1), 11–14.

Hwang, W.-S., Chae, S.-M., Kim, S.-W., and Woo, G. (2010). Yet another paper ranking algorithm advocating recent publications. In *Proceedings of the 19th ACM International Conference on World Wide Web (WWW)*, 1117–1118.

Jin, B., Liang, L., Rousseau, R., and Egghe, L. (2007). The R- and AR-indices: Complementing the h-index. *Chinese Science Bulletin*, 52(6), 855–863.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5), 604–632.

Kong, S. and Feng, L. (2011). A tweet-centric approach for topic-specific author ranking in micro-blog. In *Advanced Data Mining and Applications,* 7120, 138–151.

Liu, Y. and Rousseau, R. (2007). Hirsch-type indices and library management: The case of Tongji University library. In *Proceedings of the 11th International Conference of the International Society for Scientometrics and Informetrics (ISSI)*, 11, 514–522.

Nie, Z., Zhang, Y., Wen, J.-R., and Ma, W.-Y. (2005). Object-level ranking: Bringing order to web objects. In *Proceedings of the 14th ACM International Conference on World Wide Web (WWW)*, 567–574.

Osaka, K., Huang, H.-H., and Kawagoe, K. (2012). Searching and ranking researchers by keywords based on author and paper matrix rank. In *Proceedings of the 2nd International Conference on Innovative Computing Technology (INTECH)*, 340–344.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford University.

Romero, D. M., Galuba, W., Asur, S., and Huberman, B. A. (2011). Influence and passivity in social media. In *Machine Learning and Knowledge Discovery in Databases*, 6913, 18–33.

Sidiropoulos, A., Katsaros, D., and Manolopoulos, Y. (2007). Generalized Hirsch h-index for disclosing latent facts in citation networks. Scientometrics, 72(2), 253–280.

Tunkelang, D. (2009). A Twitter analog to PageRank. The Noisy Channel Blog, http://thenoisychannel.com/.

Walker, D., Xie, H., Yan, K.-K., and Maslov, S. (2007). Ranking scientific publications using a model of network traffic. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(06), 6-10.

Weng, J., Lim, E.-P., Jiang, J., and He, Q. (2010). TwitterRank: Finding topic-sensitive influential Twitterers. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM)*, 261–270.

Yamaguchi, Y., Takahashi, T., Amagasa, T., and Kitagawa, H. (2010). TURank: Twitter user ranking based on user-tweet graph analysis. In *Web Information Systems Engineering – WISE 2010*, 6488, 240–253.

Yan, S. and Lee, D. (2007). Toward alternative measures for ranking venues: A case of database research community. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, 235–244.