



ELSEVIER

Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

Information Processing and Management 41 (2005) 289–312

www.elsevier.com/locate/infoproman

**INFORMATION
PROCESSING
&
MANAGEMENT**

A new perspective to automatically rank scientific conferences using digital libraries

Antonis Sidiropoulos^{*}, Yannis Manolopoulos

Data Engineering Lab, Department of Informatics, Aristotle University, Thessaloniki 54124, Greece

Received 21 April 2003; accepted 11 September 2003

Available online 14 October 2003

Abstract

Citation analysis is performed in order to evaluate authors and scientific collections, such as journals and conference proceedings. Currently, two major systems exist that perform citation analysis: Science Citation Index (SCI) by the Institute for Scientific Information (ISI) and CiteSeer by the NEC Research Institute. The SCI, mostly a manual system up until recently, is based on the notion of the ISI Impact Factor, which has been used extensively for citation analysis purposes. On the other hand the CiteSeer system is an automatically built digital library using agents technology, also based on the notion of ISI Impact Factor. In this paper, we investigate new alternative notions besides the ISI impact factor, in order to provide a novel approach aiming at ranking scientific collections. Furthermore, we present a web-based system that has been built by extracting data from the Databases and Logic Programming (DBLP) website of the University of Trier. Our system, by using the new citation metrics, emerges as a useful tool for ranking scientific collections. In this respect, some first remarks are presented, e.g. on ranking conferences related to databases.

© 2003 Elsevier Ltd. All rights reserved.

Keywords: Citation analysis; Ranking; Digital libraries; Impact factor

1. Introduction

Currently, two major systems exist that perform citation analysis: Science Citation Index (SCI) by Institute for Scientific Information (ISI) and CiteSeer by the NEC Research Institute. The idea of performing citation analysis has evolved in early 60's. Then, in 1972 it started to be used by the SCI for the evaluation of journals spanning many scientific fields, computer science included. The

^{*} Corresponding author.

E-mail addresses: asidirop@delab.csd.auth.gr (A. Sidiropoulos), manolopo@delab.csd.auth.gr (Y. Manolopoulos).

ISI impact factor (Garfield, 1972, 1994) was the main metric used by SCI for journal evaluation and ranking, a necessary task to make decisions on tenure, funding, salary levels etc. This notion is computed for each particular year during a journal's lifetime based on the number of citations made to the papers published in it during the previous k years, where k is usually equal to 2 or 5. (However, it can be computed for any k value, if necessary for a specific scientific domain or for a statistical purpose.) On the other hand, the CiteSeer system is a modern system and constructs the citation graph from publications acquired from the web (Lawrence, Giles, & Bollacker, 1999a). CiteSeer is also based on the ISI Impact Factor for ranking conferences and journals (Goodrum, McCain, Lawrence, & Giles, 2001).

Citation analysis is based on the notion of *citation graphs*, which are graphs representing papers as nodes, whereas an edge from node x to node y represents a citation from paper x to paper y . Citation graphs can be used to derive useful statistical information related to evaluating and ranking several entities, such as authors, publications in scientific conferences and journals, as well as conferences and journals as scientific collections.

In particular, citation graph analysis is similar to web-graph analysis. Notable is the PageRank algorithm by Brin and Page (1998), which is used by the Google search engine. This algorithm computes a score for a page as a summary of the fractions of the scores of the referrer pages. Thus, it ranks the webpages returned to the user according to their relevance to the user query. It has been derived that the statistical distribution of the PageRank metric follows the familiar inverse polynomial law reported for webpage degrees (Dhyani, Bhowmick, & Ng, 2002). Recently, the PageRank formula has been further analyzed (Pretto, 2002).

Except ranking, other operations can also be performed on citation graphs by using graph-theoretic and data mining techniques (An, Janssen, & Milios, in press). For example, assuming a set of scientific collections, related books, conferences, journals or/and technical reports—with regard to the specific area—can be categorized by using clustering. In an analogous manner, authors can be grouped in clusters in order to find and establish their communities, i.e. authors that co-operate and cite each other and, in an analogous manner, to find *hubs* and *authorities*, i.e. clusters of authors that mostly cite and mostly get cited, respectively. Two important works on this area are the Kleinberg's HITS (Hyperlink Induced Topic Search) algorithm (Kleinberg, 1999; Kleinberg, Kumar, Raghavan, Rajagopalan, & Tomkins, 1999), which computes a Weighted score for the above notions. A further study of the web as a graph and the hubs/authorities notions has appeared in Meghabghab (2002).

The structure of this paper is as follows. In the next section we review the SCI and CiteSeer systems and compare their advantages and disadvantages. Also, we review the literature on several efforts towards citation analysis and ranking specific scientific collections. In Section 3, we investigate new alternative notions besides the ISI Impact Factor, in order to provide a novel approach aiming at evaluating and ranking scientific journals and conferences. Also, we present the basic functionalities of a web-based system that is called *Scientific Collection Evaluator by using Advanced Scoring* (SCEAS). Our system has been built by extracting data from the Data Bases and Logic Programming (DBLP) website ¹ of the University of Trier. This system, by using the new citation metrics, emerges as a useful tool for ranking scientific col-

¹ <http://www.informatik.uni-trier.de/~ley/db/>

lections, such as conferences or/and journals. In Section 4, some first remarks are presented in the respect, e.g. about ranking conferences related to databases. The last section concludes the paper.

2. Major systems for citation analysis

As mentioned, currently there exist two major systems that perform citation analysis: SCI and CiteSeer. Here, we will examine closer these systems in order to see their “weak” points and motivate the research of this paper.

Beforehand, it is important to notice that SCI has served the whole academic community for several decades by providing useful information, in the lack of anything better. However, nowadays the system disadvantages and limits are apparent. For example, the main disadvantages of the SCI system are:

- (1) Each scientific field is divided in certain areas, which remain static over the years and do not reflect the scientific evolution and, in particular, the dramatic evolution of computer science.
- (2) In each area, only a set of journals is selected for journal evaluation. Thus, the representative value of the selected journals is questionable.
- (3) Although any such set is dynamic and updated periodically, this update is done in a subjective way, which might also trigger questions about when, why how and by whom.
- (4) In some cases, irrelevant journals (e.g. technical vs. popular) are grouped in a certain area leading to erroneous results.
- (5) Scientific conferences, books and technical reports are not taken into consideration.
- (6) It is/was manually constructed and, therefore is an expensive system to built and maintain.
- (7) It is not for free neither for libraries nor for individuals.

On the other hand, CiteSeer is a modern system and constructs the citation graph from publications acquired from the web (Lawrence et al., 1999a). More specifically, it is an autonomous system that collects computer science papers by crawling the web. Then, from the format each paper is stored (i.e. postscript or pdf), it detects and exports the bibliographic information (e.g. title, authors, etc.) as well as the included citations to construct the underlying citation graph (Lawrence, Bollacker, & Giles, 1999b). The advantages of the CiteSeer system are:

- (1) It is automatic and transparent; thus, it is objective as human intervention is limited.
- (2) It takes into consideration all kinds of scientific publications, books and technical reports. Under the current circumstances, this is quite important since we face a gradual and steady increase of the quantity of high quality scientific information disseminated via conferences, workshops, symposia and white technical papers.
- (3) It is sensitive to the important fact that there are some highly competitive conferences with acceptance ratio which is much higher than that of several relevant journal publications. For example, consider the cases of the ACM SIGMOD Conference on Management of Data

and the International Conference on Very Large Databases (VLDB), where during the last 20 years we remark a typical acceptance ratio ranging from 1:5 up to 1:7.

- (4) It uses the citation graph to perform several sophisticated tasks, such as: ranking search page results (based on the in-degree value of each match), investigate relevant papers (based on co-citation analysis), etc.
- (5) It is delivered for free through Internet, assisting thus the research and administrative work of the academic community.

However, the limitations of CiteSeer are:

- (1) It concerns only the field of Computer Science and, thus, it is not useful outside of this academic discipline.
- (2) It does not really focus on conference or journal evaluation/ranking. There exist only one ranking² in CiteSeer which (a) includes data from DBLP only, (b) mixes-up conferences and journals, and (c) groups together various scientific areas.
- (3) Finally, it is based on the notion of ISI Impact Factor that, although for a long period it has played an important role in evaluating journals (and, subsequently academic authors), it is quite rigid and cannot be used to perform deeper qualitative analysis.

The present work is motivated from the latter point. To comprehend the limits of the ISI Impact Factor, it is necessary to notice that this notion cannot be applied in the case that we want to rank/evaluate conferences for a specific year. More specifically, a journal is published several times a year. On the other hand, a conference is held once per year or even less often. In such a case, what should be the k value of the ISI Impact Factor for a conference? By using a k value we actually evaluate the conference for the period of k previous years. But how could we rank all the conferences that were held for example in 1996? Therefore, it is apparent that it is not safe to use the ISI Impact Factor for conference evaluation. In addition, one could argue against the “flat” nature of the ISI Impact Factor. For example,

- Is it fair to count a citation from Professor “Well-known” as equivalent to a citation from Professor “Unknown”?
- Is it fair to count a citation from Journal “The-top” as equivalent to a citation from Journal “The-bottom”?
- Or, finally, is it fair to count a citation from Paper “The-best” as equivalent to a citation from Paper “The-worst”?

From these simple questions, it is apparent that it is necessary to embed some kind of weighting to answer such questions. In this work we will investigate some new ideas for ranking scientific collections and we will try to put the task of citation analysis and journal or conference evaluation in a more generalized perspective.

² <http://citeseer.nj.nec.com/impact.html>

3. Literature survey

Except SCI and CiteSeer several other digital libraries and indexing systems exist, which perform some kind of citation analysis. As already mentioned, the DBLP website maintained since 1993 by Michael Ley at the University of Trier is a rich digital library focusing in the field of computer science, and in particular in the areas of Data Bases and Logic Programming (Ley, 2002). More specifically, as of March 2003, the DBLP website contains bibliographic data about 240 000 authors, 1250 conferences, 300 journals and 360 000 papers, articles or books, with links to personal pages, research groups, publishing houses, etc. Navigation through the DBLP content makes seeking information an easy task for the academics, researchers and professionals working in the areas of Data Bases and Logic Programming. The basic DBLP functionalities are author, conference, journal and term searching. In addition, for a great part of the indexed publications it provides full text retrieval and lists of citations. Although DBLP is mainly an indexing and searching system, it is interesting to note that it provides a ranked list of the most cited papers in the above areas of computer science. However, this ranked list is based on the DBLP data only, i.e. on the citation lists per publication, which are not exhaustive.

Recently, another indexing prototype has appeared in the literature (Bradshaw & Hammond, 2001), called *Rosetta*, which is a digital library system for scientific literature related to computer science articles. Rosetta indexes research articles based on the way they have been described when cited in other documents. The concise description that occur in citations is similar to the short queries people typically form when searching. Rosetta provides a user interface that presents users with an automatically generated directory of the information space surrounding the query. It is reported that the Rosetta corpus contains over 37 000 indexed documents using roughly 450 000 references.

Another interesting system is AuthorLink by Lin, White, and Buzydlowski (2003), which is a visualization prototype aiming at enhancing author searching. This is achieved by author co-citation analysis. In particular, given a query for a specific author, the system constructs interactive author maps in real time from a database of 1.26 million records related to Arts and Humanities and supplied by ISI. These maps contain the 24 authors that are mostly co-cited to the query name, along with some data about counts. The user by clicking to any of these 24 names can proceed and have other maps constructed. In fact this facility helps in several occasions during author searching in a specific narrow area.

Finally, PubSearch is a system developed by He and Hui (2002) aiming at illustrating author co-citation analysis by using a data mining methodology. The authors use a similar technique to that of CiteSeer to collect bibliographic data by crawling the web. The collected entries are inserted into a data warehouse and an agglomerative hierarchical clustering is performed to construct author maps, showing authors with similar interests to a given query name. The system is experimentally tested with data from the ISI Social Science Citation Index (SSCI). More specifically, 1466 Information Retrieval related papers, which appeared during the period 1987–1997 in 367 journals, with 44 836 citations were used for system evaluation.

Finally, in Ding, Chowdhury, and Foo (2001) a co-word analysis system is presented. The authors of this paper selected 2012 Information Retrieval related papers from SCI and SSCI, which appeared during the period 1987–1997, and extracted 193 keywords, with 5.09 keywords

per paper in order to perform co-word analysis and reveal patterns of the specific area evolution with time.

As mentioned, more or less the purpose of the above systems is indexing, citation analysis and visualization; however, their purpose is not ranking. With regards to ranking we meet several efforts in the literature spanning several disciplines except computer science.

For example, a citation analysis based ranking study (Baumgartner & Pieters, 2000) is reported for Marketing journals. In essence, the authors performed a manual extraction of citations of the 1996–1997 issues of 49 marketing journals (26 of them not contained in the SSCI), where the collections of the 49 titles was largely based on a survey. A non citation analysis based ranking was reported in Hult, Neese, and Bashaw (1997), where the authors rank marketing related journals according to the respondents affiliation (doctoral vs. non doctoral institutions) based also on a survey.

In Tahai and Rigsby (2002) the authors' concern is the area of Accounting. They obtained data from the SSCI (i.e. they extracted 351 articles from 8 journals published during the period 1992–1994 with 11 746 citations) and considered the notion of ISI Impact Factor and proposed variations of it based on the statistical distribution of the number of citations over time. In Korobkin (1999) the author criticizes the theory and methodology of ranking journals having in mind Law related journals.

Mylonopoulos and Theoharakis (2001) focused in the area of Information Systems and performed an on-line survey for 87 journals with 1000 respondents approximately. Although, this work is not a citation analysis based ranking study, it is interesting in the sense that it performs a ranking according to readers (and authors) perception as a function of the geographic location of the respondents.

Having the same motivation with our paper, Keijnein and Groenendaal focus in the area of Information Systems and try to rank journals, conferences and books, i.e. a larger set of publications in comparison to the SCI practice (Kleinjnen & Groenendaal, 2000). Using sampling, bootstrapping and classification, they classify their set of publications in six ranking categories according to the number of citations received. However, the limitations of this work are:

- it is restricted to a very specific domain of computer science,
- it is a manual method that uses only a small set of publications (e.g. only 123 journal articles and 82 proceedings articles), which collectively cite 6901 publications (i.e. 3128 journal articles, 1532 proceedings articles, 1577 books and 664 other publications),
- like SCI mixes up diversifying publications (e.g. technical vs. managerial),
- it uses the same notion with ISI impact factor. Notably, it uses an infinite time window (instead of $k = 2$ or 5 years as ISI proposes) in the course of Impact factor computation. This may give seriously misleading results by not capturing the dynamic nature of science evolution and author preferences, since a journal may be ranked high in the past, but low currently.

4. The SCEAS system

We built a system called *Scientific Collection Evaluator by using Advanced Scoring* (SCEAS). Our system imports DBLP XML records into a MySQL database system. We have used the

specific software since it is light, fast and fits our needs as the transactions sent to the server are mainly reads rather than updates. The SCEAS system is available on the Internet ³, and thus the user can easily access it, postqueries, get answers, and extract useful information.

In our model, the main entities are:

Publications which could be articles, in-proceedings (according to the latex terminology) etc.

Collections such as conferences, books, journals etc.

Persons which could be authors or editors.

Each publication belongs to a collection (or more, e.g. conference publications belong to a conference and to a proceedings collection). A collection may be a part of another one, e.g. VLDB'97 is a collection and it is part of the VLDB collection. Persons can be related to publications as authors or to collections as editors (e.g. for proceedings). Finally, publications can be related to each other with the “citation” relation.

Based on the DBLP database, we built the citation graph of the collection, which includes journal as well as conference publications. Using this graph we derived two Collection Citation Graphs, the Conference Citation Graph, and the Journal Citation Graph. In the same way, any other type of semantic grouping of the publications can be used to derive analogous citation graphs (e.g. Book Citation Graph). Fig. 1 is a small sample of a Conference Citation Graph. It consists of four nodes standing for the conferences SIGMOD, VLDB, PODS and ICDE of all years. The weight of the arc from node i to node j depicts the number of citations from all the publications of the first node to all the publications of the later node.

Scientific collection evaluation, and conference evaluation in particular, being our first concern, we tried to investigate alternative ways for such a task. The basic idea for the ranking, is that not all the citations should have the same weight. For example, the weight should depend on two factors: (a) the quality of the conference, where a citation to another conference is made, and (b) the scientific domain of the conferences, e.g. if they belong in the same domain. Thus, there exist

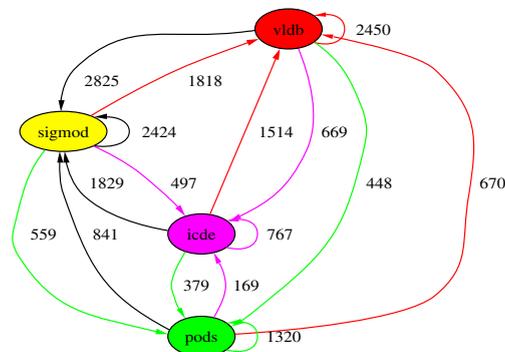


Fig. 1. Conference citation graph.

³ <http://delab.csd.auth.gr/sceas>

two tasks that should be done. First, we must specify the scientific domains and then perform the ranking.

In particular, we have performed the following tasks:

- (1) Cluster the conferences based on the conference citation graph after a preprocessing phase that uses some keywords appearing in the conference titles.
- (2) Cleansing, since the data of many conferences in the DBLP database are not complete. Therefore, we try to exclude the collection subset that inserts noise into our algorithms.
- (3) Finally, ranking each conference cluster separately. We have performed ranking by taking into consideration the whole lifetime of all conferences and every specific year of each conference. The former does not produce useful results from the statistics point of view, since there exist a lot of factors affecting ranking. Thus, we focused in the latter case and for every distinct year we produced:
 - rankings using Plain Scoring,
 - rankings using Weighted Scoring, and
 - rankings using the Inverted Impact Scoring,
 - rankings using Weighted Inverted Impact Scoring.

All these new notions introduced above will be explained in the sequel. Please, notice that ranking is performed by using several algorithms, which will be presented in the sequel as well.

4.1. Clustering conferences according to topics

Based on the conference citation graph (as the example of Fig. 1), first we performed a clustering operation. As a utility for clustering the conferences we used the hMetis (Han, Karypis, Kumar, & Mobasher, 1997; Karypis, Aggarwal, Kumar, & Shekhar, 1997), the leading hyper-graph partitioning tool for large hyper-graphs. hMetis has been successfully used in applications related to VLSI circuits, data mining and numerical analysis.

Fig. 2 shows an example of a conference citation graph, where nodes represent conferences, whereas edges represent citations. The edges are directed and weighted. An edge from node A to node B with weight w , means that there are w citations from publications of conference A to publications of conference B .

To perform clustering, we need to minimize the sum of the edge weights that cross from one cluster to another (this is computed by hMetis). For this purpose we do not need a directed graph. Thus, we convert the later to an undirected graph (not avoiding some loss of information). In our example, Fig. 3 is produced from Fig. 2. In this graph, the weight on an edge that connects two nodes represents the total number of citations that these conferences make to each other. The derived graph cannot be used for ranking, which is our main purpose, but only for the clustering step. Actually, this is the graph type that we feed to hMetis for the clustering.

Clustering results were not perfect, since part of the citation graph is incomplete (i.e. no citations are included in some DBLP records for a few conference publications). Therefore, we re-cluster the conferences, after performing some preprocessing based on keyword matches in the conference titles. In particular, we predefine 4 clusters:

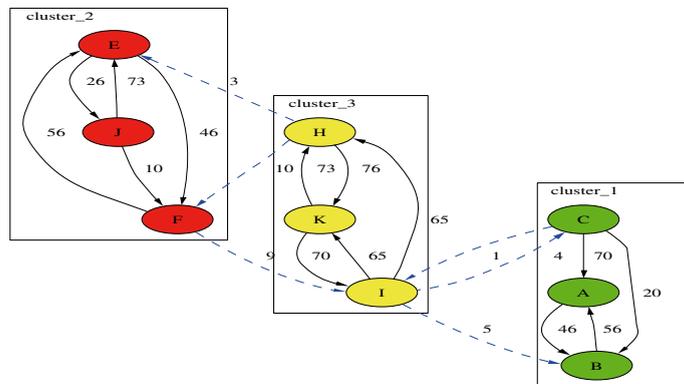


Fig. 2. Conference citation directed graph.

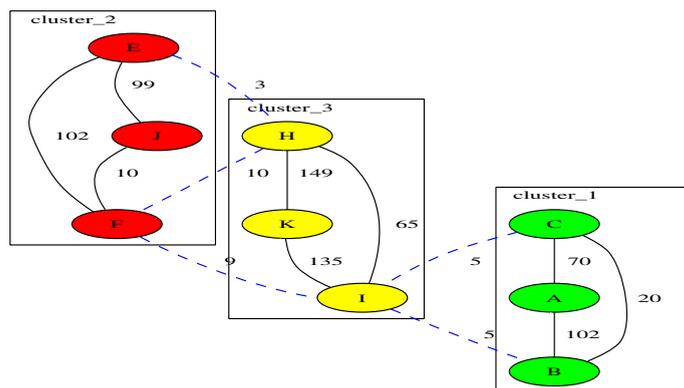


Fig. 3. Equivalent conference citation undirected graph.

- *Cluster 1: Databases.* The keywords used to identify the conferences that belong in this domain where: data base, database, digital library, information retrieval, information system, mining, and geographic.
- *Cluster 2: Logic Programming.* There were identified by the keywords: AI, artificial, intelligent, knowledge, logic, and algorithm.
- *Cluster 3: Networks and Distributed Systems.* distributed, network, parallel, web, www, and w3c.
- *Cluster 4: Operating Systems, Software Engineering, Compilers and Languages.* for the respective areas.

After defining the above 4 clusters with some conference members in them, we feed this pre-defined partitions to hMetis to continue with the unclustered conferences.

4.2. Cleansing the clusters

As mentioned above, the DBLP database is incomplete. For example, for some conferences or journals, 1 or 2 publications are only included (i.e. the most important ones). This would lead our

ranking algorithm to produce erroneous results, since the main metric is the average number of citations per publication.

In this step we exclude from the conference set, which will be used for ranking, the conferences that:

- contain less than 3 publications,
- are held only once, and
- have average number of publications per year less than 0.5.

For these conferences we set a flag meaning that they will not be ranked, but we do not delete them from our database. Thus, any citations included in them do count.

4.3. Definition of the metrics

Here, we introduce the new metrics in order to establish a new perspective for conference and journal evaluation using the citation graph. These metrics are defined as follows:

4.3.1. Plain Score

If C is the set of all the conferences, then the *Plain Score*, S_c , that is the *Score* for conference c , is defined as:

$$S_c = \frac{1}{P_c} \sum_{\forall i \in C} N_{i \rightarrow c} \quad (1)$$

where $N_{i \rightarrow c}$ is the number of citations made from conference i to conference c , whereas the normalizing factor P_c is the number of publications in conference c .

The rank is computed by ordering the conferences' scores. In case of a tie, the conference with the fewer publications precedes. Actually, this score is exactly the same as the in-degree of the corresponding node in the conference citation graph, divided by the number of publications included in this conference. This score is the simplest one, and it is basically used as a first approach for ranking. Actually, although this metric carries some information with respect to ranking, it has the disadvantage that conferences with “long” history are more likely to have more citations. Thus, it can only be used to compare and rank a set of conferences that have exactly the same life-time.

4.3.2. Weighted Score

Here we introduce the idea of the weighted ranking. This means that the citations do not count the same. Eq. (2) shows abstractly how the *Weighted Score* for conference c , defined as WS_c , can be computed.

$$WS_c = \frac{1}{P_c} \frac{\sum_i W_i * N_{i \rightarrow c}}{\sum_i W_i} \quad (2)$$

How can we compute the weights? Which conferences are “The-Top” that should have larger weights and which are the “The-Worst” conferences? Here, arises the need for recursive computing. This computation is performed by using the following formula:

$$\begin{aligned}
 WS_{c,l} &= \frac{1}{P_c} \frac{\sum_i W_{i,l-1} * N_{i \rightarrow c}}{\sum_i W_{i,l-1}} \quad l \geq 1 \\
 W_{i,0} &= 1 \quad \forall i \in C
 \end{aligned}
 \tag{3}$$

Initially all the weights are set equal to 1 (at level 0). Thus, we can compute the ranking for the next level, based on the weights computed in the previous one. The ranking we get at level 1, is equivalent with the Plain Score ranking (since we used weights of 1 for all entities). After computing the scores for level 1, we can compute the weights. This is achieved with another clustering algorithm. In Section 4.5 a detailed discussion on the computation of weights can be found.

After computing the weights for level 1, we continue computing the scores for the next levels by applying the same procedure until the ranking remains unchanged. This is our termination condition. The computation is repeated $\forall l \geq 1$ until L , where the ranking for level L is equivalent with that of the level $L - 1$. Alternatively, if while at level L we get the same weights as in level $L - 1$ ($W_{i,L} = W_{i,L-1} \forall i \in C$), then it is obvious that the ranking for $L + 1$ would be the same with the one computed at level L . Thus, an alternative stop condition is a “no change” in the computed weights. Then $\forall l \in \{L \dots \infty\}$ the condition: $WS_{i,l+1} = WS_{i,l}$ is true, and we set:

$$WS_c = WS_{c,\infty} = WS_{c,L}$$

This type of ranking, similarly to the Plain Score, cannot be used for conference evaluation without risk. Despite the refinement of computing the average score per publication by means of the citations’ weights, not all conferences have the same life-time, whereas some are only held only once per two or three years. Therefore, conferences with “longer” history are more likely to have more citations. This ranking can be used only for the conferences that have exactly the same life-time.

4.3.3. Plain Score per Year

Adapting the notion of the Plain Score in order to rank conferences for each distinct year, we introduce the *Plain Score per Year* metric as:

$$SY_{c,y} = \frac{1}{P_{c,y}} \sum_{\forall i \in C} N_{i \rightarrow c,y}
 \tag{4}$$

where $SY_{c,y}$ is the score for conference c in the year y , $N_{i \rightarrow c,y}$ is the number of citations from conference i to conference c that was held in year y and $P_{c,y}$ is the number of publications of conference c during the year y . In particular, a more detailed expression that we used for our computations is:

$$N_{i \rightarrow c,y} = \sum_{z=y}^{\text{last_year}} N_{i,z \rightarrow c,y} \stackrel{(4)}{\Rightarrow} SY_{c,y} = \frac{1}{P_{c,y}} \sum_i^{\forall i \in C} \sum_{z=y}^{\text{last_year}} N_{i,z \rightarrow c,y}
 \tag{5}$$

where $N_{i,z \rightarrow c,y}$ is the number of citations made from conference i in year z to conference c held in year y . The variable `last_year` is set to the maximum valid year in our collection (normally the current year). This ranking can be used to compare conferences that were held in the same year.

4.3.4. *Weighted Score per Year*

By combining WS (Weighted Score) and SY (Plain Score per Year), for $l \geq 1$ we produce the WSY, i.e. the *Weighted Score per Year* metric:

$$WSY_{c,y,l} = \frac{1}{P_{c,y}} \frac{\sum_i^{\forall i \in C} (W_{i,y,l-1} * N_{i,y \rightarrow c,y} + \sum_{z=y+1}^{last_year} W_{i,z,\infty} * N_{i,z \rightarrow c,y})}{\sum_i^{\forall i \in C} (W_{i,y,l-1} + \sum_{z=y+1}^{last_year} W_{i,z,\infty})} \tag{6}$$

The same way as above we set:

$$W_{i,z,0} = 1 \quad \forall i \in C \text{ and } \forall z \in \{valid_years\}$$

The computation is made for each year by starting from the last year in reverse order. Therefore, when computing scores for year Y , all the weights are known for years $\{Y + 1 \dots max_year\}$. For each year, the procedure is repeated $\forall l \geq 1$ until L , where the ranking does not change or the condition $W_{c,y,L} = W_{c,y,L-1}$ is true $\forall c \in C$. Then $W_{c,y,\infty} = W_{c,y,L}$ and we set:

$$WSY_{c,y} = WSY_{c,y,\infty} = WSY_{c,y,L}$$

4.3.5. *Inverted Impact Score per Year*

Garfield (1994) defined the ISI Impact Factor by using the following example for the 1992 year:

- Ⓐ = total cites in 1992
- Ⓑ = 1992cites to articles published in 1990–1991
- Ⓒ = number of articles published in 1990–1991

then

$$\mathbb{D} = B/C = 1992ISI \text{ Impact Factor} \tag{7}$$

If J is the set of journals and j is a specific journal, then Eq. (7) is equivalent to (8) in a general form:

$$IF_{j,y} = \sum_{z=y-k}^{y-1} \frac{\sum_i^{\forall i \in J} N_{i,y \rightarrow j,z}}{P_{j,z}} \tag{8}$$

This metric cannot be applied directly to conferences for ranking per year. This is due to the fact that when we compute the ISI Impact Factor for a conference c for year y , we actually evaluate the events of c that were organized during the previous k years. For example, in order to compute the ISI Impact Factor of VLDB’95, we actually evaluate VLDB’94 and VLDB’93. This way, two distinct events (that have been organized in different continents) of a specific conference are grouped and evaluated together. Perhaps, we could evaluate VLDB’95 by computing the ISI Impact Factor for VLDB’96 and VLDB’97. In such a case, our results are not affected from the VLDB’95 success only, but from the success of VLDB’96 and VLDB’97 as well. Thus, for the case of conferences, the ISI Impact Factor cannot be used to evaluate a specific conference c held in year y .

For the above reasons, we “revert” the concept of ISI Impact Factor and instead of counting the citations made to the k previous years, we count the citations made during the next k years

(Eq. (9)). This way we count the “Impact” that a specific conference has during the next 2 years. Let this factor be the “Inverted Impact Factor” or “I-Impact Factor”. The *I-Impact Score per Year* is defined as follows:

$$\text{IISY}_{c,y} = \frac{1}{P_{c,y}} \sum_i^{\forall i \in C} \sum_{z=y}^{y+k} N_{i,z \rightarrow c,y} \quad (9)$$

Eqs. (8) and (9) may be semantically different but they are qualitatively similar as they both count the impact of a collection. In (8) the “impact” is computed during a specific year (e.g. What is the impact of VLDB during 1998—actually the impact of VLDB97 and 96). In (9) the impact is computed for a specific year of the conference (e.g. What is the impact of VLDB97?). This way we can rank individual conferences and for example we can get the information: which was the most “successful” conference in 1997. Actually, this is the reasoning why the VLDB Foundation established the 10-years best-paper award.

The Inverted Impact Score (Eq. (9)) metric is a sub-case of the Plain Score per Year algorithm if we set $\text{last_year} = y + k$, where the usual value for k used by ISI is 2 or 5. Since the notion of ISI Impact Factor is widely accepted, we use this metric in our tests as the basic metric to compare with. We cannot use for comparison the ISI Impact Factor as it is exactly defined by Garfield (1994) because it is semantically different from the metrics presented here.

4.3.6. Weighted I-Impact Score per Year

The same way, if in Weighted Score per Year (Eq. (6)) we set $\text{last_year} = y + k$, where $k = 2$ or 5 , then we get the I-Impact Score in a weighted manner, let it be $\text{WIISY}_{c,y}$. This has the advantages of the I-Impact Score metric, plus the advantages of a weighted metric.

4.4. The ranking algorithm

The ranking algorithm is shown in Fig. 4. This algorithm is used for all 4 types of scores. The Plain Scores per Year are the results of the algorithm of level 1. The Weighted Scores are the results of the last reached level, and we store them in the results array with a level index set to -1 (instead of ∞ for practical reasons). The I-Impact Scores can be computed by the same algorithm by setting the variable last_year to $y + 2$.

4.5. The weight set

For every distinct ranking ⁴ we need to define a set of sets:

$$G = \{G_1, G_2, \dots, G_n\} \quad (10)$$

⁴ At level l and for year y , or just at level l when computing ranking for all years.

```

last_year=max_year;
L = -1;
for(y=max_year;y>=min_year;y--) {
  l=0;
  for(c=0; c<n_confs; c++) {
    W[c,y,l] = 1;
  }
  ok=0;
  while(!ok) {
    l++;
    for(c=0; c<n_confs; c++) {
      WSY[c,y,l] = 0;
      for(i=0; i<n_confs; i++){
        WSY[c,y,l] += W[i,y,l-1] * N[i,y,c,y];
        for(z=y+1; z<=last_year; z++) {
          WSY[c,y,l] += W[i,y,L] * N[i,z,c,y];
        }
      }
      W[0..n_confs,y,l] =
        compute_weights(WSY[0..n_confs,y,l]);
      ok = check_if_done(l, y);
    }
  }
  for(c=0;c<n_confs;c++) {
    WSY[c,y,L] = WSY[c,y,l];
  }
}

```

Fig. 4. General ranking algorithm.

where

$$G_i = \{\text{cluster of conferences}\} \quad \text{for } 1 \leq i \leq n$$

$$G_1 \cup G_2 \cup \dots \cup G_n = C$$

$$G_i \cap G_j = \emptyset \quad \text{for } 1 \leq i, j \leq n, i \neq j$$

We have to assign a specific weight value to each set G_i , for $1 \leq i \leq n$. Thus, we have to define the set:

$$W = \{W^1, W^2, \dots, W^n\}$$

At this point, it is necessary to introduce two important parameters. The *number of clusters* ($\equiv n$) and the *range for the weights*. For our tests, we have set the number of clusters equal to 5 (meaning: very strong, strong, average, very weak). This leads us to have 5 distinct weights and 5 clusters in the conference ranking.

The selection of the weight range is also important as it affects the results in the sense that it tunes the importance of a citation from a “very strong” conference in comparison to the importance of a citation from a “very weak” conference. We decided to use the range 1–5 and specifically the weights $W^1 = 1$, $W^2 = 2$, $W^3 = 3$, $W^4 = 4$, $W^5 = 5$, in order to emphasize the difference to the Plain Score. For instance, selecting the range 1–2 (i.e. 1, 1.2, 1.4, 1.6, 1.8, 2) did not make any difference.

Actually, since the scores are normalized by dividing with the sum of weights, the important factor is the fraction of the weights divided with the minimum one, and not the absolute values.

Thus, it is safe to accept a minimum weight of 1. It is obvious that there is no sense in using a negative or zero weight.⁵

Also, we defined that the conferences which belong in a different scientific domain than the one the ranking is computed for, to be members of the G_1 set. In addition to that, conferences that have zero score ($\Leftarrow 0$ citations to them), are set by the classification algorithm in group G^1 , as well.

4.6. Clustering conferences according to citations

The clustering algorithm is a hierarchical clustering algorithm applied on one-dimension points (Jain, Murty, & Flynn, 1999). Initially, a number of clusters N is defined, where N is the number of conferences for ranking.

$$G_1 = \{S_1\} \quad G_2 = \{S_2\} \quad \dots \quad G_N = \{S_N\}$$

For each cluster we set G_x^A as the average value of G_x members. In every step of the algorithm, we find two sets G_i and G_j , for which the difference of their average values ($|G_i^A - G_j^A|$) is the minimum of any other pair. We define a new set $G_k = G_i \cup G_j$ and we delete the sets G_i and G_j . The procedure is repeated until the number of groups reaches n . If, when reaching n , there exists a pair with zero difference of their average values ($|G_i^A - G_j^A| = 0$)⁶, we continue joining the clusters, until we get $|G_i^A - G_j^A| > 0 \forall G_i, G_j \in G$. In Fig. 5 we show the clustering algorithm.

4.7. Weight refinement

The Weighted Score algorithm, as described above, is open to deadlocks. This is due to the fact that there is no guarantee that a conference will not move from one cluster to another at some point during the algorithm execution. We illustrate this situation with a simple example of two conferences A and B for which:

$$\begin{aligned} P_A &= P_B = x (= 10) \\ N_{A \rightarrow B} &= 4 & N_{B \rightarrow B} &= 0 \\ N_{B \rightarrow A} &= 3 & N_{A \rightarrow A} &= 0 \end{aligned}$$

In this case:

$$\begin{aligned} \text{level : 1} & \left. \begin{array}{l} WS_{A,1} = 0.15 \\ WS_{B,1} = 0.2 \end{array} \right\} \Rightarrow \left. \begin{array}{l} W_{A,1} = 1 \\ W_{B,1} = 2 \end{array} \right\} \Rightarrow \\ \text{level : 2} & \left. \begin{array}{l} WS_{A,2} = 0.2 \\ WS_{B,2} = 0.13 \end{array} \right\} \Rightarrow \left. \begin{array}{l} W_{A,2} = 2 \\ W_{B,2} = 1 \end{array} \right\} \Rightarrow \\ \text{level : 3} & \left. \begin{array}{l} WS_{A,3} = 0.1 \\ WS_{B,3} = 0.26 \end{array} \right\} \Rightarrow \left. \begin{array}{l} W_{A,3} = 1 \\ W_{B,3} = 2 \end{array} \right\} \end{aligned}$$

⁵ A weight can be set to zero, iff the appropriate conference does not exist. This happens in the case of computing the rankings per year, where not all conferences are present within a specific year, either because they were not organized or they are just absent from our database.

⁶ This occurs only when ALL the members of the two groups have exactly the same score.

```

function compute_weights(array scores[0..n_confs]) {
    max_groups=5; min_weight=1; max_weight=5;

    groups = new array;
    for(g=0;g<n_confs;g++) {
        groups[g].avg = res[g].score;
        groups[g].members.add(g);
    }
    groups = sort {(a.avg<=>b.avg)} groups;
    while(count(groups) > max_groups) {
        (MinDiff,ToJoin) = find_MinDiff(groups);
        groups[ToJoin] = union(groups[ToJoin], groups[ToJoin-1]);
        delete groups[ToJoin-1];
    }
    while(MinDiff == 0) {
        (MinDiff, ToJoin) = find_MinDiff(groups);
        if(MinDiff == 0) {
            groups[ToJoin] = union(groups[ToJoin], groups[ToJoin-1]);
            delete groups[ToJoin-1];
        }
    }
    weight = min_weight;
    step = (max_weight-min_weight+1)/max_groups;
    for(g=0; g<count(groups); g++) {
        groups[g].weight = weight;
        weight += step;
    }
}
function find_MinDiff(array groups) {
    // Find the minimum difference by comparing the
    // groups[i].avg values and return the index of the
    // second element belonging to the mindiff-pair.
    return(MinDiff, ToJoin);
}

```

Fig. 5. Compute clusters and set weights algorithm.

This leads to an infinite loop since at level 4 we will get exactly the same results as at level 2. In order to avoid this case, after the computation of the clusters G_1, G_2, \dots, G_n at level l , and before we assign the weights (W^1, W^2, \dots, W^n) for each conference, we check if the same condition has been raised in a previous level d . If there is a level $d < (l - 1)$ for which $W_{c,d} = W^k$ ($c \in G_k$), $\forall c \in C$ ⁷, then we do not set $W_{c,l} = W^k$ (as we should do), but instead we set:

$$W_{c,l} = \text{avg}(W_{c,d} \dots W_{c,l-1}) = \frac{\sum_{p=d}^{l-1} W_{c,p}}{l-d}$$

This way, $W_{c,l} \xrightarrow{l \rightarrow \infty} x$, where x is a real number. Actually, since we have distinct weights, we reach x very fast.

In the previous example the next steps should be:

⁷ If $d = l - 1$ then the termination condition is held.

$$\begin{array}{l} \text{level : 3} \Rightarrow \left. \begin{array}{l} W_{A,3} = 1.5 \\ W_{B,3} = 1.5 \end{array} \right\} \Rightarrow \\ \text{level : 4} \left. \begin{array}{l} WS_{A,4} = 0.15 \\ WS_{B,4} = 0.2 \end{array} \right\} \Rightarrow \text{termination} \end{array}$$

5. Results

First, we note that the ranking is made for only one out of the four clusters that have been presented in Section 4.1. That is, we focus in the Database cluster as it is the most complete cluster in the DBLP database. The database contains conferences from 1959 to 2003 (but “complete” data for these conferences exist only for the year 1980 and afterwards). Thus, we find the ranking for each year separately by using:

- the Plain Score per Year,
- the Weighted Score per Year,
- the Plain I-Impact Score per Year,
- the Weighted I-Impact Score per Year.

For all runs, the algorithm shown in Fig. 4 is used. The Plain Score is the result of the algorithm at level 1 (all the weights at level 0 are equal to 1). The results of a weighted ranking are the results of the last reached level. The I-Impact Score is a sub-case of the previous ones and, therefore, we reach it if we set the variable *last_year* equal to $y + 2$.⁸ Thus, we compute for 44 years the Weighted Score and the Weighted I-Impact Score, a total of 88 individual rankings (the plain ranking is a sub-result of the weighted one).

An important concern is the computational cost, e.g. how many times we have to repeat the computation in order to get the termination condition “no change” in the course of ranking. As shown in Fig. 6, in most of the cases, two levels were enough to get the final ranking, and only in three instances (once for the Weighted Score and twice for the Weighted I-Impact Score) we had to reach level four.

5.1. Rank comparisons

In order to visualize the comparison of the various ranking results, we use q–q plots (quantile–quantile plots), which illustrate the quantiles of one univariate distribution against the corresponding quantiles of another (in our case, we compare the rankings). Therefore, for comparing the type *A* ranking to type *B* ranking, for each conference *c* in our rank table, we put a dot in the graph at point (x, y) where *x* is the position of *c* by using type *A*, whereas *y* is the position of *c* by

⁸ All results are accessible at the web location: <http://delab.csd.auth.gr/sceas>.

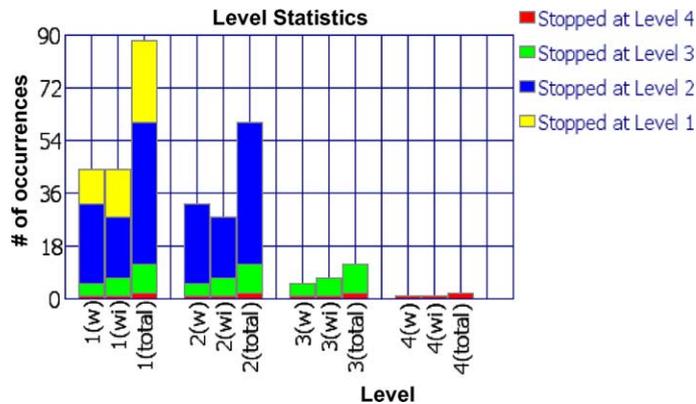


Fig. 6. Level statistics.

using type *B*. Thus, the *x*-axes represent positions computed by *A* and *y*-axis positions computed by *B*. The two rankings would be equivalent iff $y = x$ for every point in the graph. It is easy to notice from the *q*-*q* plots (Figs. 7 and 8), that the various results do not differ substantially for the “very strong” or “very weak” conferences but mainly for “average” cluster.

In all *q*-*q* plots that compare *I-Impact Score* ranking (either Weighted or Plain) and *Score* ranking (either Weighted or Plain) (Figs. 7a and 8a, b), there are some outliers (marked as ■ (blue squares in the web version)), for which $x \ll y$, meaning that they have much better rank position by using *Score* than *I-Impact Score*. This is due to the nature of the *Impact Score* notion, where only citations made in the next *k* (2 in our tests) years are taken into account. Thus, these specific conferences do not have big “Impact”, meaning that do not have a lot of citations during the next 2 years, but they have citations until “now”.

Specifically in Figs. 8a and b the outlier that lies above the line $y = x$ is the CPM’96 Conference (Combinatorial Pattern Matching⁹). The specific conference does not get any citations during the next 2 years, so its *I-Impact Score* is low. However, if we have to evaluate its overall contribution to the academic community, we have to see the *Score Ranking*.

In an analogous way, the outliers for which $y \ll x$, are conferences with big *I-Impact Score* but low *Score*. These conferences get a lot of citations during the next *k* years, but the citations are decreasing over time, meaning that they do not contain “citation classics” papers. In cases of Figs. 7b and 8c and d, the outliers are very close to the line $y = x$ and quantitatively few. This means, that there is no radical repositioning in the plain ranking by adding the notion of weight, although the fraction W^5/W^1 we have used is high ($= 5$). There are some re-orderings, which help in refining the ranking. The conferences of Fig. 8c, for which $x \neq y$, are shown in detail in Table 1. We see that the HT Conference (=ACM Conference on Hypertext) and the SPIESR Conference (=Storage and Retrieval for Image and Video Databases) have swapped positions after computing the Weighted *Score*. In the Plain *Score* the scores for these two conferences are very close.

⁹ The specific conference does not strictly belong to the database cluster, but it has been placed in this cluster by the algorithm as we have not defined deliberately any other closer scientific domain.

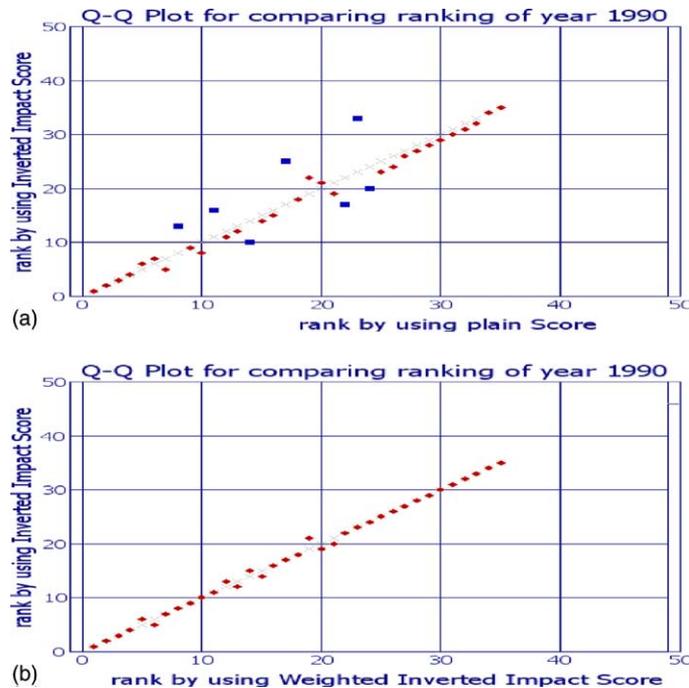


Fig. 7. Comparison of the ranking for year 1990.

The Weighted Score for “HT” is greater than the one of “SPIESR”, as it has more citations from “very strong” conferences.

5.2. Rank results

Besides the introduction of the new metrics for citation analysis, in this paper we report some first rankings of database conferences.¹⁰ The presentation of the ranking results derived from the SCEAS system are made by using two ways:

- By a Rank table, assuming a specific type of ranking and a selected year. For example, in Table 2 we present the ranking by using the Weighted Score for year 1995.
- By a Historical chart, where we can view the whole history of a conference for any specific type of ranking.

In Fig. 9, the history of ranking of VLDB conference is presented, according to all types rankings. Each bar consists of three parts. The bottom part (black; blue in the web version) gives the percentage of conferences that have a lower ranking, the top part (grey; red in the web version)

¹⁰ Full presentation of the results is available at the web location <http://delab.csd.auth.gr/sceas/>. However, the results shown there may be slightly different than the ones presented here since the database is continuously updated.

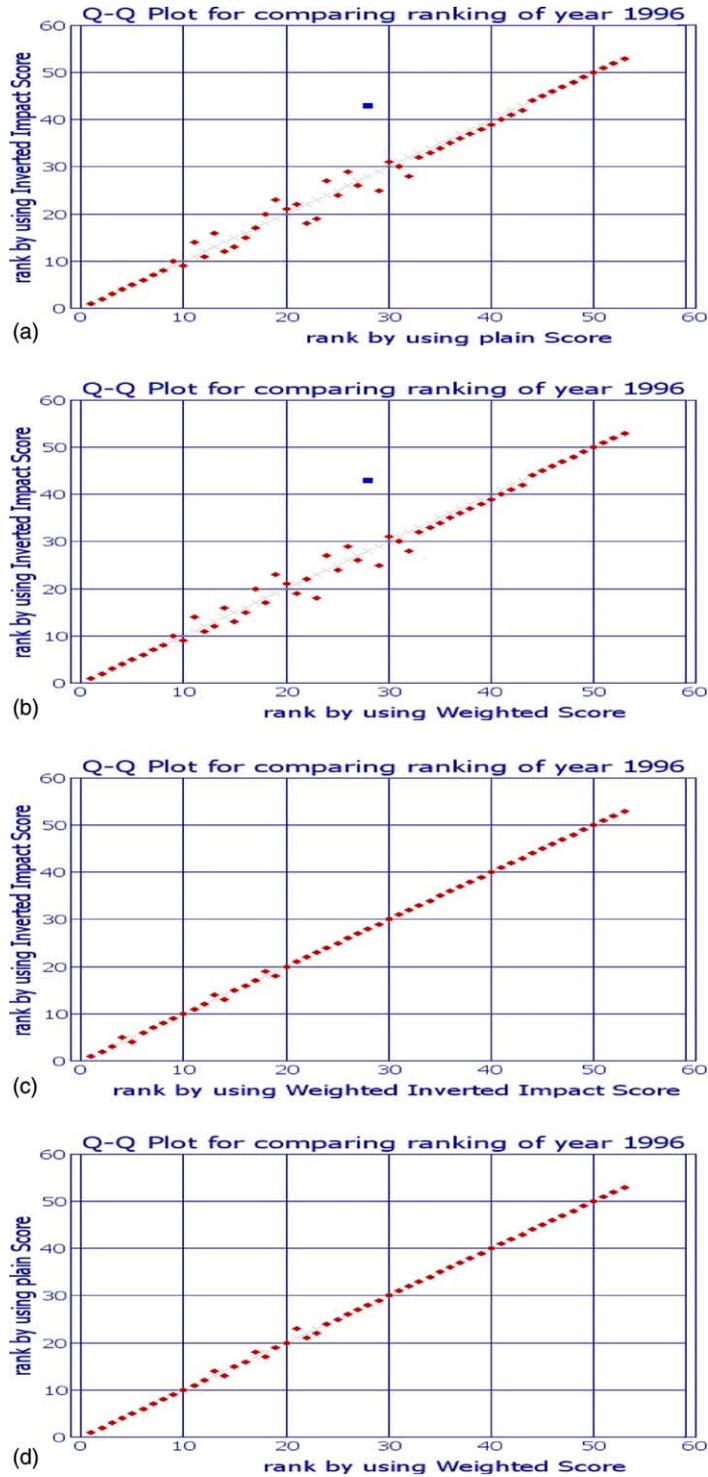


Fig. 8. Comparison of the ranking for year 1996.

Table 1
Detailed comparison of Plain Score vs. Weighted Score for the 1996 year

Weighted Score		Plain Score		Conference
Pos	Score	Pos	Score	
13	0.404208	14	0.287565	ACM Conference on Hypertext
14	0.382026	13	0.287772	Storage and Retrieval for Image and Video Databases (SPIE)
17	0.236508	18	0.168072	Database and Expert Systems Applications (DEXA)
18	0.223758	17	0.168552	Digital Libraries
21	0.160647	23	0.110927	Advances in Databases and Information Systems (ADBIS)
22	0.158213	21	0.119178	Australasian Database Conference (ADC)
23	0.154697	22	0.116530	British National Conference on Databases (BNCOD)

Table 2
Rank with Weighted Score for the 1996 year

Pos	Score	#Papers	Weight	Conference
1	10.82818	64	5	ACM SIGMOD Conference
2	7.52698	72	4	Very Large Data Bases (VLDB) Conference
3	6.14474	26	4	Symposium on Principles of Database Systems (PODS)
4	3.71273	27	3	Conference on Parallel and Distributed Information Systems (PDIS)
5	3.61389	81	3	International Conference on Data Engineering (ICDE)
6	2.60681	47	3	International Conference on Extending Database Technology (EDBT)
7	1.65844	17	2	Research Issues in Data Engineering (RIDE)
8	1.19002	74	2	Knowledge Discovery and Data Mining (KDD)
9	0.68100	23	2	International Conference on Cooperative Information Systems (CoopIS)
10	0.59669	28	2	Statistical and Scientific Database Management (SSDBM)

gives the percentage of conferences that have a higher ranking, whereas the middle part (light grey; green in the web version) gives the percentage of conferences that have equal ranking. In addition, the ratio below the x -axis gives the relative rank for each year. A different position occurs for several years, but all the graphs are very similar since the specific conference is clearly a “very strong” one during all years.¹¹

Note that the last 2 years ranking (i.e. 2001 and 2002) could not be considered as reliable, since there are no citations in our database to conferences organized during these years. The scores for all conferences in 2001 and 2002 are zero and, thus, the rank is up to the number of publications. Also, it is too early to evaluate conferences that were held in 2000. Thus, we should ignore the last three years of the rank history.

¹¹ VLDB and SIGMOD are not directly comparable to the other references since they includes “industrial” publications, which can be excluded only in manually. In other words, in reality the distance between these two conferences from the third one is greater.

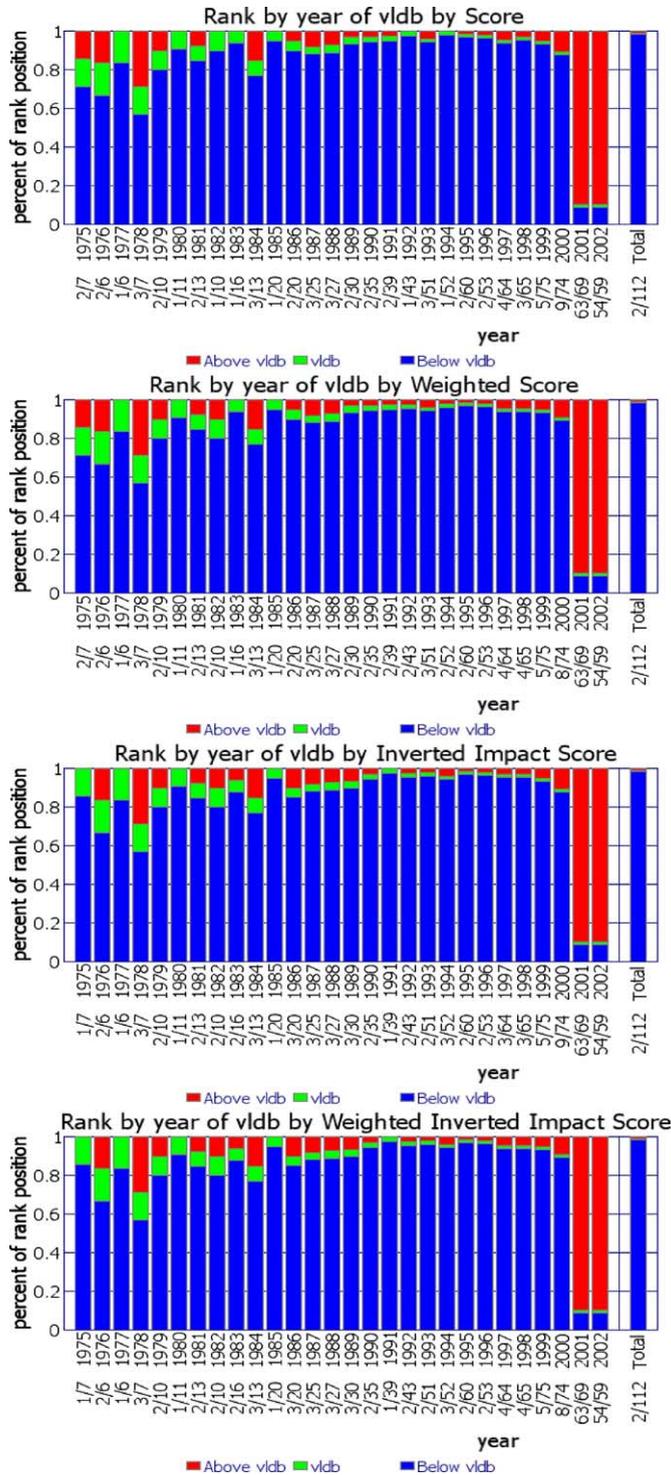


Fig. 9. The ranking history of VLDB.

6. Conclusion and future work

In this paper we first presented an overview of two major current systems for conference and journal ranking by using citation analysis, CiteSeer and SCI. A weak point of these systems is that they are based on the ISI Impact Factor, thus, considering citations in a flat way, e.g. without paying attention to the quality of the respective publication. Therefore, we introduced four new metrics in order to cure this deficiency, which are suitable for considering both journal and conference publications. These new metrics are used by a system that we have built, the SCEAS system (Scientific Collection Evaluator by using Advanced Scoring). The system is autonomous and has the following characteristics:

- it imports the DBLP bibliography records into a local database (it could be extended to import any other scientific collection of publications),
- it partitions the imported collection into clusters according to the topic of the conference and performs a cleansing step to provide reliable information,
- it performs the ranking by all four metrics for the conferences that focus in databases.

The web user of SCEAS system has access to all the results produced at any stage of the rank process, can compare the various rank metrics and can study the rank results in order to derive useful information regarding the quality of the database conferences.

In the future we plan to extend the system by:

- Computing more variations of the weighted metrics in which self-citations of the collection could be excluded or taken into account multiplied with a smaller weight.
- Perform detailed citation analysis of each article and compute aggregated results for each collection.
- Extending the ranking in more collections/scientific domains. This could give us the ability when ranking a cluster (e.g. DB conferences) to take into account “weighted” citations from other type of collections belonging in the same scientific field (e.g. DB Journals and Books).
- Improving the clustering according to scientific domain by allowing one entity to be a member of more than one clusters.

References

- An, Y., Janssen, & J., Milios, E. (in press). Characterizing and mining the citation graph of the computer science literature. *Knowledge and Information Systems*.
- Baumgartner, H., & Pieters, R. (2000). The influence of marketing journals: a citation analysis of the discipline and its sub-areas (Technical Report). Tilburg University. Available: <http://greywww.kub.nl:2080/greyfiles/center/2000/doc/123.pdf>.
- Bradshaw, S., & Hammond, K. (2001). Using citations to facilitate precise indexing and automatic index creation in collections of research papers. *Knowledge Based Systems*, 14(1/2), 29–35.
- Brin, S., & Page, L. (1998). The Anatomy of a large-scale hypertextual web search engine. In *Proceedings 7th www conference* (pp. 107–117).
- Dhyani, D., Bhowmick, S., & Ng, W. K. (2002). Deriving and Verifying Statistical Distribution of a Hyperlink-based Web Page Quality Metric. In *Proceedings 13th dexa conference* (pp. 19–28).

- Ding, Y., Chowdhury, G., & Foo, S. (2001). Bibliometric cartography of information retrieval research by using co-word analysis. *Information Processing and Management*, 57(6), 817–842.
- Garfield, E. (1972). Citation analysis as a tool in journal evaluation. *Essays of an Information Scientist*, 1, 527–544.
- Garfield, E. (1994). The Impact Factor. Available: <http://www.isinet.com/isi/hot/essays/journalcitationreports/7.html>.
- Goodrum, A., McCain, K., Lawrence, S., & Giles, C. L. (2001). Scholarly publication in the Internet age: a citation analysis of computer science literature. *Information Processing and Management*, 57(2), 661–675.
- Han, E., Karypis, G., Kumar, V., & Mobasher, B. (1997). Clustering based on association rule hypergraphs. In *Proceedings dmkd'97 workshop*.
- He, Y., & Hui, S. C. (2002). Mining a web citation database for author co-citation analysis. *Information Processing and Management*, 55(4), 491–508.
- Hult, T., Neese, W., & Bashaw, E. (1997). Faculty perception of marketing journals. *Journal of Marketing Education*, 19(1), 37–52.
- Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys*, 31(3), 264–323.
- Karypis, G., Aggarwal, R., Kumar, V., & Shekhar, S. (1997). Multilevel hypergraph partitioning: Application in VLSI domain. In *Proceedings 34th design automation conference* (pp. 526–529).
- Kleinberg, J. M. (1999). Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5), 604–632.
- Kleinberg, J. M., Kumar, R., Raghavan, P., Rajagopalan, S., & Tomkins, A. (1999). The web as a graph: measurements, models, and methods. In *Proceedings 5th cocoon conference* (pp. 1–17).
- Kleinjn, J., & Groenendaal, W. V. (2000). Measuring the quality of publications: New methodology and case study. *Information Processing and Management*, 36(4), 551–570.
- Korobkin, R. (1999). Ranking journals: Some thoughts on theory and methodology (Technical Report). University of Illinois. Available: <http://www.law.fsu.edu/journals/lawreview/downloads/264/koro.pdf>.
- Lawrence, S., Giles, C. L., & Bollacker, K. (1999a). Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6), 67–71.
- Lawrence, S., Bollacker, K. D., & Giles, C. L. (1999). Indexing and retrieval of scientific literature. In *Proceedings 1999 acm cikm conference* (pp. 139–146).
- Ley, M. (2002). The DBLP computer science bibliography: Evolution, Research Issues, Perspectives. In *Proceedings spire symposium* (pp. 1–10).
- Lin, X., White, H., & Buzydlowski, J. (2003). Real-time author co-citation mapping for online searching. *Information Processing and Management*, 39(5), 689–706.
- Meghabghab, G. (2002). Discovering authorities and hubs in different topological web graph structures. *Information Processing and Management*, 38(1), 111–140.
- Mylonopoulos, N., & Theoharakis, V. (2001). Global perceptions of IS journals. *Communications of the ACM*, 44(9), 29–33.
- Preto, L. (2002). A theoretical analysis of Google's PageRank. In *Proceedings spire symposium* (pp. 131–144).
- Tahai, A., & Rigsby, J. (2002). Information processing using citations to investigate journal influence in accounting. *Information Processing and Management*, 34(2/3), 341–359.