An Automatic and Generic Framework for Ranking Research Institutions and Scholars based on Publication

Jie Ren
University of California, Irvine
jie@ics.uci.edu

Richard N. Taylor
University of California, Irvine
taylor@uci.edu

June 2004

ISR Technical Report # UCI-ISR-04-5

Institute for Software Research
ICS2 210
University of California, Irvine
Irvine, CA 92697-3425
www.isr.uci.edu

An Automatic and Generic Framework for Ranking Research Institution and Scholars based on Publications

Jie Ren, Richard N. Taylor
Institute for Software Research
University of California, Irvine
Irvine, CA 92697-3425
{jie, taylor}@ics.uci.edu

ISR Technical Report # UCI-ISR-04-5

June 2004

Abstract:

We describe an automatic and generic framework for ranking research institutions and scholars based on publications. Compared to other publication-based ranking methodologies, our framework is automatic and can apply to a larger selection of publication venues over a longer time span without any further manual work. Our framework is generic so it can apply not only to computer science and related fields but also to other scholarly disciplines. We used our framework to automatically rank research institutions and scholars in both computer science and software engineering. The quality of rankings based on the framework depends on the abundance of bibliographic data. The framework can be improved by incorporating citation counts as additional quality measures.

An Automatic and Generic Framework for Ranking Research Institutions and Scholars based on Publications

Jie Ren, Richard N. Taylor
Institute for Software Research
University of California
Irvine, CA 92697-3425
1-949-824-2776
{jie,taylor}@ics.uci.edu

ISR Technical Report #UCI-ISR-04-5
June 2004

ABSTRACT

We describe an automatic and generic framework for ranking research institutions and scholars based on publications. Compared to other publication-based ranking methodologies, our framework is automatic and can apply to a larger selection of publication venues over a longer time span without any further manual work. Our framework is generic so it can apply not only to computer science and related fields but also to other scholarly disciplines. We used our framework to automatically rank research institutions and scholars in both computer science and software engineering. The quality of rankings based on the framework depends on the abundance of bibliographic data. The framework can be improved by incorporating citation counts as additional quality measures.

1. INTRODUCTION

It is important to rank academic and industry research institutions and scholars in order to identify the best organizations and individuals in a given discipline. This can publicize outstanding institutions and scholars, so future students or researchers can better decide where they want to study or work, and employers can know where to recruit future employees. Rankings can also help both internal and external administrators make influential decisions, such as how funding should be allocated and who should be promoted and compensated.

Such rankings can be conducted in different ways. The most well known rankings use a significant portion of subjective polls in addition to objective indicators like budget and enrollment. Since 1988, each year US News and World Report publishes its ranking on US doctoral programs, including computer science, partly based on subjective polls from university administrators [12]. The National Research Council ranked over one hundred US doctoral programs in 1993, based on a similar methodology that combines objective indicators and subjective polls from computer science department chairs [11]. The National Research Council is planning a similar ranking for 2004. However, poll-based ranking draws opinions from individuals who cannot possibly be familiar with every program. Consequently, opinions of institutions that are not very well known may vary [5].

While poll-based raking could be subjective, publication-based ranking has the potential to be a more objective alternative. Generally, a publication-based ranking chooses a research field, selects a group of publication venues that are considered prestigious, representative, and influential for the field, gives a score to each paper an institution or author has published, and ranks institutions and authors using their sums of the scores.

In the Computer Science field, the latest publication-based ranking of different institutions was finished in 1996 by Geist et al. [5]. They selected 17 archival research journals published by ACM or IEEE. Each paper appeared in each journal from January 1990 to May 1995 received one point. If the paper was a multi-author paper, the one point was apportioned equally among coauthors. Author scores were attributed to the authors' institutions at the time of publication, regardless of their current locations. Institutions were ranked by the total sum of their respective authors' scores. However, the ranking only covered academic research programs in US because "the journals chosen for the study are more likely to be primary journals for U.S. researchers than for researchers in other countries, with the possible exception of Canada." [5]

In the Systems and Software Engineering field, the Journal of Systems and Software (JSS) has been publishing an annual assessment of scholars and institutions since 1994. (The assessment will be referred as the JSS ranking henceforth). Each year the assessment was based on papers published in the five years prior to the assessment. The latest ranking of the series was based on publications from 1998 to 2002 [6]. The rankings used six journals selected by a 1991 survey of the editorial board of the Journal of Systems and Software. The JSS ranking believed that the results of the survey represented "a knowledgeable, active, and unbiased source of judgment" [6]. The ranking used a scoring scheme that gave one point to the author of a single-author paper. To avoid penalizing multi-author papers, authors of tw0-author papers received 0.7 points each, authors of three-author papers received 0.5 points each, and authors of four-or-more-author papers received 0.3 points each. An institution's score was the sum of the score of every author who was a member of that institution during the publication time. The JSS ranking did not consider page length of papers in scoring. It gave papers from different journals the same weight. Moreover, it only counted refereed research papers published in the chosen journals.

These publication-based rankings provide new data points to assess the quality of scholars and institutions more objectively than poll-based rankings. However, existing rankings have several limitations. The biggest limitation is that they had to be performed manually. As a result, both the number of journals considered and the time span over which they were published were limited, reducing the scope of such rankings. Ranking manually might also be the reason for considering journals only and excluding other important sources of academic communication such as conference proceedings. This omission could affect the quality of rankings. An additional limitation is that the reported rankings were field specific. Each new research field will require performing a new ranking that repeats the same basic procedure manually. The two limitations yield the third one: inflexible criteria. Each ranking made different decisions about what journals were included and how each paper was scored. While the decisions were made with legitimate reasons, the criteria cannot be altered without repeating the entire labor-intensive process. These limitations hinder applying publication-based ranking more flexibly and widely.

To overcome these limitations, we propose and implement a framework that facilitates automatic and generic publication-based ranking. It utilizes electronic bibliographic data to process many journals and conferences that span a long period. The framework can accommodate many policy choices. We have successfully used this framework to repeat the two manual rankings reported above.

We will describe the design of our framework in section 2, demonstrate its usage to the field of computer science and software engineering in section 3, discuss its limitations in section 4, outline steps to apply the framework in section 5, and conclude with section 6.

2. FRAMEWORK DESIGN

We have two primary goals in designing the framework. One is to automatically rank institutions and scholars based on publications. The other is to rank with flexible policy choices. With the automatic and generic support, evaluator can easily rank with policies appropriate to their specific needs.

The biggest difficulty in automatic ranking is the availability of bibliographic data that contain the institution with which an author was affiliated when the paper was published. While there are several digital bibliographic services such as DBLP [3], Computer Science Bibliography [1], and ACM Digital Library [10], only INSPEC [8] consistently provides the author affiliation information. A limitation of INSPEC is that it only records the affiliation information for the first author. INSPEC also backs the IEEE Explore digital library [4]. We thus chose INSPEC as the source of data when designing the framework and conducting experiments. However, we stress that with only minimal change the framework can be easily adapted to other bibliographic services that provide affiliation information once the services become available.

2.1 Accommodating Flexible Policies

The general steps in a publication-based ranking are:

1. choose a field,
2. select representative publication venues for the field,
3. set the time range for consideration,
4. assign a score to each published paper,
5. divide the score among multiple authors if the paper has more than one author,
6. sum the scores for each scholar and each institution, and finally
7. rank the scholars and institutions based on sums of their scores.

The relevant policy decisions involved in this process are as follows:

What field to rank? The field can be the whole field of computer science or it can be a sub field like Systems and Software Engineering, as in [5] and [6], respectively. Our framework supports both choices. The framework can even rank a non-computer related discipline, as long as publications are considered effective assessment of scholarship in that field and bibliographic data for that field are available.

Selecting a field for ranking and correctly explaining the results is crucial to the validity of the ranking. For example, two of the top five universities from the US News and World report ranking [12] and the National Research Council ranking [11] did not appear in the top 15 institutions of the JSS ranking [6]. The authors of the JSS ranking believed this was because that even informed opinions from fellow academics might not relate well to publication frequency [6]. However, another possible explanation for the absence could be that the schools under question do not have a research program on “Systems and Software Engineering”, even though they are excellent in other fields of computer science.

What entities to rank? The framework supports ranking a wide range of entities. It can rank both scholars and institutions, handle both academic and industry institutions, and cover scholars and institutions from not only the United States but also from other geographical regions.

What journals and conferences are considered important in the field? This is the key decision of the ranking process, because selecting different journals and conferences will result in significantly different results. None of the previous rankings included proceedings of conferences or workshops [5, 6]. We feel proceedings from these meetings are important academic communication channels, and they are especially relevant for a field such as computer science that is rapidly developing. Since different fields have different representative journals and conferences, and even for the same field different evaluators may have different opinions on what the best journals and conferences are, our framework does not impose any restriction on conference or journal selection. It allows evaluators to make decisions based on their own criteria.

How many years of publications should be included in the ranking? Both rankings selected publications of the previous five years [5, 6]. This is a reasonable time range for assessing the current quality of a scholar or an institution. However, a more comprehensive ranking would span a longer time. Even for the intermediate range, another evaluator might prefer a different number of years. Our framework allows an evaluator to use any preferred year range.

What weight should papers from different journals or conferences receive? In previous rankings [5, 6], papers from different journals always receive the same weight. Since those evaluators only selected the most prestigious referred journals for their respective fields, these decisions are rational. However, different evaluators might disagree about what the most prestigious publication venues are. With the inclusion of many journals and conferences enabled by the automatic and generic nature of our framework, it is almost inevitable to treat papers from different publication venues differently. The framework gives evaluators much freedom in assigning different weights to different journals and conferences. They can treat their selections equally or differently. If they treat the selections differently, the difference can be either minor or significant.

How should the score be distributed among co-authors for a multi-author paper? After the score of a paper has been assigned based on the venue, the ranking in [5] apportioned the score equally among the authors, and the ranking in [6] gave each author a little bit more than a simple equal share of the score to avoid penalizing a multi-author paper. A third scheme is that each author is given the same score for any paper, whether the author is the primary contributor or just a co-author. A fourth scheme is distributing the score unequally among the authors and giving slightly more share to the authors listed first and progressively assigning fewer share to authors listed latter. A final scheme is each institution or author receives the same score for each paper, disregarding the number and order of authors from the institution. Our framework supports all these schemes. If a chosen distribution scheme gives credits to authors other than the first author, currently manual editing of the bibliographic data is necessary to provide affiliations for other authors, because INSPEC only records the affiliation of the first author.

2.2 Special Treatments

The framework is based on bibliographic data retrieved from INSPEC. Due to the nature of the data, several special treatments are needed in the framework. Research papers should be differentiated from non-research items. Aliases of an institution should be accredited to the same entity. Finally, the address of institutions should be normalized.

Differentiate research papers from no-research items. The bibliographic data may include entries of non-research items. For example, some conference proceedings contain entries for tutorials and posters, and some journals have special columns for editorial introductions and errata. A manual ranking can accurately identify such papers, but an automatic framework cannot achieve perfection. This problem can be mitigated by imposing a minimum page limit for a paper to be classified as a research paper, because generally the non-research papers are rather short. The page limit also enables ranking criteria that differentiate longer papers from shorter ones, even though neither of the rankings in [5, 6] adopted this scheme.

Identify aliases. INSPEC may uses different names for the same institution. Sometimes the different names come from the inconsistency in the compilation process; sometimes this is because the institution went through a name change. For example, here are the different names used for Bell Labs at Murray Hill:

AT&T Bell Labs., Murray Hill
AT&T Bell Lab., Murray Hill
Bell Labs., Murray Hill
AT&T Bell Labs, Murray Hill
Bell Telephone Labs. Inc., Murray Hill
Lucent Technol. Bell Labs., Murray Hill

The framework provides a mechanism to identify these aliases, because at the start of a ranking an evaluator probably does not even know what aliases INSPEC has used. The framework also provides a method to group these aliases so all papers published by the same institution can be correctly accredited to it. This grouping method also provides flexibility in ranking a multi site organization. For example, the nine University of California campuses are generally considered independent research institutions. However, a company like IBM might prefer to group its various research locations such as the Watson Center and the Almaden Center together for the ranking purposes [6].

Normalize addresses. Some institutions have special addresses. For example, the San Diego campus of the University of California is actually in the city of La Jolla, not in the city of San Diego. The framework provides a mechanism to translate address quirks such as “University of California, San Diego, La Jolla, California, USA” into the canonical format: University, City, State/Province, Country.

This translation mechanism provides another flexibility in accurately accrediting institutions. For example, in the field of software engineering, both the School of Computer Science and the Software Engineering Institute located in Carnegie Mellon University contribute significantly. However, the Institute is not an academic department in the usual sense. The mechanism can separate these university-operated government labs from regular academic departments, allowing more ranking criteria [6].

3. VALIDATION

To validate our framework, we used it to perform two rankings. The first ranking assessed US computing graduate programs. We adopted similar criteria as used in [5] and reached comparable results. The second ranking evaluated institutions and scholars in software engineering. We adopted similar criteria as used in the JSS ranking [6]. Our results were different. We will discuss possible reasons for the difference.

3.1 Ranking of US Computing Graduate Programs

We used our framework to repeat the ranking of [5], based on publication data from 1995 to 2003. Other than the different time range, the only other different criterion was that we selected a scoring scheme that gives credit only to the first author, while in [5] the score was distributed equally among multiple authors. We adopted this policy because INSPEC bibliographic data only records the affiliation of the first author and we decided not to perform manual editing for this ranking. The resulting top 50 US computing graduate programs are listed in Table 1. The first column is the rank from our ranking. The second column is the rank reported in [5]. Overall the two rankings agree with each other, confirming the effectiveness of our framework. However, our ranking is performed automatically.

Table 1, Top 50 US Computing Graduate Programs

#	[5]	Score	University
1	2	87	Massachusetts Institute of Technology
2	1	79	University of Maryland, College Park
3	6	78	Carnegie Mellon University
4	19	73	Georgia Institute of Technology
5	7	70	Stanford University
6	3	64	University of Illinois, Urbana-Champaign
7	4	63	University of Michigan, Ann Arbor
8	5	61	University of Texas, Austin
9	10	59	Purdue University
10	11	47	University of California, Berkeley
11	26	46	University of California, San Diego
12	12	45	University of Massachusetts, Amherst
13	30	44	Rutgers University, New Brunswick
14	9	43	University of Southern California
14	15	43	University of Washington, Seattle
16	21	40	Cornell University
16	13	40	University of California, Santa Barbara
18	32	39	Michigan State University
19	20	38	University of California, Irvine
20	16	36	University of Minnesota, Minneapolis
21	8	35	University of Wisconsin, Madison
22	31	31	Columbia University
22	22	31	Princeton University
24	14	29	Ohio State University
24	34	29	University of Florida, Gainesville
26	18	28	Pennsylvania State University
26	27	28	Texas A&M University
26	37	28	University of Pennsylvania
29	50	27	University of Texas, Dallas
30	29	26	State University of New York, Stony Brook
31	93	25	Oregon State University
31	17	25	University of California, Los Angeles
31	51	25	University of Virginia
34	65	24	California Institute of Technology
34	25	24	University of Arizona
34	24	24	University of Illinois, Chicago
37	43	23	State University of New York, Buffalo
38	40	21	Louisiana State University
38	38	21	Rice University
38	68	21	Washington University in St. Louis
41	56	20	Harvard University
42	63	19	Southern Methodist University
42	51	19	University of Iowa
42	87	19	University of South Florida, Tampa
45	53	18	Boston University
45	69	18	Rensselaer Polytechnic Institute
47	71	17	North Carolina State University
47	41	17	University of California, Davis
49	36	16	University of Colorado, Boulder
49	23	16	New York University

There are several noteworthy points about this ranking. First, both ACM and IEEE launched several new journals since [5] was published. A coverage based on more journals that are current would be a more comprehensive measurement. However, since not all bibliographic data were available and we wanted to make our results comparable to those of [5], we adopted the same collections used in [5]. Second, we adopted a slightly different scoring scheme than that of [5]. The two schemes will result in different scores if the authors are not from the same university. Since most papers are written by researchers from the same university, we believe this would not significantly affect the comparability between the two rankings. Finally, because of the multi-year delay between the reception of the manuscripts and their eventual publication, a Ph.D. student author may have graduated during the period. The recorded affiliation will be the current employer of the student, unless the journals can otherwise indicate the original affiliation. These limitations might affect the accuracy of the ranking, but we hope, and we believe, the ranking still conveys useful information.

3.2 Ranking in Software Engineering

3.2.1 Criteria and Results

We ranked another field, software engineering. We chose two journals and two conferences that are considered most prestigious in the field. The journals are ACM Transactions on Software Engineering and Methodologies (first published in 1992) and IEEE Transactions on Software Engineering (first published in 1975). The conferences are International Conference on Software Engineering (first held in 1975) and Foundations of Software Engineering (first held in 1993). We gave each paper the same score of one point. To compare with the JSS ranking [6], we adopted the score distribution scheme used in that ranking. To support this scheme, we manually edited the bibliographic data to include affiliation information for multiple authors. Based on data from 1996 to 2003, the resulting top 50 institutions and scholars are listed in Table 2 and Table 3, respectively. The first column in each table is the rank from our ranking. The second column in each table is the rank reported the JSS ranking, if it can be found in [6].

Table 2, Top 50 Software Engineering Institutions

#	[6]	Score	University
1	1	34.00	Carnegie Mellon University
2	13	32.90	Politecnico di Milano, Italy
3	11	31.79	University of Maryland, College Park
4		26.79	University of Massachusetts, Amherst
5		25.29	University of California, Irvine
6	4	24.49	Bell Lab, Murray Hill
7		23.90	University of Washington, Seattle
8		22.79	Bell Lab, Naperville

9		21.49	Massachusetts Institute of Technology
10		19.70	University of British Columbia, Canada
11		19.50	Oregon State University
12		19.00	University of Pittsburgh
13	6	18.80	Fraunhofer-IESE, Germany
13		18.80	Imperial College London, UK
15		18.59	IBM Thomas J. Watson Research Center
16	14	18.10	Georgia Institute of Technology
17		16.50	University of Texas, Austin
18		16.39	Ohio State University
19		16.00	University of Colorado, Boulder
20		15.10	Katholieke Universiteit Leuven, Belgium
21		14.50	Washington University in St. Louis
22		14.29	Kansas State University
23		13.49	University of California, San Diego
24		13.09	University of Virginia
25		12.90	Michigan State University
26		12.70	University of Waterloo, Canada
27		12.60	Rutgers University, New Brunswick
28		12.30	University of Illinois, Urbana-Champain
29	9	11.80	AT&T Labs - Research, Florham Park
30		11.60	Purdue University
31		10.40	Carleton University, Canada
32		10.19	University of Michigan, Ann Arbor
33		10.00	University of California, Santa Barbara
34		9.90	University of Paderborn
35	12	9.77	City University of London, UK
36		9.60	University College London, UK
37		9.19	Osaka University, Japan
38		9.10	Stanford University
39		9.00	Naval Research Lab
39		9.00	University of Karlsruhe, Germany
41		8.90	Software Engineering Institute
42		8.80	University of Torino, Italy
43		8.40	University of Southern California
44	5	7.90	National University of Singapore, Singapore
44		7.90	University of Hawaii
44		7.90	West Virginia University
47		7.80	IRISA, France
48		7.50	University of Bologna, Italy
49		7.40	University of Wisconsin, Madison
50		7.30	University of New South Wales, Australia

Table 3, Top 50 Software Engineering Scholars

#	**[6]**	**Score**	**Author (Last Name First Initial)**
1		10.89	Harrold M
2		10.60	Rothermel G
3		9.40	Murphy G
4	9	8.80	Briand L
5	12	8.10	Basili V
6		7.90	Devanbu P
6		7.90	Notkin D
8		7.50	Jackson D
9		7.30	Kramer J
10		6.90	Wolf A
11		6.80	Roman G-C
12		6.50	Clarke L
13		5.90	Rosenblum D
13		5.90	Soffa M
15		5.80	Griswold W
16		5.60	Dillon L
17	1	5.50	El Emam K
17		5.50	Herbsleb J
19		5.49	Dwyer M
20		5.40	Mockus A
20		5.40	Sullivan K
22		5.00	van Lamsweerde A
23		4.88	Magee J
23		4.88	Medvidovic N
23		4.88	Porter A
26		4.80	Corbett J
26		4.80	Osterweil L
28		4.70	Reiss S
29		4.69	Taylor R
30		4.50	Avrunin G
31		4.38	Weyuker E
32		4.30	Pezze M
33		4.29	Bertolino A
34		4.20	Picco G
35		4.09	Kemmerer R
36		4.00	Batory D
36		4.00	Cook J
36		4.00	Votta L
39		3.99	Ernst M
40		3.80	Robillard M
40		3.80	Snelting G
42		3.70	Padberg F
43		3.50	Le Metayer D
43		3.50	Michail A
45		3.40	Fuggetta A
45		3.40	Leveson N
45		3.40	Melo W
45		3.40	Zeller A
49		3.30	Egyed A
49		3.30	Emmerich W

As can be seen from Table 2 and Table 3, most of the top scholars and institutions are US-based, but a significant amount of them come from Europe. We thus believe the ranking is representative of the entire field, not just US centric. This is expected given that the international nature of the conferences and journals (Foundations of Software

Engineering is held every other year with European Conference on Software Engineering).

3.2.2 Reasons for the Difference

The result of our ranking is significantly different from that of the JSS ranking. The second column in Table 2 shows that only five of the top fifteen institutions from the JSS ranking are among the top fifteen of our ranking. The second column in Table 3 shows that only two of the top fifteen scholars from the JSS ranking are among the top fifteen of our ranking. Three factors could contribute to the difference. First, we chose a longer time span (1996-2003) while the JSS ranking used the period of 1998-2002. Second, we included two conferences in our ranking but the JSS ranking did not consider any conference. Last and most importantly, our ranking and the JSS ranking selected different journals, and the journals contributed differently in the two rankings. We will elaborate this last factor in the rest of this section.

The journals selected in the JSS ranking were:

- Information and Software Technology (IST),
- Journal of Systems and Software (JSS),
- Software Practice and Experience (SPE),
- IEEE Software (SW),
- ACM Transactions on Software Engineering and Methodologies (TOSEM), and
- IEEE Transactions on Software Engineering (TSE).

Of these we selected only TOSEM and TSE. The JSS ranking had several problems. First, it included a magazine (SW). Second, it relied heavily on JSS and IST. Finally, it received almost no contribution from TOSEM.

The first issue is the inclusion of a magazine, IEEE Software. IEEE Software has an editorial process markedly different from that of an archival transaction such as TSE, where a paper must be thoroughly peer reviewed before publication. A similar magazine, the Communications of the ACM (CACM), whose editorial process is different from that of TOSEM, was exclude from the ranking of [5]. The JSS ranking noticed the exclusion of CACM but it still included IEEE Software.

The second issue of the JSS ranking is its heavy reliance on papers published in JSS and IST. This fact is especially noteworthy since the journals used in the ranking were chosen by the editorial board of JSS and both JSS and IST are published by Elsevier Science. The reliance can be seen from Figure 1, which lists the number of the top scholars and institutions that have published in each journal. It shows that IST and JSS host most top scholars and institutions. Figure 2, which depicts how the scores received by the top scholars are distributed among each journal, illustrates that JSS and IST contributed almost 60% of the total scores for those top scholars.

The third problem is that TOSEM, the major research archival journal on software engineering from a major professional society, had almost no influence on the JSS ranking. Figure 2 shows that TOSEM contributed less than one percent of the total scores received by the top scholars in the JSS ranking. Figure 1 shows that only one top scholar published in TOSEM and more than half of the top institutions from the JSS ranking did not publish in TOSEM.

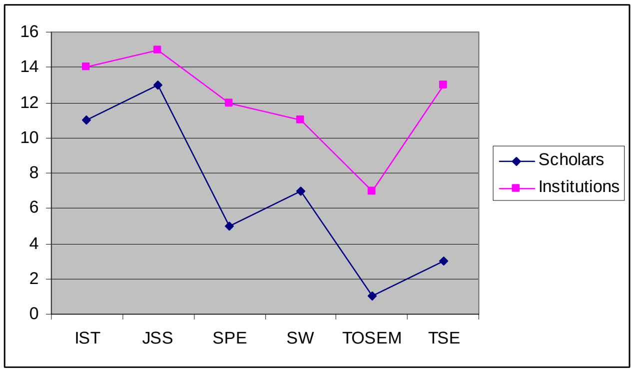

Figure 1, Scholars and Institutions published in each Journal

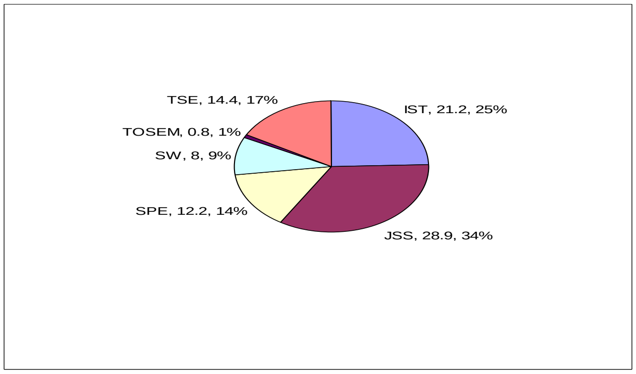

Figure 2, Score Distribution of Top Scholars in each Journal

4. DISCUSSION

In this section, we identify some limitations on the current framework, such as not differentiating papers from the same venue and relying on English bibliographic data. We also outline using citation as additional quality assessment and incorporating complete affiliation information to improve the framework.

Limitation: not differentiating papers from the same venue. Previous rankings gave papers from different journals the same weight [5, 6]. However, our framework supports giving different weight to papers from different publication venues. Papers from the same journal or conference still receive the same score based on the general quality of the journal or conference. More often than not, an outstanding conference paper in computer science will later turn into a journal paper. Our framework will count these papers at both occasions. This can mitigate the lack of discreetness among papers from the same venue to a certain extent.

Limitation: relying on English bibliographic data. Our framework can be used to rank not only US universities but also industry institutions and European organizations. Currently the bibliographic data comes from INSPEC and the current implementation of the framework is based on an English bibliographic format, so the framework in its current incarnation has a language dependence on English. However, we believe the framework can be easily adapted to other language settings with minimal change, once the bibliographic data for the new language is available.

Improvement: using citation. While a ranking should try to measure the quality of research, a ranking based on publication counts really measures quantity [6]. One possible measurement of quality is the citation count for a paper. The difficulty involved in applying citation count prevents the wider usage of this measurement. Science Citation Index [7] provides a service for citation count, but it is not easily available, and it only covers major journals. CiteSeer is a citation service using automatic citation index based on citations found in web resources [2]. It covers many types of technical works, including conference proceedings. Both the ACM Digital Library [10] and IEEE Explore [4] have started to provide citation information. We are investigating the possibility of incorporating citation counts from these sources to improve the accuracy of our framework.

A very important decision in a publication-based ranking is selecting journals and conferences representative of a given field. To reduce controversy and improve accuracy, journals can be ranked by citations to decide what journals are among the best of a given field [9]. The citation count can be used, in addition to recommendations from field experts, as another indicator of the relative contribution from each journal and conference.

Improvement: incorporating complete affiliation. INSPEC only lists the affiliation of the first author. If the chosen scoring scheme for a ranking gives credit to multiple authors, the bibliographic data about papers written by authors from different institutions need additional manual editing to overcome this limitation. During our software engineering ranking, we manually edited about 800 entries of the total 1900 entries. The framework itself can handle multiple authors automatically after the data is ready. Since the ACM Digital Library has started providing complete affiliation information of each author for new publications, we will explore utilizing the library in future rankings.

5. USING THE FRAMEWORK

The ranking framework is a GUI application written in Java. It contains a collection of reusable classes. The major ones are Reference, Bibliography, InstitutionScore, and Ranking. The Reference class stores authors, affiliations, the title, and the origin of each publication. The Bibliography class contains all references used for the current ranking. The InstititionScore class collects scores earned by each institution based on references in the bibliography. The Ranking class decides what policies to use for the current ranking and conducts ranking based on those policies.

There are four steps to use this framework to perform a ranking. The first step is to prepare bibliographic data. The data can be retrieved from the INSPEC. We used the INSPEC interface available through the library of our university. The data retrieved are saved as text files. Each entry saved contains the title, the authors, the affiliation, and the origin of a paper. If the ranking requires a score distribution scheme giving credits to authors other than the first author, then the affiliation needs to be manually edited to include multiple affiliations. This can be a time consuming process. Otherwise, the saved data can be used directly without editing.

The second step is to load the bibliographic data into the framework. The data forms the basis for a ranking.

The third step is to determine the ranking. An evaluator chooses particular policies for the ranking, such as what the relative weights of the conferences and journals are, whether institutions or authors are ranked, what scoring distribution scheme is used, and how many years should be considered. The framework conducts the ranking according to these choices using the bibliographic data obtained in the second step.

To explore a different set of ranking policies, the evaluator can keep the current bibliography and simply create a new ranking. The framework automatically performs a new ranking in accordance with the new policies. The evaluator can continue experimenting using different choices until satisfactory results are achieved.

The final step is to save the ranking. The saved result contains the score and position of each institution or scholar. The bibliography used and the publications for a selected institution can also be saved.

6. CONCLUSION

Ranking based on publications is only one data point in a comprehensive evaluation process [5]. We provide an automatic and generic framework that can be used to rank research institutions and scholars based on publications. It can save labor for evaluators and provide them with more flexibility.

This framework is an enabler for future rankings. It serves as a platform allowing evaluators to experiment in different settings. It cannot replace the evaluators' methodology and professionalism in an objective ranking.

The framework and data used in this paper can be downloaded from http://www.isr.uci.edu/projects/ranking/.

7. ACKNOWLEDGEMENTS

The authors would like to thank Scott Hendrickson for fruitful discussions.

8. REFERENCES

[1] Computer Science Bibliography, http://liinwww.ira.uka.de/bibliography/

[2] CiteSeer, http://citeseer.nj.nec.com/cs

[3] DBLP, http://www.informatik.uni-trier.de/~ley/db/

[4] IEEE Explore, http://ieeexplore.ieee.org

[5] Geist, R., et al., *Computing Research Programs in the U.S.* Communications of the ACM, 1996. **39**(12): p. 96-99.

[6] Glass, R.L. and T.Y. Chen, *An Assessment of Systems and Software Engineering Scholars and Institutions (1998–2002).* Journal of Systems & Software, 2003. **68**(1): p. 77-84.

[7] Science Citation Index, http://www.isinet.com/products/citation/sci/

[8] INSPEC, http://www.iee.org/Publish/INSPEC/

[9] Katerattanakul, P., B. Han, and S. Hong, *Objective Quality Ranking of Computing Journals.*

Communications of the ACM, 2003. **46**(10): p. 111-114.
[10] ACM Digital Library, http://portal.acm.org
[11] Osuna, J.A., *Nrc Releases Data on Cs Program Rankings.* COMPUTING RESEARCH NEWS, 1995. **7**(5): p. 1.
[12] U.S. News & World Report, http://www.usnews.com/usnews/edu/college/rankings/rankindex_brief.php