Ecole Polytechnique Federale de Lausanne

Computer Science Department

Master Thesis

Ranking Scientific Publications Based on Their Citation Graph

Student:

Ludmila Marian

Supervisors:

Dr. Martin Rajman
EPFL - IC/IIF/LIA

Ing. Jean-Yves Le Meur
CERN - IT/UDS/CDS

April 3, 2009

Contents

List of Figures

List of Tables

Abstract

CDS Invenio is the web-based integrated digital library system developed at CERN. It is a suite of applications which provides the framework and tools for building and managing an autonomous digital library server. Within this framework, the goal of this project is to implement new ranking methods based on the bibliographic citation graph extracted from the CDS Invenio database. As a first step, we implemented the Citation Count as a baseline ranking method. The major disadvantage of this method is that all citations are treated equally, disregarding their importance and their publication date. To overcome this drawback, we consider two different approaches: a link-based approach which extends the PageRank model to the bibliographic citation graph and a time-dependent approach which takes into account time in the citation counts. In addition, we also combined these two approaches in a hybrid model based on a time-dependent PageRank. In the present document, we describe the conceptual background behind our new ranking methods, detail their implementation and provide a comprehensive analysis of the results obtained with the citation graph extracted from the CDS Invenio database. Our main contributions are: (i) a study of the currently available ranking methods based on a citation graph; (ii) the development of new ranking methods that correct some of the identified limitations of the current methods, such as considering all citations of equal importance, not taking time into account, or considering the citation graph as complete; (iii) a robust and scalable implementation of the aforementioned ranking methods; (iv) a detailed study of their key parameters. Our study reveals why the dumping factor used by the PageRank algorithm is not suited for ranking bibliographic data and why adding even a week time decay factor still has a strong impact on the final ordering of the documents.

Acknowledgments

I would first like to thank all the persons who made this Master Project at CERN possible: Dr. Martin Rajman for supervising my Thesis, Jean-Yves Le Meur, for accepting me in his team, and Martin Vesely for proposing this project to the EPFL students.
Secondly, I would like to express my deepest appreciation to my EPFL Master Thesis Supervisor, Dr. Martin Rajman for all the discussions that we had along the duration of the project and for continues feedback.
I would also like to express my gratitude to all CDS members, who quickly integrated me in the team, and gave me technical support in various areas. I am especially indebted to Jean-Yves Le Meur and Martin Vesely for the insightful talks about the conception of the new ranking methods, and to Tibor Simko, for fruitful discussions on the implementation details of my project.
Last but not the least, I would like to thank Travis Brooks for all his feedback regarding the results of my work.

Chapter 1

Introduction

CDS Invenio is the integrated digital library system developed and used at CERN. It is a suite of applications which provides the framework and tools for building and managing an autonomous digital library server. CDS Invenio can manage a very large collection of documents. An important part of the interaction between the end user and the system is the search engine, which gives access to all the publications. Having in mind that the average length of a query is 2 words, and that we are dealing with a large amount of data, it is relatively easy to see that the average recall per query cannot be analyzed by the users. Thus, in order to facilitate access to the desired documents, we must offer the user different choices in terms of ranking. Although there has been some research in the area of ranking based on the citation graph, no relevant implementations exist. Hence, the objective of this project is to construct and analyze new ranking methods based on the citation graph. In this chapter we elaborate further on the context of this project and on its main objectives.

The rest of the report is organized as follows. In Chapter 2 we review some of the work that has been conducted in the domains of citation analysis and ranking scientific publications. In Chapter 3 we give a detailed analysis of the CERN and Inspire bibliographic data sets and the extracted citation graphs. In Chapter 4 we analyze the Citation Count ranking method in order to have a baseline algorithm for our future experiments, since this is the only ranking algorithm based on the citation graph that is not influenced by any parametrization. In Chapter 5 and Chapter 6 we develop several link-based ranking methods by adapting the PageRank algorithm to the task of ranking scientific publications. In order to take into account time in the citation graph, we developed a time-dependent ranking method in Chapter 7. We also develop a hybrid ranking method based on a time-dependent PageRank in Chapter 8. Finally, we conclude in Chapter 9 by analyzing the results of the different algorithms and by identifying important data characteristics. By this, we try to provide insights and solutions concerning the task of ranking bibliographic data.

1.1 Context

1.1.1 CERN

"CERN, the European Organization for Nuclear Research, is one of the worlds largest and most respected centers for scientific research. Its business is fundamental physics, finding out what the Universe is made of and how it works. At CERN, the worlds largest and most complex scientific instruments are used to study the basic constituents of matter the fundamental particles. By studying what happens when these particles collide, physicists learn about the laws of Nature. The instruments used at CERN are particle accelerators and detectors. Accelerators boost beams of particles to high energies before they are made to collide with each other or with stationary targets. Detectors observe and record the results of these collisions. Founded in 1954, the CERN Laboratory sits astride the FrancoSwiss border near Geneva. It was one of Europes first joint ventures and now has 20 Member States. " [2]

What would be also important to add to this description is that more than 8000 scientists, half of the world's particle physicists, collaborate with CERN for their research. They represent more than 500 Universities and over 60 nationalities.

The World Wide Web (WWW) is one of the important discoveries made at CERN. The WWW was first thought by Tim Berners-Lee in 1989 as a solution to the problem of loss of information due to the high turnover of people at CERN. His original proposal [10] suggested to use Hypertext to link information systems across network such that anyone could have access to any important documents produced at CERN. The web has now extended worldwide and has become a primary component of IT.

1.1.2 CDS Invenio

CDS Invenio is the integrated digital library system developed and used at CERN [5]. It is a suite of applications which provides the framework and tools for building and managing an autonomous digital library server. CDS Invenio is developed and maintained by the CERN Document Server (CDS) section, which is part of the User and Document Services (UDS) group. Besides CERN, CDS Invenio is currently installed and in use by over a dozen scientific institutions worldwide, including EFPL [3].

At CERN, CDS Invenio manages over 500 collections of data, consisting of over 900,000 bibliographic records, covering preprints, articles, books, journals, photographs, and more, including 360,000 fulltext documents [1]. CDS Invenio is OAI-compliant to support the open dissemination of the documents. CDS Invenio also uses MARC 21 (and its XML derivative MARCXML) to store and process bibliographic meta-data. The collections of documents can have customizable portals to support community building. Additionally, CDS Invenio provides collaborative tools such as baskets or alerts for new specific documents.

CDS Invenio is a free, open source application (under the terms of the GNU General Public License) and it covers all aspects of digital library management. Its flexibility and performance make it a comprehensive solution for the management of document repositories of moderate to large size.

To get a better understanding of the CDS Invenio software, one must look at its architecture. *"The key feature of CDS Invenio's architecture lies in its modular logic. Each module embodies a specific, defined, functionality of the digital library system. Modules interact with other modules, the database and the interface layers. A module's logic, operation and inter-operability are extensible and customizable."* [5] A detail map of the CDS Invenio's workflow can be seen in Figure 1.1.

For a better understanding of the Figure 1.1, here is a brief description of

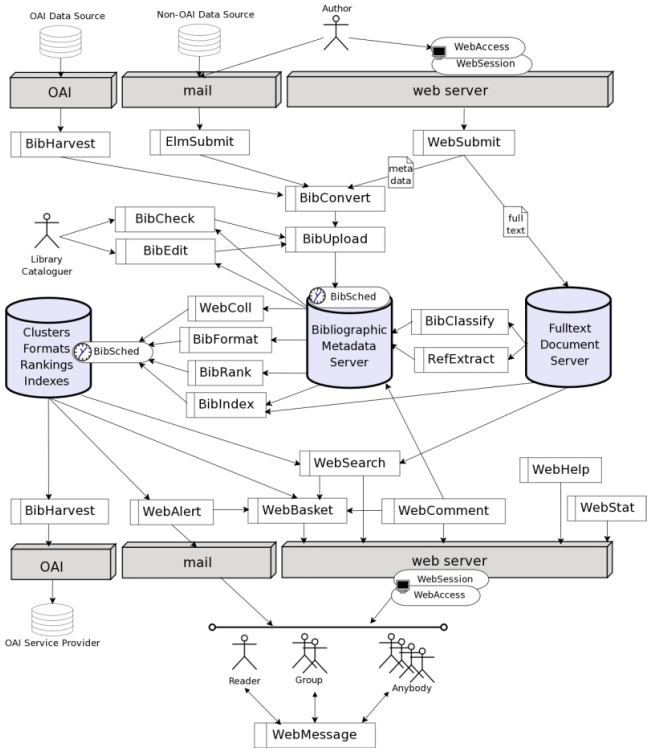

Figure 1.1: Document workflow in the CDS Invenio system [5]

some of the modules. More details can be found at
`http://invenio-demo.cern.ch/help/hacking/modules-overview`.

- **BibCheck** permits administrators and library cataloguers to automate various kind of tests on the metadata to see whether the metadata comply with quality standards.
- **BibClassify** allows automatic extraction of keywords from fulltext doc-

uments based on the frequency of specific terms, taken from a controlled vocabulary.

- **BibConvert** allows metadata conversion from any structured or semi-structured proprietary format into any other format, typically the MARC XML that is natively used in CDS Invenio.
- **BibEdit** permits one to edit the metadata via a Web interface.
- **BibFormat** is in charge of formatting the bibliographic metadata in numerous ways.
- **BibHarvest** represents the OAi-PMH compatible harvester allowing the repository to gather metadata from fellow OAi-compliant repositories and the OAi-PMH repository management.
- **BibIndex** does the word and phrase indexation of metadata, references and full text files.
- **BibMatch** compares new entries with the already existing ones for avoiding duplication of records.
- **BibSched** is the central unit of the system and it allows all other modules to access the bibliographic database in a controlled manner.
- **BibUpload** enters the new bibliographic data into the database.
- **ElmSubmit** is an email submission gateway that permits for automatic document uploads from trusted sources via email.
- **MiscUtil** is a collection of miscellaneous utilities that other modules are using.
- **WebAccess** is responsible for granting access to users for performing various actions within the system.
- **WebAlert** announces the user whenever a new document matching his/her personal criteria is inserted into the database.
- **WebBasket** enables the user of the system to store the documents he/she is interested in, in a personal basket or a personal shelf.
- **WebComment** provides a community-oriented tool to share information between the readers.
- **WebSearch** module handles user requests to search for certain words or phrases in the database.
- **WebSubmit** is a comprehensive submission system allowing authorized individuals (authors, secretaries and repository maintenance staff) to submit individual documents into the system.

1.1.3 BibRank

The ***BibRank*** module allows the user to set up various ranking criteria to be used by the search engine, when returning the query results. The methods currently implemented in CDS Invenio are the following:

Single tag rank One example of the use of this method is the journal impact factor ranking method which ranks documents using a configurable knowledge base.

Word Similarity/Similar Records This ranking method is weighting the records according to the importance of their content with respect to the query.

Word Frequency This ranking method assigns weights to the records based on the frequency and placement of the query terms.

1.2 Mission

1.2.1 Motivation

As described in the previous section, CDS Invenio can manage a very large collection of documents. A very important aspect of the interaction between the end user and the system is the search engine, that gives access to all the publications. Having in mind that the average length of a query is 2 words and also that we are dealing with a large amount of data, it is straight forward to conclude that the average recall per query is above the means of the user to analyze it. We can see a basic example in the Figure 1.2.

It is without any doubt that the user will end up looking at maybe the first 2 or 3 pages (30-40 results) if he/she is patient, if not, maybe just the first page, searching for the wanted document(s). If the wanted document is not there, there is the possibility of the user redefining the query, but there is also the possibility of the user starting to look for the wanted information in some other repositories. This is why we need to give the user the possibility of choosing the order in which the results are displayed (the ranking).

Since CDS Invenio has a rather large collection of documents with information regarding their citations, this is a perfect opportunity for experimenting with different ranking methods based on the citation graph. They will provide the CDS Invenio user with different possibilities of ranking the results of the search engine; they will add new features to the BibRank module; and, the last but not the least, they will provide interesting results for the scientific community. Although there has been some research conducted in the area of ranking based on the citation graph, nothing relevant has been implemented so far for the bibliographic data.

1.2.2 Objectives

This project has two main objectives:

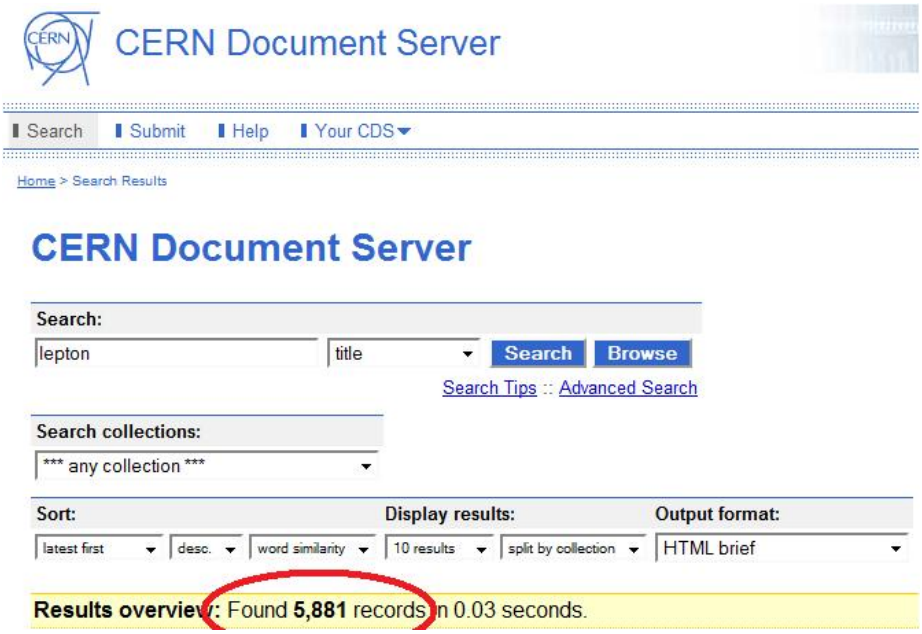

Figure 1.2: Screenshot of the CDS Invenio search engine

1. Construct and analyze new ranking methods based on the citation graph.
2. Implement and integrate the newly constructed ranking methods in the CDS Invenio software.

We defined several tasks for each of the aforementioned objectives:

Construct and analyze new ranking methods based on the citation graph

- analyze what has been done so far in the domain;
- analyze the citation graph;
- develop new ranking methods based on the citation graph;
- analyze the performance of the newly constructed ranking methods;
- construct a small web platform for comparing different ranking methods.

Implement and integrate the newly constructed ranking methods in the CDS Invenio software

- understand the architecture and implementation of the CDS Invenio software;

- implement the new ranking methods in Python;
- integrate the newly developed methods with the BibRank module;
- write the documentation for the source code.

Chapter 2

Related Work

In this chapter, we review some of the work previously conducted in the areas of citation analysis and ranking scientific publications. Since there is a high similarity between the bibliographic citation graph and the WWW graph, we first look at the current state of raking the results recalled in response to a user query. Second, we survey the current state of ranking in the other bibliographic repositories and existing analysis of the bibliographic citation graph. Last, but not least, we look at the current state of ranking scientific publications based on the citation graph.

2.1 State of the Art - Link-based ranking

First of all we should mention that the idea of this project started from the work of H. Wang and M. Rajman [19], on implementing a novel ranking method that combines the PageRank method with the TFIDF score of a web page with respect to a query. This work proves that combining linking information with content based weighting can give better results then each of them taken alone. Also, it presents insightful information about the theory behind the PageRank model and the link matrix.

Knowing the high similarity between the bibliographic citation graph and the WWW graph, when given the task of classifying publications one must look at the current state of raking the results recalled in response to a user query. One of the most popular ranking methods, also due to the impressive results achieved by the search engine that uses it, is PageRank. The method proposed by S. Brin, L. Page in [22] is based on a random surfer model as a stationary distribution of a Markov chain. The PageRank solution is a principal eigenvector of a linear system that can be found via the power method. There has been a lot of research done on the Google's ranking model. It is not our intention to provide a comprehensive study of this work. Still, we would like to mention the work of P. Berkhin, "A survey on PageRank Computing" [9] that presents insightful information about the theoretical foundations of the

PageRank method and different techniques for computing the weights.

There has been a lot of work done in the field of bibliometrics in general. Here we must recall the pioneer work of E. Garfield in citation analysis and citation indexing [17], [18], [16]. On a more recent note, there is an article published in the Journal of Neuroscience 2008 that presents an overview of the different ranking methods that extend Google's PageRank, based on the citation graph. They consider the works of Chen et al. [15] and Walker et al. [12] as being relevant for the task of ranking scientific publications based on the citation graph.

P. Chen et al. in [15] apply the Google PageRank algorithm on the citation graph to assess the relative importance of all publications in the Physical Review family of journals from 1893-2003. They identify some exceptional papers or "gems" that are universally familiar to physicists but have a lower number of citations with respect to other publications. They prove with different examples that applying PageRank is better at finding important publications then the simple citation count. They also argue about using a different dumping factor than the one used in the original PageRank algorithm.

Further analysis on the usefulness of PageRank method in the context of ranking scientific publications is carried out in Chapter 5.

2.2 State of the Art - Time-dependent ranking

Also in the context of our project is the work of D. Walker et al. [12]. They introduce a new ranking method called CiteRank to account for strong aging characteristics of citation networks. They modify Google's PageRank algorithm by initially distributing random surfers exponentially with age, in favor of more recent publications. By this, they try to model the behavior of researchers in search for new information. They test their model on all American Physical Society publications and the set of high-energy physics theory (hep-th) preprints. They find the parameters for their model by trying to maximize the correlation between the CiteRank output and the download history.

There has been some research activity also in the area of "temporal link analysis", mostly done on WWW pages. In [7] the authors present several aspects and uses of the time dimension in the context of Web IR. K. Berberich et al. [8] argue that the freshness of web content and link structure is a factor that needs to be taken into account in link analysis when computing the importance of a page. They provide a time-aware ranking method and through experiments they conclude on the improvements broth by it to the quality of ranking web pages. They test their approach on the DBLP data set but with the scope of ranking researchers rather than publications.

Further analysis on the usefulness of a time-dependent ranking method in the context of ranking scientific publications is carried out in Chapter 7 and Chapter 8.

2.3 Physics Bibliographic Citation Graph

There has been several analysis of the Physics bibliographic citation graph mainly done by S. Redner in [21] and [20]. In [21] he examines the distribution of citations for papers published both in 1981 in journals which are catalogued by the Institute for Scientific Information (783.339 papers), and also in 20 years of publications in Physical Review D (24.296 papers). He concludes that the number of citations to a given paper versus its citation rank is consistent with a power-law dependence for leading rank papers, with exponent close to -1/2. In [20], the author presents a detailed study on the Physical Review data, including the accuracy of data and the distribution and age structure of citations. He also gives multiple examples of interesting individual citation history, highlighting the importance of the citations age.

Another important repository of physics publications aside Invenio is the NASA Astrophysics Data System (ADS). In [14] and [13] they present insightful information about their system and the different ranking method that they are using. They combine different information such as citations, and content with readership logs and they create the Second Order Bibliometric Operators. They argue that these operators improve substantially the accuracy in literature queries. Although their objective is different then ours, in the sense that they are ranking scientists rather than publications, their contribution is quite interesting and definitely worth mentioning in the context of our work.

Also in the area of ranking authors rather then ranking scientific papers is the work done by D. Mimno and A. McCallum in [11]. They present a probabilistic model that ranks authors based on their influence in particular areas of scientific research. Their model combines several sources of information: citation information between documents as represented by PageRank scores, authorship data gathered through automatic information extraction, and the words in paper abstracts.

2.4 Conclusions

There has been a lot of research conducted in the areas of citation analysis and ranking scientific publications. We consider that having a new data set unexplored so far (the citation graph extracted from CDS Invenio), gives as the opportunity to further investigate the possible advantages and disadvantages of applying different ranking methods to the bibliographic citation graph. In this sense, we continue the work done on applying PageRank to rank scientific publications. We consider that, so far, only the advantages have been truly emphasize while the possible shortcomings have been neglected. We carry out a more detailed analysis in Chapter 5 and Chapter 6. We also focus on applying a time-dependent ranking method to the bibliographic citation graph. We think that the work done so far on this approach, although very insightful, lacks a complete analysis of the time factor and its influence on the ranking results. Further details can be found in Chapter 7 and Chapter 8.

Chapter 3

Experimental Framework

3.1 Introduction

For our experiments, we used two large sets of bibliographic data (not completely disjoint): Inspire (`http://inspire.cern.ch`) containing 500,000 High Energy Physics (HEP) documents and CDS WebDev (`http://cdswebdev.cern.ch`) containing 200,000 CERN and HEP documents. The main difference between these two sets of data is that the former is a human edited repository while the latter is automatically generated.

These sets of data contain Physics publications as well as the citations between them. In this work, we refer at the *citation graph* as having the publications as nodes and the citations as directed links.

The data analysis we emphasized two main aspects: the first is *data connectivity*, i.e. the number of publications that have no citations, the number of publications that have no references, and the second is *data quality*, i.e. the number of publications missing from the data set. All these characteristics are very important in our work, because they determine the correctness and robustness of different ranking methods.

3.2 Data

3.2.1 Inspire data

Inspire is the project name of a new High Energy Physics information system which will integrate present databases and repositories to host the entire corpus of the HEP literature and become the reference HEP scientific information platform worldwide. It is a common project between CERN, DESY, FERMILAB and SLAC. More details can be found at `http://inspire.cern.ch`.

The Inspire data contains almost half a million publications, with a total

number of eight million citations. Close to 25% of these papers[1] are not cited by any other paper in the system. Also, 16% of these papers do not reference any other paper in the system. Since, in general, we are dealing with a bibliographic subset of data, we are aware of the fact that we can not have a complete citation graph. So, having only 16% of the papers with no reference could be considered a fairly complete system.

One thing worth mentioning when talking about Inspire data is that it is a human edited repository, meaning that the entries are validated by an authorized person. This means that the error rate for the citation extraction is close to 0 % which makes this data set a real asset and our primary testing environment.

In terms of how many papers are missing from the data set, we discovered that on average a paper is missing 9 references. We were able to compute this information because, even if the system does not have all the references from one paper, it is still displaying them on the web interface. Comparing this number (9 missing references/paper) with the average number of references that are in the system (20 references/paper) we could say that in terms of quality this data set is fairly good.

Here is a detailed analysis of the Inspire data:

Total number of papers	490.730
Total number of citations	7.976.155
Papers that are not cited	123.550
Papers that have no references	80.653
Average number of received citations per paper	21.70
Average number of references per paper	19.45
Average number of references not in the system per paper	9.12

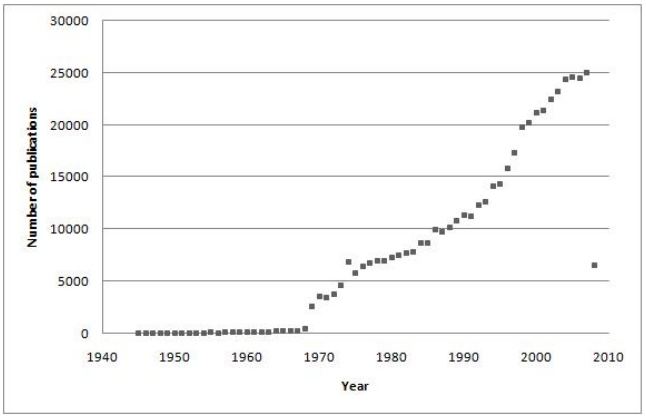

Figure 3.1: Number of publications in Inspire per year

[1]In this report we are using for simplicity also the name of "paper" when referring to a publication.

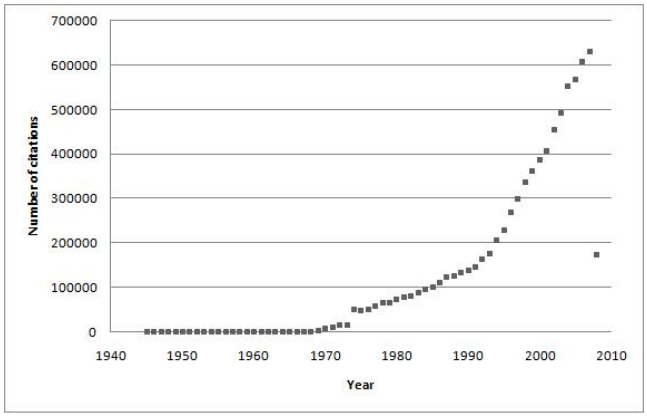

Figure 3.2: Number of citations in Inspire per year

3.2.2 CDS WebDev data

CDS is the CERN Document Server that contains documents about CERN and High Energy Physics. Out of more than 900,000 bibliographic records indexed by CDS, there is a subset of documents considered for different experiments with ranking methods. Details about different ongoing experiments can be found here: `http://cdswebdev.cern.ch`.

This subset of data contains 200,000 documents with 1,4 million citations. Out of these papers, around 35% of them contain no references and 20% of them are not cited by any other paper in the system. This percentage is caused by either very new papers or by very old papers. For measuring the quality of the data, we calculated the number of references that are missing from the system. On average, each paper is missing 28 references. Compared with the number of existing references per paper (9) the number of missing links is quite high. One reason for this is that currently CDS is using an automated references extractor and no human editing.

Here is a detailed analysis of the CDS WebDev data:

Total number of papers	204.230
Total number of citations	1.371.568
Papers that are not cited	38.570
Papers that have no references	57.689
Average number of received citations per paper	8.28
Average number of references per paper	9.36
Average number of references not in the system per paper	28.73

3.2.3 Atlantis data

Atlantis Institute of Fictive Science constitutes the data for a small demo site put in place for the CDS platform users: `http://invenio-demo.cern.ch`. This data set contains only 20 papers, connected by 18 citation. Although it is a rather small sample set of the CDS data, it is perfect for preliminary tests and theory proofs.

A snapshot of the citation graph generated with the Atlantis set of bibliographic data can be seen in the Figure 3.3.

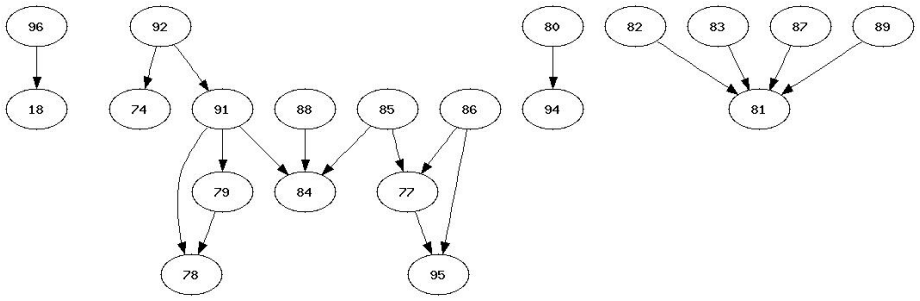

Figure 3.3: Citation graph based on the Atlantis bibliographic records

We tried the same graphical approach also for analyzing the two main data sets: the Inspire data and the CDS WebDev data. Unfortunately, we could not find a suited software that was able to compute a comprehensive graphical image of a such large graph. Just to give you an idea on the difficulty of the task, a directed graph generated with 500 random links (citations) from CDS WebDev can be seen in the Figure 3.4.

One of our major concerns when analyzing the data was the *correctness* of it. While the intuition is that we are dealing with a directed acyclic graph (DAG), we discovered that this is not entirely true. Since the system contains preprints (drafts of scientific papers that have not yet been published in a peer-reviewed scientific journal) as well as published papers and conference proceedings, it might happen in some cases that future work is cited. On top of this, there are also some cases where a paper is citing itself. We try to eliminate these last types of anomalies as often as possible. Still, the first class of problems is harder to permanently eliminate, and even though theoretically impossible, the "future work" citation is sometimes legitime. For these reasons, we build our algorithms on top of a general directed graph and not on top of DAG.

Since the Inspire set of data is the best available set, and since our work in ranking methodologies will be integrated in the Inspire project, our further experiments were intensively tested on this set of bibliographic data, unless otherwise stated.

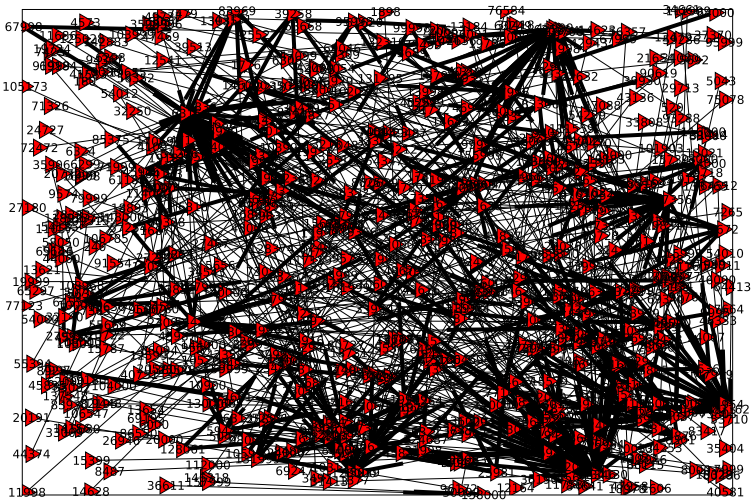

Figure 3.4: Citation graph based on 500 random citations from CDS WebDev

3.3 Conclusions

In this section, we analyzed our experimental data sets, Inspire and CDS WebDev.

The Inspire data set contains almost half a million publications, with a total number of 8 million citations. Approximately 25% of the documents are not cited by any other document in the system, while approximately 16% of the documents have no references. Since, in general, we are dealing with a bibliographic subset of data, we are aware of the fact that we can not have a complete citation graph. In terms of the number of papers missing from the data set, we discovered that on average a paper is missing 9 out of 30 references. So, with only 16% of the documents without references and a relatively small number of missing documents, Inspire can be considered a fairly complete system.

On the other hand, the CDS WebDev data set contains 200,000 documents with 1.4 million citations. Approximately 20% of these documents are not cited by any other document in the system while 35% of the documents have no references. On average, each document is missing 28 out of 37 references. One reason for this low number of available references is that currently CDS is using an automated references extractor and no human editing. Combined, these results indicate the low quality of this data set. Still, since the future of bibliographic repositories is the automation of the data extraction, one must consider these drawbacks in the development and analysis of the ranking methods based on the citation graph.

Concerning data correctness, we discovered that, while the general intuition is that the citation graph is a directed acyclic graph, this is not true for the two data sets in question. For this reason, our ranking algorithms are designed for general directed graphs.

Chapter 4

Baseline Ranking: Citation Count

4.1 Introduction

One of the most frequent measures of a scientist's reputation was the ***Citation Count***. It is one of the simplest ways of ranking scientific work, and, at the same time, it is also a good way of discovering appreciated publications in the scientific community. In this chapter, we present the concepts behind the Citation Count method and the results generated by applying this method to our main data set.

We present this method in order to provide a baseline algorithm for our future experiments. Note that this is the only ranking algorithm based on the citation graph that is not influenced by any parametrization.

4.2 Conceptual Background

Having a directed graph $G = (V, E)$, where the vertices,V, represents the publications and the edges,E, citations between the publications, the question is what is the weight associated with each $v_i \in V$?

$$W(v_i) = \sum_{j=1}^{j=|V|} W(e_{ji}),\ \forall i = \overline{1, |V|}$$

$W(e_j)$ represents the weight of each edge, and it is defined as follows:

$$W(e_j) = \begin{cases} 1 & if\ j \rightarrow i, \\ 0 & otherwise \end{cases}$$

4.3 Results

After applying this ranking method to our data set, we were able to compute a ***Top of Most Cited Publications*** (see Table 4.2 for Top 10 Most Cited Publications). If we take a closer look at this table, we can see that the second and the third publications have similar citation counts. Interestingly, the third ranked paper is almost 25 years younger than the second ranked paper, thus it had a much smaller time window to accumulate citations. So, one might argue, that the third paper should be ranked higher than the second one. But, when looking at the distribution of the citations for the second paper (`http://hep-inspire.net/record/37116/citations`) we see that, although it was published over 35 years ago, it continues to be heavily cited. For this reason, one might argue that this paper is more important for the scientific community than the third one. Therefor, in order to get a clear picture of the influence of a publication in its domain, we must also take into consideration not just the citation counts but also the different characteristics of the citations, like their publication date or their importance.

The time distribution of the most cited papers reveals that the ranking is not necessarily biased towards publications from a specific period of time. As it can be seen in Figure 4.1, the top 100 papers are from a time range of over 50 years.

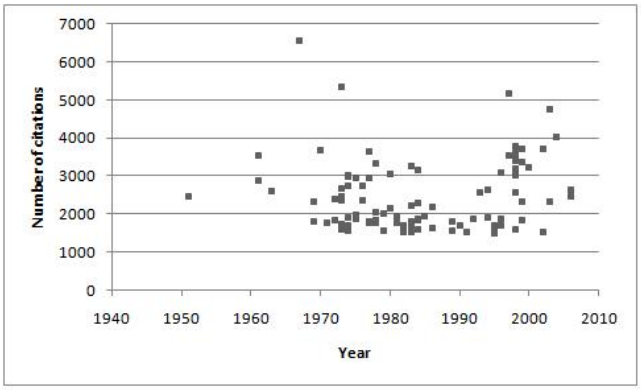

Figure 4.1: Time distribution for the top 100 most cited papers

Citation Count ranking revealed that although the top cited papers accumulate thousands of citations, there is a quite high number of papers with a very low number of citations. In the Table 4.1 you can see the number of papers from the Inspire data set that have less than 10 citations. This high number is caused by either really new papers that did not have enough time to acquire a high number of citations or by really old papers that are forgotten. The total number of publications with less than 10 citations represents approximately 70% of the data set. These publications will not only remain undiscovered due

Number of Citations	Number of Publications
0	123550
1	61630
2	39188
3	28939
4	22737
5	18304
6	15649
7	12893
8	11454
9	10147

Table 4.1: Number of papers with a low number of citations

to the insignificant number of accumulated citations, but also be very hard to differentiate due to the small difference in the citation count. In order to find the truly relevant publications, one needs to take also into consideration different characteristics of the citations, like their importance or their publication date.

Rank	***Publication***	***Citation Count***
1	A Model of Leptons: Weinberg, Steven (1967)	6565
2	CP Violation in the Renormalizable Theory of Weak Interaction: Kobayashi, Makoto (1973)	5351
3	The Large N limit of superconformal field theories andsupergravity: Maldacena, Juan Martin (1997)	5162
4	First year Wilkinson Microwave Anisotropy Probe (WMAP) observations: Determination of cosmological parameters: Spergel, D.N. (2003)	4752
5	Review of particle physics. Particle Data Group: Eidelman, S. (2004)	4033
6	Measurements of Omega and Lambda from 42 high redshift supernovae: Perlmutter, S. (1998)	3787
7	A Large mass hierarchy from a small extra dimension: Randall, Lisa (1999)	3719
8	Review of particle physics. Particle Data Group: Hagiwara, K. (2002)	3711
9	Weak Interactions with Lepton-Hadron Symmetry: Glashow, S.L. (1970)	3671
10	Asymptotic Freedom in Parton Language: Altarelli, Guido (1977)	3649

Table 4.2: Top 10 publication by citation count

4.4 Conclusions

Although the Citation Count ranking is a natural measure of a publication's impact, there are many cases where this method fails to reveal a good picture of the influence of a publication in its domain. We can identify at least two main shortcomings of using the Citation Count as the main ranking method for publications:

- It treats all the citations equally ignoring the differences in importance of the citing papers.
- It does not take into account time, i.e. it underestimates the importance of newly acquired citations, and overestimates the importance of the older ones.

These drawbacks motivated us to study alternative metrics. To overcome the first shortcoming, we designed ranking methods derive from PageRank (see Chapter 5 and Chapter 6) that, therefor, take into consideration the notion of citation importance. For correcting the second drawback, we designed various time dependent ranking methods: a time-dependent Citation Count (see Chapter 7) and a hybrid method based on a time-dependent PageRank (see Chapter 8).

Chapter 5

Link-based Ranking: PageRank on the Citation Graph

5.1 Introduction

In order to find a ranking method for scientific publications that takes into consideration the importance of the citations, one must take a step back and analyze the nature of the citation graph. The citation graph is a network similar to the World Wide Web (WWW). A successful solution to the problem of ranking the web pages is PageRank, an algorithm introduced in 1998 by Brin and Page. The PageRank algorithm is based on a random surfer model, and may be viewed as a stationary distribution of a Markov chain. The solution is a principal eigenvector of a linear system that can be found via the power method.

In the previous section, we noted that one of the major shortcomings of the Citation Count method is the fact that it treats all citations equally. We consider PageRank as being suited for the task of ranking publications because it overcomes this drawback by giving higher weight to publications that are cited by important papers. Because of this, the algorithm identifies a large number of modestly cited publications that contain important results for the scientific community.

In this section, we discuss the advantages and disadvantages of applying this method to the bibliographic data set.

5.2 Theoretical Background

In this section we discuss about the theory behind the PageRank algorithm. We introduce some general concepts of stochastic processes and Markov chains, and

we discuss about their application in the PageRank case. We follow the theory by Allen [6] and by Papoulis/Pillai [4].

5.2.1 Stochastic Processses

Definition 1. (Stochastic process) *A family of random variables* $\{X(t), t \in T\}$ *is called a* ***stochastic process****.*

Definition 2. (Index set) *For a stochastic process* $\{X(t), t \in T\}$ *the set* T *is called the* ***index set*** *of the process.*

Definition 3. (State space) *The* ***state space*** *of a stochastic process* $\{X(t), t \in T\}$ *is the set of values that any of the random variables* $X(t)$ *can assume.*

The distinct values in the state space are are actually the states of the stochastic process. The index states and the space states can be discrete or continuous. This gives birth to four categories of stochastic processes. Further in our work, we consider stochastic processes with discrete index set and discrete state space, if not otherwise stated.

5.2.2 Markov Chains and Random Walks

Definition 4. (Markov property) *A stochastic process is said to have the* ***Markov property*** *if for any set of* $n+1$ *values out of the index set* $t_1 < t_2 < ... < t_{n+1}$ *and any set of* $n+1$ *states* $\{x_1, x_2, ..., x_{n+1}\}$ *out of the state space, the following holds*

$$P[X(t_{n+1}) = x_{n+1} | X(t_1) = x_1, X(t_2) = x_2, ..., X(t_n) = x_n] = P[X(t_{n+1}) = x_{n+1} | X(t_n) = x_n]$$

In other words, having the Markov property means that, given the present state (t_n), future states (t_{n+1}) are independent of the past states ($t_i, i = \overline{0, n}$). The description of the present state fully captures all the information that could influence the future evolution of the process. Future states will be reached through a probabilistic process instead of a deterministic one. At each step the system may change its state from the current state to another state, or remain in the same state, according to a certain probability distribution. The changes of state are called *transitions*, and the probabilities associated with various state-changes are called *transition probabilities.*

A stochastic process having the Markov property is called a ***Markov process***. A special kind of Markov process is a Markov chain where the system can occupy a finite or countably infinite number of states such that the future evolution of the process, once it is in a given state, depends only on the present state and not on how it arrived at that state. Both Markov chains and Markov processes can be discrete-time or continuous-time.

Markov processes are named after A. A. Markov (1856-1922), who introduced this concept for discrete parameter systems with a finite number of states

(1907).

The transition probability $P[X_{n+1} = j | X_n = i]$ describes the probability that the process being in state i at time n transitions to state j at time $n+1$. Thus the transition probability possibly depends on the time n. We only consider *stationary* transition probabilities that do not depend on time.

Definition 5. (Time homogeneous Markov chain) *A Markov chain with stationary transition probabilities is called time homogeneous.*

For a time homogeneous Markov chain, the transition probabilities can be defined using a square matrix P which is then called transition probability matrix. The entry P_{ij} of the matrix is the probability of making a transition from state i to state j at any time. Consequently, the matrix P has to be row-stochastic, meaning that every row must describe a valid probability distribution. Therefore, all entries of P must be positive, $P_{ij} \geqslant 0$ and each row i must satisfy:

$$\sum_{j=0} P_{ij} = 1$$

Definition 6. (Irreducible Markov chain) *A Markov chain is irreducible if for every pair of states i and j there exist the integers n and m, such that $P_{ij}^{(n)} > 0$ and $P_{ji}^{(m)} > 0$.*

The recurrence values of a state i is defined as the k-s, for which $P_{ii}^{(}k) > 0$ holds. In other words, it exists a finite number of states for which the process returns to the starting state with a positive probability. The period $d(i)$ of a state i is defined as the greatest common divisor of all recurrence values. If no recurrence values exist (it is imposable to return to the initial state), then $d(i) = 0$. States having a period of 1 are called *aperiodic*, and states having a period d(i) ¿ 1 are called *periodic*.

Definition 7. (Aperiodic Markov chain) *A Markov chain is aperiodic if all its states are aperiodic. In other words, the greatest common divisor of their recurrence values is 1.*

Theorem 1. ***(Ergodic Markov chain with finite number of states)*** *A discrete time Markov chain with a finite number of states that is irreducible and aperiodic is* ***ergodic****.*

Ergodicity has useful ramifications on the long-run behavior of a Markov chain. We call a probability distribution π (described as a vector) the *stationary state probability* of the Markov chain defined by the transition probability matrix P, if $\pi = \pi \cdot P$. Thus the vector describing the probability distribution is an eigenvector of the transition probability matrix, associated with the eigenvalue 1. Since, according to the largest eigenvalue of a stochastic matrix is always 1, this eigenvector is the *principal eigenvector* of the transition probability matrix. Consequently, if the initial state distribution is chosen to be π, the state probabilities after n steps do not change while n increases and are all equal to π.

Theorem 2. *For an ergodic Markov chain, unique stationary state probabilities π exist and they are independent of the initial state distribution $\pi^{(0)}$s.*

The unique stationary state probabilities can be obtained as the solution to the equation system of balance equations,

$$\pi_j = \sum_i \pi_i \cdot P_{ij}$$

$$\sum_i \pi_j = 1$$

Instead of solving the above equation system, we can solve the eigenvector problem $\pi = \pi \cdot P$. The power method is an iterative method for solving the eigenvector problem.

$$x^{\langle k+1 \rangle} = x^{\langle k \rangle} \cdot P$$

The iterates $x^{\langle k \rangle}$ are guaranteed to converge to the principal eigenvector of the matrix P. In practice, the multiplication will be stop when the difference between two consecutive iterates is below a certain threshold. The previous equation can be also written as follows:

$$x^{\langle k+1 \rangle} = x^{\langle 0 \rangle} \cdot P^k$$

where $x^{\langle 0 \rangle}$ is the initial state distribution These states (probabilities) will converge to a unique probability distribution, namely the unique stationary state probabilities, which is independent of the initial state distribution explaining the almost arbitrary choice of $x^{\langle 0 \rangle}$.

An example of a Markov chain is a *simple random walk* where the state space is a set of vertices of a graph and the transition steps involve moving to any of the neighbors of the current vertex with equal probability (regardless of the history of the walk).

5.2.3 PageRank

PageRank relies on the uniquely democratic nature of the web by using its vast link structure as an indicator of an individual page's value. In essence, Google interprets a link from page A to page B as a vote, by page A, for page B. But, Google looks at more than the sheer volume of votes, or links a page receives; it also analyzes the page that casts the vote. Votes cast by pages that are themselves "important" weigh more heavily and help to make other pages "important".[Google]

Although the PageRank model was introduced much earlier, it was made famous in 1998 by the founders of Google.com. It assigns weight to pages proportional with the importance of the papers that link to them. In this section we will look at some theory behind the PageRank method [9].

Let $G(V, E)$ be a directed graph with V being the publications, and E the citations between the publication, in the sense that $i \rightarrow j$ means i *cites* j. The

edges E can be associated with a sparse matrix $P^{n\times n}$, where $n =\mid V \mid$ is the number of publications. An element p_{ij} of the matrix can be defined as follows:

$$p_{ij} = \begin{cases} 1 & if\ i \to j, \\ 0 & otherwise \end{cases}$$

Note. In this work, the notation $i \to j$ means that there is a direct path from node i to node j, or in terms of bibliographic records, publication i is citing publication j.
The *out-degree* of a node i is the number of outgoing links:

$$deg(i) = \sum_{j=1}^{n} p_{ij}$$

To transform the matrix P in a transition matrix all the entries must correspond to probabilities. In this scope, we transform the entries of the matrix P as follows:

$$p_{ij} = \begin{cases} p_{ij}/deg(i) & if\ deg(i) > 0, \\ 0 & if\ deg(i) = 0 \end{cases}$$

In other words, this means that from a certain publication, one can follow with an equal probability any of the references.

For the i-s with $deg(i) > 0$, the matrix P is row-stochastic, meaning that the i-row elements sum to one.

Let us consider the following *random surfer model*. A surfer travels along the directed graph (for the simplicity of writing we will consider a female surfer). If at a step k she is located in a node i, then at the next step $k+1$ she moves uniformly at random to any out-neighbor of i, j. Nodes can be considered as n states of a Markov chain with a transition matrix P. Given a distribution

$$p^{\langle k\rangle} = p_i^{\langle k\rangle}$$

of probabilities for a surfer to be at a page i at a step k, the probability for being at a page j on the next step is proportional to

$$p_j^{\langle k+1\rangle} = \sum_{i\to j} p_i^{\langle k\rangle}/deg(i) = \sum_i p_{ij} p_i^{\langle k\rangle}$$

If we look at this globally:

$$p^{\langle k+1\rangle} = P^T p^{\langle k\rangle} \tag{5.1}$$

(If all pages would have out-links, $p_j^{\langle k+1\rangle}$ would be indeed a probability distribution.) From this equation we see that the more incoming links a page has, the higher its importance. The importance of a page j is also proportional to the importance of its in-neighbors and inversely proportional to the neighbors out-degrees.

Definition 8. (PageRank vector) *A **PageRank vector** is a stationary point of the transformation (5.1) with nonnegative components (a stationary distribution for a Markov chain).*

$$p = P^T p$$

The problem with this definition is that the matrix P has zero rows for *dangling nodes* (nodes that have no outgoing links), since for them $deg(i) = 0$, while a Markov chain transition matrix has to be row-stochastic. Dangling nodes serve as *sinks*: they accumulate weight without distributing it to others. Different remedies have been explored. One approach suggests getting rid of dangling nodes, since they do not influence any other node. Unfortunately, deleting dangling nodes generates new dangling nodes. Another alternative is to consider adding deleted dangling nodes on final iterations. Another option is to add a self-link to each dangling node and, after computation, to adjust the constructed PageRank. Another approach suggests introducing an ideal node (sink) with a self-link to which each dangling node links. The most widely used method, also proposed in [22] modifies the matrix P by adding artificial links that uniformly connect dangling nodes to all the nodes in the graph. This would mean:

$$P' = P + t \cdot v^T \tag{5.2}$$

with

$$t = \begin{cases} 1 & if\ deg(i) = 0, \\ 0 & if\ deg(i) > 0 \end{cases}$$

and $v = (1/n)e^T$, where e^T is the row vector of all 1s and n is the number of nodes in the graph.

With this trick, any dangling node is now linked to any other node in the graph, and the probability for reaching this nodes is equally distributed.

After doing all this, the question is when does the process in 5.1, also known as *power method* or *power iteration* converge to the solution? Or, in other words, is the Markov chain defined by P ergodic? As a short reminder from the previous theory, it is sufficient to show that the Markov chain is *irreducible*, *time homogeneous*, *aperiodic* and *finite in the number of states* to conclude that it is ergodic, and so the power method converges to a principal eigenvector that has positive elements, and the principal eigenvalue $\lambda_1 = 1$.

Time Homogeneous The transition probabilities are stationary by definition, which implies that the Markov chain is time homogeneous.

Finite The number of states of the Markov chain is equal to the number of nodes in the graph, which is obviously finite.

Aperiodic Due to the random jump every state has a small looping transition probability, and this means, the period of each state, which is the greatest common divisor of all recurrence values, is 1.

Irreducible A Markov chain is irreducible if all its states are mutually reachable, or in other words, there exists a directed path from each node to any other node. Up to now, our transition probabilities do not guarantee this condition.

To satisfy the *irreducible* property, there is another trick that needs to be done, to have a connection between any two nodes:

$$P'' = dP' + (1-d)ev^T \tag{5.3}$$

This transformation conserves the stochastic property (it transforms a probability distribution into another probability distribution). According to 5.3, from a non-dangling page a surfer follows one of the local out-links with probability d and jumps to some $j \in V$ with probability $(1-d)$.

d is called *damping factor* and in the literature concerning the web graph it usually has values in $[0.85, 1)$. d is a free parameter that controls the performance of the PageRank algorithm. d gives the fraction of random walks that continue to propagate along the links, while $(1-d)$ is the fraction of random walks re-injected into the network. The parameter d prevents all the influence on concentrating on the oldest papers. As we were mentioning earlier, the most used value for d is 0.85, value prompted by the observation that an random surfer will typically follow 6 hyperlinks $(1 - d = 1/6 = 0.15)$ before getting bored and starting a new search.

Chen et al. [15] demonstrate the use of PageRank in bibliographic networks. In this sense, PageRank can be seen as modeling a researcher who moves from paper to paper in the document collection. At each paper the researcher either follows a randomly chosen reference from the current paper or, with a probability of $(1-d)$ chooses a random paper from the collection. The PageRank of a given paper can be interpreted as the probability that a researcher will be reading that paper at any given moment. In their work, the authors suggested the use of a $d = 0.5$, representing the assumption that readers of scientific papers walk shorter paths on the citation graph, of average length 2.

To conclude, this is the formula for calculating the PageRank number of a certain paper i:

$$PR(p_i) = \frac{1-d}{n} + d \sum_{j, p_j \to p_i} \frac{PR(p_j)}{deg(j)} \tag{5.4}$$

5.3 Implementation

Problem: Given a graph $G(V, E)$, where $\mid V \mid = 490.730$ nodes, and $\mid E \mid = 7.976.155$ links, what are the PageRank weights for each node?

Having discussed about the theory behind PageRank, a solution for the problem above is to apply the iterative method. The constraints are related to *scalability* and *robustness*. We take advantage of the fact that the transition probability matrix is a very sparse one, so instead of storing the entire matrix, we are using dictionaries to keep the information. Here is the algorithm:

The iterative method proposes a multiplication between a matrix and a vector. This is usually done in $O(n^2)$. Since we are deling with a sparse matrix,

foreach i, j *in* V, $j \rightarrow i$ **do**
  $sparseDictionary[(i,j)] = dampingFactor \times \frac{1.0}{deg(j)}$
end
foreach i *in* V, $deg(i) = 0$ **do**
  add i to $danglingVector$
end
initialize weights
$converged = False$
while *not converged* **do**
  $newWeights = 0$
  foreach (i,j) *in* $sparseDictionary$ **do**
    $newWeights_i \mathrel{+}= sparseDictionary[(i,j)] \times weights_j$
  end
  $newWeights \mathrel{+}= \frac{dampingFactor}{n} \times \sum_{j \in danglingVector} weights_j + \frac{(1-dampingFactor)}{n} \times \sum_{j=1}^{n} weights_j$
  $W = newWeights - weights$
  $\epsilon = \frac{\sqrt{\|W\|}}{n}$
  if $\epsilon < convergeceThreshold$ **then**
    $converged = True$
  end
end

Algorithm 1: The PageRank algorithm

as mentioned before (16% of the rows and 25% of the columns are zeros), we were able to reduce the complexity of the problem to $O(n)$.

5.4 Results

For determining the best values for our parameters we experimented with different settings. One of the parameters that has the highest impact on the results is the *damping factor*. As discussed in the *Theoretical Background* section, there are different opinions in the literature for what value the damping factor should have. While a value of 0.85 (corresponding to 6 links on the link graph) seems to be preferable for the WWW, a value of 0.5 (corresponding to 2 links on the citation graph) seems better for the bibliographic citation graph. In this case, we present the results for three distinct values of the *damping factor*: 0.85, 0.70, and 0.50. We present the ***Top 10 publications*** for each of the values mentioned before in Tables 5.1, 5.2, and 5.3.

Analyzing these three tables, we can see that there is a serious difference between the PageRank outcome and the Citation Count outcome, at least in the top 10 results. The only publication that maintains its rank, no matter the ranking algorithm, is the first ranked paper. The PageRank method keeps in the top 10 positions only 2 (or 3, depending on the damping factor) publications

Rank	***Publication***	***Citation Count***	***Rank by Citation Count***
1	A Model of Leptons: Weinberg, Steven (1967)	6565	1
2	Ultraviolet Behavior of Nonabelian Gauge Theories: Gross, D.J., (1973)	2379	44
3	Weak Interactions with Lepton-Hadron Symmetry: Glashow, S.L., (1970)	3671	9
4	Reliable Perturbative Results for Strong Interactions?: Politzer, H.David, (1973)	2390	43
5	Confinement of Quarks: Wilson, Kenneth G., (1974)	3023	26
6	Radiative Corrections as the Origin of Spontaneous Symmetry Breaking: Coleman, Sidney R., (1973)	2472	40
7	Broken symmetries, massless particles and gauge fields: Higgs, Peter W., (1964)	1454	111
8	Axial vector vertex in spinor electrodynamics: Adler, Stephen L., (1969)	2332	47
9	Field Theories with Superconductor Solutions: Goldstone, J., (1961)	739	462
10	Supergauge Transformations in Four-Dimensions: Wess, J., (1974)	1373	130

Table 5.1: Top 10 publication by PageRank, when damping factor = 0.85

Rank	***Publication***	***Citation Count***	***Rank by Citation Count***
1	A Model of Leptons: Weinberg, Steven (1967)	6565	1
2	Weak Interactions with Lepton-Hadron Symmetry: Glashow, S.L., (1970)	3671	9
3	Confinement of Quarks: Wilson, Kenneth G., (1974)	3023	26
4	Ultraviolet Behavior of Nonabelian Gauge Theories: Gross, D.J., (1973)	2379	44
5	Reliable Perturbative Results for Strong Interactions?: Politzer, H.David, (1973)	2390	43
6	Radiative Corrections as the Origin of Spontaneous Symmetry Breaking: Coleman, Sidney R., (1973)	2472	40
7	CP Violation in the Renormalizable Theory of Weak Interaction: Kobayashi, Makoto, (1973)	5351	2
8	Pseudoparticle Solutions of the Yang-Mills Equations: Belavin, A.A., (1975)	1978	56
9	Axial vector vertex in spinor electrodynamics: Adler, Stephen L., (1969)	2332	47
10	Broken symmetries, massless particles and gauge fields: Higgs, Peter W., (1964)	1454	111

Table 5.2: Top 10 publication by PageRank, when damping factor = 0.70

Rank	***Publication***	***Citation Count***	***Rank by Citation Count***
1	A Model of Leptons: Weinberg, Steven (1967)	6565	1
2	Confinement of Quarks: Wilson, Kenneth G., (1974)	3023	26
3	Weak Interactions with Lepton-Hadron Symmetry: Glashow, S.L., (1970)	3671	9
4	CP Violation in the Renormalizable Theory of Weak Interaction: Kobayashi, Makoto, (1973)	5351	2
5	Ultraviolet Behavior of Nonabelian Gauge Theories: Gross, D.J., (1973)	2379	44
6	Radiative Corrections as the Origin of Spontaneous Symmetry Breaking: Coleman, Sidney R., (1973)	2472	40
7	Reliable Perturbative Results for Strong Interactions?: Politzer, H.David, (1973)	2390	43
8	Pseudoparticle Solutions of the Yang-Mills Equations: Belavin, A.A., (1975)	1978	56
9	Maps of dust IR emission for use in estimation of reddening and CMBR foregrounds: Schlegel, David J., (1997)	3556	13
10	Axial vector vertex in spinor electrodynamics: Adler, Stephen L., (1969)	2332	47

Table 5.3: Top 10 publication by PageRank, when damping factor = 0.50

that are also in top 10 by Citation Count while the rest are publications with a lower number of citations.

To be able to have a global vision of the differences between these ranking methods, we used the ***Spearman's rank correlation coefficient*** to calculate the correlation between PageRank and Citation Count as well as the correlation between PageRank with different damping factors:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

where

$d_i = x_i - y_i$ = the difference between the ranks of corresponding values X_i and Y_i,

n = the number of values in each data set (same for both sets).

You can find the correlation coefficients in the Tables 5.4 and 5.5.

Ranking Methods	***Spearman Coefficient***
Citation Count vs. PageRank(d=0.85)	0.93
Citation Count vs. PageRank(d=0.70)	0.93
Citation Count vs. PageRank(d=0.50)	0.94

Table 5.4: The Spearman correlation coefficients between Citation Count and PageRank

Ranking Methods	***Spearman Coefficient***
PageRank(d=0.50) vs. PageRank(d=0.85)	0.99601
PageRank(d=0.50) vs. PageRank(d=0.70)	0.99759
PageRank(d=0.70) vs. PageRank(d=0.85)	0.99701

Table 5.5: The Spearman correlation coefficients between PageRank with different dumping factor

Studying the Spearman correlation coefficients between the different rankings generated with the three chosen values for the dumping factor, we can conclude that at the global scale, the difference is almost undetectable. A Spearman coefficient so close to the value of 1 signifies almost identical rankings. This conclusion is also sustained by the Top 10 results identified with each of the rankings (Tables 5.3, 5.2, 5.1). As it can be observed, there is a bit of reordering among the Top 10 papers, but the difference is not impressive. What strikes, is the difference between the PageRank results and the Citation Count results. Here we can really see some interesting movements of publications in the Top 10 lists. Still, as we can see from Table 5.4, the overall rankings are not really that different.

Another interesting thing to analyze is the distribution of the ranks over the time. Since we know that a higher damping factor is boosting old papers rather than new ones, we are interested to see if we can detect this kind of behavior also in our data. For this, we plotted the distribution in time for the Top 100 papers ranked with PageRank. The results are displayed in Figure 5.1. Indeed, we can see that for a damping factor corresponding to two links on the citation graph, the age of the top 100 papers decreases. If for a damping factor corresponding to six links on the citation graph we have a large concentration of top papers in the 1970-1980 period, when decreasing the number of the likes followed by the random surfer in the citation graph, we see a shifting of the top papers towards the 1990-2000 period, which is indeed what we wore hoping to see. For this reason, we consider that a dumping factor of 0.5 is more appropriate for ranking scientific publications than a factor of 0.85.

In terms of performance, the PageRank weight converge faster for a lower value of the damping factor. In the Table 5.6 you can see the number of steps needed for each of the settings to converge. Still, for any of the configurations, the weights vector converges in an impressively low amount of time: 3min30s on an average computer.

d	***iterations***	ϵ
0.50	4	8.28e-05
0.70	6	5.85e-05
0.85	7	8.30e-05

Table 5.6: The number of iterations needed for reaching a stable state

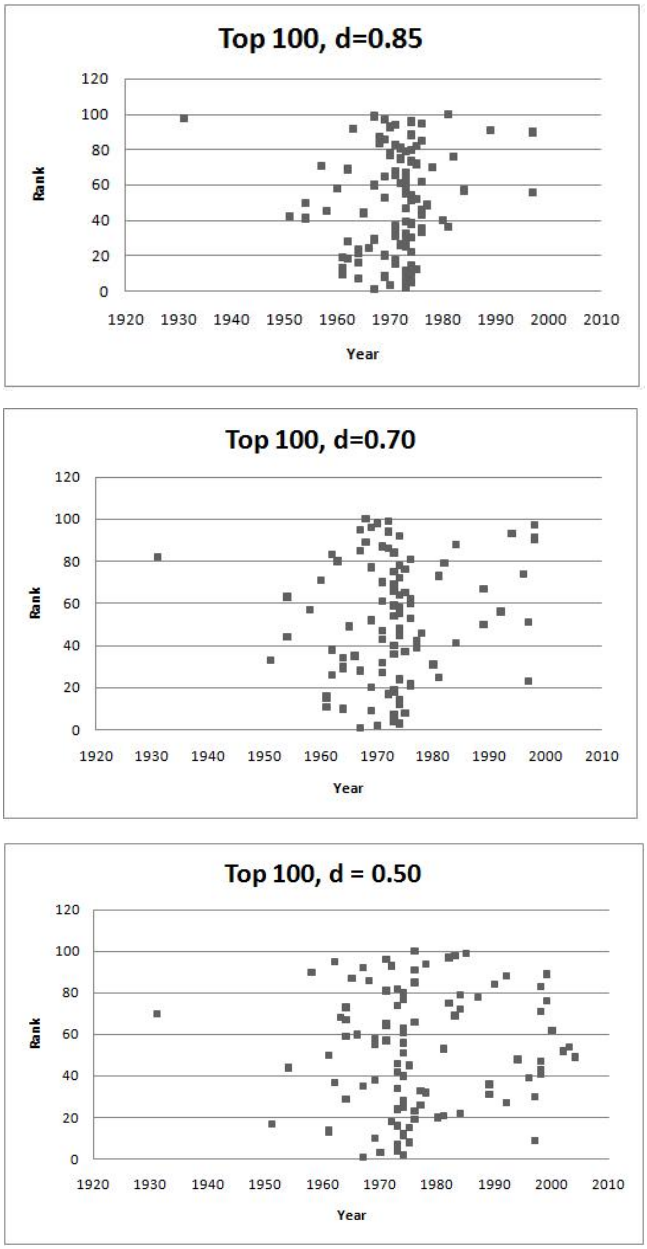

Figure 5.1: Time distribution for the top 100 publication, ranked with PageRank

5.5 Conclusions

PageRank is one of the most popular link-based ranking methods. In order to make it better suited for the bibliographic citation graph, we first modified the setting of the dumping factor. Indeed, when ranking web pages, it is usually considered that a random surfer follows on average 6 links in the link before getting "bored" and restarting the search. Such a setting does not really apply for the scientific community. It is rather believed that a scientist follows on average 2 links in the citation graph before restarting the search. Although our experiments show that the general ranking results remain quite similar when using different dumping factors, a dumping factor corresponding to 2 links in the citation graph should still be preferred for scientific publications, due to the fact that it is not entirely biased towards older publications.

One of the main advantages of this method is the fact that it weighs the publications accordingly to the importance of their citations, bringing to light some very insightful publications that would not have been discovered with the Citation Count method. Thus, it associates each publication with an "all-time achievement" rank. On the other hand, we can also identify two shortcomings of this ranking method:

- Because we are dealing with an *incomplete* citation graph, some anomalies occur. For example, a paper that only has one reference in the citation graph, will propagate all its weight to the lucky reference. If this paper is also a very highly regarded paper (it has a high PageRank weight), then the only reference will accumulate a lot of artificial weight. For this reason, one must think of some corrections for the algorithm that remove this artificial boost in terms of weight. We propose a fix of this drawback by extending the actual method to account for the missing citations (see Chapter 6).

- Because it calculates an all-time achievement weight, the method tends to give older papers more popularity, to the detriment of highly regarded new papers. In order to achieve a ranking where new publications can be ranked as high as older ones, one must design an algorithm that accounts also for the age of each publication/citaton. For this, we designed time dependent ranking methods (see Chapter 7 and Chapter 8).

Chapter 6

Link-based Ranking with External Citations

6.1 Introduction

As discussed previously, the PageRank algorithm was initially designed for the WWW. Several distinctions between the WWW and the citation graph must be made starting with the fact that while WWW is a closed graph (contains all the existing links, or at least a very large part of them), the citation graph is quite incomplete. As we sow in Chapter 3, the Inspire data set is missing on average 9 out of 30 references per paper while the CDS WebDev data set is missing on average 28 out of 37 references per paper. While for the Inspire data, these missing links represent just a small percentage, for the CDS WebDev data they represent almost 75%. In the context of applying the PageRank algorithm, this means that instead of distributing the weight to 37 references, a node is distributing its weight only to 9. This further means that these 9 papers receive much more weight then expected. So, we end up with a phenomena of "artificial inflation of weights".

For fixing this error we developed a new ranking method that accounts for the external citations. This new method assumes the existence of an "external authority" that accumulates weight from all the nodes in our graph, proportionally with the missing citations, and also feeds back into the network a certain percentage of its weight. With this method, we assure the correct propagation of the weight through the network.

6.2 Theoretical Background

Before discussing the theory behind our model, let us prove its usefulness with an example. First, let us look at a graphical demonstration of the PageRank

model, taken from `http://en.wikipedia.org/wiki/PageRank` (Figure 6.1)

The example in the Figure 6.1 illustrates the benefic effect of PageRank:

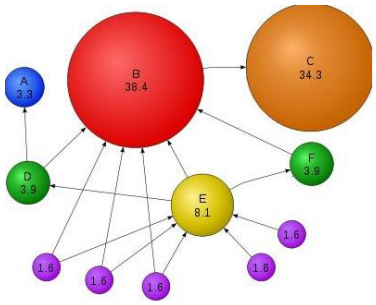

Figure 6.1: PageRank model for a simple network

you do not need to be cited a lot, you just need to be cited by important nodes. Still, the problem is a bit more complicated when applying this method on the citation graph: we want to give high weights to papers cited by authorities but in the same time, we do not want the "artificial inflations" of the weights generated by the fact that a really powerful publication has only a small percentage of its references in the database, and sow, it ends up distributing its weight just to them. In our illustrated example, a case of artificial inflation is node C.

As discussed previously, for fixing this error we assume the existence of an "external authority" that accumulates weight from all the nodes in our graph, proportionally with the missing citations, and also feeds back into the network a certain percentage of its weight. With this method, the node C from our illustrated example will disappear.

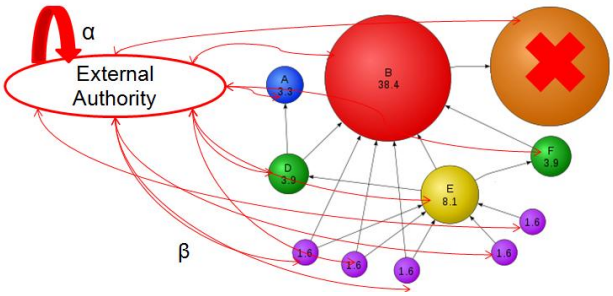

Figure 6.2: Correction of the PageRank model for a simple network

More formally, let us consider a graph $G = (V, E)$ and a matrix $M^{n \times n} = m_{ij}$ non-negative, where $n = \mid V \mid$. M is defined as quasi-stochastic, meaning that for each line i, either $\sum_{j=1}^{n} m_{ij} = 1$ or $\sum_{j=1}^{n} m_{ij} = 0$. The last possibility,

combined with the fact that the matrix is positive, implies that all the elements from the line are 0s. We define a stochastic vector $A^{1\times n} = a_j, \sum_{j=1}^{n} a_j = 1$.

Let us also consider a variable $a \in (0,1)$ and a vector $B^{n\times 1} = b_i$ such that,

$$\forall i \begin{cases} b_i = 1, & if \sum_{j=1}^{n} m_{ij} = 1 \\ b_i = 0, & otherwise \end{cases}$$

We define a matrix $T^{(n+1)\times(n+1)} = t_{ij}$ as follows:

$$T = \begin{bmatrix} t_{11} & A^{1\times n} \\ B^{n\times 1} & M^{n\times n} \end{bmatrix}$$

$$\begin{array}{lll} t_{11} & = 1-a & \\ t_{1j} & = a \times a_{j-1}, & \forall j > 1 \\ t_{i1} & = b_{i-1}, & \forall i > 1 \\ t_{ij} & = (1-b_{i-1}) \times m_{(i-1)(j-1)} & \forall i,j > 1 \end{array}$$

We can demonstrate that matrix T is *stochastic*.

$$i = 1 \Rightarrow \sum_{j=1}^{n+1} t_{ij} = (1-a) + \sum_{j=2}^{n+1} a \times a_{j-1} = 1 - a + a \times \sum_{j=1}^{n} a_j = 1 - a + a = 1$$

$$\forall i > 1 \Rightarrow \sum_{j=1}^{n+1} t_{ij} = b_i + \sum_{j=1}^{n} ((1-b_i) \times m_{ij}) = b_i + (1-b_i) \sum_{j=1}^{n} m_{ij} = 1$$

This proves that each row of the matrix T sums up to 1, which implies that T is stochastic. More, T can be considered a transition probability matrix. If we can prove that the Markov chain defined by T is ergodic, we prove that the power method converges to a principal eigenvector that has positive elements, and the principal eigenvalue $\lambda_1 = 1$. Just as a reminder, to prove that the Markov chain defined by T is ergotic, we have to prove that the it is: *time homogeneous, finite, aperiodic and irreducible.*

Time Homogeneous The transition probabilities are stationary by definition, which implies that the Markov chain is time homogeneous.

Finite The number of states of the Markov chain is equal to the number of nodes in the graph plus one, which is obviously finite.

Aperiodic By construction, $t_{11} = 1 - a, a \in (0,1)$, so we have at least one $t_{ii} > 0$.

Irreducible A Markov chain is irreducible if all its states are mutually reachable, or in other words, there exists a directed path from each node to any other node. If we can assure that the vectors A and B are strictly positive, then this will be enough for the Markov chain to be irreducible (all the nodes will reach node 1, and node 1 will be connected with all the nodes).

Going back to our ranking method, and to the Figure 6.2, we can explain the matrix T as follows:

- Matrix $M^{n\times n} = m_{ij}$ represents the adjacency matrix for our graph G.

$$m_{ij} = \begin{cases} 1, & if\ i \to j \\ 0, & if\ i \nrightarrow j,\ but\ deg(i) > 0 \\ 1, & if\ deg(i) = 0 \end{cases}$$

- The first entry in the matrix, t_{11} represents the "External Authority" node.

$$t_{11} = 1 - \alpha, \alpha \in (0, 1)$$

- Vector $A^{1\times n} = a_j$ represents the weights given by the "External Authority" to the nodes in the graph. We consider that each node should receive an equal amount of weight, $a_x = a_y, \forall x, y = \overline{1, n}$.

$$a_j = \frac{1}{n}, \forall j = \overline{1, n}$$

- Vector $B^{n\times 1} = b_i$ represents the weights received by the "External Authority" from the graph's nodes. Let us consider e_i being the number of external references for the publication i.

$$b_i = \beta \times max\{1, e_i\}$$

Note: There has been a slight change in the original formula of matrix B in order to guarantee the irreducibility, in the sense that we will still guarantee T to be stochastic, but the terms of B will have to be strictly positive.

Now we can construct the final T such that it is stochastic:

$$\begin{array}{lll} t_{11} & = 1 - \alpha & \\ t_{1j} & = \frac{\alpha}{n}, & \forall j > 1 \\ t_{i1} & = \frac{\beta\times max\{1,e_i\}}{\beta\times max\{1,e_i\}+\sum_{j=1}^{n} m_{ij}}, & \forall i > 1 \\ t_{ij} & = \frac{m_{ij}}{\beta\times max\{1,e_i\}+\sum_{j=1}^{n} m_{ij}}, & \forall i, j > 1 \end{array}$$

Intuitively, α quantifies how much of the external weight is re-injected into the network and β represents the fraction between an external citation and an internal one. We consider that, if a publication is not in the data set, it means that it values less then the ones already inserted in the database. We know that

this is not the case for all the missing citations, and we apologize if we hurt someone's feelings. But since we know nothing about the missing citations, except their number, it is fair give them lower weight in comparison with the existing ones.

In the further sections, we will present the implementation of this method and its results.

6.3 Implementation

In the "Theoretical Background" section above we proved that T can be considered a transition probability matrix and that the Markov chain defined by T is ergodic. Having proven this, we implicitly demonstrated that the power method converges to a principal eigenvector that has positive elements, and the principal eigenvalue $\lambda_1 = 1$. The eigenvector values are nothing else than the weight corresponding to each of the nodes in the graph.

The constraints of the implementation are related to *scalability* and *robustness*. We take advantage of the fact that the transition probability matrix is a very sparse one, so instead of storing the entire matrix, we are using dictionaries to keep the information.

The iterative method proposes a multiplication between a matrix and a vector. This is usually done in $O(n^2)$. Since we are deling with a sparse matrix, as mentioned before (16% of the rows and 25% of the columns are zeros), we were able to reduce the complexity of the problem to $O(n)$.

$sparseDictionary[(0,0)] = 1 - \alpha$
foreach j *in* V **do**
  $sparseDictionary[(j+1,0)] = \frac{\alpha}{n}$
  if $j \nrightarrow ExtAuthority$ **then**
    $sparseDictionary[(0,j+1)] = \frac{\beta}{n+\beta}$
  end
  if $j \rightarrow ExtAuthority$ **then**
    if $deg(j) > 0$ **then** $sparseDictionary[(0,j+1)] = \frac{\beta \times ext_j}{\beta \times ext_j + deg(j)}$
    if $deg(j) = 0$ **then** $sparseDictionary[(0,j+1)] = \frac{\beta \times ext_j}{\beta \times ext_j + n}$
  end
end
foreach i, j *in* V, $j \rightarrow i$ **do**
  $sparseDictionary[(i+1,j+1)] = \frac{1 - sparseDictionary[(0,j+1)]}{deg(j)}$
end
foreach i *in* V, $deg(i) = 0$ **do**
  $danglingVector[i] = \frac{1 - sparseDictionary[(0,i+1)]}{n}$
end
initialize weights
$converged = False$
while *not converged* **do**
  $newWeights = 0$
  foreach (i,j) *in* $sparseDictionary$ **do**
    $newWeights_i \mathrel{+}= sparseDictionary[(i,j)] \times weights_j$
  end
  $newWeights[1 : n+1] \mathrel{+}=$
  $\sum_{j \in danglingVector} danglingVector[j] \times weights_j$
  $W = newWeights - weights$
  $\epsilon = \frac{\sqrt{\|W\|}}{n}$
  if $\epsilon < convergeceThreshold$ **then**
    $converged = True$
  end
end

Algorithm 2: The Link-based Ranking with External Citations algorithm

6.4 Results

As discussed in the "Theoretical Background" section, this algorithm is tailored for data sets that have an incomplete citation graph. Although the Inspire data set is not really a candidate, for consistency reasons we present here the Top 10 results when ranking with this method in Table 6.1.

Comparing these results with the ones in Table 5.1 we see that they are actually quite similar, which is somehow normal. A data set that is almost

Rank	*Publication*	*Citation Count*	*Rank by Citation Count*
1	A Model of Leptons: Weinberg, Steven (1967)	6565	1
2	Ultraviolet Behavior of Nonabelian Gauge Theories: Gross, D.J., (1973)	2379	44
3	Weak Interactions with Lepton-Hadron Symmetry: Glashow, S.L., (1970)	3671	9
4	Reliable Perturbative Results for Strong Interactions?: Politzer, H.David, (1973)	2390	43
5	Radiative Corrections as the Origin of Spontaneous Symmetry Breaking: Coleman, Sidney R., (1973)	2472	40
6	Broken symmetries, massless particles and gauge fields: Higgs, Peter W., (1964)	1454	111
7	Confinement of Quarks: Wilson, Kenneth G., (1974)	3023	26
8	Axial vector vertex in spinor electrodynamics: Adler, Stephen L., (1969)	2332	47
9	Field Theories with Superconductor Solutions: Goldstone, J., (1961)	739	462
10	Supergauge Transformations in Four-Dimensions: Wess, J., (1974)	1373	130

Table 6.1: Top 10 publication by Link-based Ranking with External Citations

complete is very little influenced by the external citation, and so, the weight of the publications is influenced mainly by the citation graph. We want to remind you that this method is basically designed for data sets that lack a high number of citations in order to correct "artificial inflations" of the weight.

A really important step in defining a final version of our algorithm is finding the right values for the tow parameters α and β. In order to analyze how they influence the final outcome of the ranking we calculated the Spearman Correlation Coefficient (SCC) between our new ranking method with different settings of α and β (between 0 and 1 with 0.1 step), and the PageRank, for the CDS WebDev data set. *Note:* The SCC between the PageRank and the Citation Count is 0.81 for the CDS WebDev data set.

Table 6.2 presents the aggregated results after 200 experiments (for each $\alpha, \beta \in (0, 1)$, with a step of 0.1). As we can see, the α parameter can be chosen arbitrary, because it does not seem to have much influence in the ranking order. One thing to consider when choosing this parameter is the convergence speed. While for $\alpha = 0.1$ the algorithm converges in approximately 15 steps (15 iterations), for $\alpha = 0.5$ it converges in approximately 5 steps. Also, looking at the exact number for SCC, when $\alpha = 0.5$ the correlation is either the highest (for $\beta = 0.1$) or the lowest (for $\beta = 1.0$), both with respect to the PageRank method.

α	β	SCC with PageRank	SCC with Citation Count
$\alpha \in (0,1)$	$\beta = 0.1$	0.97	0.89
$\alpha \in (0,1)$	$\beta = 0.2$	0.94	0.91
$\alpha \in (0,1)$	$\beta = 0.3$	0.92	0.92
$\alpha \in (0,1)$	$\beta = 0.4$	0.91	0.92
$\alpha \in (0,1)$	$\beta = 0.5$	0.89	0.93
$\alpha \in (0,1)$	$\beta = 0.6$	0.88	0.93
$\alpha \in (0,1)$	$\beta = 0.7$	0.87	0.93
$\alpha \in (0,1)$	$\beta = 0.8$	0.87	0.93
$\alpha \in (0,1)$	$\beta = 0.9$	0.86	0.93
$\alpha \in (0,1)$	$\beta = 1.0$	0.85	0.93

Table 6.2: Spearman Correlation Coefficient between PageRank/Citation Count and Ranking with External Citations

6.5 Conclusions

In this chapter we describe our extension of the PageRank algorithm. The novelty of this new ranking method is that is specifically tailored for the citation graphs that are missing a significant number of citations. Since most of the bibliographic databases are incomplete, we believe that this method is better suited for ranking scientific publications.

In order to correct one of PageRank's shortcomings, namely the artificial inflation of some of the nodes weight we introduce a new node called "external authority" that is the place holder for all the missing citations. The addition of this node introduces two new parameters: α, the weight that the external node redirects into the network and β, the fraction between an external citation and an internal one. We consider that, if a publication is not in the data set, it means that it values less then the ones already inserted in the database. Our experimental analysis showed that α only influences the rate of convergence of the iterative algorithm and has little impact on the general reordering while β is the one that truly makes a difference in the outcome of the ranking method. For $\beta \in [0.1, 0.5)$ the outcome of the new ranking method is highly correlated with the PageRank results, and less correlated with the Citation Count results. On the other hand, for $\beta \in [0.5, 1]$ the correlation with the PageRank method drops, while the correlation with the Citation Count remains approximately constant. We advise for the use of a β lower than 0.5 since in this case the results will be less correlated with the citation counts and enough correlated with the PageRank as to assume that the artificial inflation problems are resolved.

We believe Link-based Ranking with External Citations to be a better candidate than Citation Count or PageRank for the task of ranking scientific publications because: (i) it inherits from PageRank its ability to take into account the citations with weights representing their importance, and thus, fixing one of the main shortcomings of the Citation Count method; (ii) it further corrects one of PageRank's shortcomings, namely the artificial inflation of some of the

weights. In the end, our new ranking method is enough correlated with the PageRank method as to assume that it inherits its usefulness and in the same time it corrects its shortcomings.

Chapter 7

Time-dependent Citation Counts

7.1 Introduction

One of the two main drawbacks of the Citation Count method is that it does not take into account the time dynamics of the citation graph, i.e. it underestimates the importance of newly acquired citations. To overcome this shortcoming, we introduce the notion of *time-dependent citation counts*. In this context, the weight of a publication i is defined as: $weight_i = \sum_{j,j \to i} e^{-w(t_{present} - t_j)}$ where $t_presen t$ is the present time and t_j is the publication date for document j^{th}.

Furthermore, this introduces the time decay parameter ($w \in (0, 1]$), which quantifies the notions of "new" and "old" citations (i.e. publications with ages less than the time decay parameter would be considered "new"; publications with ages larger than the time decay parameter would be considered "old"). The larger the time decay parameter is, the faster we "forget" old citations.

7.2 Theoretical Background

The new formula for calculating the weigh of each node depends on the present time, so when a new publication is added to the database, the weights of the publications that are cited by it will have to by recalculated from scratch. Since this adds a lot of computational overhead, we would like to be able to calculate these weights iteratively.

Let us consider $F_n(t)$ to be the weight of document D at the moment when it has n citations:

$$F_n(t) = \sum_{i=1}^{n} e^{-w(t-t_i)}$$

$$\begin{aligned} &= \sum_{i=1}^{n}(e^{-wt}e^{wt_i}) \\ &= e^{-wt}\sum_{i=1}^{n}e^{wt_i} \\ &= e^{-w(t-t_n)}\sum_{i=1}^{n}e^{-w(t_n-t_i)} \\ &= e^{-w(t-t_n)}G(n) \end{aligned} \tag{7.1}$$

where n is the current moment in time, i are the past moments when citations occurred and $G(n) = \sum_{i=1}^{n} e^{-w(t_n-t_i)}$ is what we call the "*citation function*".

$$\begin{aligned} F_{n+1}(t) &= e^{-w(t-t_{n+1})}\sum_{i=1}^{n+1}e^{-w(t_{n+1}-t_i)} \\ &= e^{-w(t-t_{n+1})}(\sum_{i=1}^{n}e^{-w(t_{n+1}-t_i)} + 1) \\ &= e^{-w(t-t_{n+1})}(\sum_{i=1}^{n}e^{-w((t_{n+1}-t_n)+(t_n-t_i))} + 1) \\ &= e^{-w(t-t_{n+1})}(e^{-w(t_{n+1}-t_n)}\sum_{i=1}^{n}e^{-w(t_n-t_i)} + 1) \\ &= e^{-w(t-t_{n+1})}(e^{-w(t_{n+1}-t_n)}G(n) + 1) \\ &= e^{-w(t-t_{n+1})}G(n+1) \end{aligned} \tag{7.2}$$

In short, combining Equations 7.1 and 7.2 we obtain:

$$F_n(t) = \sum_{i=1}^{n} e^{-w(t-t_i)} = e^{-w(t-t_n)}G(n)$$

where:

$$G(1) = 1$$
$$G(n+1) = 1 + e^{-w(t_{n+1}-t_n)}G(n), \forall n > 1$$

For a document D, in order to iteratively calculate its weight we have to store: t_{last} = time of the last citation for the document D, and G_{last}, the citation function used for calculating the current weight of the document $F_{last}(t)$. If a new citation arrives at t_{new} we have two possibilities (considering the fact that in a digital library a paper might be added after a long time after it is published,and because of this, the citations that he creates might be older than the citations that are already in the system):

$$\begin{aligned} if\ t_{new} \geq t_{last} &\Rightarrow \begin{cases} G_{last} \leftarrow 1 + e^{-w(t_{new}-t_{last})}G_{last} \\ t_{last} \leftarrow t_{new} \end{cases} \\ if\ t_{new} < t_{last} &\Rightarrow G_{last} \leftarrow G_{last} + e^{-w(t_{last}-t_{new})} \end{aligned}$$

This demonstrates that there is an iterative way of calculating the weight of a document when a new citation arrives, without having to recalculate it from scratch.

7.3 Results

An important aspect that needs to be carefully considered when using a time-dependent ranking method is the time decay parameter, w. It is the only quantifier for the "freshness" of the results. In this section we try to analyze the impact of the decay factor on the stability of the final rankings and also on the stability of certain ranges of ranks.

Let us first take a look at the Spearman Correlation Coefficient (SCC) between Citation Count with and without time decay (Table 7.1).

Methods	**SCC**
Citation Count vs. Citation Count with time decay = 1 year	0.792
Citation Count vs. Citation Count with time decay = 2 year	0.821
Citation Count vs. Citation Count with time decay = 5 year	0.887

Table 7.1: Spearman Correlation Coefficient between different settings of Citation Count

In this case the Spearman Correlation Coefficient is not really helpful, in the sense that, it just tells us that the Citation Count with a high time decay is converging to the Citation Count without any time decay. This is already ensured by the definition of the time decay factor: the higher the time decay is, the less we "forget" old citations. A more powerful measure in this case would be the "*locality of changes*". This should tell us if adding a time decay has a global impact on the ranking (i.e. the tail is promoted to the head and the other way around) or if it is rather local (i.e. there are certain windows in the ranking where there is some reshuffling).

Let us consider s as being the *stability factor*:

$$s = \frac{\mid \{d \mid rank_d(t), rank_d \in window\} \mid}{windowSize}$$

where $rank_d(t), rank_d$ are the ranks of publication d, the first when using a time-dependent ranking method and the last when using the non-decayed ranking method.

Using the stability factor we want to determine what windows of the ranking are suffering the most from different time decay parameters. For this we are building dynamic windows as follows: we are splitting the rank range by consecutive powers of 2, until we either reach a rank window of size less than 100 or the stability factor goes below a certain minimum threshold (0.3 in our

experiments). We should mention that we remove the publications with 0 citations, since their weight and rank will not be influenced by any ranking method. We constructed a chart for each value of the time decay factor(1 year, 2 years, 5 years, 10 years, 20 years, 40 years) (Figure 7.1). The interpretation of these charts is that whenever we have a zone with a lot of activity (a lot of points), that zone is quite stable at a high level and needs to be broken into small intervals to reach the instability threshold. On the other hand, when we have a zone with low activity, that means that the stability of the corresponding window is low also at a high level, so if we would split it in smaller windows, the stability will drop even lower then the threshold.

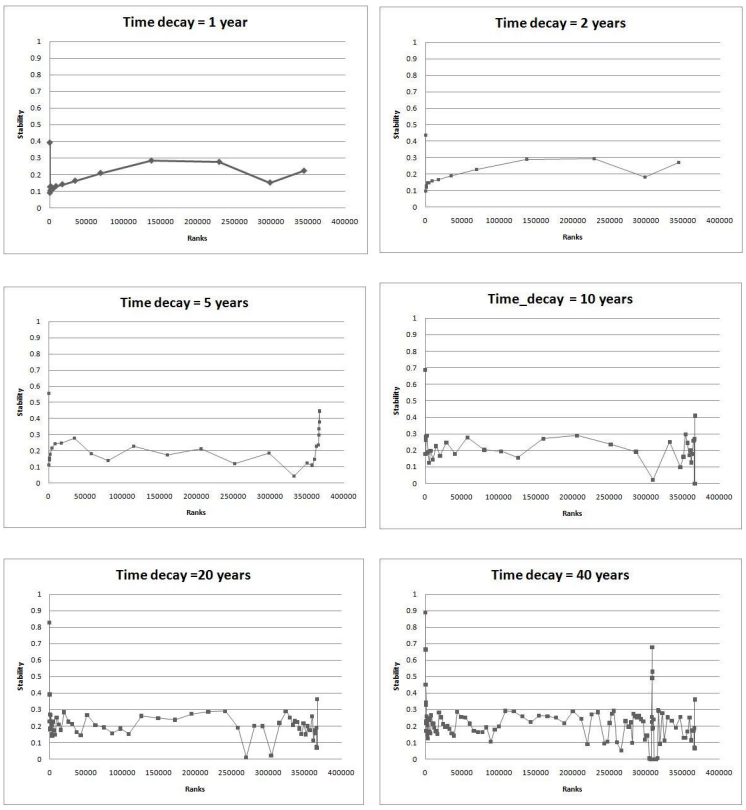

Figure 7.1: Stability of Time-dependent Citation Count

An interesting thing to observe in the Figure 7.1 is that the head of the ranks is usually more stable than the rest. Also, when looking at the ranks obtained with a time decay of 40 years, we see that the new ranks are quite stable at the high level. But, interestingly, even with such a large time decay the ranks are still reshuffled, but in small windows.

Up to now we sow that the stability between a non-decayed ranking and a decayed one is really low, meaning that the order of the documents is seriously modified. Taking the non-decayed citation count as reference, we are further trying to see how much are the documents promoted/demoted when using time decayed methods. For this we analyze the cumulative curves generated by the difference between rankings of the same papers (Figure 7.2). A value in these charts represents the number of nodes in the graph that have been promoted/demoted with less then a certain value. The numbers have been normalized to the dimension of the graph (in our case they are actually normalized with the number of nodes i, for which $deg(i) > 0$).

The first thing that we observe in the Figure 7.2 is that the *time-dependent ranking methods are promoting more publications than demoting.* Secondly, and as a consequence of the first observation, the *time-dependent ranking methods are demoting strongly than promoting.* Also, we can easily observe that, when increasing the time decay parameter the amount by which the papers are demoted/promoted is decreasing. Further, we observe that even with a high time decay parameter we still see promotions and demotions, meaning that we will have to go even further with the time decay parameter to see the ranks converging. We generated Tables 7.2 and 7.3 to get a clearer view on the actual numbers concerning the demotion/promotion.

Table 7.2 contains information about the percentage of publications promoted or demoted. As you can see, the stable percentage (publications that have the same position in both time decayed and non-decayed rankings) is almost the same regardless the time decay parameter. This means that even a

Time decay	Promotion	Demotion	Stable
1 year	54.1%	45.8%	0.1%
2 years	55.8%	44.1%	0.1%
5 years	60.6%	39.3%	0.1%
10 years	62.7%	37.2%	0.1%
20 years	63.4%	36.5%	0.1%
40 years	60.0%	39.9%	0.1%

Table 7.2: Promotion vs. Demotion [%]

Time decay	90% Promotion	90% Demotion	Avg. Promotion	Avg. demotion
1 year	155440	-195265	74888	-88568
2 years	131642	-179833	63961	-81019
5 years	84801	-154946	43342	-66846
10 years	50917	-112646	27317	-45919
20 years	26744	-64231	14293	-24783
40 years	12743	-29728	7039	-10584

Table 7.3: Promotion vs. Demotion [values]

slight time decay will probably reshuffle the results.

Table 7.3 shows exact values for the average promotion/demotion, 90% promotion/demotion (the value with which 90% of the documents are promoted / demoted). The maximum promotion / demotion is +/-367179.

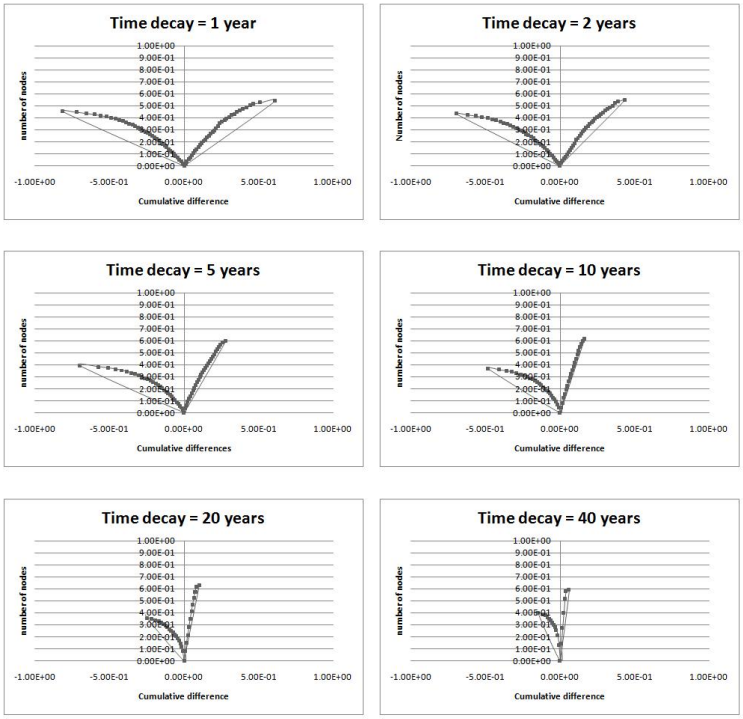

Figure 7.2: Cumulative differences between ranking with and without time decay

7.4 Conclusions

The Time-dependent Citation Count method is a tentative solution to one of the main drawbacks of the Citation Count method, namely that it does not take into account the time dynamics of the citation graph. The new time-based ranking method has one important parameter that controls the freshness of the top results: the time decay parameter. A close analysis of this parameter revels several interesting properties:

- Adding a week time decay factor (in our experiments, as week as 40 years) to a ranking method will still have an impact on the final ordering of the documents.

- Adding a strong time decay factor to a ranking method reveals the most popular publications at the current moment in time.
- Adding any time decay factor to a ranking method results in having more publications increasing their rank than decreasing it. At the same time, the gain in rank for the promoted publications is less strong than the decrease in rank for the demoted ones.

We consider the Time-dependent Citation Count to be a better candidate for the task of ranking scientific publications than Citation Count, because it takes into consideration the age of the citations. Thus, adding even a weak time decay factor, the time-dependent ranking can still differentiate between an important old publication that acquired a large number of citations over a long period of time, and a new publication, that, although important for the scientific community, did not have enough time to acquire as many citations as the old one, in the favor of the latter. Still, this method inherits one major shortcoming from the Citation Count method, i.e. it does not take into consideration the different importance of each citation. To overcome this, we developed the Time-dependent Link-based Ranking as a combination of Time-dependent Citation Counts and Link-based Ranking (Chapter 8).

Chapter 8

Time-dependent Link-based Ranking

8.1 Introduction

Until now, we focused on developing new ranking methods for the bibliographic data set that overcome one of the two main disadvantages of the Citation Count method, either that all the citation are treated equally disregarding their importance (Link-based Ranking: Chapter 5, Chapter 6), either that it does not account for the time dynamics of the citation graph (Time-dependent Ranking: Chapter 7). In this section, we concentrate on combining the Link-based Ranking methods with the Time-dependent Citation Count in order to develop a ranking method capable of correcting both these drawbacks.

The idea of the Time-dependent Link-based Ranking method is to distribute the random surfers exponentially with age, in favor of more recent publications. Every researcher, independently, is assumed to start his/her search from a recent paper or review and to subsequently follow a chain of citations until satisfied. In this way the effect of a recent citation to a paper is greater than that of an older citation to the same paper.

We consider the weight of each publication as being inversely proportional with its age: the younger the publication is, the more its citations will value. In this case, the initial probability of selecting the i^{th} paper in a citation graph will be given by:

$$p_i = e^{-w(t-t_i)}$$

where t is the present time, t_i is the publication date for document i^{th} and w is what we call the *time decay parameter*.

8.2 Implementation

As a reminder, this is the PageRank number of a certain document i:

$$PR(i) = \frac{1-d}{n} + d \sum_{j,j \to i} \frac{PR(j)}{deg(j)}$$

Adding the time decay, this formula becomes:

$$PR(i,t) = \sum_{x=1}^{n} \left(\frac{1-d}{n} \times p_x(t) \right) + d \sum_{j,j \to i} \left(\frac{PR(j)}{deg(j)} \times p_j(t) \right)$$

where $p_x(t)$ is the initial probability of selecting the x^{th} node in the citation graph.

As seen in the Chapter 5, a solution for the problem above is to apply the iterative method. The constraints are again related to *scalability* and *robustness*. We take advantage of the fact that the transition probability matrix is a very sparse one, so instead of storing the entire matrix, we are using dictionaries to keep the information.

foreach i *in* V **do**
  $p_i = e^{-w(t_{present} - t_i)}$
end
foreach i, j *in* V, $j \to i$ **do**
  $sparseDictionary[(i,j)] = dampingFactor \times \frac{p_j}{deg(j)}$
end
foreach i *in* V, $deg(i) = 0$ **do**
  add i to $danglingVector$
end
initialize weights
$converged = False$
while *not converged* **do**
  $newWeights = 0$
  foreach (i,j) *in* $sparseDictionary$ **do**
    $newWeights_i += sparseDictionary[(i,j)] \times weights_j$
  end
  $newWeights += \frac{dampingFactor}{n} \times \sum_{j \in danglingVector} weights_j \times p_j + \frac{(1-dampingFactor)}{n} \times \sum_{j=1}^{n} weights_j \times p_j$
  $W = newWeights - weights$
  $\epsilon = \frac{\sqrt{\|W\|}}{n}$
  if $\epsilon < convergeceThreshold$ **then**
    $converged = True$
  end
end

Algorithm 3: The Time-dependent PageRank algorithm

The iterative method proposes a multiplication between a matrix and a vector. This is usually done in $O(n^2)$. Since we are deling with a sparse matrix, as mentioned before (16% of the rows and 25% of the columns are zeros), we were able to reduce the complexity of the problem to $O(n)$. Having a time-dependent method increases the computations done in the PageRank method because, while in PageRank all the initial probabilities nonzero were 1, now they are inverse proportional with the age of the citations.

The idea of distributing the random surfers exponentially with age, in favor of more recent publications was also presented in the work of Walker et al. [12]. Their motivation is that researchers typically start "surfing" scientific publication from a rather recent publication that caught their attention on a daily update of a preprint archive or a recent volume of a journal. This algorithm simulates the dynamics of a large number of researchers looking for new information. Every researcher, independent of one another, is assumed to start his/her search from a recent paper or review and to subsequently follow a chain of citations until satisfied. In this way the effect of a recent citation to a paper is grater than of an older citation to the same paper.

8.3 Results

Since we already did a comprehensive analysis of the time decay parameter in Chapter 7, in this section we will focus on identifying the advantages and disadvantages of applying the time-dependent PageRank. First of all, let us take a look at the Top 10 results generated by the Time-dependent PageRank with a time-decay of 5 years (Table 8.1).

One might find it a bit strange that the 1^{st} publication is older than the rest (which are mainly published in 2006 - 2007). For finding the explanation for this fact we should look at the evolution of the citations for this particular publication (Figure 8.1) (`http://hep-inspire.net/record/268027/citations`).

Although this publication is rather old, with respect to the other publications in the Top 10 results, it acquired a large part of its citations recently. This is the reason for being highly ranked by the time-dependent ranking method. The identification of this kind of behavior can definitely be consider an advantage for this new ranking method. This is a good example of what a user looking for "hot trends" might want to see.

Looking further down in the top 100 best ranked publications when using the time-dependent link-based ranking method, we notice something strange (Table 8.2).

The two publications presented in Table 8.2 have less than 20 citations, and thus, are ranked really low with the Citation Count ranking method. How is it possible to be so highly ranked with our new ranking method? Further investigations showed that the problem comes from the fact that these two papers are citing each other, and thus, are part of a cycle. Because of this and

Rank	*Publication*	*Citation Count*	*Rank by Citation Count*
1	On the Phase Structure of Vector-Like Gauge Theories with Massless Fermions: Banks, Tom (1981)	314	2316
2	Review of Particle Physics: Yao, W.-M. (2006)	2480	39
3	New Tests for Quark and Lepton Substructure: Eichten, E. (1983)	691	541
4	Another odd thing about unparticle physics: Georgi, Howard (2007)	130	9990
5	Collider signals of unparticle physics: Cheung, Kingman (2007)	108	13251
6	Unparticle physics: Georgi, Howard (2007)	149	8016
7	Wilkinson Microwave Anisotropy Probe (WMAP) three year results: implications for cosmology: Spergel, D.N. (2006)	2652	34
8	Some phenomenologies of unparticle physics: Luo, Mingxing (2007)	94	15874
9	The Large N limit of superconformal field theories andsupergravity: Maldacena, Juan Martin (1997)	5162	3
10	Unparticle physics on direct CP violation: Chen, Chuan-Hung (2007)	82	18889

Table 8.1: Top 10 publication by Time-dependent PageRank

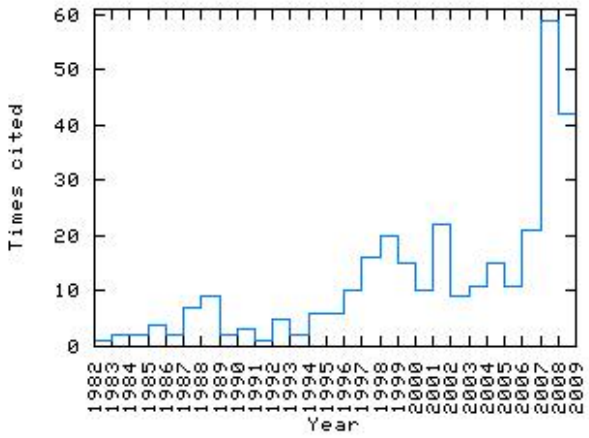

Figure 8.1: Citations history for 1^{st} ranked publication with time-dependent PageRank

Rank	*Publication*	*Citation Count*	*Rank by Citation Count*
31	Gauge symmetry and supersymmetry of multiple M2-branes: Bagger, Jonathan (2007)	19	90786
32	Comments on multiple M2-branes: Bagger, Jonathan (2007)	18	94900

Table 8.2: Snapshot from Top 100 publications by Time-dependent PageRank

of the link-based ranking which iteratively propagates the weight in the graph, when a strong time decay factor is used (in our case, a 5 year time decay), the newly published documents that are part of a cycle accumulate artificial weight. Unfortunately, this makes the time-dependent link-based ranking method unsuitable for data sets that allow cycles.

8.4 Conclusions

In order to overcome the two main deficiencies of the Citation Count algorithm, we combined the Link-based Ranking methods with the Time-dependent Citation Count creating a new method, the Time-dependent Link-based Ranking. Unfortunately, this algorithm has one major disadvantage due to the fact that it is time-dependent: it overestimates the weight of the younger publications that are part of a cycle. As discussed in Chapter 3, even the bibliographic data sets can allow cycles due to certain inconsistencies in the publication dates or in the listing of references. Since some of the publications are not dated, the identification/removal of the cycles is almost impossible due to the computational overhead. Because of this and of the link-based ranking which iteratively propagates the weight in the graph, when a strong time decay factor is used, the newly published documents that are part of a cycle accumulate artificial weight. Thus, this method is not suited for data sets that allow cycles.

Chapter 9

Conclusions

The Citation Count is a very popular measure of the impact of a scientific publication. Unfortunately, it has two main disadvantages: it gives all the citations the same importance and it does not take into account time. These drawbacks motivated our study of alternative approaches: Link-based Ranking methods(Chapter 5 and Chapter 6) and Time-dependent Ranking methods (Chapter 7 and Chapter 8).

The link-based ranking methods were developed to take into account the importance of the citing papers. We started with the PageRank algorithm originally designed for ranking web pages. In order to make it better suited for the bibliographic citation graph, we first modified the setting of the dumping factor. Indeed, when ranking web pages, it is usually considered that a random surfer follows on average 6 links in the link before getting "bored" and restarting the search. Such a setting does not really apply for the scientific community. It is rather believed that a scientist follows on average 2 links in the citation graph before restarting the search. Although our experiments show that the general ranking results remain quite similar when using different dumping factors, a dumping factor corresponding to 2 links in the citation graph should still be preferred for scientific publications, due to the fact that it is not entirely biased towards older publications.

Furthermore, we adjusted the PageRank model by adding an "external authority" node that represents a place holder for all the missing citations. In particular, this additional node prevents some publications from getting artificially boosted simply because of the incompleteness of the citation graph. We believe Link-based Ranking with External Citations to be a better candidate than Citation Count or PageRank for the task of ranking scientific publications because: (i) it inherits from PageRank its ability to take into account the citations with weights representing their importance, and thus, fixing one of the main shortcomings of the Citation Count method; (ii) it further corrects one of PageRank's shortcomings, namely the artificial inflation of some of the weights. In the end, our new ranking method is enough correlated with the PageRank

method as to assume that it inherits its usefulness and in the same time it corrects its shortcomings.

The time-dependent ranking methods were developed to take into account time dynamics of the citation graph. More precisely, we first introduced time-dependent citation counts, taking into consideration the lifetime of the citations. This method introduces the time decay parameter ($w \in (0, 1]$). The larger the time decay parameter, the faster we "forget" old citations.

Finally, we combined the two previous ranking methods, creating the Time-dependent Link-based Ranking. Unfortunately, this algorithm is not well suited for the citation graphs that are not DAG, due to the fact that it tends to overweight the young publications that are part of a cycle.

In terms of future work, there are several directions that can be explored. One of them would be to continue experimenting with the different parameters for a better tuning of the rankings. Another interesting direction might be to conduct more types of evaluation of the results including query evaluation.

Appendix A

Integration, Testing and Performance Evaluation

One of the major objectives of this project was the implementation and the integration of the newly crated ranking models in CDS Invenio. As described in the introduction, "the key feature of CDS Invenio's architecture lies in its modular logic. Each module embodies a specific, defined, functionality of the digital library system. Modules interact with other modules, the database and the interface layers. A module's logic, operation and inter-operability are extensible and customizable." [5] In this framework, our ranking methods must be added to the *BibRank module* and the interaction with the other modules, the data base and the interface needs to be assured.

Since we already discussed about the implementation of each model in the related chapters, we will only describe here details about the integration, the testing procedures and the performance evaluation. The implementation was entirely done in *Python*.

A.1 Integration and Testing Procedures

In order to have a fully running ranking method, we first need the input data. There are two possibilities for extracting the data (citations, external citations, and dates): one is reading it from a file (in case someone wants to heavily test with one of the ranking methods) and the second one is querying the database.

Citations can be retrieved by querying the rnkCITATIONDATA table. This table stores the citations in a form a dictionary *recid:citedBy list*.

External Citations can be retrieved by querying the tables associated with the MARC tag for references. In out case this tag is $999C5$ and the tables are $bibrec\_bib99x$ and $bib99x$. There is also the possibility of querying the rnkCITATIONEXT table in order to obtain the external citations, but

after doing some testing, it seems that the accuracy is better when using the MARC tables.

Dates can be retrieved by querying the tables associated with the MARC tag for the date of the publication, in our case either 260_*c* (for the year of the publication) or 269_*c* (for the complete date). Since we are only using the year of the publication in our computations, we went for the first tag. In the cases where the date of the publication is not available we consider the date of insertion in the data base (961_*x* tag). If neither of the information is available for a certain publication, we assign it an average date (computed with the available dates).

After retrieving the necessary data from various sources and running the chosen ranking procedure we store the newly calculated weights in the database, in the *rnkMETHOD* table, as a dictionary of type *recid:weight*. Since there exists the possibility of having different publications with the same weight we set up also a second ordering, based on the date of the publication. So, in the case of equal weight, a newer paper will be ranked higher than an older one. There is also the possibility of exporting the ranks in a file, for the testing purpose.

Since the running of the module requires setting several parameters, we created a configuration file for each ranking method, available in the Appendix. We propose the user three different ranking methods: time-dependent citation count, link-based ranking (PageRank) with or without the external citations and time-dependent link-based ranking.

CDS Invenio has a uniform testing technique, for which Python unittest module is used. Tasting has been a continues activity while implementing the new ranking methods. Every part of the code was intensively tested, first for monitoring the correctness of the algorithms with respect to the data sets and secondly for catching implementation errors. Still, several test cases have been designed to assure proper behavior of the core functionality of the new ranking methods. Among them, one tests for the correct extraction and manipulation of the citation data, and one tests the behavior of the PageRank method.

A.2 Performance Evaluation

As described in the previous chapters, different ranking methods require the use of the iterative method (successive multiplications between a matrix and a vector until the vector's values become stable). The constraints when developing this kind of algorithms are related to *scalability* and *robustness*. The scalability is an important requirement because we are dealing with an increasing data set, and while today we have close to half million papers, in a few years the number might double.

The iterative method has usually a complexity of $O(n^2)$. Since we are deling with a sparse matrix, 16% of the rows and 25% of the columns are zeros, we

were able to reduce the complexity of the problem to $O(n)$. Having a time-dependent method increases the computations done in the PageRank method because, while in PageRank all the initial probabilities nonzero were 1, now they are inverse proportional with the age of the citations. This is mainly the reason why the time-dependent PageRank algorithm runs more slowly than the classical PageRank. Still, the running speed is less then expected. The PageRank algorithm runs in approximately 4 minutes on an average equipped machine and the time-dependent PageRank runs on average in 8 minutes. At this running speed we also need to add the time for querying the data base, which does not necessary depend on our module, but still is an overhead that needs to be consider.

Appendix B

Configuration Files

citerank_citation_t.cfg

[rank_method]
function = citerank

[citerank]
##citerank_method - defines the method to use for ranking
citerank_method = citation_time

##time_decay - measures how fast we 'forget' old citations:
##0.0(never); 0.2(after 5 years); 0.1(after 10 years)
time_decay = 0.2

##file_with_citations - defines if the citations are to be read
##from an external file. (Default is to use the Invenio database.)
##The external file format must be: x[tab]y where x cites y; x,y are recids
#file_with_citations = /path/to/file/containing/citations

##output_ranks_to_filename - defines whether to also output the results to
##an external file. (Default is to only write them to the Invenio database.)
#output_ranks_to_filename = /path/to/where/to/write/citations

##output_rank_limit - defines the number of ranks to be written to the external
##file denoted by 'output_ranks_to_filename' argument.
##It can be either a number or 'all'. (Default is 'all' - output all the ranks.)
#output_rank_limit = all

##file_with_dates - defines if the publication years are to be read from an
##external file. (Default is to use the Invenio database.) The external file format
must b:e x[tab]y, where x is a recid and y is the publication year
#file_with_dates = /path/to/file/containing/dates

relevance_number_output_prologue = (
relevance_number_output_epilogue =)

citerank_pagerank_c.cfg

```
[rank_method]
function = citerank

[citerank]
##citerank_method - defines the method to use for ranking: pagerank.
citerank_method = pagerank_classic

##check_point - defines the frequency for calculating the stability of the
##weight vector
check_point = 1

##conv_threshold - defines the stability threshold for the weight vector.
conv_threshold = 0.0001

##damping_factor - measures in what depth the citation graph is
##influencing the ranking: 0.85(6 links), 0.7(3 links), 0.5(2 links)
damping_factor = 0.50

##file_with_citations - defines if the citations are to be read
##from an external file. (Default is to use the Invenio database.)
##The external file format must be: x[tab]y where x cites y; x,y are recids
#file_with_citations = /path/to/file/containing/citations

##output_ranks_to_filename - defines whether to also output the results to
##an external file. (Default is to only write them to the Invenio database.)
#output_ranks_to_filename = /path/to/where/to/write/citations

##output_rank_limit - defines the number of ranks to be written to the external
##file denoted by 'output_ranks_to_filename' argument.
##It can be either a number or 'all'. (Default is 'all' - output all the ranks.)
#output_rank_limit = all

##file_with_dates - defines if the publication years are to be read from an
##external file. (Default is to use the Invenio database.) The external file format
## must be: x[tab]y, where x is a recid and y is the publication year
#file_with_dates = /path/to/file/containing/dates

##use_external_citations - defines weather the ranking method should use the external
##link information. (Default is 'no'.)
#use_external_citations = yes

##ext_citation_file - defines if the external citation are to be read from an
##external file. (Default is to use the Invenio database.) The external file format
##must be: x[tab]y, x is a recid, y is the corresponding number of ext citations
#ext_citation_file = /path/to/file/containing/external/citations

##ext_reference_tag - defines the MARC tag corresponding to references
ext_reference_tag = 999C5

##ext_alpha - defines the fraction of the external node's weigh that goes back
##into the network.
ext_alpha = 0.1
```

##ext_beta - defines the proportion between the weight of an external link and
##the weight of an internal link.
ext_beta = 0.1

relevance_number_output_prologue = (
relevance_number_output_epilogue =)

citerank_pagerank_t.cfg

[rank_method]
function = citerank

[citerank]
##citerank_method - defines the method to use for ranking: time decayed pagerank.
citerank_method = pagerank_time

##check_point - defines the frequency for calculating the stability of the
##weight vector
check_point = 1

##conv_threshold - defines the stability threshold for the weight vector.
conv_threshold = 0.000001

##damping_factor - measures in what depth the citation graph is
##influencing the ranking: 0.85(6 links), 0.7(3 links), 0.5(2 links)
damping_factor = 0.50

##file_with_citations - defines if the citations are to be read
##from an external file. (Default is to use the Invenio database.)
##The external file format must be: x[tab]y where x cites y; x,y are recids
#file_with_citations = /path/to/file/containing/citations

##output_ranks_to_filename - defines whether to also output the results to
##an external file. (Default is to only write them to the Invenio database.)
#output_ranks_to_filename = /path/to/where/to/write/citations

##output_rank_limit - defines the number of ranks to be written to the external
##file denoted by 'output_ranks_to_filename' argument.
##It can be either a number or 'all'. (Default is 'all' - output all the ranks.)
#output_rank_limit = all

##file_with_dates - defines if the publication years are to be read from an
##external file. (Default is to use the Invenio database.) The external file format
must be: x[tab]y, where x is a recid and y is the publication year
#file_with_dates = /path/to/file/containing/dates

relevance_number_output_prologue = (
relevance_number_output_epilogue =)

Bibliography

[1] Cds invenio. `http://cds.cern.ch/`.

[2] Cern. `http://www.cern.ch`.

[3] Epfl infoscience. `http://infoscience.epfl.ch/`.

[4] S.U. Pillai A. Papoulis. *Probability, Random Variables and Stochastic Processes, 4th edition*. Mc. Graw Hill, 2002.

[5] M. Gracco J-Y. Le Meur N. Robinson T. Simko A. Pepe, T. Baron and M. Vesely. Cern document server software: the integrated digital library. `http://doc.cern.ch/archive/electronic/cern/preprints/open/open-2005-018.pdf`, 2005.

[6] A. Allen. *Probability, Statistics and Queueing Theory with Computer Science Applications*. Academic Press Inc., 1972.

[7] E. Amitay. Trend detection through temporal link analysis. In *J. of the American Society for Information Science and Technology*, pages 1–12, 2004.

[8] K. Berberich. T-rank: Time-aware authority ranking. In *WAW*, pages 131–142, 2004.

[9] P. Berkhin. A survey on pagerank computing. In *Internet Mathematics, Vol 2*, pages 73–120, 2005.

[10] Tim Berners-Lee. Information management: A proposal. `http://www.w3.org/History/1989/proposal.html`.

[11] A. McCallum D. Mimno. Mining a digital library for influential authors. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, 2007.

[12] D. Walker et al. Ranking scientific publications using a simple model of network traffic. 2006.

[13] M. J. Kurtz et al. The nasa astrophysics data system: Sociology, bibliometrics, and impact. In *J. of the American Society for Information Science and Technology*.

[14] M. J. Kurtz et al. Worldwide use and impact of the ansa astrophysics data system digital library. In *J. of the American Society for Information Science and Technology*, pages 36–45, 2005.

[15] P. Chen et al. Finding scientific gems with google. In *J.Informet. 1*, pages 8–15, 2007.

[16] E. Garfield. Citation indexes for science: A new dimension in documentation through association of ideas. In *Science, 122(3159)*, pages 108–111, 1955.

[17] E. Garfield. Citation indexing for studying science. In *Nature, 227 (5259)*, pages 669–671, 1970.

[18] E. Garfield. Citation analysis as a tool in journal evaluation. In *Science, 178 (4060)*, pages 471–479, 1972.

[19] Y. Guo B. Feng H. Wang, M. Rajman. Newpr - combined tfidf with pagerank. In *ICANN*, pages 932–942, 2006.

[20] S. Redner. How popular is your paper? an empirical study of the citation distribution. In *The European Physical Jurnal B*, pages 131–134, 1998.

[21] S. Redner. Citation statistics from 110 years of physical review. In *American Institute of Physics*, 2005.

[22] L. Page S. Brin. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, pages 107–117, 1998.