ACE: Big Data Approach to Scientific Collaboration Patterns Analysis

Andrei Zammit, Kenneth Penza, Foaad Haddod
Charlie Abela, and Joel Azzopardi

Department of Artificial Intelligence,
University of Malta,
Msida, Malta

Abstract. The characteristics of scientific collaboration networks have been extensively analysed and found to be similar to other scale-free networks. Research has furthermore focused on investigating how collaboration patterns between authors evolved over time, by providing insights into different fields of research. Numerous bibliographic datasets, such as DBLP and Microsoft Academic Graph, provide the basis for investigations and analysis of such networks. This paper presents ACE (Academic Collaboration analyzEr); an interactive framework that uses big data technologies and allows for scientific collaboration patterns to be analysed and visualised. Through ACE it is possible to reveal the key authors in particular fields of research, the topological features of the collaboration network, the network trends over time and the relationships between authors and co-authors. Furthermore, ACE allows for the discovery of potentially new collaborations between authors in the same field of research as well as fields where scientists can conduct future joint-research work.

Keywords: graph analysis, big data, collaboration patterns, collaboration networks

1 Introduction

Bibliometrics and scientometrics are two closely related research fields intended to measure and analyse scientific publications and science. Collaboration analysis is the study in which author collaborations in scholarly articles are used to establish relationships between authors and/or fields of study. This analysis is intended to provide insight into the evolving communities of authors and scholarly publications, the collaborations between authors, and the evolution of areas of knowledge over time. A high impact factor is partly determined by the number of citations to articles within a particular journal. If an article is published in a journal with a high impact factor, the publishing profile of the author is raised. The number of citations to that article over time is also a measure of the impact of that author.

Collaboration Networks are typically visualized as graphs whereby a vertex represents some entity and an edge represents some property relating multiple

vertices. In a collaboration graph, vertices can typically represent *authors*, *papers* as well as *keyword*. Different types of edges can be used to represent different interactions between these entities; for instance in the case of an author and a paper, edges can represent the *authoredBy* relation, whilst in the case of a keyword and a paper, an edge can represent the *usedBy* relation. Depending on the schema adopted, both vertices and edges can have an arbitrary number of properties. Furthermore, separating entities into different vertices can aid the visualisation and analysis tasks.

In a network, particular nodes might be more important than others due to the preferential attachment characteristic highlighted by [3]. The notion of importance can mathematically computed using graph metrics such as closeness, betweenness and degree centrality. Closeness centrality is defined as the geodesic distance, which is the shortest path between two nodes [6, 13]. The closeness centrality is computed by dividing the number of reachable nodes by the sum of the geodesic distance to each accessible vertex. Betweenness is a centrality measure computed on shortest paths [6, 13, 29, 15]. A vertex has a higher betweenness if more geodesic shortest paths, pass through this vertex. On the other hand vertices with a higher degree centrality have a higher probability to be part of a dense network [6, 13, 29, 15]. In other graph analysis algorithms such is PageRank [16], the importance of a node in the network depends on the number of times a random surfer visits the same page. In the case of websites, if the site has a high in-degree the probability of revisiting the same site is higher.

An interesting aspect in collaboration analysis is the identification of potential collaborators for a given author. Turker et al [27] report about various studies that have been performed to analyse co-authorship networks from the perspective of the research disciplines involved and the journals to which the research was submitted. In this work mathematical techniques were used to identify strong collaborations and the authors that were more likely to collaborate with others. Another approach to co-author prediction reported by [24] uses random walks and graph metric to perform author suggestion. The model uses a set of criteria to select potential candidates including, authors that collaborated with different authors, authors that already collaborated and authors with common authors.

The provisioning of bibliographic datasets such as DBLP[1] and Microsoft Academic Graph[2], together with large-scale graph processing technologies such as Apache Spark[3] and Neo4j[4] offer new research opportunities in the bibliometrics and scientometrics fields.

In recent years, a number of studies have analysed such collaboration networks in search for emerging trends and communities of interest by leveraging on big data technologies. Sreenivas et al proposed a data discovery and knowledge recording tool called SEEKER [23] that uses big data technologies to help users

[1] http://dblp.uni-trier.de/xml/
[2] http://aka.ms/academicgraph
[3] http://spark.apache.org/
[4] https://neo4j.com/

quickly assimilate knowledge from diverse data sources with different formats, hosted across different infrastructures. SEEKER provides collaborative knowledge management tools and access to a data warehouse via a query interface to provide results via a variety of visualisations. Another big data approach proposed by [28], analysed the social network of eleven years of publications in engineering education and their authors. The bibliometric analysis was based on grouping authors by the research areas, disciplinary backgrounds and geographical locations.

In this paper, we present the Academic Collaboration analysEr (ACE)[5] interactive framework, which enriches collaboration networks resulting from the Microsoft Academic Graph and the DBLP datasets with keywords extracted from the publications. ACE uses big data technologies, Apache Spark and Neo4J to allow the user to identify research trends and communities in the collaboration networks. Furthermore, ACE permits the analysis of the networks using different perspectives which include *author*, *keyword* and *publication year*. Through ACE the user can identify potential collaborators for a given author and the evolving community of researchers around specific keywords.

The rest of the paper is structured as follows: in the next section we present literature related to different collaboration network analysis tools. Then in the methodology section go in detail through the various steps used to build ACE. We discuss the challenges that were encountered and explain how we addressed them. In the experiments section, we report about the findings from using ACE to answer a specific set of queries related to particular authors and keywords. In the final section we present some conclusions and ideas how ACE can be extended.

2 Literature Review

Nowadays, measuring the scientific output of researchers is becoming increasingly important to support research assessment decisions related to accepting research projects, contracting researchers and/or awarding scientific prizes [10]. Despite the recent advances in scientific impact prediction and more specifically, paper citation prediction, it is still unclear and even controversial whether one should depend on the reliability and bound of the prediction accuracy of a long-term citation prediction model. A number of measures, such as the g-index [8], h-index [12, 1] have become popular measures to gauge journals, scholars, labs, departments, and institutes [11]. Other tools such as Microsoft Academic Search[6], Rexplore [19], ArnetMiner[25], and Saffron[7] provide a variety of visualizations that can be used for trend analysis, such as publication trends and co-authorship paths among researchers. We can also find several systems for

[5] https://youtu.be/kzXOIzddEa4
[6] https://academic.microsoft.com/
[7] http://saffron.insight-centre.org/

exploring and making sense of research data such as Google Scholar[8], Faceted-DBLP[9] and CiteSeerX [22].

2.1 h-index and g-index

As citation data have become more available, new metrics for analysis have been developed. The best known metrics include the h-index [12] and g-index [8] which are aimed at facilitating the comparisons of the impact or importance of individual researchers. The h-index is considered to be a way to assess the impact of an individual author without the skewed citation distribution affecting the results. This index reflects both the overall publications as well as the level of citation of those publications. While evaluation at the level of individuals is useful, the evaluation at the journal level is more practical for large scale assessment of research outputs, such as those carried out by universities and funding agencies [21]. The easiest method to calculate the h-index is to first rank papers in a table in descending order by the number of citations they have received. The h-index can be applied to journals as well as researchers [8].

The g-index was introduced as an improvement of the h-index to measure the global citation performance of a set of articles and it inherits all the good properties of the h-index and, in addition, takes into account the citation scores of the top articles. This yields a better distinction and order of the scientists from the point of view of visibility [8]. A measure which should indicate the overall quality of a scientist or of a journal should deal with the performance of the top articles and hence their number of citations should be counted. This can be accomplished by modifying the h-index so that the above described disadvantage was addressed while keeping all advantages of the h-index and, at the same time, the calculation of the new index is as simple as that of the h-index [9].

2.2 Microsoft Academic Search

Microsoft Academic Search (MAS) provides a variety of visualizations, including co-authorship graphs, publication trends, and co-authorship paths between authors [19]. The coverage of MAS at the beginning was limited to the computer science and technology fields, but this was extended in March 2011 to other categories thus turning MAS into a platform oriented to the identification of the top papers, authors, conferences and organisations in 15 fields of research and more than 200 sub-fields. It provides both the bibliographic description of the publications and their citation counts. In short it offers everything required to identify the most relevant research and to carry out comparative performance assessments [18]. MAS is a scientific web database which gathers bibliographic information from the main scientific editorials (such as Elsevier[10] and Springer[11])

[8] https://scholar.google.com
[9] http://dblp.l3s.de
[10] https://www.elsevier.com/
[11] www.springer.com/

and bibliographic services (such as CrossRef[12]). It roughly contains 38.9 millions of documents and 22 million profiles. Amongst other features, MAS presents a personal profile which provides not only the authors list of publications but also relevant bibliometric indicators (publications, citations), the disciplinary areas of interest and other rosters showing the most frequent co-authors, preferred journals and a few important keywords [17].

2.3 Google Scholar

Google Scholar (GS), constituted a great revolution in the retrieval of scientific literature, since for the first time bibliographic search was not limited to the library or to traditional bibliographic databases. Instead, because it was conceived as a simple and easy-to-use web service, GS enabled simple bibliographic search for everyone with access to the web. GS is freely accessible and it indexes data from publishers only if the publisher is willing to provide at least the abstract of the paper freely. The data comes from other sources as well, like freely available full text from preprint servers or personal websites as well, thus in many cases the full text is freely available for all users [4]. GS uses web crawlers to retrieve scholarly material from journal websites, university repositories, and authors personal websites. Scholarly documents are identified by means of automatic format inspection such as the title in large font at the front page, authors' names right below the title, and the presence of a section titled "References" or "Bibliography"). Indexing is done automatically by parsers that identify bibliographic data in the selected documents. It has been argued that because of its automatic inclusion process, GS is susceptible to errors in metadata and to indexing of non-scientific works [7].

2.4 ArnetMiner

ArnetMiner offers different visualizations and provides support for expert search and trend analysis [19]. This system mainly consists of five main components: extraction, integration, storage and access, search, and mining.

i. *Extraction:* Focuses on extracting researcher profiles from the Web automatically by identifying relevant pages from the web and collecting publications from existing digital libraries [25, 26].
ii. *Integration:* Integrates the extracted researchers profiles and the extracted publications by using the researcher name as the identifier. A probabilistic framework has been proposed to deal with the name ambiguity problem in the integration. The integrated data is stored into a Researcher Network Knowledge Base (RNKB)s [25, 26].
iii. *Storage and Access.* Provides storage and index for the extracted and integrated data in the RNKB [25, 26, 2, 5].

[12] https://www.crossref.org/

iv. *Search.* Provides three types of search activities; person search, publication search, and conference search. It also provides other services, e.g., author interest finding and academic suggestion [25, 26].
v. *Mining.* Provides five mining services; expert finding, people association finding, hot-topic finding, sub-topic finding, and survey paper finding [25].

2.5 Rexplore

Rexplore[13] is a tool that integrates statistical analysis, semantic technologies, and visual analytics to provide effective support for exploring and making sense of scholarly data [19]. The semantic relationships among authors and topics are at the heart of many new functionalities of Rexplore. These relationships are in particular used for computing novel kinds of similarities and ranking metrics that take in consideration the semantic characterization of research areas. Furthermore, the semantic relationships improve the ability of Rexplore to interpret user queries and enable a novel graph-based navigation technique, which combines both the semantic relationships and automatically computed metrics to generate links between the elements of the domain [14].

Rexplore supports users effectively by enabling them to detect and make sense of the important trends in one or more research areas. Additionally, users are able to identify researchers and analyse their academic trajectory and performance in one or multiple areas, according to a variety of fine-grained requirements. Furthermore, Rexplore users can discover and explore a variety of dynamic relations between researchers and topics and rank specific sets of authors, generated through multi-dimensional filters, according to various metrics [19]. Other important features of Rexplore include:

1. *Data Integration:* Rexplore integrates a variety of data sources in different formats, including: the MAS API2, DBLP++3 and DBpedia4.
2. *Topic Ontology and Klink:* while most systems use keywords as proxies for research topics, Rexplore relies on an OWL ontology, which characterizes research areas and their relationships.
3. *Multi-criteria Search:* Rexplore offers fine-grained search functionality for authors, publications and organizations with respect to detailed multi-dimensional parameters.
4. *The Graph View:* the graph view is an interactive tool to explore the space of research entities and their relationships using faceted filters. It takes as input, authors, organizations, countries or research communities and generates their relationship graph, allowing the user to choose among a variety of connections, ranking criteria, views and filters.
5. *Community Detection:* Rexplore integrates a novel algorithm called TST (Temporal Semantic Topic-Based Clustering), which identifies communities of researchers who appear to follow a similar research trajectory.
6. *Author and Group Analysis:* every author in Rexplore has a personal page which offers a variety of metrics and visualizations to analyse the authors performance, trends and collaborations.

[13] https://technologies.kmi.open.ac.uk/rexplore/

3 Methodology

In this section we describe the challenges that we had to address within ACE, from pre-processing to the integration of different big data technologies,

3.1 Dataset selection and pre-processing

Two important reference datasets for bibliographic information about major computer science publications are DBLP and the Microsoft Academic Graph. The Microsoft Academic Graph dataset is much larger than DBLP and the structure of the two datasets is completely different. The first challenge was the choice of the dataset to use for ACE and the related experiments. The Microsoft Academic Graph was considered to be too extensive to be processed on a typical personal computer and hence DBLP was the choice of the dataset. The DBLP dataset is based on XML and there are two types of records; articles and in-proceedings. Table. 1 shows the record structure.

DBLP structure	
Article	In Proceedings
Title	Title
Year	Page
Volume	Volume
EE	EE
URL	URL
Journal	Year
	Book Title

Table 1: DBLP structure

Microsoft Academic Graph enrichment		
Paper	Keywords	DOI DBLP
Paper ID	Paper ID	DOI
Title	Keyword	
Venue	Field of study	
Author ID		
Affiliation ID		
DOI		
Journal ID		
Conference ID		

Table 2: DBLP structure

The structure clearly showed that the dataset suffered from missing information in order to execute the experiments with ACE, namely the keywords, field of study and abstract. To source this information, a web crawler and parser were developed to consume the Digital Object Identifier (DOI) provided in the EE XML tag. The DOI is a standard used to cite and link permanently to electronic documents. The DOI would typically direct to a specific page of a publication house which contains the title, author, abstract and keywords of a particular journal or research paper. A Python script was written to extract the EE XML tag and dump it to a text file. The crawler and parser were instantiated to target different publication houses and any locations which could not be parsed were stored locally for retry at a later stage. Using this method of sourcing the missing data, failed after just nearly one thousand records (papers/journals) processed because the website of the publication house tracked this activity and blocked the IP of the machine which launched the crawler and parser. An alternative option that was explored was to use the journals publishers' developer API.

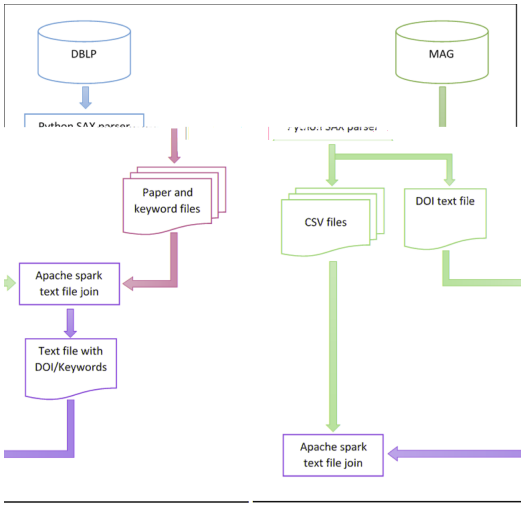

Fig. 1: Enriching DBLP with MAG keywords

When consumed, these APIs would allow an entity to query a webservice using the DOI and retrieve meta data such as the abstract and keywords. However, usage of these APIs was limited to a small number of calls to the service per day. Given the sheer amount of the DOI required to be fetched and the search limits imposed it was not feasible to get a reasonable number of abstracts in a short timeframe. Finally, the unavailability of the abstract and keyword data was mitigated through the use of the Microsoft Academic Graph dataset. This dataset is more comprehensive than DBLP and it contains all the required information. The Microsoft Academic Graph data is a tab delimited text file and is structured as illustrated in Table. 2. Using DataFrames found in Apache Spark, three schemas were created; one for the DOI file, and the other two for the Paper and Keyword files from the Microsoft Academic Graph dataset. The two data-frames originating from the Microsoft dataset were then joined together via the Paper ID field and in turn this joint data-frame was linked to the DOI data-frame via the DOI key. The process used to enrich the DBLP using the MAG keywords is illustrated in Figure. 1.

3.2 Graph Database

We evaluated three different graph database setups; Neo4j, Apache Spark with GraphX and Apache Spark with GraphFrames. Apache Spark offers high scalability and parallel graph processing. Data manipulation is performed via Scala. Neo4j is a robust graph database and uses a SQL like language called Cypher to manipulate data. One of the aims of ACE is to be a portable application which can be executed on typical everyday personal computers. Hence, one of

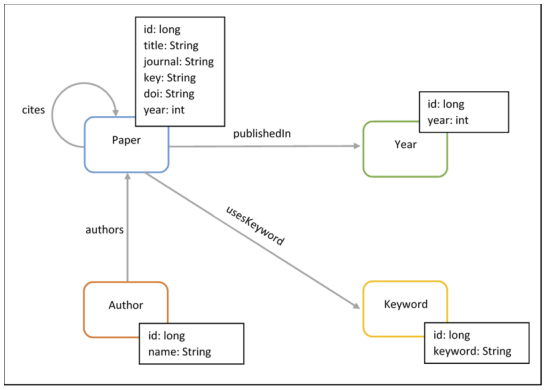

Fig. 2: ACE graph schema

the requirements was that the graph database did not require special hardware to operate and offers interface APIs. Neo4j is a mature product, backed with detailed documentation and official client APIs for different languages. This graph database has a large community and is widely used in industry. For ACE, Neo4j was deemed to be the ideal backend candidate. ACE was to be implemented using the .NET Framework and C# as a language. The main factor for this decision was that there are currently no official .NET API for Spark and on the other hand Neo4j has its official client API. Currently, the only possible binding using C# with Apache Spark is via Mobius, which is still in early beta stage. Furthermore, Spark required much more hardware resources than Neo4j which made Spark impossible to execute on personal computers. Considering these restrictions, Neo4j and Apache Spark with GraphFrames were chosen to perform experiments using parallel operations and resilient distributed datasets (RDD).

3.3 Schema

The enriched dataset consists of information about papers, their authors, author selected keywords and year of publication. The schema was defined to map the information in the dataset into a number of vertices and edges. The details of the entity were stored in the vertex as properties. A number of edge types were used to link vertices; for example an author authors a 'paper' whilst a 'paper' has a 'keyword'. When querying the graph, the edge type can be defined to identify the relation type being requested. For example, the number of outgoing edges of type authors amount to the number of papers authored. Similarly, the number of incoming edges in a keyword vertex amounts to the number of papers using that keyword. Figure 2 illustrates the graph database schema used in ACE.

```
MATCH (a:Author)-[r:Authors]->(p:Paper)-[s:Uses_keyword]->(k: Keyword)
WHERE a.name = ″{0}″
with distinct a as a, p.journal as journal, k, id(a) as currauthorid
MATCH (colla:Author)-[collr:Authors]->(collp:Paper)-[colls:Uses_keyword]->(k)
WHERE collp.journal = journal and id(colla) <>currauthorid
RETURN distinct(colla.name) as collaborator, a.name as author,
count(k) as keywordmatch, id(a) as idSource, id(colla) as idTarget,
id(k) as idTarget2 ORDER BY keywordmatch DESC;
```

Fig. 3: Cypher query for potential collaborators

```
MATCH (y:Year)<-[r1:Published_In]-(p: Paper)-[r: Uses_keyword]->(k: Keyword)
where k.keyword = ″0″
return id(k) as idSource, id(p) as idTarget, id(y) as idTarget2,
p.journal as journal, k.keyword as keyword, y.year as year;
```

Fig. 4: Cypher query for community analysis

3.4 Data import and querying

The data was extracted in CSV format in the pre-processing phase and was loaded into Neo4j. Before the data was loaded a number of constraints were created to speed up the loading. The load command was set to commit every 2500 records to avoid performance issues as recommended by Neo4j bulk load guide. Additional indexes were created after the loading to speed up cypher queries. For the Apache Spark database, Scala was used to perform queries and launch parallel operations. Within ACE potential author collaborators have common keywords and publication journal. Keywords are used to correlate the author's areas of interest. The rules used by ACE are the following:

- Identify the keywords used by a given author;
- Find authors that have used the same keywords;
- Select only authors that have authored papers in the same journal;
- Return list of authors, ranked by keyword matches.

The cypher query shown in Figure 3 finds potential collaborators for particular authors.

A group of authors that have a common area of interest are considered to be a community. A research domain is identified around a given keyword, for example data mining. Communities have a dynamic nature as they build up, remain stable or decrease, around topics and journals with time. The cypher query displayed in Figure 4 is used to find such communities.

3.5 Visualization

ACE was designed to present query results graphically using two types of graphs; a force-directed graph and a force-directed graph with time slider. User query

Fig. 5: ACE year, title and keyword relationship

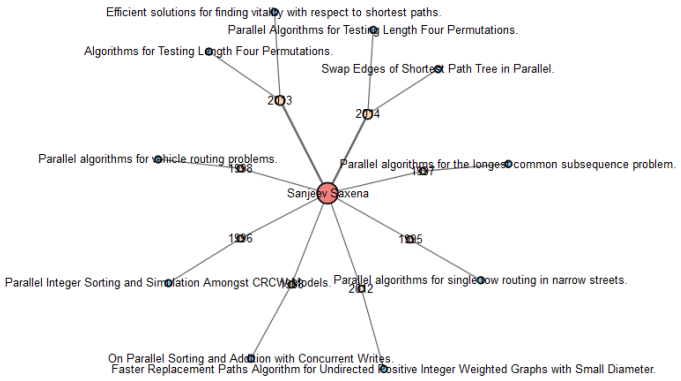

Fig. 6: ACE keyword, title and author relationship

results are stored in a CSV file and visualised using the D3[14] visualisation library. The ACE user can interact with the graph by clicking on the nodes to visualise more information about the node as shown in Figures 5-9.

[14] https://d3js.org/

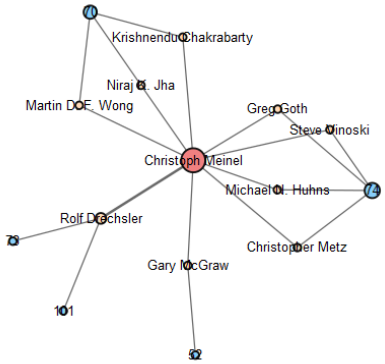

Fig. 7: ACE author, collaborator year relationship

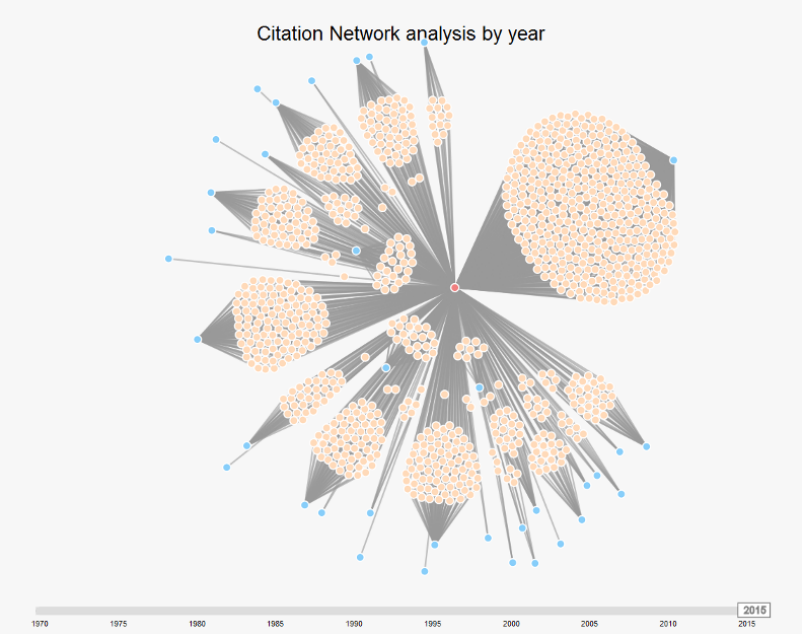

Fig. 8: ACE illustrating timeline for a keyword

Experiments and Evaluation

3.6 Apache Spark

Apache Spark with Graphframes utilises dataframes to store edges and vertices. The loading process entails loading the contents of the text files to a dataframe.

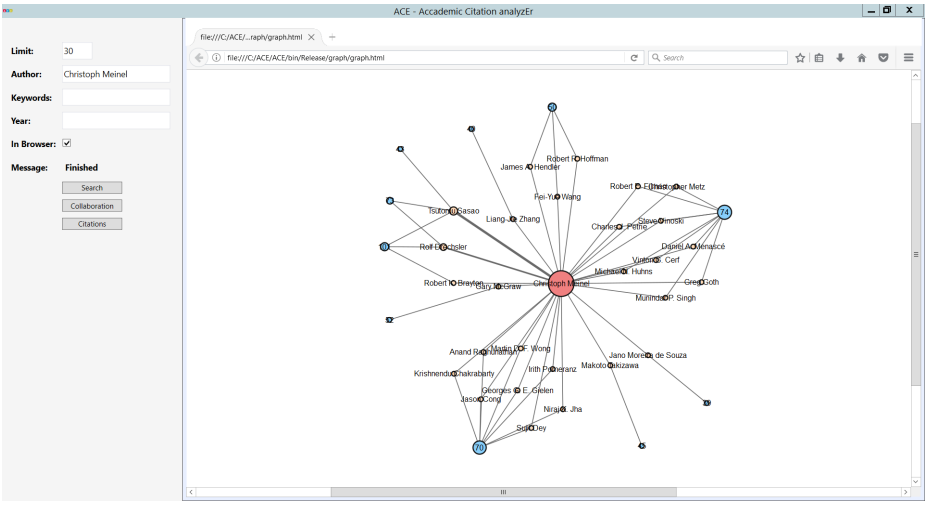

Fig. 9: ACE query interface

The vertex dataframe must have a numeric column with unique values called id. The edge dataframe must have two columns with the source and destination id of the vertices named src and dest respectively. The PageRank algorithm was used to traverse the graph and find the most important nodes within the graph. The results from PageRank are reported in Table 3.

3.7 ACE

The ACE front-end allows the user to perform several queries interactively:

- Query the authors and co-authors that collaborated in each year;
- Author collaboration;
- Papers that contain a user given keyword;
- Author collaboration suggestion;
- The evolution of the community around a user given keyword.

ACE presents the results as a graph that the user can interact with. In case of the community evolution across time, via a slider the users can visualize the evolution of the communities. Data extracted from the ACE system was verified against the DBLP online search provided accessible from the DBLP site.

4 Conclusion and Future Work

The emergence of big data technologies and bibliographic datasets have opened new possibilities in the research areas of bibliometrics and scientometrics. In this paper, the DBLP dataset was analyzed to extract communities using the

Table 3: Page Rank Results

Top 10 Authors	Top 10 Keywords	Top 10 Papers
Hans Jrgen Schneider	data mining	A Relational Model of Data for Large Shared Data Banks.
Jarkko Kari	internet	The IceProd Framework: Distributed Data Processing for the IceCube Neutrino Observatory.
Ehsan Khamespanah	real time	The Entity-Relationship Model - Toward a Unified View of Data.
Stefan Szeider	satisfiability	Length Sensing and Control in the Virgo Gravitational Wave Interferometer.
Richard R. Muntz	feature extraction	Finding needles in haystacks: linking scientific names, reference specimens and molecular data for Fungi.
Helmut Alt	neural network	EVpedia: a community web portal for extracellular vesicles research.
Derek Coleman	algorithms	System R: Relational Approach to Database Management.
Reiji Nakajima	computer science	Further Normalization of the Data Base Relational Model.
Matthieu Perrinel	bioinformatics	The Notions of Consistency and Predicate Locks in a Database System.
Soma Chaudhuri	mathematical model	The Design and Implementation of INGRES.

PageRank algorithm on Apache Spark. These experiments were executed on server hardware and operating systems. At a later stage, ACE was developed using the .NET framework and Neo4j as graph database. ACE is a portable tool that can be executed on any typical personal computer. Communities and the evolution of the collaboration network can be analyzed visually. Queries can be executed and results are processed in a considerable short period of time, giving the user a truly interactive experience. In ACE, communities are discovered by traversing the graph according to the input provided by the user. Apache Spark was used to perform pre-processing and initial analysis. Apart from PageRank, other community detection algorithms such as Triangle Counting, Connected Components and Label Propagation Algorithm can be executed on Apache Spark. On the outset ACE was intended to be an online interactive tool to allow users to explore collaboration patterns. Potential co-authors for a given author was determined by finding similar authors in the communities for a given author. This implementation design transitioned the implementation focus from Apache Spark to Neo4j. The main drivers for this decision were the infancy of the .NET connectivity for Apache Spark and integration with the visualisation part. Further work is required to investigate how ACE can be transformed into an web application using Apache Spark with automated visualisation. The graph schema used in graph analysis provides the required granularity to fulfill the ACE requirements. A number of changes can be made to improve the results obtained from graph analysis. The schema should be revised to include edges from the keyword to the paper and from the paper to the

author. The journal publishing the paper is currently an attribute in the paper's vertex. Extracting journals as separate vertex with the respective edges would allow computing journal importance. This data can be correlated to determine whether importance is gained from keyword, authors or both. Currently, ACE matches author names and keywords using string matching. Analysis on similarity search would improve system usability. In order to be able to correlate collaborations ACE should be extended to support multiple keyword searches. An implementation enhancement that merits further investigation would be the ability to plugin other datasets using linked data techniques. Enriching ACE using linked data requires rewriting the pre-processing phases to allow ACE to read data sources defined using standard ontologies. A linked data version of ACE would boost the data available and improve the overall functionality of ACE namely the author suggestion. Through the linked data approach more context on the authors involved on the collaboration can be mined. Furthermore, the topics of collaboration for a given author and the conferences and journals to which he/she submits research, tend to change over time. Through linked data the mapping can be preserved and more information can be attained from the collaboration network.

References

1. Acuna, D.E., Allesina, S. and Kording, K.P., Future impact: Predicting scientific success. Nature, 489(7415), (2012) pp.201-202.
2. Baeza-Yates, R. and Ribeiro-Neto, B., Modern information retrieval (Vol. 463). (1999) New York: ACM press.
3. Barabasi, A. and Albert, R., Emergence of Scaling in Random Networks. Science 286 , no. 5439 (1999) pp.509-512.
4. BarIlan, J., Which h-index? A comparison of WoS, Scopus and Google Scholar. Scientometrics, 74(2), (2008) pp.257-271.
5. Carroll, J.J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A. and Wilkinson, K., Jena: implementing the semantic web recommendations. In Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, (2004) pp. 74-83. ACM.
6. Day, M. Y., Shih, S. P. and Chang, W. D., Social network analysis of research collaboration in Information Reuse and Integration. IEEE International Conference on Information Reuse & Integration, (2011) pp.551556.
7. De Winter, J.C.F. and Zadpoor, A. A. and Dodou, D., The expansion of Google Scholar versus Web of Science: a longitudinal study. Scientometrics, 98(2), (2014) pp.1547-1565.
8. Egghe, L., Theory and practise of the g-index. Scientometrics, 69(1), (2006) pp.131-152.
9. Egghe, L., An improvement of the h-index: The g-index. ISSI newsletter, 2(1), (2006) pp.8-9.
10. Franceschet, M., PageRank: Standing on the Shoulders of Giants. Commun. ACM 54, 6, (2011) pp.92-101.
11. Fuyuno, I. and Cyranoski, D., Cash for papers: putting a premium on publication. Nature, 441(7095) (2006) pp.792.

12. Hirsch, J.E., An index to quantify an individual's scientific research output. Proceedings of the National academy of Sciences of the United States of America, (2005) pp.16569-16572.
13. Hou, H., Wang, C., Luan, C., Wang, X. and Zhuang, P., The Dynamics of Scientific Collaboration Networks in Scientometrics. Collnet Journal of Scientometrics and Information Management, (2013).
14. Motta, E. and Osborne, F, Making sense of research with rexplore. Proceedings of the 2012th International Conference on Posters & Demonstrations Track-Volume 914 (2012).
15. Mutschke, P. and Mayr, P., Science models for search: a study on combining scholarly information retrieval and scientometrics. Scientometrics, (2015).
16. Page, L., Brin, S., Motwani, R. and Winograd, T., "The PageRank citation ranking: Bringing order to the Web." Paper presented at the meeting of the Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia, (1998).
17. Ortega, J.L., Influence of co-authorship networks in the research impact: Ego network analyses from Microsoft Academic Search. Journal of Informetrics, 8(3), (2014) pp.728-737.
18. Ortega, J. L. and Aguillo, I. F., Microsoft academic search and google scholar citations: Comparative analysis of author profiles. Journal of the Association for Information Science and Technology, 65(6), (2014) pp.1149-1156.
19. Osborne, F., Motta, E. and Mulholland, P., Exploring scholarly data with rexplore. In International semantic web conference (2013) pp.460-477.
20. Osborne, F. and Motta, E., Understanding research dynamics. In Semantic Web Evaluation Challenge (2014) pp. 101-107. Springer International Publishing.
21. Rosenstreich, D. and Wooliscroft, B., Measuring the impact of accounting journals using Google Scholar and the g-index. The British Accounting Review, 41(4), (2009) pp.227-239.
22. Salatino, A., Early Detection and Forecasting of Research Trends. DC@ISWC, (2015).
23. Sukumar, S.R. and Ferrell, R.K., Big Data collaboration: Exploring, recording and sharing enterprise knowledge. Information Services & Use, 33(3-4), (2013) pp.257-270.
24. Sun, X., Lin, H., Xu, K. and Ding, K., How we collaborate: characterizing, modeling and predicting scientific collaborations. Scientometrics, (2015).
25. Tang, J., Zhang, J., Zhang, D., Yao, L., Zhu, C. and Li, J., Arnetminer: An expertise oriented search system for web community. In Proceedings of the 2007 International Conference on Semantic Web Challenge-Volume 295 (2007) pp. 1-8. CEUR-WS. org.
26. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L. and Su, Z., Arnetminer: extraction and mining of academic social networks. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (2008) pp.990-998.
27. Türker, I. and Çavuşoğlu, A., How we collaborate: characterizing, modeling and predicting scientific collaborations. Scientometrics, (2016).
28. Xian, H. and Madhavan, K., Anatomy of Scholarly Collaboration in Engineering Education: A Big-Data Bibliometric Analysis. J. Eng. Educ., 103, (2014) pp.486514.
29. Zhao, Y. and Zhao, R., An evolutionary analysis of collaboration networks in scientometrics. Scientometrics, 107(2), (2016) pp.759772.