Towards Supporting Complex Retrieval Tasks Through Graph-Based Information Retrieval and Visual Analytics

Aleksandar Bobic[1,2], Jean-Marie Le Goff[1] and Christian Gütl[2]

[1]*CERN, Espl. des Particules 1, Meyrin, 1211, Switzerland*
[2]*Graz University of Technology, Rechbauerstraße 12, Graz, 8010, Austria*

Abstract
The retrieval result analysis approaches of existing retrieval solutions tend to be either too simple, provide too few features for exploring retrieval results or are very narrowly focused. We present an enhanced approach that attempts to address these issues and help the wider community to get more insight from their retrieved data. To this end, this paper presents an enhanced graph-based retrieval prototype built on the Collaboration Spotting platform. It combines information retrieval and visual analytics concepts to provide an advanced solution for data retrieval and exploration. It enables users to retrieve information, explore it from different perspectives using a graph representation and perform further searches based on their navigation and selection interactively. Compared to traditional retrieval solutions, a search action in CS can reveal more detailed aspects/techniques when visually analysing the search output. To gain initial feedback, we interviewed five domain experts in related fields. Findings reveal that the developed retrieval approach provides users with helpful ways of exploring search results and provides mechanisms of connecting features that are not explicitly linked otherwise. Furthermore, several research directions and improvements have been identified for future work, which should be addressed.

Keywords
information retrieval, visual analytics, knowledge discovery, visualization system

1. Introduction

With the recent digitalisation efforts and steadily growing data piles, the amount of generated information rapidly increased over a short period. This increase in data quantity made the need for efficient retrieval and visual analytics tools apparent. This need is also reflected in multiple works which identified the necessity for IR applications that would enable users to carry out complex retrieval tasks, visualise hidden connections by leveraging interaction and visualisation and extract implicit insights from retrieved data automatically [1, 2]. Examples of such complex retrieval tasks could include retrieval of institution collaborating in a specific field, identification of author collaboration networks, retrieval of upcoming research topics connected to existing topics and more.

As one example, the need for the above-mentioned features to analyse data and grasp connections is also present in bibliometric data. For this application scenario, data are traditionally gathered, indexed and made accessible by services such as Google Scholar [3], Microsoft Academic [4] and ArXiv [5] which present search results as an ordered list based on assumed relevance and do not offer advanced analytics approaches which would support analysing correlations between papers. This ordered list format does not help users to extract complex relationships and gain deeper insights from large retrieval results [6, 7, 1]. In the context of bibliometric data, examples of data retrieval insights might include identifying author collaboration networks, identifying trending research areas in recent years, and discovering common concepts shared among fields. User-centred interactive analysis of bibliometric data can lead to better insights, novel research projects, and more informed decision-making [8, 9].

A variety of visual analytics (VA) tools and visualisation approaches were created as a result of the above-outlined needs for supporting bibliometric data exploration, and analysis workflows by different interest groups [10, 8, 6]. A straightforward and broad division can be made between solutions created for bibliometric mapping and general-purpose VA tools [6]. Both groups leverage multiple visualisation techniques to provide users with an insightful exploration process and reveal hidden connections which can not be easily inferred from an ordered list of retrieval results. A common approach to representing large connected datasets is displaying and analysing them as a connected graph. The potential of the graph representation has been apparent to researchers and tool creators for quite some time [11].

Another example of a graph-based representation is Collaboration Spotting (CS). It is a graph-based visual analytics (VA) platform created to address the limitations of existing graph-based exploration tools such as limited leveraging of interactivity and network visualisations,

DESIRES 2021 – 2nd International Conference on Design of Experimental Search Information REtrieval Systems, September 15–18, 2021, Padua, Italy
aleksandar.bobic@cern.ch (A. Bobic);
jean-marie.le.goff@cern.ch (J. L. Goff); c.guetl@tugraz.at (C. Gütl)
0000-0001-5403-8475 (A. Bobic); 0000-0001-9589-1966 (C. Gütl)

and visualisation of explicit and implicit connections between features [12]. It enables users to explore sizeable connected datasets by navigating through or changing perspectives[1] and contexts[2].

To enable users to execute complex retrieval tasks and gain further insight into their retrieval results, and based on existing work, we develop an enhanced CS-based retrieval system as a prototype. However, as an example and due to large amounts of available data, we focus on bibliometric data. As our main contribution, we integrate an enhanced retrieval mechanism in the CS platform's main version. We combine graph-based VA and information retrieval (IR) by introducing an enhanced IR system that retrieves data from a search provider and leverages an interactive graph representation to display the search results. It also provides a mechanism for further search refinement through simple graph interactions. Furthermore, to identify the needs of experts, understand how to develop a system supporting users at multiple steps of their retrieval tasks and potentially expanding the system for broader use, we interview five experts with a semi-structured approach.

This paper is structured in the following manner: Section 2 introduces briefly related concepts and related work. Section 3 describes the requirements, architecture, technical details and the user interface (UI) of the retrieval system. Section 4 describes three sample case studies with real-world data and presents how the retrieval solution in CS could be used to gain further insight into bibliometric data. Additionally, it also describes the feedback gathered from experts and discusses potential future research directions. The paper concludes with Section 5 where we discuss the current implementation and future work.

2. Related Work

2.1. Visual Analysis Bibliometric Approaches

A variety of modern solutions such as search engines [3, 4, 13], repositories [14, 15, 5] and services [16, 17] collect, create and retrieve large amounts of bibliometric data which can potentially provide new insights. This data can be analysed using VA, which is a science that aims to provide explainable insight into large abstract data through interactive data visualisation [18]. It can be combined with IR approaches to provide a deeper insight into retrieval results by visualising them and enabling the use of advanced tools for their analysis [19, 20, 21].

Bibliometric data analysis is usually demanding, tedious and time-consuming and can overwhelm novice researchers and the broader community. Even though various natural language processing (NLP) and IR approaches can be applied to bibliometric data, they might not produce insightful results [10]. Therefore, multiple visualisation techniques, VA tools and bibliometric-oriented solutions were created to provide better insights into the increasing amount of bibliometric data.

A commonly used graph-based visual analysis tool with a broad application range, including analysis of bibliometric data, is Gephi [11]. An example of a more narrowly focused bibliometric data tool is Galex which represents disciplines, areas and institutions as an interactive galaxy [22]. BiblioViz focuses specifically on table and graph visualisation to enable users to investigate bibliometric data from various perspectives [23]. Another tool for analysing publication data is VISPubComPAS which focuses on the analysis of institutions and authors [24]. Additionally, a solution for exploring university bibliometric data for driving strategic decisions is presented by [9].

As a result of this research area's growing popularity, multiple surveys were created covering different aspects and solutions. [10] provides an overview of interactive VA approaches for patent and publication data. Next, [8] report on approaches for extracting and visualising bibliometric data. Finally, [6] identify multiple solutions and two common workflows for processing and visualising publication data. These surveys indicate the potential of reviewed approaches but also identify multiple open challenges, such as lack of applications leveraging user interaction for analysis, lack of empirical research regarding the effectiveness of visualisation techniques and tools, visualisation of relationships between different data features and more.

2.2. Bibliometric-Oriented Search Systems

Although some of the aforementioned search engines and repositories provide further insight into author influence, relations between papers, and more, their main focus is still related to representing content as an ordered list. This almost never-ending list of results ranked by assumed relevance does not provide a way of gaining in-depth insights into data [6, 7, 1]. As identified by [2] IR systems should enable the execution of elaborate retrieval tasks, which might lead to more significant insights and drive decision making processes by leveraging visualisation methods to display connections in the retrieved data. Multiple approaches have been created to mitigate the issues of traditional bibliometric search engines by combining VA with IR. An example that leverages the above-mentioned connections is Rexplore, an analytics tool that enables retrieval of research publication data via facets and sorting of results [19]. Another example

[1] Data features represented as the graph nodes.
[2] Data features represented as graph edges.

is PivotSlice, which focuses on searching and analysis of retrieval results using a combination of filters and facets [20].

An example from industry is Connected Papers[3] which visualises retrieval results as connected graphs where papers are connected based on their similarity. Another similar solution is Open Knowledge Maps which visualises retrieval results as a multi-level bubble chart where papers are grouped based on text similarity [21]. Although there are many existing approaches and services, [7] identify the limits of these tools, focusing on providing search results in the form of individual papers or focusing on bibliometric analysis and provide a conceptual solution.

2.3. Collaboration Spotting

The approaches above are, to the authors' knowledge, either not actively developed anymore, are not accessible, cannot be used on large scale data, or too simple to provide users with advanced analytics insights.

As a possible alternative, CS is a graph-based VA platform that enables users to analyse large quantities of connected data through the use of filters, facets, and contexts [12]. Unlike other approaches, it can be used to analyse a wide variety of datasets and enables users to change the graph structure dynamically. A separate CSC version was developed to explore how to provide users with complete retrieval and analytics experience [25]. However, this version did not enable users to manipulate their subsequent searches with a finer granularity (for example by combining their selected nodes that represent the search result features with Boolean operators) since it relied on document embeddings and was never implemented in the primary CS version. Furthermore, it did not explicitly combine graph interactions with the retrieval process.

Based on insights and data analysis requirements, we aim to incorporate a prototype IR system into the primary CS platform to support users in performing complex retrieval tasks. As part of this process, we introduce a novel way of performing searches by exploring intrinsic graph patterns and selecting graph nodes from combinations of different features using the prototype. Furthermore, we add new connections to external services in CS and an analytics integration to perform empirical evaluation studies. Finally, we discuss use cases in bibliometric data analysis, describe possible approaches to analysing such data with CS, report feedback from expert interviews and discuss potential future research directions.

[3] https://www.connectedpapers.com/

3. Design and Implementation

3.1. Prototype Requirements

Based on the identified gaps and needs outlined in the previous sections, our goal is to build an enhanced graph-based retrieval and exploration prototype based on the existing CS system. As an example application scenario, we chose to use bibliometric data due to its vast accessibility. To this end, the retrieval system should provide sufficient flexibility to enable CS users to search via multiple queries and through a wide variety of data. Additionally, the retrieval system should leverage users' interactions to provide an efficient retrieval and exploration workflow. Furthermore, the system should enable the investigation of implicit connections between the entities of a dataset (e.g. Institution collaborations based on co-authorship). Finally, the expanded CS system should be ready for empirical analysis studies and gather interaction data. High-level requirements can be summarized as:

1. Support integration of multiple datasets and search providers.
2. Support exploration of implicit and explicit entity connections.
3. Collect user interaction data for empirical studies.
4. Visualise complex search results using various visual cues.
5. Enable exploration of search results using graph interactions.
6. Enable search query refinement through graph interactions.
7. Support complex search query creation.
8. Provide explainable report generation.
9. Enable visual creation of retrieval queries and filtering steps.
10. Enable graph analysis approaches to gain further insight.
11. Enable usage of graphs for knowledge retrieval.

As part of the initial prototype we focus on requirements 1 to 6.

3.2. Prototype Architecture

To address the novel combination of graph interaction and retrieval concepts described in this work and based on the above-listed requirements we enhanced the existing architecture seen in Fig. 1 with new components. The architecture is split into multiple conceptual components for clarity. However, in reality, the Graph Calculation, API Request Handlers and the Search are one module. This architecture is set to change once the move from a prototype to a production system is made. The *Graph Vis. & Interaction* (Fig. 1 a) component and the *Menus &*

Side-Panels (Fig. 1 b) component handle interactions such as selecting a search source, entering search queries, navigating graphs and selecting graph elements for search refinement. Once users start a new search, the *Search Handler* (Fig. 1 c) sends a request with their query and selected dataset to the *API Request Handler* (Fig. 1 d), which forwards the information to the *Search Source & Provider Selector* (Fig. 1 e) component. Here the request is parsed, and the appropriate data search provider is selected based on a project environment variable.

The currently supported data search providers include Elasticsearch[4], Whoosh[5] and the ArXiv API. Users who aim to perform an initial shallow exploration with a small amount of data and no advanced pre-processing can use the ArXiv API or an API from another existing hosted search provider. However, the introduction of new search providers would require implementing a new Python search component that would communicate with the search providers. On the other hand, users who aim to get a deeper insight into their data and perform a more thorough exploration can use an existing search provider like Elasticsearch or Whoosh. Furthermore, the latter search providers enable the use of time-demanding pre-processing and pre-analytic steps outside of the expanded CS system. For example, a user might wish to extract named entities or add additional data features before importing them into the system.

The selected dataset, query and search operator are sent to the *Search Handler* (Fig. 1 f) component, where the search is executed using the previously selected *search provider* (Fig. 1 g), and the results are transformed into a CS-specific format. Results represent a network of data out of which a graph corresponding to users selection is built using the *Graph Calculation* (Fig. 1 h) module, which retrieves the graph id from the newly generated graph. The id is then sent back to the Front-end to retrieve the newly generated graph. The users' retrieval and exploration process is enhanced by features such as navigation through the result graph and multiple iterations on their search. The interactions users perform on the Front-end are tracked using Matomo[6] as part of the *Analytics System* (Fig. 1 i) component for user behaviour and engagement analysis.

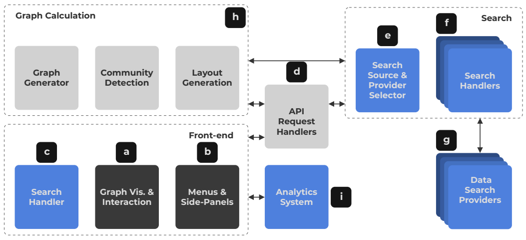

Figure 1: Simplified prototype architecture diagram. The filled rectangles represent conceptual modules in which the code is grouped. Light-grey rectangles represent previously existing modules; dark-grey rectangles represent previously existing modules that were updated, while the blue rectangles represent newly introduced modules. The arrows represent a simplified data flow between components. The dashed rectangles represent groupings of conceptual modules.

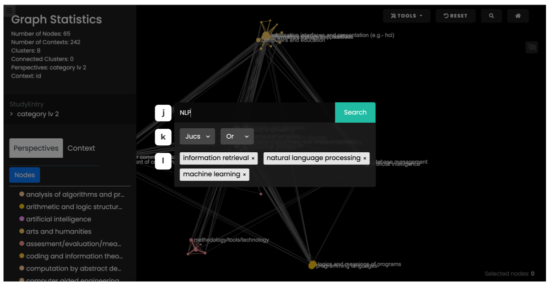

Figure 2: Prototype search interface. Users start by writing queries in the search field (j). They then select a search source and the binding operator for their queries (k). Next, they can inspect their search terms and delete terms from the search term list (l). Finally, they run their search by pressing the search button.

3.3. User Interface

To retrieve information on an initial dataset, users perform searches by opening the search modal seen in Fig. 2. Once they enter their queries, they can select one of the available data sources visible in the left drop-down (Fig. 2 k) and select a binding Boolean operator for the queries using the right drop-down. Additionally, they can also remove the queries from the query list (Fig. 2 l) by hovering over them and clicking the "x" button. They access the results by expanding the corresponding node into a graph, selecting the graph parameters of their choice from the menu, and navigating through the network. Furthermore, users can select one or multiple nodes, communities or connected components to perform another search. Once they selected the relevant nodes, they can open the search modal, whose search box is populated with the labels of the selected nodes as keyphrases. If they select a community or a connected component, only the most significant (based on size) three nodes will be retained for the search. Furthermore, users can navigate through the data network by selecting multiple perspectives and contexts and exploring either explicit connections in the dataset or identifying new implicit connections.

[4] https://www.elastic.co/elasticsearch
[5] https://whoosh.readthedocs.io/en/latest/index.html
[6] https://matomo.org/

3.4. Data Preparation

The dataset should be appropriately pre-processed to leverage the prototype's features effectively. The example dataset is retrieved from the Journal of Universal Computer Science (J.UCS) [26] since the authors had full access to it's detailed metadata. The data includes the *doi, title, abstract, authors, affiliations, author keyphrases and publication categories*. Since the author-defined keyphrases might be biased and reflect only on a subset of the paper content, we extract additional keyphrases using the keyphrase extraction tool YAKE! [27] to provide an alternative view on the paper content. Additionally, we split the publication categories into level 1, 2 and 3 to provide users with the possibility of exploring categorical data through graph navigation. We extend this data with data from Scopus[7] by extracting the affiliation name, affiliation city and affiliation country. Finally, we convert the data into a format that the CS platform can process. Once a graph is generated from the data, users can explore which authors and institutions collaborate, identify authors' focus categories, and more.

4. Case Studies and Evaluation

The focus of the case studies is on bibliometric data from the J.UCS journal as described above.

4.1. Case Studies

4.1.1. Potential Reviewers

A journal editor would like to identify potential reviewers for an IR and NLP paper. Using the prototype system, they search for "IR" and "NLP" on the J.UCS dataset. They first filter out the resulting journal categories that do not fall into one of the two above mentioned topics. Next, they navigate to a new key phrase graph where they select phrases closely related to NLP or IR and navigate to the author view. The authors are connected if they have joint publications. The editor can now identify potential candidates who are likely knowledgeable in the fields mentioned above and avoid authors who have previously published papers with the submission author.

4.1.2. Identification of Potential Collaborators

A company executive searches for online education using the prototype system to identify potential collaborators in online education. They explore the results from keyphrases' perspective to identify relevant phrases and use them to perform a search. Next, they explore and filter out countries that are not easily accessible from their location. They then explore institutions that are connected if their representatives wrote a joint paper. Using this view, the executive can identify institutions where they might know someone and establish a collaboration.

[7] The data was downloaded from Scopus in winter of 2020-2021 using the Python library Pybliometrics [28]

Table 1
Frequency of Special Characters

What is your occupation, and what are your daily tasks?
Where do you see the strengths of CS?
Where do you see the weaknesses of CS?
What could be improved in CS?
Did you identify any other uses-cases for the system?

4.1.3. Introduction to a New Topic

A novice researcher in software engineering explores the J.UCS categories using the prototype system. They further explore the author keyphrase of papers in the software engineering category to identify points of interest relevant to their research. They notice that software engineering is connected to formal methods and decide to investigate both topics' authors. Only a few authors published in J.USC about these topics, so they return to the previous author keyphrase graph and search for the same topics using the ArXiv API. The search results represent a more diverse set of documents that can be used to identify prominent authors in the field of interest by observing the node sizes.

4.2. Expert Evaluation

4.2.1. Study Environment

To identify further potential users' needs, we organised individual interviews with five experts from different domains who could benefit from using CS. The interviews were semi-structured to gain quick feedback that will guide further research and development efforts and potentially enable the discovery of additional edge cases that the authors might not have identified yet. Furthermore, we aimed to identify how to implement future versions of CS in particular in a way which will enable users to perform complex retrieval and analysis tasks, support users at multiple steps of the retrieval process and gain potential users' view for shaping future system features. As part of the interview, which was held as an online meeting, we presented the enhanced CS system, discussed the three use cases mentioned earlier and demonstrated how users could use CS for the first use case using a dataset from J.UCS as an example through screen sharing. Finally, the experts were asked the five questions depicted in Table 1. During the interview, they could ask to view specific sections of CS again and asked further questions about how the system works.

4.2.2. Study Participants

The first participant was a librarian with more than 30 years of experience who also had experience in database usage and is leading the library services for the last 11 years. The next participant was a computer scientist and doctoral student focusing on learning environments and learning analytics. The third participant was a postdoctoral researcher focusing on computer science and psychology who participated in research projects focusing on VA, UI design, mitigation of cognitive biases and more. The fourth participant was a senior data scientist who analyses literature based on clients' requirements and implements machine learning algorithms for various datasets based on this analysis. The final participant was a Knowledge Transfer Officer, who, among other things, focuses on patent and research paper exploration and retrieval. All participants were previously vaguely familiar with the project but did not know how it works or the details of how it can be used and what are its features.

4.2.3. Study Results

A commonly identified strength of the prototype compared to traditional web search systems is that users can explore results efficiently and avoid fine-tuning precision and recall through keyphrases by navigating through the graph. Additional strengths include the ability to identify relations between fields and authors, the visual feedback provided through the node sizes, the ability to make sense of information that would be difficult to analyse with simpler representations and the ability to explore implicit connections. An expert also mentioned that *"Navigation is the door to serendipity"*. In the context of the prototype system, navigation is well supported by enabling different perspectives and contexts.

We also identified much room for improvements. Suggestions include visualising other data relationships such as the impact of papers on different fields, using a wider variety of visual cues to display new dimensions and avoid node label overlap. The need to handle visualisations of multidimensional datasets was also identified by [10, 8]. Moreover, data should also be presented with traditional charts to give the user a familiar overview of the data. Furthermore, more quantitative details about the retrieved data and more insightful details such as the largest clusters and what they include were among the suggestions. Experts also proposed exploring ways of integrating financial data and general impact data[8] to increase the added value of data exploration. A similar conclusion was reached by [6] who suggest using social media for the expansion of scientific datasets. Furthermore, similarly to what was concluded in [10] experts suggested the use of other data types such as source code and multimedia attached to scientific work. Menus could be improved by including wording which calls for action[9] and is understandable for the general public. Additionally, it was proposed that they should take up less space. To simplify the graph search and exploration, the system should support natural language queries that can be automatically translated into search and exploration actions. The UI could be additionally improved by providing an onboarding tutorial with short introductory examples, introducing an advanced UI mode with the complete set of features and a simple UI mode that can be used to navigate through predefined templates and presenting a traditional list view of results alongside the graph view. The accommodation of novice users was recognised as a critical feature also by [6] who suggested that the amount of data shown should be adjustable in order not to overwhelm novice users. Furthermore, it was mentioned that creating reports based on the performed actions and enabling easy graph export with the search and navigation history and the option to customise the background colour to better fit in professional reports would be beneficial.

We also identified additional use cases such as creating yearly reports about larger institutions' publications, code analysis evaluation where concepts used and bugs encountered by each user could be visualised, and analysis of personal email corpora. A use case that two experts mentioned is the visualisation and exploration of employee skills and project participation inside companies.

In conclusion, the combination of IR and VA helps facilitate user exploration through graph navigation and helps avoid fine-tuning keyphrases for relevant results.

4.3. Future Research Directions

Based on the expert feedback, literature survey, initial requirements and our own experience, we identified several future research directions. Some of the identified directions are listed below.

IR aspects include the use of retrieved graphs not only for gaining analytical insights but also for advanced knowledge retrieval for example by exploiting graph patterns for further retrieval processes. Furthermore, we need to identify how to support user groups to perform multi-user retrieval and analysis tasks together. Another broad question identified by [1] is how to support users in complex retrieval tasks.

Graph analysis aspects may include content summarization of larger graph clusters, entity generation from

[8]For example, if a solution is mentioned in news articles without an explicit citation it should still count as a mention which contributes to the general impact of a work.

[9]For example *"Select by:"*

graph patterns and identification of improved clustering and layout techniques which might be more appropriate for the dynamic nature of the graphs in this work.

Machine learning aspects contain an exploration of conversational IR approaches to enhance users analytical abilities of result graphs as well as generate user models based on user interactions which could aid users in the retrieval process [1, 2].

Engineering aspects of future work include improved connection generation and system refactoring. The current system is not scalable and should be rewritten in modern technologies with modularity in mind. Furthermore, the connection calculation process should be refactored to avoid implying connections between points that might not directly connect in the retrieved dataset.

Evaluation aspects which represent the final key aspect and are a prevalent issue in VA systems are concerned with efficient quantitative evaluation, which will provide a clearer picture about the usefulness of the system [1].

5. Conclusion and Future Work

This paper describes a graph-based visual analytics and IR prototype that enables the search and exploration of data through a combination of IR and VA approaches. The solution is built as an enhancement to the CS system. As part of the IR process, users perform a traditional search whose results are then presented as an interactive graph that can be explored or used to perform multiple additional searches. To investigate how the introduced solution could help users in their retrieval process, identify users needs and ideas for future system development, we held interviews with five experts. Their answers indicate that the prototype does provide a helpful workflow for analysing data but that there is also room for improvement. Among the areas of improvement, we identified enrichment of the dataset using data from other domains, UI simplifications, the introduction of new interaction approaches and displaying the search result data in traditional and graph form. Furthermore, visualisations could be enhanced by additional visual cues. We also discuss future research directions that would be beneficial for the proposed system. We plan to improve and refactor the system and conduct an empirical study to gain further insight into how this approach can help support users in their retrieval process.

Acknowledgments

We want to thank the five interviewed experts for their time and for contributing valuable feedback. We would also like to thank André Rattinger for scraping and supplying the primary J.UCS dataset.

References

[1] Z. Chen, X. Cheng, S. Dong, Z. Dou, J. Guo, X. Huang, Y. Lan, C. Li, R. Li, T.-Y. Liu, et al., Information retrieval: a view from the chinese ir community, Frontiers of Computer Science 15 (2021) 1–15. doi:10.1007/s11704-020-9159-0.

[2] J. S. Culpepper, F. Diaz, M. D. Smucker, Research frontiers in information retrieval: Report from the third strategic workshop on information retrieval in lorne (swirl 2018), SIGIR Forum 52 (2018) 34–90. URL: https://doi.org/10.1145/3274784.3274788. doi:10.1145/3274784.3274788.

[3] P. Jacsó, Google scholar: the pros and the cons, Online information review 29 (2005) 208–214. doi:10.1108/14684520510598066.

[4] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. Hsu, K. Wang, An overview of microsoft academic service (mas) and applications, in: Proceedings of the 24th international conference on world wide web, 2015, pp. 243–246. doi:10.1145/2740908.2742839.

[5] P. Ginsparg, Arxiv at 20, Nature 476 (2011) 145–147. doi:10.1038/476145a.

[6] M. E. Bales, D. N. Wright, P. R. Oxley, T. R. Wheeler, Bibliometric visualization and analysis software: State of the art, workflows, and best practices (2020).

[7] J. P. Bascur, N. J. van Eck, L. Waltman, An interactive visual tool for scientific literature search: Proposal and algorithmic specification., in: BIR@ ECIR, 2019, pp. 76–87.

[8] J. Liu, T. Tang, W. Wang, B. Xu, X. Kong, F. Xia, A survey of scholarly data visualization, Ieee Access 6 (2018) 19205–19221. doi:10.1109/ACCESS.2018.2815030.

[9] P. Rosenthal, N. H. Müller, F. Bolte, Visual analytics of bibliographical data for strategic decision support of university leaders: A design study., in: VISIGRAPP (3: IVAPP), 2019, pp. 297–305. doi:10.5220/0007396302970305.

[10] P. Federico, F. Heimerl, S. Koch, S. Miksch, A survey on visual approaches for analyzing scientific literature and patents, IEEE transactions on visualization and computer graphics 23 (2017) 2179–2198. doi:10.1109/TVCG.2016.2610422.

[11] M. Bastian, S. Heymann, M. Jacomy, Gephi: an open

source software for exploring and manipulating networks, in: Proceedings of the International AAAI Conference on Web and Social Media, volume 3, 2009. doi:10.13140/2.1.1341.1520.
[12] A. Agocs, D. Dardanis, R. Forster, J.-M. Le Goff, X. Ouvrard, A. Rattinger, Collaboration spotting: A visual analytics platform to assist knowledge discovery, ERCIM News (2017) 46–48.
[13] S. Fricke, Semantic scholar, Journal of the Medical Library Association: JMLA 106 (2018) 145–147. URL: http://jmla.pitt.edu/ojs/jmla/article/view/280. doi:10.5195/jmla.2018.280.
[14] J. F. Burnham, Scopus database: a review, Biomedical digital libraries 3 (2006) 1–8. doi:10.1186/1742-5581-3-1.
[15] M. Ley, Dblp: some lessons learned, Proceedings of the VLDB Endowment 2 (2009) 1493–1500. doi:10.14778/1687553.1687577.
[16] S. Ovadia, Researchgate and academia. edu: Academic social networks, Behavioral & social sciences librarian 33 (2014) 165–169. doi:10.1080/01639269.2014.934093.
[17] V. Henning, J. Reichelt, Mendeley-a last. fm for research?, in: 2008 IEEE fourth international conference on eScience, IEEE, 2008, pp. 327–328. doi:10.1109/eScience.2008.128.
[18] J. J. Thomas, K. A. Cook, A visual analytics agenda, IEEE computer graphics and applications 26 (2006) 10–13. doi:10.1109/MCG.2006.5.
[19] F. Osborne, E. Motta, P. Mulholland, Exploring scholarly data with rexplore, in: International semantic web conference, Springer, Springer Berlin Heidelberg, 2013, pp. 460–477.
[20] J. Zhao, C. Collins, F. Chevalier, R. Balakrishnan, Interactive exploration of implicit and explicit relations in faceted datasets, IEEE Transactions on Visualization and Computer Graphics 19 (2013) 2080–2089. doi:10.1109/TVCG.2013.167.
[21] P. Kraker, C. Kittel, A. Enkhbayar, Open knowledge maps: Creating a visual interface to the world's scientific knowledge based on natural language processing, 027.7 Zeitschrift für Bibliothekskultur 4 (2016) 98–103. doi:10.12685/027.7-4-2-157.
[22] Z. Li, C. Zhang, S. Jia, J. Zhang, Galex: Exploring the evolution and intersection of disciplines, IEEE transactions on visualization and computer graphics 26 (2020) 1182–1192. doi:10.1109/TVCG.2019.2934667.
[23] Z. Shen, M. Ogawa, S. T. Teoh, K.-L. Ma, Biblioviz: a system for visualizing bibliography information, in: Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation-Volume 60, volume 60, Citeseer, 2006, pp. 93–102. doi:10.1145/1151903.1151918.
[24] Y. Wang, M. Yu, G. Shan, H.-W. Shen, Z. Lu, Vispubcompas: a comparative analytical system for visualization publication data, Journal of Visualization 22 (2019) 941–953. doi:10.1007/s12650-019-00585-2.
[25] A. Rattinger, J.-M. Le Goff, C. Guetl, Collaboration spotting cite: An exploration system for the bibliographic information of publications and patents., in: Proceedings of the 11th International Joint Conference on Knowledge Discovery, volume 1, 2019, pp. 548–554. doi:10.5220/0008366105480554.
[26] N. Baloian, J. A. Pino, G. Zurita, V. Lobos-Ossandón, H. Maurer, Twenty-five years of journal of universal computer science: A bibliometric overview, JUCS - Journal of Universal Computer Science 27 (2021) 3–39. URL: https://doi.org/10.3897/jucs.64594. doi:10.3897/jucs.64594. arXiv:https://doi.org/10.3897/jucs.64594.
[27] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, A. Jatowt, Yake! collection-independent automatic keyword extractor, in: European Conference on Information Retrieval, Springer International Publishing, Cham, 2018, pp. 806–810.
[28] M. E. Rose, J. R. Kitchin, pybliometrics: Scriptable bibliometrics using a python interface to scopus, SoftwareX 10 (2019) 100263. doi:https://doi.org/10.1016/j.softx.2019.100263.