



UNIVERSITÀ
DEGLI STUDI
DI UDINE

Università degli studi di Udine

A Web application for creating and sharing visual bibliographies

Original

Availability:

This version is available <http://hdl.handle.net/11390/1127628> since 2018-07-24T13:21:01Z

Publisher:

Springer International Publishing

Published

DOI:10.1007/978-3-030-01379-0_6

Terms of use:

The institutional repository of the University of Udine (<http://air.uniud.it>) is provided by ARIC services. The aim is to enable open access to all the world.

Publisher copyright

(Article begins on next page)

A Web application for creating and sharing visual bibliographies

Marco Corbato, Antonina Dattolo

SASWEB Lab, DMIF, University of Udine, Italy
marco.corbato@uniud.it, antonina.dattolo@uniud.it

Abstract. The increasing availability of repositories of peer-reviewed scientific literature, like Scopus and WOS, and the growth of Open Access data providers and specialized search engines offer to the users a huge amount of data, which is often difficult to connect and aggregate in sharable formats. Furthermore the deriving bibliographic metadata represent an interesting resource for research evaluation and for highlighting the groups and the progress in specific areas.

In order to build reasoned bibliographies on specific topics, key items are the papers, the authors, the interrelationships among papers and authors together with the connections given by citations.

Traditional repositories, although contain the single pieces of information, generally do not offer for them holistic, aggregated, and graphical views; each search furnishes new results, which are difficult to correlate with previous ones. Our work faces these issues proposing VisualBib, a Web application prototype: it enables user to progressively create, visualize and share bibliographies, which are represented by narrative views.

This paper presents VisualBib, a Web application realized to support the researchers who wish to create, refine and visualize bibliography; starting with a small set of significant papers or from a restricted number of authors, the users can enrich it by exploring citing/cited references. The narrative view and the explicit representation of the authorship and citing relations help users to build cohesive bibliographies and to disseminate the research on a specific topic through the sharing of personal points of view.

Keywords: Visual bibliography, Graphic organizer, Holistic view, Information visualization, Visualization design and evaluation methods, Human computer interaction (HCI)

1 Introduction

The search in scientific literature is traditionally carried on by specifying a set of words in specialized search engines. Generally a massive volume of documents are returned at every search and this involves a significant effort for the researcher in trying to map the results in a general vision of the topic. The results are generally presented in a long list that brings the user in a poor

understanding of the relationships between the documents, forcing him/her to play out complex heuristic strategies to rank the items and to build a consistent mental vision of the research topic. Using repository as Scopus or WOS is difficult to follow a paper in the time, and identify the relations among papers of different authors.

A way to discover significant documents is to start from the publications of given key authors or from a restricted set of well known papers, analysing and expanding the cited references. Also the citing references play an important role in discovering relevant documents, in understanding the direction of the research and to get inspiration for new projects.

In this paper we introduce VisualBib, a Web application which interacts with external data providers in order to retrieve bibliographic metadata, offering to researchers an interactive visual representation of the set of retrieved documents. The used representation is a time-based narrative view that shows the authors and citation relationships between papers, giving the users the opportunity to manage a bibliography, deleting documents or adding new ones, starting with an author, a document id or exploring the citing/cited references for every paper.

The idea is to give the users the opportunity to progressively build a reasoned and shareable bibliography, supported by a holistic view which highlights the collaborations between authors and help to get answers to questions like the following: who worked with whom in a certain period of time and on what; what are the sources of inspiration of a specific paper or what subsequent papers have followed a specific work.

This paper is organized as follows: Section 2 discusses related work, emphasizing open issues and challenges in graphical representation of bibliographies; Section 3 presents our prototype, VisualBib, introducing to its basic functionalities and user interface; Section 4 proposes its architecture and some implementation details; it follows in Section 5 the evaluation carried out as a comparative study between VisualBib and Scopus. Conclusions and future work end our work.

2 Related work

In the last years, several tools have been proposed to graphically represent bibliographic data and to support the researchers in analysing and exploring data and relationships.

Two recent surveys can be found in [1], where 109 different approaches, emerged between 1991 and 2016, are analysed using two dimensions for the classification, data types and analysis tasks; in [2], where authors present an interactive visual survey of text visualization techniques, which display 400 different techniques.

Some projects and tools emerged from the InfoVis2004 contest [3]: among them, BiblioViz [4] which integrates table and network 2D/3D views of

bibliographic data; PaperLens [5] where tightly couples views across papers, authors, and references are presented in order to understand the popularity of a topic, the degree of separation of authors and the most cited papers/authors.

Other visual interfaces for bibliographic visualization tools have been more recently proposed: for example, CitNetExplorer [6] provides the visualization of citation networks offering expansion and reduction operations and clustering of the publications in groups; PivotPaths [7] uses a node-link representation of authors, publications, and keywords, all integrated in an attractive interface with smoothly animations; JigSaw [8] is a visual analytic system which provides multiple coordinated views of document entities with visual connections across the documents; PaperCube [9] offers a suite of alternative visualizations based on graph, hierarchy, and timeline integrated into an analysis framework, although the project appears stopped in 2009. The genealogy of citation patterns, Citeology [10], connects papers titles organized in a chronological layout controlling the number of generations to display and the shortest path between two selected papers.

In spite of the great number of existing tools, we highlight some open challenges and lacks in the current apps, and relative to the importance of:

- creating a personal view of a bibliography, selecting authors and papers;
- saving personal views and sharing them in write or only-read modalities;
- providing a holistic narrative view of papers, authors and cited/citing relationships;
- retrieving metadata from multiple online repositories and not from static datasets;
- using an online, real-time Web app.

Our contribute focus on these objectives and propones an online Web application that distinguishes, from existing others, for these primary features.

3 Basic functionalities and user interface of VisualBib

VisualBib is an online app prototype (<http://sasweb.uniud.it/visualBib/>) conceived for supporting the researchers in the creation of reasoned bibliographies, starting from papers or authors of their interest.

Figure 1 shows a typical visual representation of a bibliography, generated by VisualBib, when searching an author by last name. In order to disambiguate among homonyms, the list of authors is enriched with their name, afference (if present and only in Scopus), subject areas (if present and only in Scopus), id and ORCID (if present) is displayed; once chosen the author, the user will see a temporally ordered list of his/her publications. The selection by the user of all (or part of) the publications, will allow him/her to visualize a narrative diagram, as shown in the bottom of Figure 1.

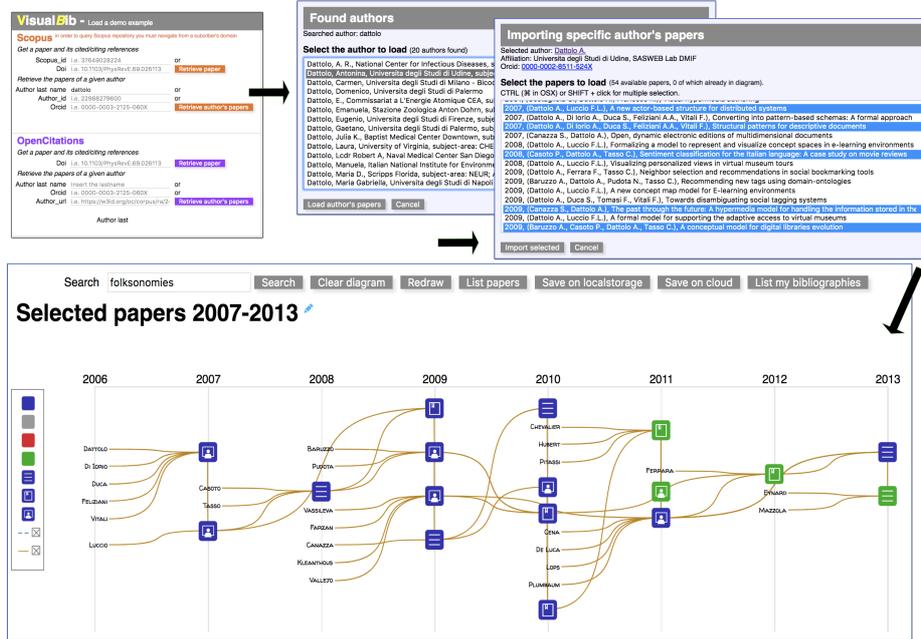


Figure 1. A look of VisualBib interface: search of an author given his/her last name.

The narrative diagram is 2-dimensional space: one dimension is the time, arranged horizontally and discretized by years; the vertical dimension is spatial and is used to properly organize authors, papers and their relationships. The coloured, round-corners square items (of blue and green colour in Figure 1) represent the publications retrieved from a data providers. The diagram includes the last names of the authors involved in at least one paper of the current set: each author is associated with a goldenrod line that connects all his/her papers, from the older to the newest, showing a sort of his/her professional path (clearly limited to the current set of imported publications) over the years.

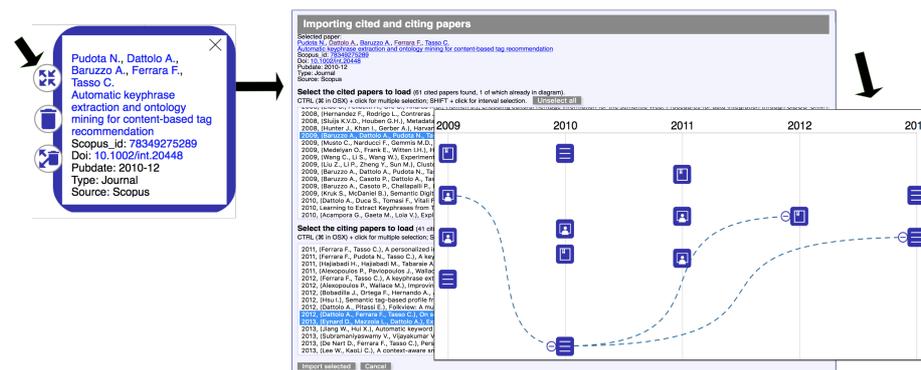


Figure 2. Adding citing/cited relationships.

The cited/citing relationships among papers are not automatically retrieved, but must be explicitly requested for the single paper: clicking over a paper icon (in the example in Figure 1, on the last (bottom) paper of the 2010), a pop-up window opens (see Figure 2, left-top) to show the complete bibliographic reference, where a list of data are linked to dedicated Web pages (in our example, each of the authors, the title, the id and the doi of the paper). A click on the four-arrows icon (see Figure 2, left-top) loads, in a separate pop-up window, the list of cited /citing papers (Figure 2, center). Among them, all the cited/citing papers, already present in the diagram, are automatically highlighted in blue. The user may select, from the two lists (cited/citing papers), the documents of his/her interest and import them (with the relative relations) in the diagram.

In the example of Figure 2, the user chooses to import the preloaded three papers and, as shown on the right of the same Figure 2, right, the cited/citing relations are visualized as blue dashed lines. Due to the potentially high number of citations, the system offers the possibility to recursively hide all cited papers through a “minus” icon situated on the left of the paper items.

Following the legend of Figure 1, expanded in Figure 3:

- the three different icons associated to publications enable users to distinguish among three paper’s typologies: journal papers; books or book chapters; conference or workshop proceedings. If the type is different or unknown, no specific icon is associated to the items;
- the icons’ colour indicates the papers state: **blue** is associated to a completely loaded paper (all the available data and metadata have been loaded); gray indicates a partially loaded paper, which has been retrieved during a cited/citing search (this operation returns only a subset of paper’s metadata); **red** is used to emphasize semantic relationships during user interaction, as described later; and, finally, **green** marks the found papers of a textual search (in Figure 1, they are the papers found looking for “folksonomies”);
- the two toggle switches allows user to hide/show respectively the connections between cited/citing papers and those jointing an author with his/her papers.

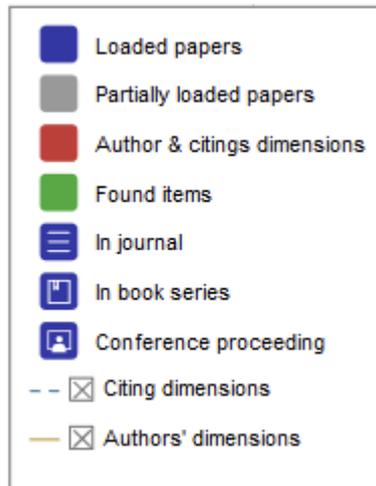


Figure 3. The legend.

Finally, moving the cursor over an *author name* the application will emphasize, in red, all his/her papers in the current view (Figure 4 - top: the author Vassileva); moving the cursor over a *paper* the application will emphasize in red all its authors' last names (Figure 4 - down left) and/or citing/cited connections (Figure 4 - down right).

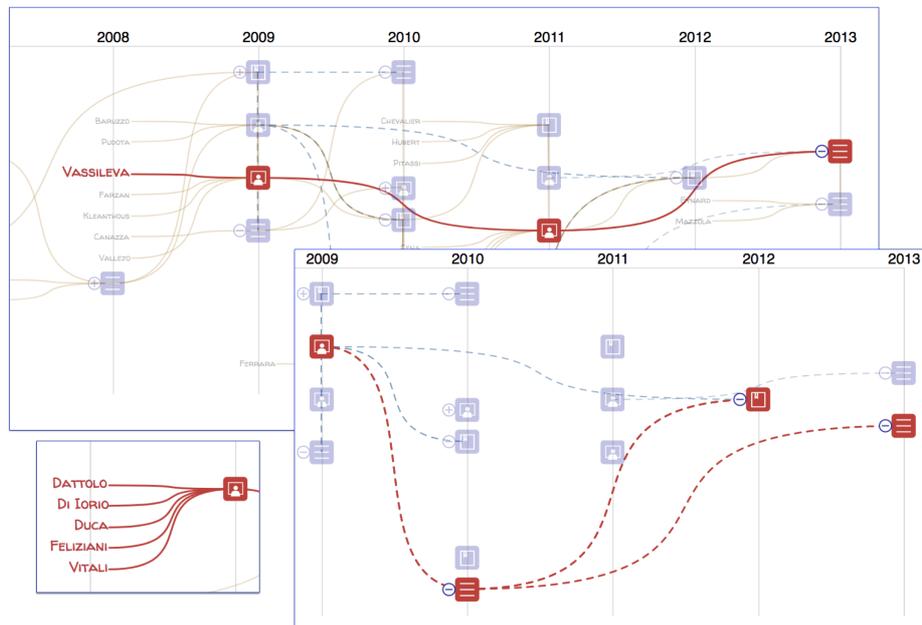


Figure 4. Focus on author(s) and citing/cited relations among papers.

4 Architecture and Implementation

VisualBib is organized as a single page Web application, based on:

- HTML5, CSS3 and SVG W3C standard languages;
- D3js [11], an efficient framework for data and DOM manipulation, and visual element management;
- AJAX techniques to perform Cross Origin Resource Sharing (CORS) calls and client-server interactions.

Figure 5 shows the architecture of VisualBib, and its main modules, described in next subsections 4.1-4.5.

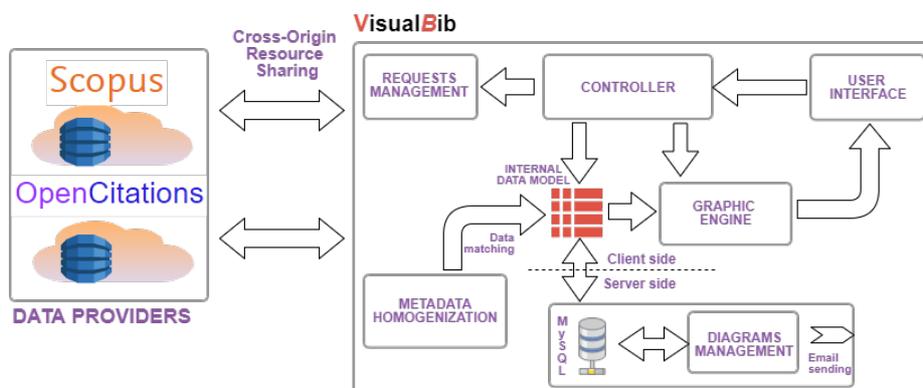


Figure 5. The architecture of the VisualBib application.

4.1 Data providers

VisualBib retrieves data from two repositories, the well-known Scopus and the open access OpenCitations (OC) [12]; other repositories will be considered in next future. Scopus [13] is the world's largest abstract and citation database of peer-reviewed research literature; it currently indexes more than 70 millions of bibliographic data, and is accessible programmatically via dedicated Application Programming Interface (API) which offers 11 different query types [14], aggregated in four groups: data search, data retrieval, other metadata retrieval, author feedback. Furthermore, each query returns data in various forms, called views, and organized in levels: each level provides a superset of the data exposed by the previous level; the access to the views is subject to restrictions due to service entitlements. To avoid misuses of data, the platform imposes some limitations [15] including selective weekly limits in the number of API calls, in the number of results returned in the response and in the number of calls per minute. To access to the service it is firstly necessary to require a personal API Key which permits the app to overcome the imposed restriction of the user agents in performing cross-origin HTTP requests. The platform manages both simple requests based on API key and

preflighted requests which generate a secured auth-token for the subsequent main request.

The OpenCitations (OC) platform [12] is the expansion of the Open Citations Corpus (OCC) [16], an open repository of scholarly citation data made available under a Creative Commons public domain; OCC is an ontology currently explorable through the use of SPARQL query language [17] and provides accurate bibliographic references harvested from the scholarly literature, that others may freely build upon, enhance and reuse for any purpose, without restriction under copyright or database law. OC aggregates different open access data sources and on December 25, 2017, published 298,797 citing bibliographic resources, 6,488,914 cited bibliographic resources and 12,652,601 citation links. The data can be freely downloaded but we chose to harvest the repository using specific SPARQL queries through the API service made available by the platform [18].

4.2 Internal data model

For each bibliography, VisualBib manages an internal representation of significant metadata:

- for papers: authors (with links), title (with link), publication year, abstract, subject areas, Scopus or OC ids (with links), doi (as link) , issn, references list. The links connect the metadata to the corresponding resource on the used repository (Scopus or OC);
- for authors: first, middle and last name (with link), preferred name, affiliation, ORCID, Scopus or OC id, and the list of his/her papers loaded into the narrative view.

A suitable data structure, represented by two multiple lists, contains cross-references, in order to replicate the many-to-many relations given by citations and authorships. The consistency of the structure is preserved during deletions of single or groups of papers, the loadings of new author's papers and citing/cited references.

4.3 Requests management and metadata homogenization

The data retrieval procedures from the two repositories differ substantially: although the availability of a common format for the data interchange (JSON), the typologies and the numbers of queries needed to get a same piece of information (for example the list of publications of an author or the paper metadata including citing/cited references) are not comparable.

The *request management module* prepares the correct sequence of queries and manages the responses and the error conditions. Due to the asynchronous nature of the AJAX calls, each query must be executed from the listen function of the previous call; furthermore, in case of fragmented responses, due to the existing limits on the number of results per single request, the internal

retrieve loop must be managed through recursive functions to assure that the next fragment is requested after receiving the previous one. The actions to be taken at the end of the process can be triggered only in the inner execution of the function, when the complete data transfer condition is observed.

The *metadata homogenization* module:

- handles the data received from each repository;
- carries out the necessary conversions of format to make the data compatible with the internal data model;
- performs the match of the incoming data, in order to map new papers and authors to those already present in the internal data model;
- builds and shows suitable forms to let the users choose the set of papers and citations to import into the current bibliography;
- merges the new items into the internal data model and create new data connections according to the detected authorship and citing relations.

The AJAX requests are triggered by the following user actions:

- *search of an author given his/her last name*. After sending the query, each fragment, received from the chosen data provider, is merged in the ordered list of authors with same last names; after the user disambiguation, the list of author publications is retrieved and visualized following the steps shown in Figure 1;
- *search of an author given his/her ORCID, Scopus id or a specific url of the OC ontology*. In this case the system will query the data provider to retrieve the list of all papers of a specific author. The list is compared with the internal data model in order to recognize which papers are already loaded in the current bibliography and which are new; a selection form, where existing papers are properly highlighted, is then presented to the user (Figure 1, top right);
- *search of a paper given its doi or Scopus id*. In this case the system verifies the existence of the requested paper in the selected data provider and subsequently retrieves all the available metadata, including the information about citing/cited papers. Each citing/cited paper is compared with current set in order to recognize possible matching papers. Then a selection form is presented to the users to allow them the selection of the papers to import (Figure 2 - center).

4.4 Graphic engine

The *graphic engine* module maps the internal data model into an interactive visual representation, called narrative view; examples are visible in previous Figure 1, Figure 2, Figure 4. The scale of the axes is dynamically computed at every change in the bibliography in order to cover, respectively, the entire temporal span and the maximum number of items per column.

The positioning of the items along the vertical dimension is critical for a

proper interpretation of the information in the diagram; a specific algorithm has been designed in order to achieve:

- no overlapping of the items in the same column;
- a balanced distribution of the papers icons along the vertical dimension;
- the correct positioning of the authors' labels on the immediate left of his/her older paper, in correspondence of the previous year;
- a regular space distribution between papers and authors' labels in every year column.

4.5 Server side management

In order to store and retrieve user generated bibliographies, VisualBib includes a server side module equipped with a MySQL database server and a Php interpreter. User diagrams are described by title, content represented in JSON format, email address relative to the owner and two unique urls: every time a new bibliography is saved (clicking on 'Save on cloud' button in Figure 1), the user is asked to specify an email address; the system generates a couple of unique urls for future accesses and/or for sharing with other users in both read-write and read-only modalities. Successively, the users may retrieve by email the list of their bibliographies, clicking on the 'List my bibliographies' button in Figure 1. VisualBib has also a simpler mechanism (activatable by clicking on 'Save on localstorage' button in Figure 1) to save a bibliography on client-side in the localstorage of the browser: it is a permanent (but erasable) not shareable space, fast and useful for frequent and temporary savings.

5 Evaluation

The evaluation has been carried out as a comparative study between VisualBib and Scopus, with the goal to collect user opinions and feelings after experiencing the platform by means of some search activities described later. It is important to clarify that the activities performed by the participants involved a subset of the functionalities of the two platforms; for this reason, the SUS values referred only to the considered aspects.

In particular, we intended to evaluate:

- the perceived usability level of the application using the well-known SUS (System Usability Scale) questionnaire [19];
- the feelings about the aesthetic and the innovative solution of the user interface, using 5-likert scales.

5.1 Study design and procedures

The participants were recruited on a voluntary basis among undergraduate students, researchers and professors of University of Udine and other

universities.

The overall participants were 67 (37 F, 30 M): 31 of them (21 F, 10 M) evaluated the Scopus platform and the other 36 (16 F, 20 M) our application VisualBib.

We prepared two short presentations (about 15 slides each) of the two platforms and distributed them to each participant in order to illustrate the interfaces and the basic functionalities available in each application.

For undergraduate students, with no significative experience in bibliographic search, we organized a live presentation to illustrate both the platforms, to clarify technical terms and to familiarize with the interfaces.

Before taking the survey, we asked the participants to perform 12 activities consisting in specific bibliographic searches in order to guide them in interacting with the platform under test. The activities were the same for the two platforms, took about 20-30 minutes to be carried out and consisted in searching the publications of some authors, counting the papers written in collaboration during a temporal interval, individuating the type of each publication, performing textual search in the metadata, searching specific citing and cited papers and the relative authors. To be sure that all the activities were executed correctly by all the participants, we asked them to fill the answers of each activity in an online form which provided the users with a negative feedback in case of wrong answer, forcing them in a loop until the right answer was acquired.

For the SUS evaluation, we adopted an Italian version of the standard SUS questionnaire, leaving out the first question: "I think that I would like to use this system frequently" [20]; this choice comes from observing that the systems under study would probably be used infrequently from the participants and the presence of that question would distort the score and probably confuse the participants.

We will refer to the this questionnaire as SUS-01: as the SUS, it is a mixed-tone questionnaire but, having dropped the first question, the odd-numbered items have now a negative tone and the even-numbered items have a positive tone.

Finally, we asked to the user to evaluate in a likert scale from 1 (not at all) to 5 (very much):

- the presence of innovative solutions in the application;
- the level of appreciation of the user interface and its aesthetic.

5.2 Data Analysis and results

Lewis and Sauro [20] studied the effects of dropping an item from the standard SUS questionnaire: specifically, when leaving out the first question, they measured a mean difference from the score the full SUS survey of -0.66 points, considering a 95% confidence interval.

Furthermore we are interested in estimating the difference between the SUS

score of the two platforms more than in absolute values.

The value of SUS-01 was computed for each participants with the formula:

$$SUS_{-01} = \left(\sum_{k=0}^4 (5 - A_{2k+1}) + \sum_{k=1}^4 (A_{2k} - 1) \right) * \frac{100}{36}$$

The distribution of SUS-01 is summarized, for the two samples, in Table 1 and in the graph of Figure 6.

Table 1. The distributions' parameters of the two samples.

Platform	Sample size	Min	1st Qu.	Median	Mean	Std. dev.	3rd Qu.	Max
Scopus	31	31.00	43.00	53.00	52.39	12.76	58.00	86.00
VisualBib	36	36.00	53.00	72.00	68.53	17.20	78.00	100.00

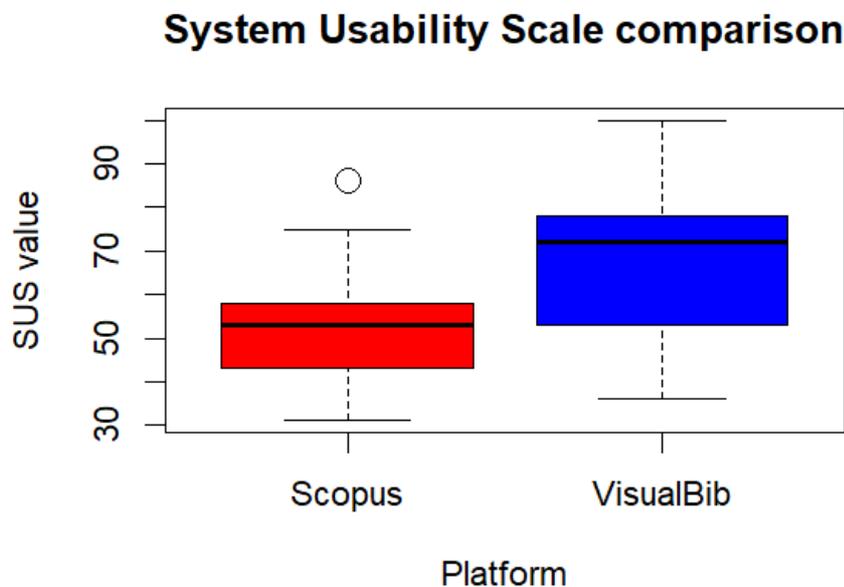


Figure 6. The evaluated SUS-01 distributions for the two platforms.

We note that the VisualBib's SUS-01 results present a mean value significantly higher in comparison with Scopus and a wider distribution. The absolute values of SUS for both platforms are relatively low; probably this fact reflects the difficulty of part of participants to face with bibliographic search tasks.

The distributions of the single SUS-01 answers, grouped per platform, are visible in the following Figures 7 and 8; for an easier data reading, in Figure 7, we reported the odd questions (negative tone questions), while in Figure 8 the

even questions (positive tone ones).

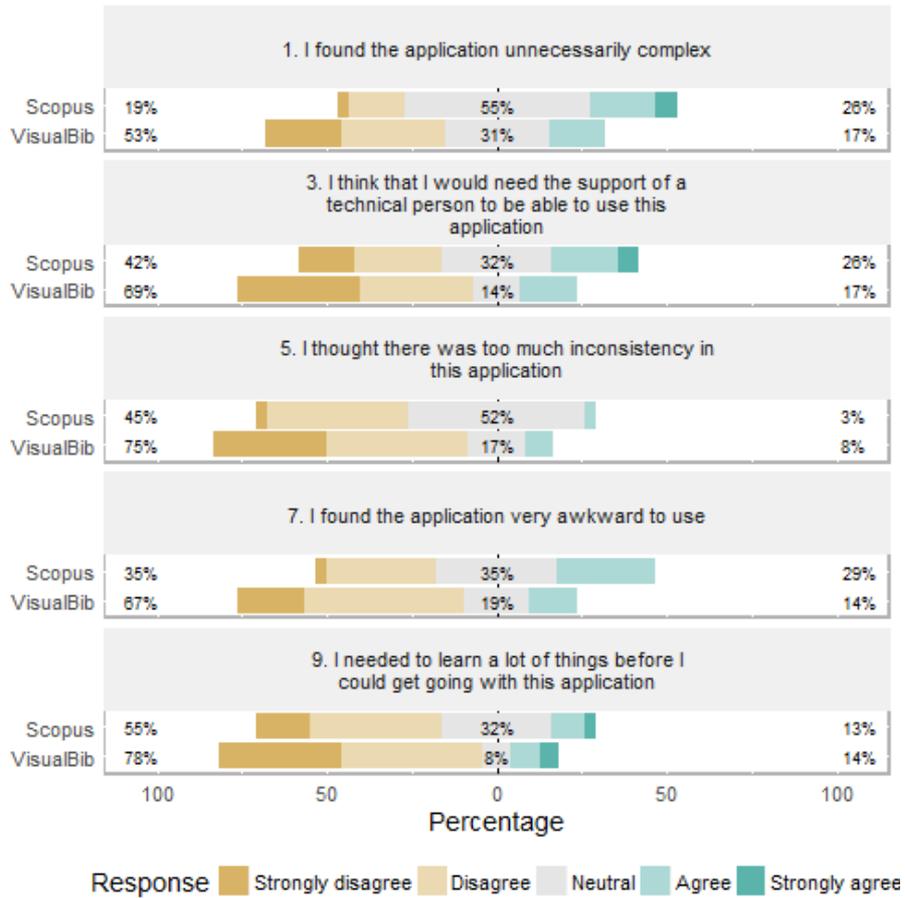


Figure 7. SUS-01: the comparative distributions of answers to the odd questions (negative tone questions) for the two platforms.

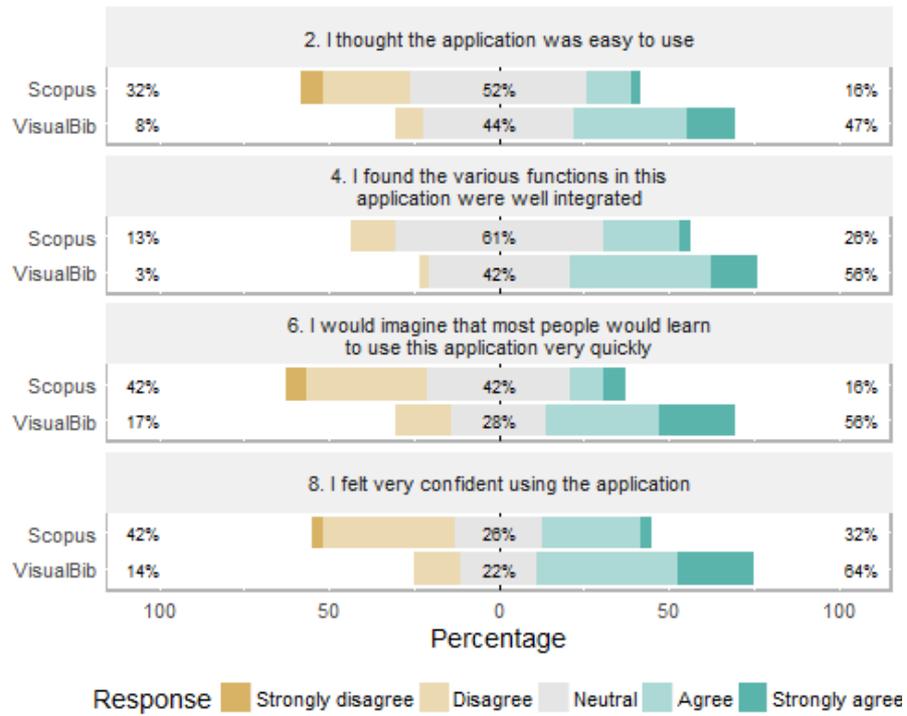


Figure 8. SUS-01: the comparative distributions of answers to the even questions (positive tone questions) for the two platforms.

Analysing the answers to the single questions, we observe an always positive difference in scores in favor of VisualBib, more pronounced in questions 1, 2, 6, 7, 8. This seems to underline the intuitiveness of the simpler interface of VisualBib which presents a smaller number of data and details in the page.

Figure 9 shows the distributions of the user opinions, grouped per platform, about the overall application and interface in a likert scale from 1 (not at all) to 5 (very much).

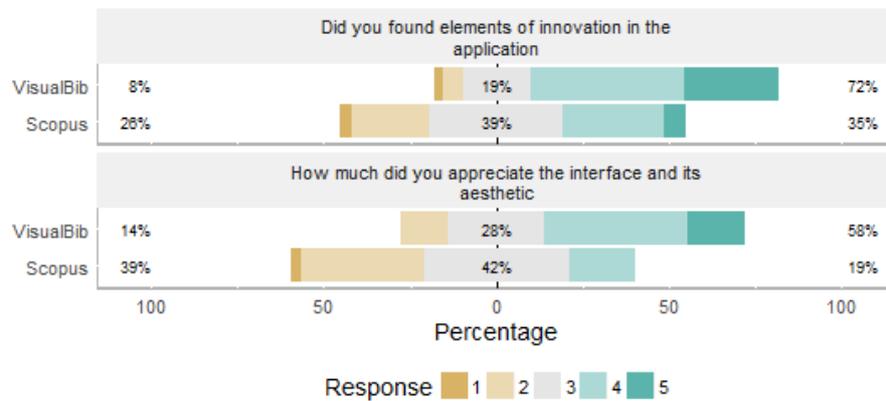


Figure 9. The evaluation of the overall application.

Also in this case the results encourage our work for the creation of this new app.

6 Conclusions and Future Work

The user experience during the search and building of reasoned bibliographies is often unsatisfactory for many reasons. In this paper we identified some critical aspects of this activity and proposed a prototype of a Web application which offers some original features to support the researchers in creating and refining a personal bibliography around a initial set of papers and authors. The system assists the user in creating a holistic view of a bibliography making visible the relationships between papers and authors and easily expand them through the exploration of citing/cited papers without losing the overview of the entire bibliography.

The partial user evaluation carried on, although still incomplete, highlights some interest from the user community for interactive visual representations of data which encourages us in proposing new applicable models and in investigating how they may help the users in reducing the cognitive overload and gathering, at a glance, significant relationships. Future work will involve improvements of VisualBib and the introduction of new features: i.e. we plan to implement the import and export of papers using the Bibtex format, to provide the editing of papers' data also to manage publications not found in current data providers, to expand the list of the repositories to query, to experiment new forms of visual representations in order to improve the harvesting of bibliographic information.

Acknowledgments

We would like to thank Meshna Koren for confirming the Elsevier interest in

our project and for having enabled us to use the Scopus API, applying specific settings; Silvio Peroni, for having enthusiastically contributed to support us in many occasions during the implementation of the remote access procedures relative to OpenCitations repository.

References

1. P. Federico, F. Heimerl, S. Koch, S. Miksch.(2017). A survey on visual approaches for analyzing scientific literature and patents. *IEEE Transactions on Visualization and Computer Graphics* 23:(9), pp. 2179–2198, doi: 10.1109/TVCG.2016.2610422.
2. K. Kucher, A. Kerren. (2015). Text visualization techniques: Taxonomy, visual survey, and community insights. *Proceedings of the 2015 8th IEEE Pacific Visualization Symposium - PacificVis 2015, Hangzhou, China, April 14-15 2015, Vol. 2015-July*, pp. 117–121, doi: 10.1109/PACIFICVIS.2015.7156366.
3. Infovis 2004 contest: dataset and tasks. (2004). <http://www.cs.umd.edu/hcil/iv04contest/info.html> (last visited, February, 1 2018).
4. Z. Shen, M. Ogawa, S. T. Teoh, K.L. Ma. (2006). Biblioviz: A system for visualizing bibliography information. *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation - APVis '06, Tokyo, Japan, February 1-3 2006, Volume 60*, pp. 93–102.35, Australian Computer Society, Inc.. <http://dl.acm.org/citation.cfm?id=1151903.1151918>
5. B. Lee, M. Czerwinski, G. Robertson, B. B. Bederson. (2005). Understanding research trends in conferences using paperlens. *Proceedings of the International Conference on Human Factors in Computing Systems - CHI '05 Extended Abstracts, Portland, Oregon, USA, April 2-7 2005*, pp. 1969–1972, doi: 10.1145/1056808.1057069.
6. N. van Eck, L. Waltman. (2014). Citnetexplorer: A new software tool for analyzing and visualizing citation networks. *Journal of Informetrics* 8:(4), pp. 802–823, doi: 10.1016/j.joi.2014.07.006.
7. M. Dork, N. Henry Riche, G. Ramos, S. Dumais. (2012). Pivot paths: Strolling through faceted information spaces. *IEEE Transactions on Visualization and Computer Graphics* 18:(12), pp. 2709–2718.
8. C. Görg, Z. Liu, J. Kihm, J. Choo, H. Park, J. Stasko. (2013). Combining computational analyses and interactive visualization for document exploration and sensemaking in Jigsaw. *IEEE Transactions on Visualization and Computer Graphics* 19:(10), pp. 1646–1663, doi: 10.1109/TVCG.2012.324.
9. P. Bergström, D. C. Atkinson. (2009). Augmenting the exploration of digital libraries with web-based visualizations. *Proceedings of the Fourth International Conference on Digital Information Management - ICDIM 2009, Ann Arbor, Michigan, USA, November 1-9 2009*, pp. 1–7, doi:

- 10.1109/ICDIM.2009.5356798.
10. J. Matejka, T. Grossman, G. Fitzmaurice, Citeology: Visualizing paper genealogy. Proceedings of the International Conference on the Human Factors in Computing Systems - CHI 2012, Austin, TX; United States; May, 5-10 2012, pp. 181–189, doi: 10.1145/2212776.2212796.
 11. D3: Data Driven Documents. <https://d3js.org/> (last visited, October, 23 2017).
 12. OpenCitations. <http://opencitations.net/>(last visited, January, 26 2018).
 13. Scopus. <https://scopus.com> (last visited, January, 26 2018).
 14. Elsevier Developers - API interface specification. https://dev.elsevier.com/api_docs.html (last visited, January, 26 2018)
 15. Elsevier Developers - API key settings. https://dev.elsevier.com/api_key_settings.html(last visited, January, 26 2018).
 16. OpenCitations corpus. <http://opencitations.net/corpus> (last visited, January, 26 2018).
 17. SPARQL Query Language for RDF. <https://www.w3.org/TR/rdf-sparql-query/> (last visited, January, 26 2018).
 18. OCC SPARQL endpoint GUI. <http://opencitations.net/sparql>(last visited, January, 26 2018).
 19. J. Brooke. (2013). SUS: a retrospective. *Journal of usability studies* 8:(2), pp. 29–40.
 20. J. R. Lewis, J. Sauro. (2017). Can I leave this one out? The effect of dropping an item from the SUS. *Journal of Usability Studies* 13:(1), pp. 38-46.