PaperQuest: a Visualization Tool to Support Literature Review

Antoine Ponsard
University of British Columbia
aponsard@cs.ubc.ca

Francisco Escalona
University of British Columbia
pax@cs.ubc.ca

Tamara Munzner
University of British Columbia
tmm@cs.ubc.ca

CHI'16 Extended Abstracts, May 07-12, 2016, San Jose, CA, USA
ACM 978-1-4503-4082-3/16/05.
http://dx.doi.org/10.1145/2851581.2892334

Abstract

The literature review is a key component of academic research, which allows researchers to build upon each other's work. While modern search engines enable fast access to publications, there is a lack of support for filtering out the vast majority of papers that are irrelevant to the current research focus. We present PaperQuest, a visualization tool that supports efficient reading decisions, by only displaying the information useful at a given step of the review. We propose an algorithm to find and sort papers that are likely to be relevant to users, based on the papers they have already expressed interest in and the number of citations. The current implementation uses papers from the CHI, UIST, and VIS conferences, and citation counts from Google Scholar, but is easily extensible to other domains of the literature.

Author Keywords

Literature Review; Sensemaking; Information Visualization

ACM Classification Keywords

H.5.2 [Information Interfaces and Presentation]: GUIs

Introduction and Related Work

The literature review is a key element of academic research, which ensures that progress is being made. It is particularly important for researchers entering a new domain, who are faced with the challenge of learning enough about it to make meaningful contributions. However, the scale of contemporary research makes this process difficult: even for a relatively well specified domain like Information Visualization, there are thousands of publications, spread across multiple conferences and journals, from authors scattered around the globe. Along with Zhang et al. [16], we believe that providing effective support for reviewing the literature can benefit the research community as a whole, and ultimately improve scientific productivity.

Two high-level approaches to visualizing the scientific literature can be distinguished. The first emphasizes the network structure of papers and citations, and often uses node-link diagram representations [1, 4]. However, as the number of papers increases, these representations turn into untangled hairballs. Showing citation relationships on demand alleviates this problem [13, 11, 7], but hides the big picture. The second approach emphasizes the multiple facets of the literature, such as authors, venue, year of publication, and keywords. PaperLens [9] and Netlens [8] provide multiple faceted views tightly linked to each other, showing either aggregate data or details about selected papers.

Yet, very little work actually addresses the specific task of a literature review. Instead of providing a top-down overview of the literature, there is a need to help users build a bottom-up understanding of a local neighborhood. To avoid wasted time and effort, only publications relevant to the researcher should be considered; so deciding what to read is a critical task. An early system was introduced by Mackinlay et al. [10], representing each paper as a “butterfly”, with its references on one wing and its citations on the other. The main concern of the authors was to handle very slow network connection to access remote databases. More recently, Zhang et al. proposed CiteSense [16], a text-based interface for searching, filtering, and organizing papers during a literature review. After finding an interesting paper, the user can see papers that cite or are cited by it, with a snippet of text providing the context of the citation. Some general-purpose sensemaking tools have also been used successfully for literature reviews, such as Jigsaw [14].

Apolo [3] builds upon the concept of the sensemaking loop [12] to explicitly support the creation of an external mental representation of the domain of interest, subdivided into several user-defined topics. From a seed paper, it fetches ten papers with a high number of citations, and tries to predict in which topic users are likely to consider them. As explained below, our own system focuses instead on finding the most relevant papers based on *all* the ones the user has expressed interest in, and provides easy access to the metadata and abstract of each paper.

Data

The scientific literature is an immense source of data, consisting of all the papers published, their metadata and relationships. In this project, we focused on the HCI and Infovis domains. Justin Matejka from Autodesk Research kindly agreed to share his own dataset with us, assembled for the Citeology tool [11], which contains papers for the CHI and UIST conferences between 1982 and 2010. We also retrieved the Visualization Publication Dataset, which contains papers from 1995 to 2014 [6].

Both datasets contain the paper title, Digital Object Identifier, year of publication, venue, authors, abstract, and references to other papers within the dataset. We extended these datasets with citation counts scraped from Google search results. With this technique, we were able to collect citation counts for respectively 81% and 83% of the papers. Finally, we traversed the network of references to compute and store the citations of each paper inside their datasets.

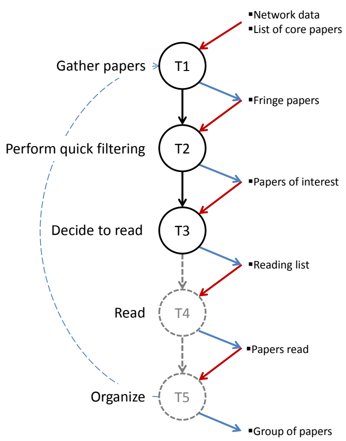

Figure 1: Task description.
T1: Gather papers from seeds.
T2: Read paper titles. Keep only the ones that have a chance of being relevant.
T3: Read the metadata, the abstract, and/or watch the accompanying video of the selected papers, to gather more detailed information on their content. Add the papers with the highest expected information gain to a "to read" list.
T4: Read papers from this list.
T5: Organize papers into different sub-categories.
PaperQuest's main strength is to improve T1, but it is not intended to support T4 and T5.

Task

We conducted a contextual inquiry with five colleagues, asking them to search online for three to five papers relevant to their current research project. We observed strong similarities in the process researchers follow to explore the literature. As it is difficult to begin a literature review effectively without an entry point in the domain of interest, researchers often start with one or more *seed papers*, usually provided by someone more knowledgeable about the field, or found by keyword search on Google Scholar. After looking up these seed papers, researchers want to discover related papers, which is classically done by browsing the references of the seed papers, a "backward search"; or the citations of these papers, a "forward search", nowadays available in many online libraries. The number of papers found in this process is potentially very large, because papers reference dozens of other papers and are sometimes cited hundreds of times. Reading each reference and citation is impossible. Therefore, a crucial step is to *filter* the papers that have been found, to identify the ones that will provide the most information relevant to the current focus.

An effective strategy is to follow a *multi-level decision process*, in which one gathers on each paper only the minimal amount of information necessary to decide whether to keep it or not for the next level (Figure 1). This process is flexible and highly iterative: after reading some papers, one gets a better understanding of the domain of interest, and gathers new papers that will eventually be filtered and read. It is also common to do small-scale iterations, such as going back to reading paper titles after reading the abstracts of a few papers. The multi-level decision process affords some sort of *batch processing*, where there is often more than one paper being considered at each step. While this is not required, we believe that the cognitive cost of task-switching repels users from processing only one paper at a time.

Finally, an important aspect underlined by Chau et al. [3] is that people build a mental representation of the domain they are exploring by classifying papers into different subtopics. This classification can happen during any of the steps described above, as soon as enough information has been collected on the paper. Yet it may change significantly towards the end of the literature review, when a better mental representation has been found. The resulting classification is commonly used to write subsections of the "Related Work" section of a paper.

Conceptual Design

A literature review is an exploration of the space of previously published papers. This space can be divided into three subspaces, as follows. The Core: papers already read, upon which a researcher build their understanding of the field. The Fringe: papers a researcher has access to, because they reference or are cited by some papers from the Core. The Unknown: an immense and terrifying abyss made of all the papers away from the Core.

As the literature review progresses, some papers from the Fringe will be added to the Core, which in turn will cause new papers to enter the Fringe. However, most of the papers will remain forever in the Unknown. To support the multi-level filtering process described above, we add another subspace: the To Read list, consisting of interesting papers gathered from the Fringe, but not read yet. This temporary buffer space is made necessary by the fact that reading papers usually takes much longer than the other steps of the filtering process, such as reading paper titles.

In the citations network, we define the Fringe as the papers that are one hop away from a paper in the Core or the To Read list. The problem of exponential explosion described in the Tasks section applies here: each paper references

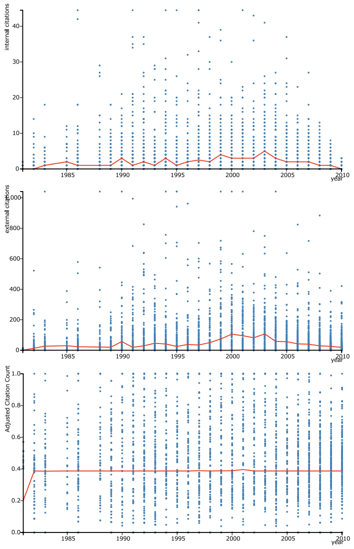

Figure 2: Internal (top) and external (bottom) citation counts follow a similar power law distribution, with a majority of papers with zero or a few citations, and a long tail of papers with very high citation counts. The scale of these distributions is however drastically different. We scale both citation counts to the [0,1] range, spreading them evenly with a square root function. Each year is then adjusted linearly so that the median citation count—shown as a red line—is consistent across years of publication (bottom).

and can be cited by many other papers. Yet, most of these papers are probably irrelevant to the current research focus. We therefore propose to order the Fringe based on a relevance score computed for each paper.

Relevance Algorithm

The purpose of the relevance algorithm is to find papers related to those that the user found interesting. Relatedness is however hard to define, and even harder to compute. Our dataset does not contain author keywords, nor any kind of hierarchical organization. Natural Language Processing techniques might be able to identify clusters of papers, but were beyond the scope of this project. Instead, we rely on one fundamental characteristic of the scientific literature: the fact that authors cite previous work that they build upon. By interpreting these citations as links in a network, we define relatedness as connectedness.

Connectedness Measure

The algorithm works on multiple sets of papers. The *Interesting* papers are the ones for which the user has expressed some interest, either by adding them to the *Core*, to the *To Read* list, or by *selecting* them on the Fringe. The *Fringe* is the one-hop neighborhood of the Interesting set: papers that either cite or are referenced by at least one interesting paper. For each paper in the Core, the To Read list, and the Fringe, we compute a *connectedness measure* as the weighted sum of all the links between this paper and other papers in the Interesting set. The weights represent the level of interest of the user for each paper contributing to the connectedness measure. We infer this interest from the set these papers belong to: 1 for Selected papers, 3 for To Read, and 5 for Core. We do not make any distinction between references and citations, as we consider both to be indicative of relatedness.

Relevance Score

To determine the relevance of a paper, we combine three quantitative metrics: an "internal" citation count from references within the dataset; an "external" citation count scraped from Google; and the connectedness measure computed by our algorithm. However, these three metrics have very different scales. We normalize them to [0,1] to allow meaningful comparisons (Figure 2).

The relevance score of a paper is then computed as the sum of its normalized connectedness and its adjusted citation count. We indeed consider high connectedness and high citation counts to be enough on their own to make a paper relevant. A paper not often cited, but strongly connected to other interesting papers, could provide pertinent insights, even though this paper may not be useful to the research community at large. Similarly, it is good to be aware of highly cited papers in one's field, even if they are only loosely connected to one's current focus. The relative weight of the normalized connectedness versus the adjusted citation count is a free parameter in our relevance algorithm. After testing our system on a set of papers related to our research interests, we decided to keep this relative weight to one, as we did not find any reason to favor one above the other.

Visualization

The relevance algorithm is the main component of PaperQuest. The visualization is therefore organized around the suggestions it provides. The Fringe shows paper titles in full, and sort them by relevance in a vertical list (Figure 3). Reading paper titles is indeed the first step of the decision process, and the most efficient for filtering out the vast majority of irrelevant papers. Compared to the traditional node-link diagram [1, 4], the list layout affords a natural top-down reading order, and helps convey which papers are the most

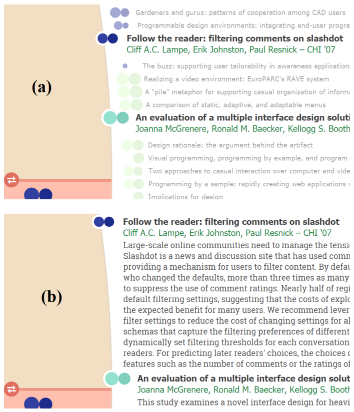

Figure 4: (a) In the first level of semantic zoom, selected papers are highlighted and their metadata is shown, while the rest of the papers are shrunk and faded. (b) At the next level, the full abstract of selected papers is shown, and other papers are hidden. Users can open the digital library entry for a given paper by clicking on the list of authors.

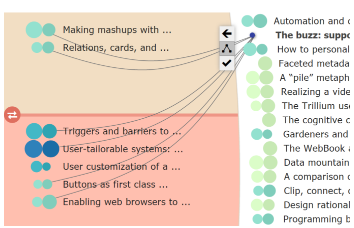

Figure 5: The curved links indicate that the selected paper cites papers in the To Read list and in the Core.

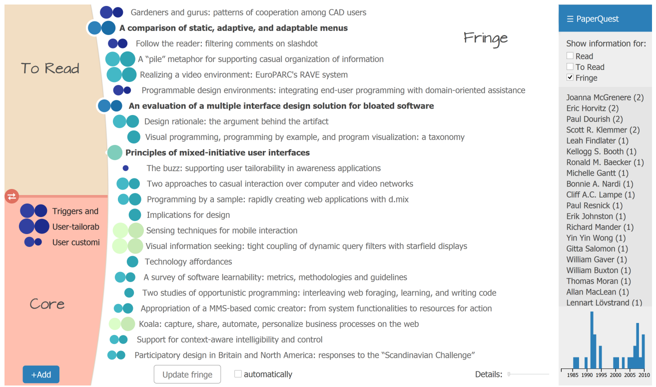

Figure 3: The main interface of PaperQuest. Three papers have been selected in the Fringe, and three more are already present in the Core. The To Read list is empty so far. Below the Fringe an update button is faded out, indicating that the Fringe is up to date. The borders between regions can be dragged, which allows users to give more screen real estate to the region of their current focus.

relevant. The main view contains two other regions: the Core and the To Read list (Figure 3). Users can move papers between these regions via the contextual menu that appears when hovering on a paper. The slight curvature of the Fringe is intended to reinforce the conceptual design of a core surrounded by a fringe, so that papers seem to move from the *outside* to the *inside* when they are moved from right to left in the interface.

Users can select and deselect papers on the Fringe simply by clicking on their titles. These actions provide additional information to the relevance algorithm, which can then reorder the Fringe more accurately. Yet continuous changes can be overwhelming, so users remain in control of when to update the Fringe: manually or automatically. The update animations are staggered: papers move first to their new position, then change color if needed.

By selecting papers on the Fringe, users convey their *degree of interest* to the system. Following the concept of Generalized Fisheye [5], we provide users with a *semantic zoom*: scrolling on the Fringe reveals more and more information about the selected papers (Figure 4). Finally, users can display all the citations and references of a paper via the contextual menu (Figure 5). Links are drawn as curved arcs, as suggested by van den Elzen and van Wijk [15], with the clockwise curvature indicating that the source makes reference to the target.

Glyph Design

The algorithm combines internal and external citation counts, as well as a custom connectedness measure, into a single score. No matter how advanced this algorithm might be, it will always make wrong predictions, or simply will not match the criteria of the user at a particular time. For this reason, we decouple the three quantitative pieces of information available for each paper, and display them as a glyph to the left of each paper title (Figure 6).

Because internal and external citation counts are semantically similar, we encode them in the same way: as the area of a disk. Indeed, our task analysis suggests that precise comparisons are not required: users simply need to quickly get a sense of how popular different papers are. The exact citation counts—normalized and not—are available on demand for each paper by hovering on the glyph. The connectedness measure is mapped to a sequential color scale with monotonically increasing luminance, retrieved and adjusted from ColorBrewer [2]. As explained above, we consider connectedness and citation counts as two orthogonal metrics, but equally relevant. We made sure to encode them in strongly separable visual channels.

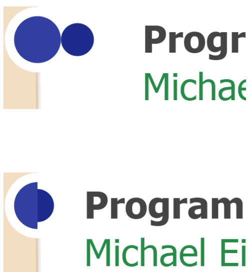

Figure 6: The glyph is a single visual entity that encodes the relevance of a particular paper. We considered two variants: “butterfly” (top) and “half-moons” (bottom). The latter is more concise, but the “butterfly” variant may allow more accurate comparisons between papers. After presenting the variants to several potential users, we did not find any reason to favor one against the other, so we kept both. An entry in the top-right menu let users specify their personal preference.

Additional Views

We provide two linked views in a sidebar (Figure 3), to display other facets of the information shown in the main view. The most frequent authors are shown as a list, sorted by the number of papers they co-authored. A histogram shows the publication years of the papers appearing in the main view. Its purpose is twofold: to provide a sense of the popularity over time of the topic the user is currently exploring; and to identify potential gaps in their set of Core and To Read papers, compared to those that appear on the Fringe. Such a mismatch would indicate that the user is not aware of a related subtrend in their domain.

Discussion and Future Work

We collected preliminary feedback from four student colleagues, one postdoctoral researcher, and three faculty members in Information Visualization. Overall, PaperQuest seems to have the potential to be a helpful tool for literature reviews. A formal evaluation is however needed to establish whether a researcher can find more relevant papers with our tool, or find them more quickly, compared to relying only on Google Scholar. Our encoding choices for the paper relevance glyph must also be validated.

The main limitation of the current prototype is the lack of scalability. Each view can show up to a few dozen papers, because we display titles in a font size that is comfortable to read, but takes up considerable space. Letting users scroll in each view could alleviate the problem to a certain extent; but a scalable solution would need to aggregate similar papers when needed. Furthermore, we could investigate other representations of papers that use less pixels. One or two keywords might be enough for users to remember the papers they have read—such as “PaperQuest” for this one. However, this approach is not applicable to the Fringe, for which full paper titles cannot be hidden without impacting the decision process of the user.

In the sidebar, the list of authors and the histogram of publication years could be used to highlight or filter out corresponding papers in the Fringe. The Core should offer a node-link diagram representation of all the papers already read, showing how they connect to each other. To further support sensemaking, users should be able to label papers and to group them into meaningful categories. The parameters of the relevance algorithm and the normalization constants of the citation counts were determined heuristically, but advanced users should be given the possibly to adjust them. For instance, a slider in the main view could be used to dynamically set the relative weight of citation counts and the connectedness measure in the relevance score, and observe its effect on the Fringe. Once PaperQuest is deployed and used, the relevance algorithm could be improved by collaborative filtering: users would get recommendations based on what other people have read.

A demo of PaperQuest is available at http://www.cs.ubc.ca/group/infovis/software/PaperQuest.

Conclusion

PaperQuest supports the literature review process by helping researchers decide which paper to read next. The three regions of the main view enable a multi-level decision process, from selecting papers on the Fringe, to adding them to a To Read list, to organizing them in the Core. Our main contribution is in presenting just as much information as needed at every step. A carefully designed glyph encodes information on each paper, and provides a visible rationale for the decisions of our custom relevance algorithm. Our goal is to offer a smarter and richer exploration of the literature than what is currently afforded by search engines and digital libraries.

References

[1] K. Borner and L. Viswanath. 2004. Major Information Visualization Authors, Papers and Topics in the ACM Library. In *IEEE Symposium on Information Visualization*. IEEE, r1–r1. DOI:http://dx.doi.org/10.1109/INFVIS.2004.45

[2] Cynthia Brewer, Mark Harrower, Ben Sheesley, Andy Woodruff, and David Heyman. 2013. ColorBrewer. Version 2.0. (2013). http://colorbrewer2.org/ (Retrieved on 2014-10-17).

[3] Duen Horng Chau, Aniket Kittur, Jason I. Hong, and Christos Faloutsos. 2011. Apolo: Making Sense of Large Network Data by Combining Rich User Interaction and Machine Learning. In *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*. ACM Press, New York, New York, USA, 167. DOI:http://dx.doi.org/10.1145/1978942.1978967

[4] Niklas Elmqvist and Philippas Tsigas. 2007. CiteWiz: a tool for the visualization of scientific citation networks. *Information Visualization* 6, 3 (Oct. 2007), 215–232. DOI:http://dx.doi.org/10.1057/palgrave.ivs.9500156

[5] George W. Furnas. 2006. A fisheye follow-up. In *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*. ACM Press, New York, New York, USA, 999. DOI:http://dx.doi.org/10.1145/1124772.1124921

[6] Petra Isenberg, Florian Heimerl, Steffen Koch, Tobias Isenberg, Panpan Xu, Chad Stolper, Michael Sedlmair, Jian Chen, Torsten Möller, and John Stasko. 2015. Visualization Publication Dataset. Dataset: http://vispubdata.org/ . (2015). http://vispubdata.org/ Published Jun. 2015.

[7] Petra Isenberg, Tobias Isenberg, Michael Sedlmair, Jian Chen, and Torsten Möller. 2014. Toward a deeper understanding of Visualization through keyword analysis. (Aug. 2014). http://arxiv.org/abs/1408.3297

[8] Hyunmo Kang, Catherine Plaisant, Bongshin Lee, and Benjamin B Bederson. 2007. NetLens: iterative exploration of content-actor network data. *Information Visualization* 6, 1 (Jan. 2007), 18–31. DOI:http://dx.doi.org/10.1057/palgrave.ivs.9500143

[9] Bongshin Lee, Mary Czerwinski, George Robertson, and Benjamin B. Bederson. 2005. Understanding research trends in conferences using paperLens. In *CHI '05 extended abstracts on Human factors in computing systems - CHI '05*. ACM Press, New York, New York, USA, 1969. DOI:http://dx.doi.org/10.1145/1056808.1057069

[10] Jock D. Mackinlay, Ramana Rao, and Stuart K. Card. 1995. An organic user interface for searching citation links. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*. ACM Press, New York, New York, USA, 67–73. DOI:http://dx.doi.org/10.1145/223904.223913

[11] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2012. Citeology: visualizing paper genealogy. In *Proceedings of the 2012 ACM annual conference extended abstracts on Human Factors in Computing Systems Extended Abstracts - CHI EA '12*. ACM Press, New York, New York, USA, 181. DOI:http://dx.doi.org/10.1145/2212776.2212796

[12] Daniel M. Russell, Mark J. Stefik, Peter Pirolli, and Stuart K. Card. 1993. The cost structure of sensemaking. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93*. ACM Press, New York, New York, USA, 269–276. DOI:http://dx.doi.org/10.1145/169059.169209

[13] John Stasko, Jaegul Choo, Yi Han, and Mengdie Hu. 2013. Citevis: Exploring conference paper citation data visually. *Proceedings of the IEEE Conference VIS 2013* (2013), 2–3. http://users.soe.ucsc.edu/~pang/visweek/2013/infovis/posters/stasko.pdf

[14] John Stasko, Carsten Görg, and Robert Spence. 2008. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization* 7, 2 (2008), 118–132. DOI: http://dx.doi.org/10.1057/palgrave.ivs.9500180

[15] Stef van den Elzen and Jarke J. van Wijk. 2014. Multivariate Network Exploration and Presentation: From Detail to Overview via Selections and Aggregations. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (Dec. 2014), 2310–2319. DOI: http://dx.doi.org/10.1109/TVCG.2014.2346441

[16] Xiaolong Zhang, Yan Qu, C. Lee Giles, and Piyou Song. 2008. CiteSense: Supporting Sensemaking of Research Literature. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*. ACM Press, New York, New York, USA, 677. DOI: http://dx.doi.org/10.1145/1357054.1357161