On the fringe: a visualization tool to support literature review

Antoine Ponsard aponsard@cs.ubc.ca
Francisco Escalona pax@cs.ubc.ca

Introduction

Researchers entering a new domain are faced with the challenge of learning enough about it so that their contribution can be meaningful. One part of this involves browsing thousands of publications to find what to read, and what to ignore. This is the process of literature review, and it is a common task for graduate students and even experienced researchers. To avoid wasted time and effort, only publications that are meaningful and valuable to the researcher should be considered, so deciding what to read is an important task in itself.

For our graduate studies in the field of Human-Computer Interaction we have faced this task, and therefore have limited but direct experience with it. Besides, we have access to colleagues who also have experience with this task, and plan use their expertise to guide our design. We believe the decision-making process can be helped by a visualization tool that allows users to identify and build up a list of candidate publications, and then to distill, categorize and prioritize them. We propose to build such a tool, to support the task of literature review in general, and in particular the decision of what to read next. We hope to create a tool that has real world impact and usage.

Data

The scientific literature is an immense source of data, consisting of all the papers published, their metadata and relationships. Our own interests lie in the fields of HCI and InfoVis; for the sake of feasibility, we will focus on these two areas. Justin Matejka from Autodesk Research has kindly agreed to share his own dataset with us, assembled for the Citeology tool [6], which contains papers and citations for the CHI and UIST conferences between 1982 and 2010. We also have access to a dataset of Infovis conference papers used for CiteVis [8], a project from the II Lab at Georgia Tech. Both datasets contain the paper title, abstract, year of publication and conference, authors, and references to other papers within the dataset. The CiteVis dataset also contains citation counts, and we are planning on acquiring the same data from Google Scholar for the Citeology dataset.

The references and citations make up a directed network of papers. Another, independent network, arises from linking papers that have authors in common, and dually authors can be seen as nodes and papers as links between them. We consider the rest of the data as attributes of the papers themselves.

Tasks

It is very difficult to begin a literature review effectively without an *entry point* in the relevant literature. Therefore, we assume that people have one or more *seed papers* available to them, usually provided by someone more knowledgeable about the field, or found by keyword search on Google Scholar. After *looking up* these seed papers, people want to discover related papers, which classically is done by browsing the references of the seed papers, a "backward search", or the citations of these papers, a "forward search" which is available today on 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 of them is impossible.

Therefore, a crucial step in the literature review process is to *filter* the papers that have been found, to identify the ones that will provide the most information relevant to the domain of interest. An effective strategy is to conduct a *multi-stage* sieving, in which we gather on each paper only the minimal amount of information necessary to decide whether to keep it for the next level or not. In practice, we have observed that people seem to follow a similar approach:

1. Read only the paper titles, and keep only the ones that have a chance of being relevant. *One of our interviewees did so by opening each paper's ACM DL page in a new tab of their web browser, by clicking on Google Scholar's search results.*
2. Read the abstract, the metadata and/or watch the accompanying video of the selected papers, then add the ones with highest expected information gain to a "to read" list. *This can be accomplished by downloading the PDFs to a folder, or adding the papers to a reference manager.*
3. Read papers from this list.

This process is generally a loop: after reading some papers people get a better understanding of the domain of interest, and gather new papers that they will eventually filter and read. It is also common to do smaller loops, such as going back to reading paper titles after adding a few papers to the "to read" list (2 -> 1). Note that we are describing 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. [2] is that people build a mental representation of the domain they are exploring by classifying papers into different sub-topics. This classification can happen during any of the three steps described above, as soon as enough information has been collected on the paper. The resulting classification is commonly used to write subsections of the "Related Work" section of a paper.

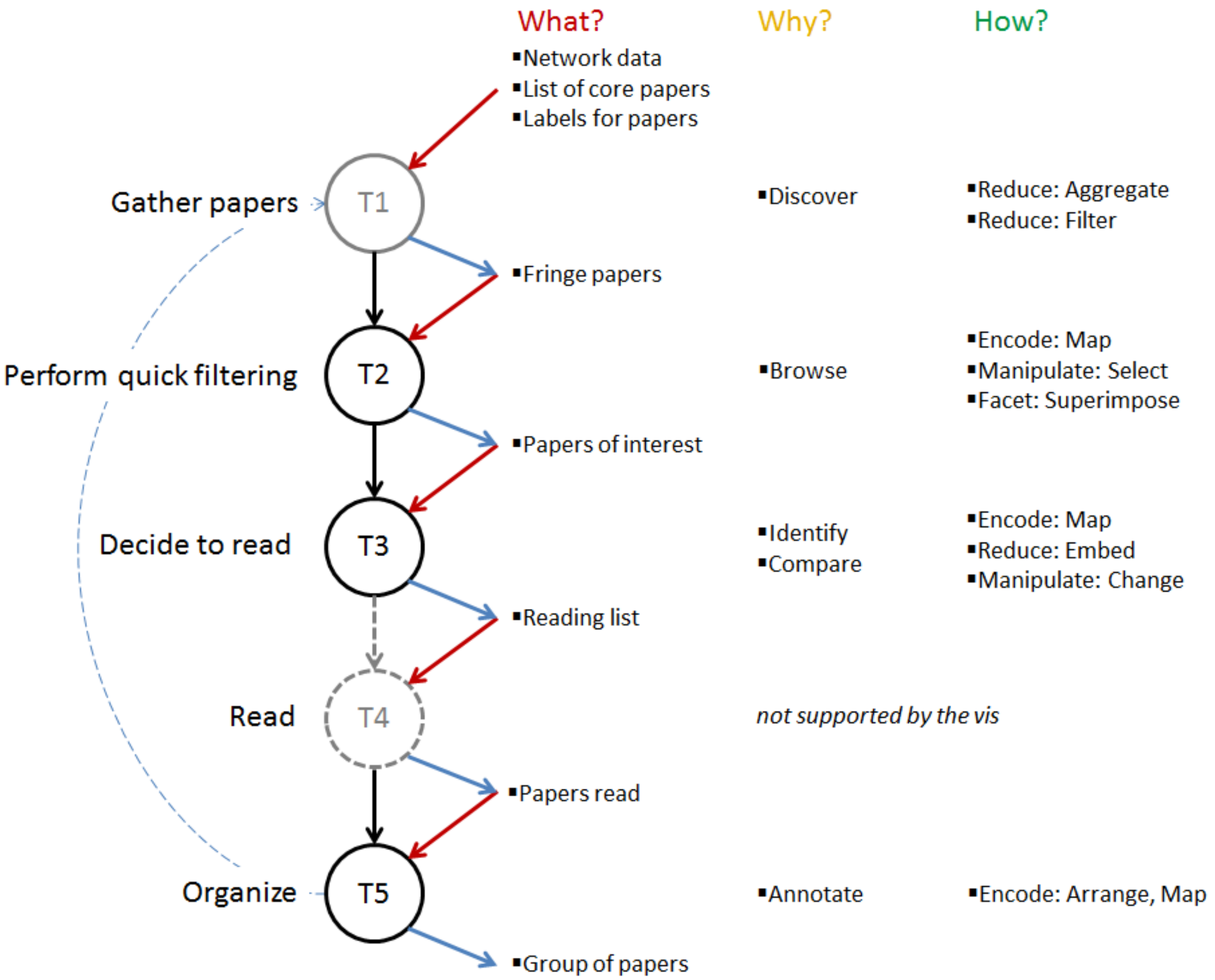


Figure 1 - Task description. T1: the algorithm aggregates papers metadata and relationships to compute a relevance score, and filters paper in the Fringe based on this score. T2: encode citation count with size, and superimpose links between papers on demand. T3: embed abstract and metadata in the paper list view, and let users reorder it. T5: classify papers into different subtopics. The dashed blue arrow shows that the literature review is iterative; we could have added similar arrows from any task to any previous task.

Proposed infovis solution

Mental model

We consider the process of a literature review as exploring the space of previously published papers, which can be divided into different subspaces:

- The Core: papers you have read, upon which you build your understanding of the field;
- The Fringe: papers you have access to, because they referenced 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 you make progress in the literature review, some papers from the Fringe will be moved to the Core, which will cause new papers to enter the Fringe. However, most of the papers will remain forever in the Unknown.

If we define the Fringe as the papers that are one hop away from a paper in the Core based on the citations network, then we face the problem of exponential explosion described in the Tasks section: each paper references and can be cited by many papers. Yet, most of these papers are actually irrelevant to your particular domain of interest. We therefore propose to order the Fringe based on a *relevance score* computed for each paper, which takes into account how many times this paper has referenced or been cited by other papers that you have expressed interest in, as well as the number of citations of this paper. See algorithm details below.

To support the multi-stage filtering process described earlier, we add another subspace: the Reading List, consisting of the papers that you have expressed interest in. It serves as a temporary space for papers that you have selected from the Fringe, but have not read yet. This space has two purposes: reading papers' abstracts and metadata, comparing them to decide which to put on the "To Read" list; organizing the "To Read" list by grouping related papers, and prioritizing it.

Algorithm

For each paper P that is one hop away of the papers that you have marked as read or to read, we compute a relevance score based on:

- the Adjusted Citation Count, or ACC, which is the total number of citations of P divided by an increasing function of the number of years since P was published;
- the count of all the papers in the Core or the To Read list that cite or are referenced by P, weighted by:
    - the level of interest that the user has expressed for these papers (star, read, to read);
    - the Adjusted Citation Count of these papers.

This algorithm has many parameters that will have to be adjusted. We plan to do so by trial and error, observing the output of the algorithm for a set of papers that we are familiar with. A better solution would be to teach the algorithm which suggestions are good by deploying it in a real setting, then using machine learning to get the optimal parameters. For the ACC, an appropriate function of the number of years since publication could probably be found in the literature. We can think of a simple linear function, or maybe the square root.

Following the exact same procedure, we could compute a relevance score in the network of papers linked by common authors. But this would add another free parameter: how does the citation score compares to the author score? The algorithm could also take into account clusters of author keywords, a notion of "similar paper" such as the one computed on Google Scholar, and even Amazon's metaphor of "other researchers who read this paper also read…" However, these further refinements appear beyond the scope of this project.

Main View

Our proposed visualization uses the spatial metaphor of concentric circles to explicitly represent these four different subspaces of the paper space. Papers farther away from the Core are considered less relevant. To maximise readability, the concentric circles have a very large radius, so that paper titles can be displayed approximately as a vertical list. Given the top-down reading order in many cultures, the algorithm that populates the Fringe displays the items with the highest relevance score at the top.

We encode each paper as a node in a node-link diagram, using by default a circle mark. However, we show the links explicitly only when necessary, because they can cause a lot of visual clutter. Moreover, the tasks that we intend to support do not generally involve understanding the exact topology of the network. Papers in the Fringe are represented as a white disk with a colored outline; papers marked as "to read" are displayed as "donuts", white disks half-filled with color; and papers that are read are fully colored disks. Users can mark items as particularly noteworthy by *starring* them; these papers are afterwards displayed using a star instead of a disk (see Figure 2).

The relevance algorithm is a key component of the proposed system, and we expect that a significant portion of the benefits of using our visualization will come from its ability to find and sort papers based on *all* the papers that you consider relevant, and not only one - as is today the case with backward and forward search. However, we do not want the visualization to simply display the results of the prediction algorithm; otherwise it would be of little value. We think the visualization should support two other purposes at this stage:

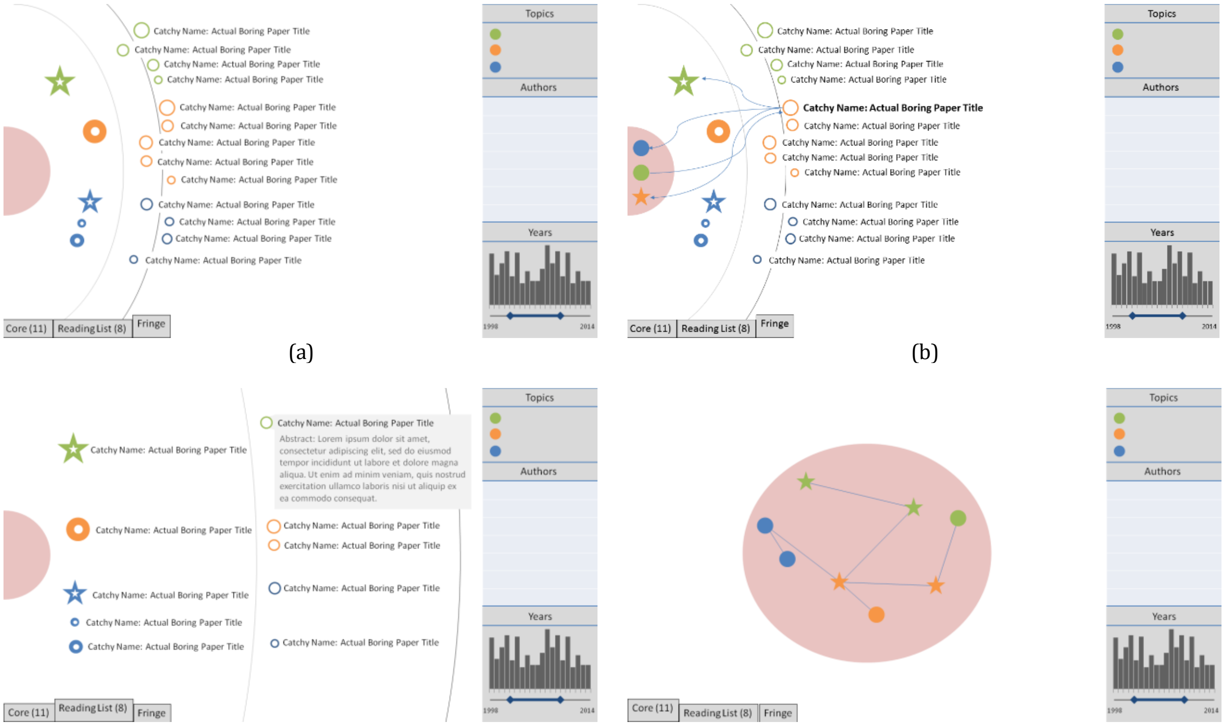
a. visually explain the *rationale* of the algorithm for promoting a paper;
b. let users make choices based on criteria that the algorithm <u>cannot</u> know, such as "I feel like reading papers about X now".

Because the algorithm takes into account the Adjusted Citation Count, we encode this number for each paper by varying the *size* of the mark. Since we are mostly interested in relative differences, we could scale the ACC values so that paper marks never get too small or too big, for instance by imposing a minimum size and using a log function of the ACC. The other component of the relevance score is the weighted sum of links to other relevant papers, which we explain visually by drawing the citation links from this paper to all the other used in the computation. By following the links, users can quickly see which papers contributed to this paper being promoted.

Displaying the links on demand also enables the second purpose of the visualization: letting users make decisions based on fuzzy criteria. By selecting one or more papers from the Core or the Reading list, they can see which papers of the Reading list are linked to these, which would allow users to explore further one aspect of the domain. Such *topics* encountered and identified during the literature review could be represented across the visualization by different colors, which are well suited to distinguish categorical data. Users could assign papers to one or more topics, and spatially organize them. The relevance algorithm could be extended with a classifier that tries to predict the topic of new papers, and partition the Fringe accordingly. Although probably useful, this approach has already been explored by Chau et al. [2] so we plan to focus on other design aspects.

Additional Views

In addition to the main view described above, we plan to show two other facets of the data in side views: a list of the top contributing authors for the selected papers, and a histogram of the distribution of papers over time, with the selected papers appearing in a different color in the histogram. These two views have linked highlighting with the main view, and can also be used to filter the papers shown, either by selecting a subset of authors or a time period. A list of topics could be used both as color labels, and a way to highlight and filter the papers displayed by topic.


Topics
Authors
Years
1998
2014
Catchy Name: Actual Boring Paper Title
Core (11)
Reading List (8)
Fringe
Abstract: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

(a) (b) (c) (d)

Figure 2 – Proposed visualization. (a) The Fringe view serves as an overview for the user. Papers of the Fringe that have been selected are moved slightly to the left, to show that they are of interest to the user. (b) When elements are highlighted their relationships are shown. (c) The reading list view shows more detail about papers. (d) The core view can be used to annotate and organize papers previously read.

Scenario

Panagiota is a PhD student in the field of Information Visualization. She has just finished reading several papers for her literature review and now needs more material. To find new papers, she goes to the Fringe Reading List website, which shows her an overview of where she was at the end of her previous session (Figure 2a). She can quickly see how many papers she has read, the ones in her reading list and an outdated list of papers in her fringe of research. She marks as read the papers she has just finished reading, which sends them into the core region at the left of the interface, and tags one of them that she considered particularly important with a star (Figure 2d). She then goes through the fresh list of papers in her updated fringe, and clicks on the ones with promising titles, which moves them to the "under consideration" list to the left. If a paper title is not clear enough on its own to decide whether the paper might be relevant, she can hover on it to get more details (authors, conference, year), and to see the connections between this paper and other that she already know of (Figure 2b). After selecting about 10 different papers, she zooms into the reading list to drill down into the details of these papers (Figure 2c). After quickly reading their abstracts, she discards 4 of the papers right away, then highlights two that are similar to compare their metadata and abstracts, and discards one of them. She adds the remaining 5 papers to her reading list, and reorders the list to prioritize the papers she wants to read at the top. She then brews a cup of coffee, gathers her courage, and goes back to her readings.

Proposed Implementation

We want to make this tool useful in the real world. For that reason we think a platform that is easily accessible is the best approach. We will use Javascript and the d3.js library, both of which are widely supported in browsers today. These technologies provide the elements necessary to create both the visualization and a rich user experience, and we can leverage d3.js network and layout algorithms. We may use other web libraries if necessary, such as jQuery or Angular.js.

Milestones

There are 6 weeks left from from November 1 to December 12. Our plan is as follows:

Week 1: Parse dataset as D3 data objects; basic node-link diagram in D3
Week 2: [no class] Retrieve citation count from Google Scholar; refine design and algorithm; start implementation
Week 3: Basic implementation in D3: different regions, papers encoded as circles or stars, simple links, algorithm
Week 4: Adjust the weights of the algorithm, add interaction
Weed 5: Add additional views if not done yet; improve rendering of the links (curves, bundling)
Week 6: Prepare presentation and report

Implementation of tagging and topics will depend on the advancement of the project, but will be given lower priority.

Related Work

Many visualizations have been created to analyze the scientific literature, in particular in the Infovis community. Borner et al. use node-link diagrams to represent papers and authors, linked by citations or coauthoring relationships [1]. The size of the nodes encodes the number of citations received by a paper or an author. However, they had to cut off papers that had less than 15 citations to avoid the hairball problem. A good solution to this problem was proposed in Citevis [8], where citation relationships are shown by coloring the nodes instead of drawing connection marks. Yet only the most cited papers of each year were shown. In contrast, the Citeology tool [6] displays three thousand CHI and UIST papers in a tiny font, stacked vertically for each year. "Parents" and "children" of a selected paper are highlighted and shown with smooth links. However, attempting to display the second or third generation usually results in an entangled set of links, especially for highly-cited papers.

Netlens [5] takes a different approach, showing aggregate data for *content* and *actors* – in our case, papers and authors. The system is highly interactive: detailed views are shown for each type of data, and selection can be passed back and forth between the two facets of the data. Jigsaw [9] is another example of a general purpose sensemaking tool that has been successfully used for this problem. More recently, Keyvis [4] aimed at visualizing trends in the Infovis literature through keyword co-occurrence analysis, displaying the results via node-link diagrams and trees.

Although many attempts have been made to visualize an entire branch of the literature, very few work addresses the task of doing a literature review. Apolo [2] is probably the closest one: 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. They build upon the concept of the sensemaking loop [7] to explicitly support the creation and reorganization of an external mental representation of the domain of interest, subdivided into several user-defined topics. Finally, the problem of browsing faceted data has been addressed often, from the influencial Flameno system [10] to the recent idea of PivotPaths [3], in which selected metadata is used to transition continuously from one facet of the dataset to another, preserving the context.

References

1. Borner, K. and Viswanath, L. Major Information Visualization Authors, Papers and Topics in the ACM Library. *IEEE Symposium on Information Visualization*, IEEE (2004), r1–r1.

2. Chau, D.H., Kittur, A., Hong, J.I., and Faloutsos, C. Apolo: Making Sense of Large Network Data by Combining Rich User Interaction and Machine Learning. *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*, ACM Press (2011), 167.

3. Dork, M., Riche, N.H., Ramos, G., and Dumais, S. PivotPaths: Strolling through Faceted Information Spaces. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2709–2718.

4. Isenberg, P., Isenberg, T., Sedlmair, M., Chen, J., and Möller, T. Toward a deeper understanding of Visualization through keyword analysis. (2014).

5. Kang, H., Plaisant, C., Lee, B., and Bederson, B.B. NetLens: iterative exploration of content-actor network data. *Information Visualization 6*, 1 (2007), 18–31.

6. Matejka, J., Grossman, T., and Fitzmaurice, G. Citeology: visualizing paper genealogy. *Proceedings of the 2012 ACM annual conference extended abstracts on Human Factors in Computing Systems Extended Abstracts - CHI EA '12*, ACM Press (2012), 181.

7. Russell, D.M., Stefik, M.J., Pirolli, P., and Card, S.K. The cost structure of sensemaking. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93*, ACM Press (1993), 269–276.

8. Stasko, J., Choo, J., Han, Y., and Hu, M. Citevis: Exploring conference paper citation data visually. *Proceedings of the IEEE Conference VIS 2013*, (2013), 2–3.

9. Stasko, J., Görg, C., and Spence, R. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization 7*, 2 (2008), 118–132.

10. Yee, K.-P., Swearingen, K., Li, K., and Hearst, M. Faceted metadata for image search and browsing. *Proceedings of the conference on Human factors in computing systems - CHI '03*, ACM Press (2003), 401.