

Visual Analysis and Dissemination of Scientific Literature Collections with SurVis

Fabian Beck, *Member, IEEE Computer Society*, Sebastian Koch, and
Daniel Weiskopf, *Member, IEEE Computer Society*

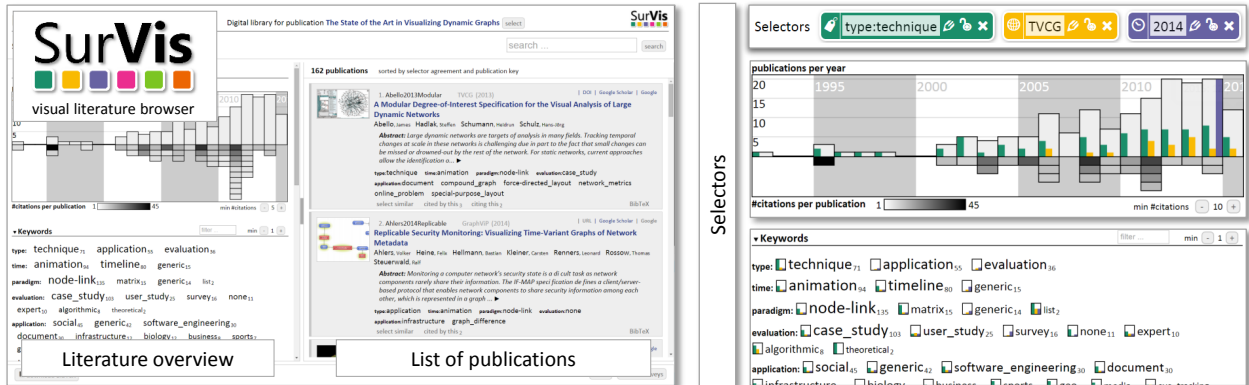


Fig. 1. User interface of SurVis, a Web-based visual analytics system to analyze and disseminate literature collections, consisting of a literature overview and a list of publications; versatile interactive selectors allow the user to query the literature collection; the system visualizes the selector agreement in a timeline and using word-sized sparkline visualizations embedded in word clouds.

Abstract—Bibliographic data such as collections of scientific articles and citation networks have been studied extensively in information visualization and visual analytics research. Powerful systems have been built to support various types of bibliographic analysis, but they require some training and cannot be used to disseminate the insights gained. In contrast, we focused on developing a more accessible visual analytics system, called SurVis, that is ready to disseminate a carefully surveyed literature collection. The authors of a survey may use our Web-based system to structure and analyze their literature database. Later, readers of the survey can obtain an overview, quickly retrieve specific publications, and reproduce or extend the original bibliographic analysis. Our system employs a set of selectors that enable users to filter and browse the literature collection as well as to control interactive visualizations. The versatile selector concept includes selectors for textual search, filtering by keywords and meta-information, selection and clustering of similar publications, and following citation links. Agreement to the selector is represented by word-sized sparkline visualizations seamlessly integrated into the user interface. Based on an analysis of the analytical reasoning process, we derived requirements for the system. We developed the system in a formative way involving other researchers writing literature surveys. A questionnaire study with 14 visual analytics experts confirms that SurVis meets the initially formulated requirements.

Index Terms—Visual analytics of documents, bibliographic data, dissemination, literature browser

1 INTRODUCTION

Research builds upon previous research and cites previous publications to document these relationships. These references are particularly important for publications that survey a field of research, such as state-of-the-art reports and research-oriented textbooks. While the referenced publications are structured and explained in the text, the list of references at the end of the publication is only added to allow the reader to retrieve the referenced literature from an analog or digital library. However, the hundred and more references usually contained in a literature survey can hardly be analyzed by just reading through the list.

We argue that the list of references, which had been carefully collected and curated by the authors of the survey, would be an even more valuable source of information if presented in an interactively ex-

plorably way and augmented with additional data, so that: (i) publications that address similar research challenges can be retrieved, (ii) historical developments and recent progress in the research area can be analyzed, and (iii) experts in the field and highly influential publications can be identified. Digital libraries might contain all referenced publications and are interactively explorable, but do not allow one to restrict the analysis to a selected list of references; adding extra information such as assigning consistent, domain-specific keywords is not possible for the authors of the survey.

For better analyzing and disseminating literature collections, we developed SurVis, a Web-based interactive system that allows the authors of a survey to consistently structure their list of references. Visualizations and knowledge discovery methods support the analytical reasoning process by efficiently retrieving and semantically abstracting the information. In contrast to many other visual analytics approaches, dissemination—being one of the core goals postulated in the visual analytics roadmap edited by Thomas and Cook [33]—is a specific focus of the system: it not only serves as an analysis technique for the authors of a survey, but at a same time as a foraging and sensemaking tool for the readers. The user interface is kept simple to maximize the accessibility of data, and hence, the outreach of the literature survey.

This paper presents SurVis as a system that integrates easy-to-understand visualizations and representations of references with a

- Fabian Beck, Sebastian Koch, and Daniel Weiskopf are with VISUS, University of Stuttgart, Germany. E-mail: {fabian.beck, sebastian.koch, weiskopf}@visus.uni-stuttgart.de.

Manuscript received 31 Mar. 2015; accepted 1 Aug. 2015; date of publication xx Aug. 2015; date of current version 25 Oct. 2015.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

powerful selector interaction concept (Fig. 1). The visualization of selector agreement provides a novel approach of using sparklines (i.e., word-sized visualizations) to analyze document collections. SurVis combines these visualizations in an interactive Web-based visual analytics system that is complemented by automatic analysis via clustering. In Section 2, we discuss differences to previous visual analytics systems for literature data and related visualization approaches. Section 3 elicits specific requirements for supporting the analysis of literature data for both authors and readers of a literature survey and identifies open issues in existing systems. We describe the design and implementation of SurVis in Section 4: in particular, we present all elements of the user interface, outline the selector interaction concept, and introduce the visualization approach of selector agreement. In a questionnaire study, we collected feedback from visualization and interaction experts who assessed the system; Section 5 reports findings of this evaluation, which confirms that SurVis meets the intended requirements.

A reference literature collection based on the latest SurVis version is available online¹ and will be used as an example throughout this paper. The reference collection presents the research literature of a state-of-the-art report on dynamic graph visualization [4]. SurVis, as a system, is published under an open-source license on GitHub.²

2 RELATED WORK

Starting with Treevis.net [30] and Timeviz.net [1], literature browsers have become popular tools within the visualization community to disseminate literature collections of survey publications. Inspired by earlier visualization collections like visualcomplexity.com [23], they present visualization techniques as a thumbnail image grid. Icons at the side enable the user to quickly filter the techniques based on their main characteristics (e.g., 2D or 3D). Similar systems are used in various other visualization literature surveys—we maintain an up-to-date list online.³ SurVis was inspired by these browsers but goes beyond them as it also supports a deeper visual analysis of the literature collections, for instance, revealing historic developments and relationships.

Faceted search approaches filter data collections based on *facets* representing different filtering dimensions. Visual literature browsers like the ones described above use variants of faceted search. Similar approaches are part of interactive visualization tools for literature data. For instance, PivotPaths [13] combines facet-like filters with a node-link representation of authors, publications, and keywords; filters modify the graph representation in smoothly animated transitions. Keshif [36] enriches faceted search with visualizations regarding keyword frequencies and temporal context in a highly interactive, versatile document browser. By contrast, SurVis uses selectors that do not narrow down the data immediately but only prioritize it, unless the user decides to apply a selector as a filter. This extended concept allows integrating fuzzy selection criteria and visualizations to analyze the relationship between selections.

Being more complex than lightweight literature browsers and faceted search systems, there are rich visualization and analysis systems to support the review of scientific literature: CiteWiz [15] integrates a citation network representation with a timeline showing citations per author and graphs for keyword co-occurrence and co-authorship. PaperCube [8] combines different citation representations based on graph, hierarchy, and timeline visualizations into an analysis framework. The Action Science Explorer [14] connects a list of references with a citation network visualization and facilitates the exploration of specific citation contexts within the texts of the publications. VisIRR [11] recommends publications based on initially selected ones and represents the publications in a scatter plot view revealing clusters. Rexplore [26] includes, among other analysis tools, a visualization of the evolution of topics in a literature collection and a graph view with faceted filters for analyzing relationships between authors. PivotSlice [37] uses a faceted approach with two sets of user-selected

filters on the axes of a two-dimensional scatter plot blending in citation links on demand. CiteRivers [17] represents the evolution of clusters of topics in a streamgraph and combine it with, among other plots, a timeline-based node-link citation visualization. Similar approaches exist for general document collections [32] or specialized to other types of documents, for instance, patent data [19, 20]. These analysis tools are powerful but usually lack the simplicity and flat learning curve to use them as tools to disseminate literature collections to a wide audience.

SurVis uses word-sized visualizations, also called *sparklines* [34], within the user interface to indicate relationships between entities, selectors, and publications. While sparklines were initially suggested as enrichments of texts and tables, they also find application in augmenting user interfaces, for instance, in integrated development environments (IDEs) in the context of software engineering [5, 6, 7]. In particular, I3 [5], an IDE extension for local code search, is related to SurVis because similarity to search results is presented as part of a sparkline. Also, sparklines have been integrated into visualization techniques, for instance, into adjacency matrices to represent dynamic graphs [10] or word clouds to represent temporal occurrences of terms [22]. In the context of document retrieval and search, the TileBars approach [16, 28] uses word-sized graphics to show term distributions within result documents. However, to the best of our knowledge, sparklines have not been used to indicate groups in document collections yet.

Among other visualization techniques, text visualization approaches [21] are related as they visualize documents as well, however, with the focus on the actual textual content, but to a lesser degree on meta-data of the documents. Visualizing documents with assigned keywords can also be considered as a set visualization problem [2], each keyword grouping together a set of documents. SurVis provides features to visualize set containment as well as to investigate set overlap, which is an additional benefit of our new interactive selector concept. Some network visualization techniques focus on the visualization of citations [12, 25, 31]. In SurVis, citation networks are integrated indirectly by providing selectors that identify all publications cited by or citing a specific publication. The system also shares some similarities with different kinds of digital libraries like literature search engines, publisher libraries, or literature managers. These are very useful tools but only provide limited support for visually analyzing the literature data. We point out further differences as part of a comparison of systems in Section 3.4.

3 REQUIREMENT ANALYSIS

Our first goal is to formulate requirements for a system to visually analyze and disseminate literature collections of survey articles. To this end, we analyze usage scenarios and the analytical reasoning process that should be supported. This analysis allows us to derive functional requirements for the system. Assessing existing systems against these finally reveals a need for developing a new system.

3.1 Usage Scenarios

We do not just support the analysis of literature data but the dissemination of the gained results as well. These results comprise a collection of literature data that has been structured as part of the analysis process. They are usually published together with a written literature survey. The goal is not to replace the report, but to augment it by providing means to make bibliographic analyses reproducible and extensible for the reader. As a consequence, our system has different types of users: On the one hand, the authors of the literature survey use it to collect, structure, and analyze the literature; we call this user type *Curator*. On the other hand, the published collection can be used by the readers of the survey. Since the intentions of people belonging to the user type *Reader* may vary significantly, we distinguish two sub-types: the *Researcher* is interested in a deep analysis of (parts of) the presented literature, whereas the *Practitioner* wants to find relevant information quickly to use it in practical application. Please note that these user types only refer to prototypical users and can be considered as *personas* of a usage analysis. In particular, real readers would often

¹<http://dynamicgraphs.fbeck.com>

²<https://github.com/fabian-beck/survis>

³http://dynamicgraphs.fbeck.com/related_surveys_dgv.html

Table 1. The analytical reasoning process according to Pirolli and Card [27] (left column) mapped to the problem of analyzing and disseminating a literature collection (middle and right column). Two linked processes are considered: in a first process, the *Curator* collects, analyzes, and publishes the literature survey, which is then filtered, sub-structured, and comprehended by the *Reader* in a subsequent process. Steps explicitly supported by SurVis are highlighted in dark gray, partially supported ones in light gray.

	<i>Curator</i>	<i>Reader</i>
Foraging loop:		
(i) External data source	Literature body	→ Literature survey
(ii) Shoebox	Relevant publications	Relevant publications
(iii) Evidence file	Keywords	Selection
Sensemaking loop:		
(iv) Schema	Structure	Substructure
(v) Hypotheses	Taxonomy	Comprehension
(vi) Presentation	Literature survey	Application

show mixed characteristics between *Researcher* and *Practitioner*, for instance, researchers from a related discipline who are interested in applying one of the surveyed techniques for their research.

These considerations already frame some general requirements of the system. First, the goal of augmenting literature surveys implies to support collections of typically 50 to 300 publications. Second, disseminating the results with the system requires a modern Web-based solution. A major challenge is the support of the targeted wide spectrum of user types. For the *Curator* and *Researcher*, the system needs to be powerful enough to gain relevant insights. However, at the same time, the solution has to be easy to use and self-explaining because the *Practitioner* should be able to quickly retrieve desired information.

3.2 Analytical Reasoning Process

We consider the steps to collect, structure, and disseminate a literature collection as an analytical reasoning and sensemaking process. Pirolli and Card [27] presented a model that breaks down a process like this into two loops: The foraging loop starts with an *external data source* from which relevant entities are retrieved and filtered to a smaller collection, called *shoebox*. Extracting information from these sources leads to an *evidence file*. In the sensemaking loop, the information is structured to build a *schema* that forms the basis for formulating *hypotheses*. The process is concluded by the *presentation* of the gained insights. As the process is described as entangled loops, it is usually not followed in a linear way but iterated in cycles and subcycles.

Applied to the above usage scenarios, we again distinguish between *Curator* and *Reader*: their reasoning processes are different as they start with different data sources (the scientific literature in general vs. the curated literature collection) and end in different targets (the curated literature collection vs. some application). The outcome of the reasoning process of the *Curator*—the literature survey—is the input of the process of the *Reader*. Table 1 describes this relationship as a pair of connected reasoning processes mapping the model of Pirolli and Card [27] to our application:

1. The *Curator* starts with foraging relevant publications from the body of scientific literature, then extracting keywords and structuring these as a start of the sensemaking process; finally, a taxonomy can be derived from the structured literature, which is presented as part of the literature survey.
2. The *Reader* tries to find relevant publications for a given context in the literature survey; extracting information of these publications results in a more specific selection; developing a substructure for this selection is the starting point of the sensemaking loop, which leads to a comprehension of the selected research and concludes in the application of the gained insights.

The goal of SurVis is to support the reasoning processes of both user types, in particular the sensemaking loop of the *Curator* and the for-

aging loop of the *Reader*. The foraging loop of the *Curator* can only be partly supported because a full support would require us to build a database of all potentially relevant research literature, which is beyond the scope of this paper. Likewise, the sensemaking loop of the *Reader* is only partly addressed because potential insights regarding the application of the surveyed research are too diverse to be covered.

3.3 Functional Requirements

The analysis of usage scenarios, user types, and analytical reasoning processes provides a basis to derive functionality requirements (RQ) for the system. An obvious first requirement is the presentation of the collected literature as an enhanced list of references.

RQ 1: Publications

- (a) The system presents all publications of the literature collection in a list.
- (b) Each entry summarizes the publication record.
- (c) The full reference is available on demand.
- (d) An optional image provides a recognizable visual representation of the publication.
- (e) Publications are retrievable through textual search.

Keywords structuring the publications are the basis for the sensemaking process of the *Curator* and the retrieval and selection of publications for the *Reader*. The representation and interactive exploration of keywords structuring the publications are a second requirement.

RQ 2: Structure

- (a) The publications are structured with keywords.
- (b) Publications are retrievable by selecting keywords.
- (c) The system presents keyword frequencies of selected publications.
- (d) Other meta-information that helps structure the publications, like author or publication series, is treated similarly as keywords.

The *Curator* needs to understand historic developments within the research area to derive a sustainable taxonomy of the surveyed research and to understand the progress made. These considerations suggest requirements regarding the temporal evolution of the field.

RQ 3: History

- (a) The system gives an overview of the historical development of the literature surveyed.
- (b) Publications are selectable based on publication year.
- (c) Publication numbers regarding a selection of publications are presented as a temporal development.
- (d) Frequently cited paper are shown in this historic context.

Finally, the system should support the sensemaking processes of *Curator* and *Reader* by revealing relationships between publications. This guides (sub)structuring the data and the development of a taxonomy or comparison of the surveyed approaches.

RQ 4: Relationship

- (a) Publications related to selected publications can be retrieved.
- (b) Publications are automatically clustered into meaningful groups.
- (c) Citation links are retrievable in both directions: publications that cite a specific paper and those cited by it.

3.4 Assessment of Related Systems

As discussed in Section 2, there are various systems and approaches available to browse scientific literature collections. Before introducing SurVis, we assess these existing systems and check whether they already meet the above requirements. For each requirement and system, Table 2 assigns a rating estimating the fulfillment of the requirement. We only included Web-based systems because dissemination

Table 2. Assessment of related systems against the requirements elicited based on a four-staged scale of requirement fulfillment: *no* ○○○, *partly* ●○○, *largely* ●●○, and *yes* ●●●.

Name	RQ1 Publications	RQ2 Structure	RQ3 History	RQ4 Relationship
Literature browsers:				
Treevis.net [30]	●●○	●●●	○○○	○○○
PivotPaths [13]	●○○	●●●	○○○	●○○
Keshif [36]	●●○	●●●	●○○	○○○
Digital libraries:				
Google Scholar	●●○	○○○	●○○	●●○
GoPubMed	●●○	●●○	●○○	●○○
ACM Digital Library	●●○	●○○	○○○	●○○
DBLP	●○○	○○○	●○○	○○○
CiteULike	●●○	●●●	○○○	○○○

is a central requirement, and group them into *literature browsers* and *digital libraries*. Due to the large number of available systems, the list cannot be complete, but we selected representatives to cover a good variety of popular systems, for instance, Treevis.net as a visualization literature browser, Google Scholar and GoPubMed as literature search engines, ACM Digital Library as a publisher library, DBLP as a publisher-independent library, and CiteULike as a Web-based literature manager; PivotPaths and Keshif are added as unique solutions. Please note that this is not a general evaluation of the systems, but a comparison restricted to the above requirements—other features that the systems support are not discussed. The ratings listed in Table 2 are subjective as they only reflect the opinion of the authors of this paper.

Regarding the representation of publications, nearly all systems largely fulfill requirement RQ1: they present a searchable list of publications with important meta-data. None of the systems, however, fully supports RQ1 because either a representative image cannot be added (digital libraries) or only the image is shown and important meta-information is only retrievable on demand (literature browsers); adding these missing features respectively, however, would be straightforward. Literature browsers provide the features to structure a literature collection by keywords and use them for interactive exploration (RQ2); digital libraries (with the exception of CiteULike) use predefined keywords or do not show keywords at all; GoPubMed automatically mines the concepts to be used as keywords through automatic semantic retrieval techniques. Only few of the assessed systems provide an overview of temporal evolution; among them, only Keshif meets RQ3 largely by providing an interactive timeline representation, but not showing citation information in temporal context. The support for retrieving relationship information (RQ4) is comparably sparse among all systems: only Google Scholar provides substantial information on citations and semantically similar publications.

In total, none of the systems meets all requirements. Our assessment identifies a particular lack of systems that would provide historic overview and reveal relationships among publications. Keshif comes closest to fulfilling the criteria; however, as a tool not specialized to publication data, it has a clear shortage supporting the retrieval of relationships among publications (RQ4).

4 THE SURVIS SYSTEM

We developed SurVis as a Web-based system to fulfill the above requirements. General design objectives that guided the development process were the following.

1. **Usability over Power:** Since the system should be usable by *Readers* of a survey article without much explanation and training, we tried to design the system as simple as possible; in doubt, we favored usability over adding more features.
2. **Curation over Automatism:** Survey authors usually invest great effort in structuring a literature collection; we assume that approaches extracting information automatically cannot compete with this manual curation regarding data quality.

3. **Everything is Selectable:** To not limit users of the system in exploring the data, every presented piece of information should be interactively selectable to filter the data and find related entities.

These design objectives are reflected in the user interface, visualizations, and interaction concept of SurVis, which is presented in the following sub-sections. They describe step by step the individual parts of the user interface shown in Fig. 2 and link individual features to the formulated requirements. Mainly addressing RQ1, a list of publications on the right side of the screen makes the individual publications explorable. Word clouds on the left provide a structure of the data according to RQ2. A timeline above the word clouds gives an overview of the history of the literature collection, meeting RQ3. A versatile interaction concept based on selectors and the visualization of agreement to currently active selectors shows relationships between individual publications, implementing mostly RQ4.

4.1 Literature Data

Before discussing the representation of data in the user interface, we first introduce the literature data on which SurVis is based. Obviously, the main data is represented by the individual publication records that comprise the literature collection. Keywords structure the publications and are themselves structured in keyword categories. Although technically optional, SurVis relies on these keywords to make the literature easily explorable. Moreover, citation data further increases the analytical power of SurVis. Technical details of data extraction are discussed as part of Section 4.8.

Publication Records: SurVis only requires standard publication records as used in typical scientific publications, which usually include title, authors, year, publication type, publication series name, and publisher. However, we recommend also including abstracts to allow the user to extract parts of the content of a publication without leaving the system and to improve the data for a reliable textual search. Moreover, a DOI (Digital Object Identifier) or Web link allows fast access of the digital version of a publication. For representing publication records, SurVis builds on the BibTeX format, a literature data format widely used in computer science.

Keywords: Keywords assigned to each publication are a flexible way to structure the publications according to subfields, applied methodology, paper type, etc. Although the authors of a publication usually provide some keywords already, we assume that the *Curators* of the literature collection assign their own keywords. These should be much more consistent and structure the surveyed field better than reused keywords from external sources. All keywords are textual—we do not use icons like other literature browsers do because some keywords would be difficult to symbolize unambiguously as an icon and *Curators* would require more time to design or choose the icons.

Keyword Categories: Keywords might address very different aspects of a publication—one could describe a detail of an evaluation approach, another may classify a publication as a system paper. To allow the *Curator* of the collection to better discern different kinds of keywords, we introduce the concept of keyword categories: indicated by a prefix, a keyword is defined to be part of a specific category (*[category]:[keyword]*, e.g., *type:system*). For each category, SurVis assumes that each publication is assigned at least one keyword, otherwise SurVis automatically assigns a ? keyword for this category (e.g., *type:?*). The use of keyword categories is optional; categories can be freely defined by the *Curator*.

Citations: For exploring relationships between publications, citation links are a good indicator of semantic similarity. Moreover, many citations by other publications hint at a high impact of a publication. To use this information, reference links can be optionally added to each publication. These links are, however, restricted to other publications within the collection: first, it would be difficult to display external citation links without building a larger database of scientific literature; second, the collected literature usually assembles a cohesive set of publications—restricting the citations to this body of literature provides a meaningful estimate of the impact of a publication within that body.

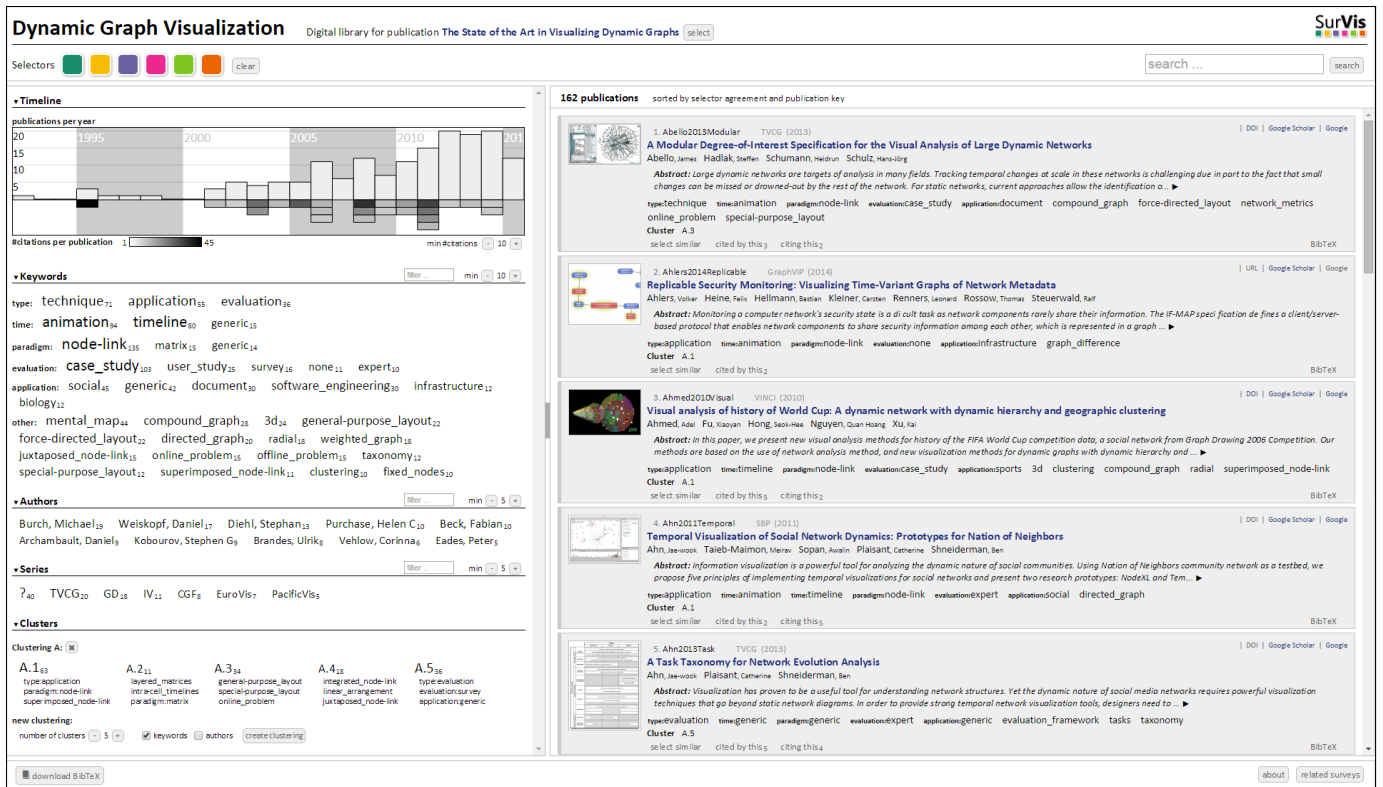
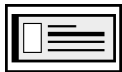


Fig. 2. SurVis user interface consisting of an overview of the literature collection on the left and a detailed list of references on the right.

4.2 List of Publications

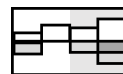
The list of publications is a central element of the user interface, filling its right side (Fig. 2; RQ 1a). While all publications of the collection are retrievable by scrolling through the list ordered by publication key, we assume that users will first select or prioritize the list based on the various selection features of SurVis described below. Like the results of a search engine, the list of publications is optimized for scanning through a small number of relevant entities providing just enough information for the user to judge whether the entity is of significant relevance.



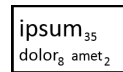
Publication: Each publication is displayed as a box containing structured text and an optional thumbnail image identifying the publication (RQ 1b, RQ 1d). The information presented most prominently is the title of the publication, followed below by the authors. The publication key as the first part of the header line is used to unambiguously identify the publication. Other meta-information is part of the header line of the box, showing publication series and year. The abstract is an important information source to estimate the relevance of a particular publication, but in full length would consume too much space. Hence, SurVis only shows the first sentences of the abstract, which can be expanded on demand. Below, all keywords of the publication are listed. The footer contains a button to open the full reference in BibTeX format (RQ 1c). Links in the top right corner of the publication box provide easy access to the digital version of the publication: they reference it directly as a PDF or DOI or indirectly by providing quick access to a search for the publication with Google Scholar and Google.

4.3 Literature Overview

The left part of the user interface provides a visual overview of the literature collection (Fig. 2). It consists of timeline and word cloud representations that can be collapsed and expanded on demand. Also, the full panel can be resized on demand to fit the visualizations to the available window space.



Timeline: The timeline displayed at the top provides a yearly overview of publications contained in the literature collection (RQ 3a). The bars at the top summarize the number of publications per year (RQ 3c). The stacked boxes below visualize the most cited publications within the literature collection. Each of the boxes represents an individual publication; the darkness of the background of the boxes encodes the citation frequency, as illustrated in a color scale legend (RQ 3d). To limit the space consumption of the diagram, a minimum number of citations required to show the respective boxes can be specified interactively by +/- buttons (the *Curator* is responsible for setting reasonable default values for this minimum frequency).



Word Clouds: We chose the concept of word clouds to summarize the keywords of a set of publications (RQ 2a): they are easy to understand, provide a good overview, and do not consume much space. The word clouds shown below the timeline represent not only keywords but also other meta-information such as authors and publication series, each type of meta-information in a different word cloud (RQ 2d). They summarize all publications currently shown in the publication list. As common in word clouds, the font size represents the frequency of each term; the value is also provided as a subscript number attached to the term (RQ 2c). When the list of terms becomes too long or should be focused to the most relevant terms, the minimum frequency required for a term to be included in the cloud can be changed using +/- buttons similar as for citations. To easily retrieve the most frequently occurring terms, the system sorts them in decreasing order of frequency. Furthermore, a small search field above the word cloud allows one to quickly filter the terms. Specific to the word cloud representation of keywords, SurVis separates the keyword categories into different sub-clouds. The *Curator* might define the meaning of keywords and categories in short descriptions, which are shown in tooltip dialogs when hovering over the respective terms with the mouse.

Clusters: In addition to structures provided through keywords and meta-information, SurVis includes an interactive clustering approach to group the currently selected publications (RQ 4b). It is based on the *k-means* clustering algorithm. The user specifies the desired number of clusters k and chooses between keywords and authors (or both) as terms that serve as input for the clustering process. A feature vector is generated based on these terms for every publication. The system shows the resulting clusters similar to a word cloud: the font size of the cluster is scaled according to the number of publications contained in it, a subscript number provides the precise value. The cluster name, however, is generic—characteristic terms need to be listed to give the cluster a meaningful label. Such terms are determined by computing the *tf-idf* measure (*term frequency–inverse document frequency*) [3] treating the clusters as documents of a collection. The three terms having the highest value are listed below the cluster name; further characteristic terms with high values can be retrieved on demand as part of a tooltip dialog for each cluster. Since SurVis supports the creation of different clusterings, the cluster naming scheme consists of two parts: first, a letter A–Z identifying the clustering, and second, a number representing the individual cluster within the clustering (e.g., B.7). Clusters are also indicated as additional keywords in the representation of the individual publications on the right.

4.4 Selectors

While the list of publications allows the user to retrieve details for any publication, the collection overview provides a visual summary of the publications. What is missing so far is a mechanism to interactively connect the two representations. We introduce the concept of selectors to provide this interactive link. Each selector defines a criterion to rank and filter the lists of the publication on the right. Selector criteria can be defined very flexibly, for instance, by a textual search query, a specific keyword, or a publication year. This concept is similar to faceted search, where different facets enable the user to filter the data. Our concept, however, extends this concept from filtering (i.e., an entity needs to meet every criterion) to a ranking approach (i.e., entities are ordered according to the extent they meet the sum of all criteria). We favor this relaxed ranking over strict filtering because filtering would too quickly narrow down the number of publications and not consider selection inaccuracies and fuzzy queries.

In this model, a selector s applied to a publication p of the literature collection provides an agreement value $\sigma_s(p) \in [0, 1]$, where 0 is assigned if the selector and publication do not match and 1 encodes a perfect agreement with the selector criterion. The combined selector agreement of multiple selectors $S = \{s_1, s_2, \dots, s_k\}$ is defined as the sum of the agreement values $\sigma_S(p) = \sum_{i=1}^k \sigma_{s_i}(p)$.

Selectors are set by clicking on entities such as keywords, author names, years in the timeline, etc., or using special inputs such as a search field. As illustrated below, the colored boxes at the top of the user interface (cf. Fig. 2) are used to represent the selectors:



An icon on the left identifies the type of the selector while a text field in the middle represents the individual search query. A pencil icon on the right ✎ allows negating the selector; the ranking criterion of the negated selector is defined as $\bar{\sigma}_s(p) = 1 - \sigma_s(p)$. Further, clicking the open lock icon 🔓, the selector is applied as a filter indicated by a closed lock 🔒: the list of publications is reduced to those at least partly matching the selector, i.e., $\{p | \sigma_s(p) > 0\}$. Finally, a single selector can be discarded using the cross icon ✕, or all selectors by pressing the *clear* button.

4.5 Selector Types

SurVis includes a wide spectrum of selector types that can be combined to versatile queries. Table 3 provides an overview of these selectors types, which we classify in three categories.

Default: The default group consists only of the 🔍 Search selector, which is based on a standard textual search (RQ 1e). The user specifies a search query in the text field in the upper right corner of the user

Table 3. Available selectors for sorting and filtering the list of entries.

Group	Name	UI elements	Matching criterion
Default	Search	Search bar	If search terms are contained in any of the fields
Meta	Keyword	Keyword cloud, list of keywords	If keyword is contained in list of keywords
	Author	Author cloud, list of authors	If author is contained in list of authors
	Series	Series cloud, publication series	If series matches publication series
	Cluster	Cluster overview, publication cluster	If cluster is contained in assigned publication clusters
	Time	Timeline, publication year	If year matches publication year
Relationship	Similarity	Button <i>select similar</i>	If similar terms used in title or abstract and similar authors or keywords
	Citation	Buttons <i>citing/cited by this</i>	If cited by or citing the publication

interface (cf. Fig. 2) and presses *search* or hits *enter* to activate the selector. For computing the selector agreement, the query is split into terms and common English stopwords are removed. The agreement is defined as the ratio of terms that is found in any of the fields of a publication. Hence, it is possible, for instance, to search for terms in the title or abstract as well as for author names or DOIs.

Meta: Another group of selectors covers specific meta-information of the publications (RQ 2b). By clicking on a keyword, author, publication series, or cluster in a word cloud or in the publication list, a respective Keyword, Author, Series, or Cluster selector is created. Time selectors can be activated similarly by clicking on a specific year in the timeline or in the publication list (RQ 3b). The selectors are binary (i.e., $\sigma_s(p) \in \{0, 1\}$) and match if the activated entity is part of, or equal to, the respective field in the publication reference.

Relationship: Relationships between publications can be explored through selectors in two ways. First, each publication box has a button *select similar*, which activates a Similarity selector (RQ 4a): it is equivalent to a search using the title, abstract, authors, and keywords of the publication as search query. Second, publications related through citations can be selected by using either the *citing this* (incoming citations within the citation network) or *cited by this* (outgoing citations) button (RQ 4c): the Citation selector is binary and checks whether the selected publication is contained in the reference list of the other publication or vice versa respectively.

4.6 Selector Visualization

Ordering and filtering the list of publications is based on the selector agreement of each publication. Beyond this, the agreement value can be generalized to other entities. For instance, a keyword represents a set of publications $\{p_1, p_2, \dots, p_l\}$ having this keyword assigned; hence, we define the agreement of a keyword to a selector s as the average value of all agreement values of the contained publications:

$$\sigma_s(\{p_1, p_2, \dots, p_l\}) = \text{avg}(\{\sigma_s(p_1), \sigma_s(p_2), \dots, \sigma_s(p_l)\}) .$$

The same can be applied to agreement values for authors, series, clusters, years, and any other entities representing a set of publications. We are, however, not primarily interested in retrieving those individuals among the entities that best match the selectors. More insight could be gained when analyzing the agreement to individual selectors than the total agreement. This would allow us, for instance, to investigate what is the area of expertise of a selected author, what were popular topics in a selected year, or what is the relationship of a selected cluster to the different publication venues. The sparkline visualizations and other visual extensions of the user interface presented in the following were designed to support these kinds of analysis. The general approach is based on assigning a recognizable color to each selector.

Sparklines: Word-sized visualizations, known as *sparklines* [34], serve as the means to visualize the agreement of entities to individual selectors. Their small scale allows us to augment every entity presented in the user interface with a sparkline illustrated in Fig. 3. The individual sparkline is a miniature bar chart, where each bar represents the agreement to one of the selectors; the y-axis always shows the range of $[0, 1]$. The selector is identified by using the same colors

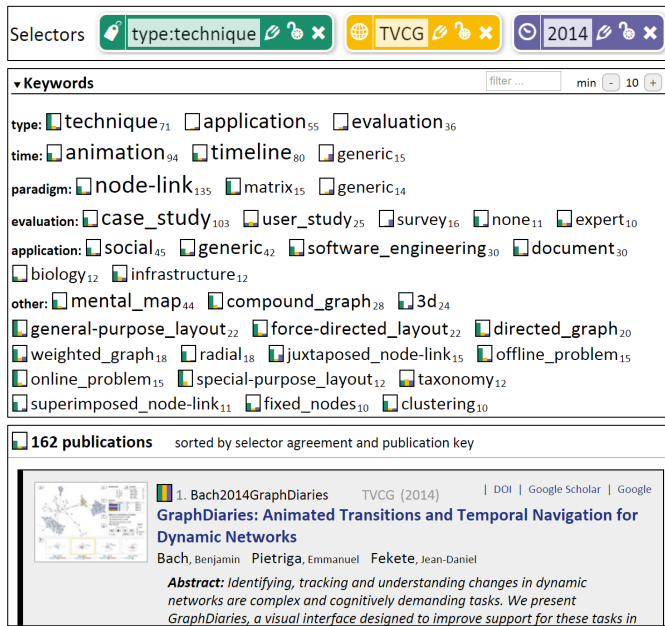


Fig. 3. User interface details showing the embedding of sparkline visualizations that encode selector agreement to currently activated selectors (top) into word clouds (middle) and the publication list (bottom), which is sorted by selector agreement (and publication key).

and order of selectors as in the selector list at the top of the user interface (cf. Fig. 2). For choosing the colors, we applied a qualitative (categorical) scale with six colors ■ ■ ■ ■ ■ ■ based on a scale created with ColorBrewer⁴; since the scale is used to colorize small areas in the sparklines, we slightly changed the brightness of some colors and color ordering to increase the luminance difference of neighboring colors. Like in traffic lights, the position of the bars redundantly encodes the color and particularly helps users with color vision deficiencies to work with the visualization. To leverage learning effects of the redundant color–position encoding, neither the order of the colors nor the colors themselves can be changed interactively. We limited the number of possible selectors to six because, first, according to our experience, even complex analyses only very rarely require the use of more than four selectors and, second, distinguishing more than six colors in the small sparklines would become difficult.

Enlarged Visualizations: The sparkline visualizations are often sufficient to investigate differences in the agreement of selectors and individual entities. More precise agreement values, however, could only be recognized in a larger representation. Enlarged diagrams also provide the space to add labels that better explain the diagram and help users with vision deficiencies in general. Since we cannot show such enlarged diagrams at the same time for all sparklines, we make them retrievable on demand in tooltip dialogs available for every sparkline. Figure 4 provides examples of tooltip dialogs belonging to a keyword and a cluster. Depending on the entity type, the tooltip contains further supportive information in textual or numeric form.

Enriched Timeline: The timeline provides the option to present the selectors in a historic context. Instead of adding sparklines, we directly enrich the bar and box representations already in use (Fig. 5). For the top part of the diagram, we subdivide the bar encoding the publication number of one year into colored bars, comparable to a stretched representation of the year as a sparkline. This shows the evolution of the selector agreement over time. Since the bottom part of the diagram specifically focuses on citations, we only add colored enrichments if a Citation selector is active: The box representing the selected publication is filled with the color of the selector. Moreover, depending on whether incoming or outgoing citations are selected, the right (incom-

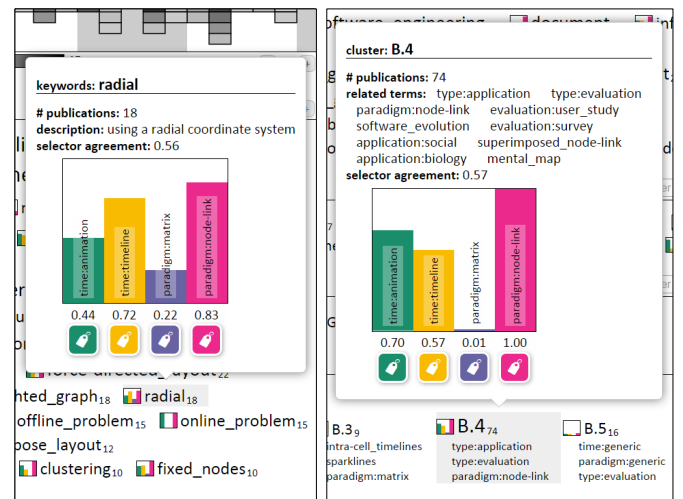


Fig. 4. Enlarged visualizations in tooltip dialogs for entities augmented with a sparkline, here, keyword *radial* (left) and cluster *B.4* (right).

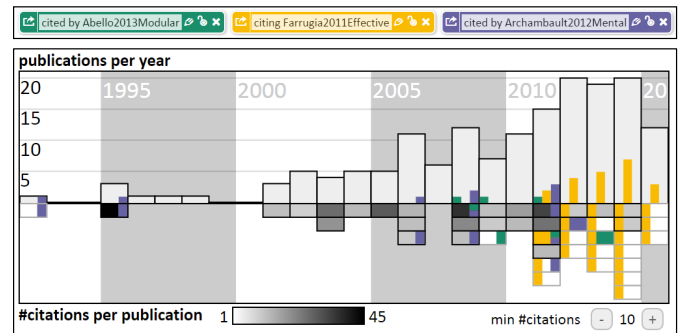


Fig. 5. Enriched timeline representation showing outgoing citations of two publications (green, purple/blue) and incoming citations of another publication (yellow) based on three Citation selectors.

ing) or left (outgoing) part of a box of a cited or citing publication is filled in the respective selector color. When multiple selected publications cite or are cited by a publication, multiple incoming or outgoing citations are depicted as stacked boxes. Publications that do not meet the minimum citation frequencies (i.e., are not represented as boxes by default) but cite or are cited by a selected publication are added to the diagram; a gray border line is used to discern them from boxes that are part of the diagram by default, having a black border.

These visualizations provide a powerful tool to analyze the literature data. They can be considered as set visualizations of publications and entities where selected entities like keywords, clusters, or citations define the sets. The sparklines relate these sets to all represented entities. For instance, Fig. 3 shows that, in 2014 (Time selector), many survey articles were published within the field (keyword *evaluation:survey*; 6 of 16 articles). Moreover, using multiple selectors, set overlap and similarity become analyzable: like Fig. 4 reveals, there is a high agreement of radial approaches (keyword *radial*) with timeline (yellow Keyword selector) and node-link techniques (pink Keyword selector), indicating an overlap of these sets in this context. Finally, the timeline provides support for retrieving temporal trends, for instance, in Fig. 5 (yellow bars, top part of the diagram), a growing impact of the work with the publication key *Farrugia2011Effective*.

4.7 Data Curation

To better support the information foraging and sensemaking process of the *Curator* of the literature collection (Table 1, left), SurVis also allows editing the data on the fly. Since modifications are usually only

⁴<http://colorbrewer2.com>

necessary for the *Curator*, there exists a specific edit mode that can be enabled, but would be usually disabled when disseminating the finalized collection. The edit mode changes the user interface of SurVis only slightly: First, the BibTeX text box that can be opened on demand for every publication becomes editable; this allows the *Curator* to change all meta-information of the publication. Moreover, it is possible to add a publication by pasting a BibTeX snippet. To assign keywords more quickly, additional keywords can be typed in a text field at the end of the keyword list of each publication. Keyword completions are offered based on the already used keywords. The changed data can be made persistent by downloading the library as a BibTeX file or saving it to the local storage of the browser. Of course, the changed data can only be updated globally with access to the server.

4.8 System Architecture and Implementation

SurVis is implemented in JavaScript on the client side. We kept the server side as simple as possible: from the perspective of the server, SurVis just consists of a single static HTML page and some JavaScript files containing all required code and data. On loading the page, all necessary information is already transferred to the client, with the exception of thumbnail images, which are only loaded on demand. Since no scripting language or database is required on the server, the system even runs on any desktop PC without the need to set up a local Web server. The literature data consists of different files in JSON format specifying the list of publications and description for tags and tag categories. For updating the data, the JSON files can be edited manually. To enable a quicker import of a library collection, we also developed a script that transforms a BibTeX file into the required JSON format. Options to configure SurVis are set in a properties file and include, among others, settings to enable optional features (e.g., citations, edit mode) and tailor the user interface to a specific literature collection (e.g., title, additional pages, customized styles).

4.9 Formative Development and Impact

SurVis was initially developed for analyzing and disseminating the literature of a state-of-the-art report on dynamic graph visualization [4], which is used as a sample literature collection throughout this paper. This direct application guided the development in a formative way and allowed us to test prototypes very early in a realistic setting. SurVis was also used for writing literature surveys on eye-tracking visualization [9] and group structure visualizations in graphs [35], which provided further valuable feedback. We also received comments on earlier versions of the system from other users. So far, the code of SurVis was distributed on request only. Already without significantly advertising the system, we received eleven individual requests from outside our institution and used SurVis in a number of internal projects. Currently, to our knowledge, six installations of SurVis databases are publicly available: in addition to the surveys on the visualization of dynamic graph [4] and group structures in graphs [35], these disseminate the literature collections of reviews on software performance visualization [18], visualization for software reuse [29], set visualization [2], and high-dimensional data visualization [24]; the links can be found on the project page of SurVis.⁵ Please note that these SurVis installations are maintained by different people and will be only gradually updated to new versions.

5 EXPERT FEEDBACK

In addition to the formative development, we evaluated the version of SurVis presented in this paper with a questionnaire study. We recruited visual analytics experts and asked for high-level feedback. In particular, we wanted to find out whether SurVis meets the requirements formulated in Section 3, how usable the system is, and which selector types are most important.

5.1 Study Design

The study consisted of a questionnaire and the online version of SurVis using the same sample literature collection as in this paper. To profit

from experience, we recruited participants who (i) are experts in visual analytics or related domains and (ii) preferably had used an earlier version of SurVis. 37 participants were invited personally by e-mail, 17 of them participated within about one week. They were asked to first explore the features of SurVis, then fill in the questionnaire. While answering the questions, they were also allowed (and reminded) to switch back to SurVis whenever necessary. We designed the study to take about 30 min.

Questionnaire: The questionnaire, an online form, consisted of 8 parts (the full questionnaire is part of the supplemental material). After an introduction (Part I), we asked to provide some background on domain expertise and experience (Part II). Then, Parts III–VI covered the four requirements described in Section 3: on a Likert-type scale from 1 (*strongly disagree*) to 5 (*strongly agree*), participants were asked whether they agree with a statement expressing the fulfillment of the requirement; below, they had the option to comment on positive and negative aspects. Part VII consisted of an analogous question regarding the usability of SurVis. Finally, Part VIII asked to rate the importance of each selector type available (cf. Table 3) on a scale from 1 (not important at all) to 5 (very important), providing the option to answer *don't know*. Below, text fields allowed commenting on missing features, a comparison to related tools, and general issues.

Participants: Based on a question asking to judge the own level of expertise in domains of information visualization, human-computer interaction, and usability (levels: *no knowledge*, *passing knowledge*, *knowledgeable*, *expert*), we focus on experts only: to be considered in the results, a participant needed to be *expert* in at least one of the domains or *knowledgeable* in at least two of the domains. 14 of the 17 participants passed this criterion. We assigned unique identifiers and refer to them as E1–E14 in the following. Among them, 11 stated to have used earlier versions of SurVis before, 8 as authors of a literature survey and 5 as readers of a survey (multiple selections possible). 9 of the experts already published work including an extensive literature survey (state-of-the-art report: 9, PhD thesis: 5; multiple selections possible). All experts were working in academia (PhD student: 9, postdoc: 4, professor: 1).

Validity and Limitations: The results of the questionnaire study are based on subjective opinions of the participants. Their expertise promises quality feedback on a high level of abstraction. However, knowing the background and authors of the system potentially biased them toward positive answers. Hence, absolute values of rating scales need to be interpreted cautiously. But as a potential bias would apply to all ratings, it is still valid to compare the numeric ratings to each other. The questionnaire does not ask the experts to solve specific tasks; some participants might not have seriously explored the system. However, most experts had reasonable experience using the system already. The number of participants is too small to interpret subtle numeric differences. Hence, the focus of analyzing the results needs to be on only substantial quantitative differences and on qualitative items (i.e., text answers). The experts are representatives of user types *Curator* and *Researcher*—user type *Practitioner* is not directly represented, but indirectly covered because it is safe to assume that the experts, based on their experience in designing visualizations and user interfaces, are able to understand the needs of *Practitioners* reasonably well.

5.2 Results

We discuss the results in order of the parts of the questionnaire. Due to limited space, we needed to condense the results. In particular, for text answers, we focus on only high-level comments that are mentioned usually by three experts at least—the raw text comments are part of the supplemental material.

Requirements: The specific statements that were asked to check whether the requirements are fulfilled are listed in Table 4 together with the agreement rating of the participants. Although potentially biased, a first indicator that the requirements are met to the satisfaction of the experts is that at least 10 of 14 experts agree with each of the statements (rating 4 or 5). The individual differences between the re-

⁵<https://github.com/fabian-beck/survis>

Table 4. Ratings of fulfilling requirements and usability on Likert-type scales from 1 (*strongly disagree*) to 5 (*strongly agree*).

RQ 1 (Publications)	RQ 2 (Structure)	RQ 3 (History)	RQ 4 (Relationship)	Usability
"SurVis supports users well in inspecting the individual publications contained in the list of publications (right part of the user interface)."	"SurVis structures and summarizes the publications well using keywords and other meta information."	"SurVis provides a good overview of the historical development of the surveyed publications."	"SurVis makes relationships between publications (e.g., similar keywords, citations) well-explorable."	"SurVis is easy to use and self-explaining."
avg: 4.1	avg: 3.9	avg: 4.3	avg: 3.9	avg: 3.8

quirements are only small, with RQ 3 (History) on the positive side (average rating: 4.3) and RQ 2 (Structure) as well as RQ 4 (Relationship) on the negative side (average rating: 3.9). The text comments on each statement provide more insights:

- RQ 1 (Publications):** Many experts consider the publication list as compact and clear (E1, E2, E10, E12), representing a good amount of information (E3, E4, E5, E8, E10, E12, E13), some, however, request an even more compact representation (E2, E7, E8, E11). Having image previews is highlighted by several experts (E2, E3, E8, E11, E12, E13, E14), sometimes together with the wish to see a bigger version of the image (E5, E8, E12) or more than one image (E4, E5). Particular helpful features are the integration of links to other services (E1, E3, E8, E12, E13, E14) and providing the BibTeX reference (E4, E5, E12).
- RQ 2 (Structure):** Structuring the information based on (interactively selectable) keywords is perceived as useful by many experts (E1, E2, E4, E7, E10, E12, E13, E14). Also, the integration of sparklines into the word clouds is well-received (E2, E7, E10, E14). While some experts positively mention the use of word clouds (E7, E8, E13), other experts express general doubts to use them (E3, E5, E12). We understand these concerns, but still believe that word clouds are a good trade-off between analytical expressiveness and compact representation.
- RQ 3 (History):** The timeline is helpful to get a historic overview or analyzing developments (E12, E13, E14), but especially, the temporal comparison regarding different selectors is mentioned as a key feature (E2, E3, E5, E6, E7, E8, E10, E13, E14). There are also negative comments criticizing different individual features, but we could not identify recurring issues.
- RQ 4 (Relationship):** Some experts describe retrieving relationships as one of the main strengths of SurVis (E1, E7, E14). Different features are identified as specifically useful: selectors in general (E5, E3, E10, E14), sparklines showing correlations between selectors (E2, E6, E13), and clustering (E1, E13, E14). Again, we could not identify common topics among the individual negative comments.

These comments further support the general assumption that SurVis fulfills the initially formulated requirements, however, having still some room for improvements. Many of the suggested extensions and changes are easy to consider in future versions of SurVis, others, however, would be more difficult or even go along with introducing other drawbacks.

Usability: As also listed in Table 4, the experts agree with the statement on usability at a comparable level as for the requirement statements (average: 3.8), however, varying somewhat more across the different ratings. The text comments on usability mostly repeat parts of above comments and only add few extra points. Neither on positive nor negative side, there are consistent statements among the experts. Positive examples of individual statements are, for instance, that "After a short period of exploration [...], it's clear how to use SurVis" (E8) and that most of the features are self-explaining (E10). Negative ones describe, for example, that some features are hidden and might be overlooked initially (E8, E10), that the display is too packed (E2) and contains too much information (E12), and that specific features are not

self-explaining enough (E8, E12). We conclude that the experts were not able to detect severe usability issues, but further usability improvements are still recommended.

Selectors: Regarding the importance ratings of selectors, on average, the 🔍 Keyword (4.4) and 👤 Author selector are considered most important, followed by 🔍 Search (3.7), ⌚ Time (3.7), 📊 Similarity (3.6), and 📄 Citation (3.6) selectors. While almost all experts agree that these selectors are at least partly relevant, the opinions are more diverse regarding the 📅 Series (2.9) and especially the 📦 Cluster (2.4) selectors. Moreover, the 📦 Cluster selector is the only type that receives a substantial number of answers *don't know* (5 answers). Still, some experts like the selector (high ratings and positive comments). Hence, clustering (and selecting publications series) is only partly considered useful, whereas all other selection criteria are rated as important by most experts.

Additional Feedback: Various extra features are suggested to extend SurVis, for instance, an *OR* combination of selectors (E1), a taxonomy visualization (E5), or support for the collaborative curation of literature collections (E8)—none, however, is named several times. In comparison to related systems, the experts highlight that SurVis has powerful features supporting the exploration and visual analysis of literature data (E1, E2, E3, E7, E14). E5 generally remarks that unlike other systems "it's clear SurVis was made by researchers, for researchers." Contrasting SurVis to visual literature browsers such as Treevis.net [30], some experts miss the image-centric literature representation (E4, E6, E8)—this representation, however, is only suitable when representative images are available, which is not true for non-visualization literature usually (E8). Within the additional comments, E4 describes SurVis as a tool that is useful beyond the visualization community and similarly E7 adds: "It is surely a tool from which many researchers can benefit." E8 concludes that SurVis is a "really convenient way to create and share a searchable, curated literature database."

6 CONCLUSION

We presented SurVis as an interactive visual analytics system for literature collections. It provides a versatile tool for authors of a literature survey to structure and analyze the references of their survey. A distinguishing feature is the specific focus of the system on disseminating the collection in order to make the analysis reproducible and extensible by the *Reader* of the survey. SurVis goes beyond what available systems provide, in particular, regarding the analysis of temporal developments and retrieving relationships among the references. The main technical contribution is the novel use of sparkline visualizations to indicate the agreement of documents to interactively activated selectors. This allows a retrieval and deep analysis of overlapping groups and relationships within the literature collection. We designed SurVis to support a meaningful visual analysis process: we modeled two connected reasoning processes of user types *Curator* and *Reader*, derived requirements from it, and checked the fulfillment of the requirements based on feedback of visual analytics experts.

ACKNOWLEDGMENTS

We thank all people providing feedback on the system, in particular, the early users of SurVis and all participants of the questionnaire study. Fabian Beck is indebted to the Baden-Württemberg Stiftung for the financial support of this research project within the Postdoctoral Fellowship for Leading Early Career Researchers.

REFERENCES

- [1] W. Aigner, S. Miksch, H. Schumann, and C. Tominski. *Visualization of Time-Oriented Data*. Springer, 2011.
- [2] B. Alsallakh, L. Micallef, W. Aigner, H. Hauser, S. Miksch, and P. Rodgers. Visualizing sets and set-typed data: State-of-the-art and future challenges. In *EuroVis - STARs*, EuroVis, pages 1–21. Eurographics Association, 2014.
- [3] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM, 1999.
- [4] F. Beck, M. Burch, S. Diehl, and D. Weiskopf. The state of the art in visualizing dynamic graphs. In *EuroVis - STARs*, EuroVis, pages 83–103. Eurographics Association, 2014.
- [5] F. Beck, B. Dit, J. Velasco-Madden, D. Weiskopf, and D. Poshyanyk. Rethinking user interfaces for feature location. In *Proceedings of the 23rd IEEE International Conference on Program Comprehension, ICPC*. IEEE, 2015.
- [6] F. Beck, F. Hollerich, S. Diehl, and D. Weiskopf. Visual monitoring of numeric variables embedded in source code. In *Proceedings of the 1st IEEE Working Conference on Software Visualization, VISSOFT*, pages 1–4. IEEE, 2013.
- [7] F. Beck, O. Moseler, S. Diehl, and G. D. Rey. In situ understanding of performance bottlenecks through visually augmented code. In *Proceedings of the 21st IEEE International Conference on Program Comprehension, ICPC*, pages 63–72. IEEE, 2013.
- [8] P. Bergström and D. C. Atkinson. Augmenting the exploration of digital libraries with Web-based visualizations. In *Proceedings of the 4th International Conference on Digital Information Management, ICDIM*, pages 1–7. IEEE, 2009.
- [9] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. State-of-the-art of visualization for eye tracking data. In *EuroVis - STARs*, EuroVis, pages 63–82. Eurographics Association, 2014.
- [10] U. Brandes and B. Nick. Asymmetric relations in longitudinal social networks. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2283–2290, 2011.
- [11] J. Choo, C. Lee, E. Clarkson, Z. Liu, H. Lee, D. H. P. Chau, F. Li, R. Kannan, C. D. Stolper, D. Inouye, N. Mehta, H. Ouyang, S. Som, A. Gray, J. Stasko, and H. Park. VisIRR: Interactive visual information retrieval and recommendation for large-scale document data. Technical report, Georgia Institute of Technology, 2013.
- [12] J. K. Chou and C. K. Yang. PaperVis: Literature review made easy. *Computer Graphics Forum*, 30(3):721–730, 2011.
- [13] M. Dörk, N. Henry Riche, G. Ramos, and S. Dumais. PivotPaths: Strolling through faceted information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2709–2718, 2012.
- [14] C. Dunne, B. Shneiderman, R. Gove, J. Klavans, and B. Dorr. Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization. *Journal of the American Society for Information Science and Technology*, 63(12):2351–2369, 2012.
- [15] N. Elmqvist and P. Tsigas. CiteWiz: a tool for the visualization of scientific citation networks. *Information Visualization*, 6(3):215–232, 2007.
- [16] M. A. Hearst. TileBars: visualization of term distribution information in full text information access. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI*, pages 59–66. ACM Press/Addison-Wesley Publishing Co., 1995.
- [17] F. Heimerl, Q. Han, S. Koch, and T. Ertl. CiteRivers: Visual analytics of citation patterns. *IEEE Transactions on Visualization and Computer Graphics*, 22(01), 2016.
- [18] K. E. Isaacs, A. Giménez, I. Jusufi, T. Gamblin, A. Bhatele, M. Schulz, B. Hamann, and P.-T. Bremer. State of the art of performance visualization. In *EuroVis - STARs*, EuroVis, pages 141–160. Eurographics Association, 2014.
- [19] S. Koch, H. Bosch, M. Giereth, and T. Ertl. Iterative integration of visual insights during patent search and analysis. In *IEEE Symposium on Visual Analytics Science and Technology, VAST*, pages 203–210. IEEE, 2009.
- [20] S. Koch, H. Bosch, M. Giereth, and T. Ertl. Iterative integration of visual insights during scalable patent search and analysis. *IEEE Transactions on Visualization and Computer Graphics*, 17(5):557–569, 2011.
- [21] K. Kucher and A. Kerren. Text visualization techniques: Taxonomy, visual survey, and community insights. In *Proceedings of the 8th IEEE Pacific Visualization Symposium, PacificVis*, pages 117–121. IEEE, 2015.
- [22] B. Lee, N. Henry Riche, A. K. Karlson, and S. Carpendale. SparkClouds: Visualizing trends in tag clouds. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1182–1189, 2010.
- [23] M. Lima. visualcomplexity. <http://www.visualcomplexity.com/>.
- [24] S. Liu, D. Maljovec, B. Wang, P.-T. Bremer, and V. Pascucci. Visualizing high-dimensional data: Advances in the past decade. In *Eurographics Conference on Visualization (EuroVis) - STARs*, EuroVis. The Eurographics Association, 2015.
- [25] J. Matejka, T. Grossman, and G. Fitzmaurice. Citeology: visualizing paper genealogy. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems, CHI*, pages 181–190. ACM, 2012.
- [26] F. Osborne, E. Motta, and P. Mulholland. Exploring scholarly data with Rexplore. In *The Semantic Web, ISWC*, pages 460–477. Springer, 2013.
- [27] P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of International Conference on Intelligence Analysis*, volume 5, pages 2–4, 2005.
- [28] H. Reiterer, G. Tullius, and T. M. Mann. INSYDER: a content-based visual-information-seeking system for the Web. *International Journal on Digital Libraries*, 5(1):25–41, 2005.
- [29] M. Schots, R. Vasconcelos, and C. Werner. A quasi-systematic review on software visualization approaches for software reuse. Technical Report ES-748/14, Federal University of Rio de Janeiro, 2014.
- [30] H.-J. Schulz. Trevis.net: A tree visualization reference. *IEEE Computer Graphics and Applications*, 31(6):11–15, 2011.
- [31] J. Stasko, J. Choo, Y. Han, M. Hu, H. Pileggi, R. Sadanaand, and C. D. Stolper. CiteVis: Exploring conference paper citation data visually. *Posters of IEEE InfoVis*, 2013.
- [32] J. Stasko, C. Görg, and Z. Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 2008.
- [33] J. J. Thomas and K. A. Cook. A visual analytics agenda. *Computer Graphics and Applications*, 26(1):10–13, 2006.
- [34] E. R. Tufte. *Beautiful Evidence*. Graphics Press, 1st edition, 2006.
- [35] C. Vehlow, F. Beck, and D. Weiskopf. The state of the art in visualizing group structures in graphs. In *Eurographics Conference on Visualization (EuroVis) - STARs*, EuroVis. The Eurographics Association, 2015.
- [36] M. A. Yalçın, N. Elmqvist, and B. Bederson. Keshif. <http://keshif.me>.
- [37] J. Zhao, C. Collins, F. Chevalier, and R. Balakrishnan. Interactive exploration of implicit and explicit relations in faceted datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2080–2089, 2013.